

TechNet

2006. DECEMBER – 2007. JANUÁR

MAGAZIN *MÉLYVÍZ, CSAK ÚSZÓKNAK!*

Microsoft®

Exchange Server 2007

TÜZETES ELLENŐRZÉSEK

Közvetlen védelem
az internetes kellemetlenségek
és fenyegetések ellen

POWERSHELL- SCRIPTEK A GYAKORLATBAN

Az objektumok feldolgozásának
kilenc fontos parancsa

KÉSZEN KAPOTT CRM

Kinek és mire jó
a Microsoft Dynamics CRM,
és hogyan mérhető
az üzleti teljesítmény?



**TARTALMAS
ÚJÍTÁSOK ÉS KIBŐVÜLT
FUNKCIONALITÁS:**

ARCHITEKTÚRA

**EXCHANGE
MANAGEMENT SHELL**

**MAGAS
RENDELKEZÉSRE ÁLLÁS**

Microsoft TechNet

Hol van már a tavalyi hó!? Nálunk idén még itt!

Évnyitó akciónk keretében valamennyi képzésünkön még tavalyi áron vehet részt!*

Néhány képzés ízelítőül az év elejéről, a teljes kínálatért kérjük,
keresse fel internetoldalunkat!

- Windows 2003 üzemeltetés (2273) – január 29-február 2., március 5-9.
- Windows 2003 Active Directory (2279) – január 29-február 2., február 19-23.
- Windows 2003 biztonság és PKI (2830) – január 29-február 2., március 19-23.
- Windows 2003 hálózatzüzemeltetés (2276/77) – február 26-március 2.
- Windows 2003 Cluster szolgáltatások (2087) – március 12-14.
- Windows 2003 hálózatszervertervezés (2278) – március 26-30.
- Windows XP terméktámogató (MCDST) képzés – február 5-9., március 19-23.
- SQL Server 2000/2005 alapok (2071) – február 5-6.
- SQL Server 2005 programozás (2779) – február 7-9.
- SQL Server 2000 programozás (2073) – február 19-23.
- Exchange Server 2003 üzemeltetés (2400) – február 19-23.
- Microsoft CRM 3.0 alkalmazások használata (8521-24) – február 15-17.
- C# és Visual Studio 2005 technológiai alapismeretek (2609E) – március 5-9.

Kedvezményes Microsoft mérnök képzéssorozatok (MCSA, MCSE, MCTS, MCITP).

A 2007-es év Microsoft tanfolyami újdonságai

- Windows Vista
- SQL Server 2005 üzleti intelligencia rendszerek
- Exchange Server 2007
- SharePoint Services 3.0
és SharePoint Portal Server 2007
- ISA Server 2006

a mi tudásunk
az **ÖN** sikere

*Az akció a 2007. március 1-ig megrendelt tanfolyamok esetében érvényes. Az akció részleteivel és 2007-ben megjelenő új képzéseinkkel és szolgáltatásainkkal kapcsolatosan internetoldalunkon találhat további információt. A képzésekre a jogszabályok szerint igénybe vehető a **szakképzési hozzájárulás** és felhasználható az **SA oktatási utalvány**.



Telefon: 203-0304/4122 m.

www.szamalk.hu/tisza



KEZDJÜK FRISSEN AZ ÚJ ÉVET!

Elkészült a Windows Vista,
az Office 2007 Rendszer
és az Exchange Server 2007.

Az új Windows-, Office- és Exchange-verziók 2007 első felétől lesznek elérhetők a felhasználók és a vállalatok számára, érdemes tehát minél hamarabb megismerkedni velük. Nem egyszerűen újabb verziók megjelenésével van ugyanis dolgunk: ezek a szoftverek igyekeznek újraértelmezni az informatikát, és azt a végfelhasználók kezébe adni, megkönnyítve a mindennapi munkavégzést. Amellett, hogy számtalan esetben okosabbak, jobbak elődeiknél, ezek az új technológiák képezik majd a következő évtizedben megjelenő további szoftverek alapját is. Nagyon sok változásról beszélünk, és ezek közül néhánynak lehet, hogy még nem látjuk értelmét, fontosságát – de idővel össze fog állni a kép.

Az elmúlt egy évben rengeteget foglalkoztunk a Windows Vista és a 2007-es Office Rendszer újdonságaival mind a Magazinban, mind eseményeken, de ehhez képest lényegesen kevesebbet – szinte semmit nem – szóltunk az új Exchange Serverről, pedig rendszergazda-szemmel nézve ez is legalább annyira fontos terület. Az Exchange Server már évek óta biztosítja a vállalatok számára talán legfontosabb csoportmunka-funkció – igen, ez a levelezés – zavartalan működését. Most látni fogjuk, hogy az Exchange túllépett régi önmagán. Teljesen újratervezték a szoftvert, sokkal modulárisabb lett, és szorosabban integrálódik a többi szervermegoldással is, valamint lényegesen több felhasználót és nagyobb terhelést képes kezelni – hála a 64 bitnek.

A fejlődésnek természetesen még nincs vége: a következő jelentősebb áttörést a System Center rendszermenedzsment-termékcsalád tavaszi megjelenése és a „Longhorn” Server év végi érkezése jelenti majd.



Budai Péter

Microsoft Magyarország

SZERKESZTŐSÉG

Főszerkesztő
Sziebig Andrea – asziebig@vogelburda.hu
Szakmai lektor
Budai Péter – i-pbudai@microsoft.com
Vezető szerkesztő
Varga János – jvarga@vogelburda.hu
Nyomdai előkészítés
Budakeszi Bejárati Kft.
Korrektor
Matula Zsolt
Lapterv és címlap
Emotion Bt.

Szerkesztőség és kiadó címe:
Vogel Burda Communications Kft.
1077 Budapest, Kéthly Anna tér 1.
Tel.: 888-3400, fax: 888-3499

KIADÓ

A Microsoft Magyarországi megbízásából kiadja
a Vogel Burda Communications Kft.

A kiadásért felel

Walitschek Csilla
cswalitschek@vogelburda.hu
Tel.: 888-3450, fax: 888-3499

A TechNetben közölt cikkek fordítása, utánnymása, sokszorosítása és adatrendszerekben való tárolása kizárólag a kiadó engedélyével történhet. A megjelent cikkeket szabadalmi vagy más védettségre való tekintet nélkül használjuk fel.

Hirdetési igazgató:

Farkas Viola – vfarkas@vogelburda.hu, tel.: 888-3459

Médiareferensek:

Harsányi Erika – eharsanyi@vogelburda.hu, tel.: 888-3452
Németh Krisztina – knemeth@vogelburda.hu, tel.: 888-3468
Rátóti Sarolta – srato@vogelburda.hu, tel.: 888-3453
Szendrey Szilvia – szendrey@vogelburda.hu, tel.: 888-3455
Fax: 888-3459

Hirdetési koordinátor:

Szöke Erika – eszoke@vogelburda.hu
Tel.: 888-3411, fax: 888-3459

Nemzetközi hirdetésfelvétel:

Eric N. Wicha – ewicha@vogelburda.com
Vogel Burda Holding
Pocistrasse 11, D-80336 München
Tel.: +49 89 74642-326, fax: +49 89 74642-325

A hirdetések körültekintő gondozását kötelességünknek érezzük, de tartalmukért felelősséget nem vállalunk.

Marketing:

Gajdos Barna – bgajdos@vogelburda.hu, tel.: 888-3494

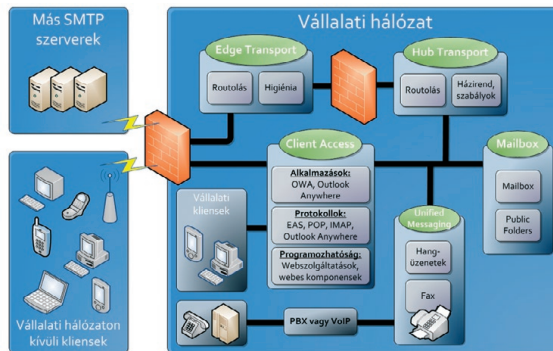
Terjesztés

Terjesztett példányszám: 3000

Nyomda:

Pauker Nyomdaipari Kft.
1047 Budapest, Baross utca 11-15.
Felelős vezető: Vértess Gábor ügyvezető igazgató
ISSN 1586-5185

Címlapon



Az Exchange Server 2007 architektúrája

(Petrényi József)

Az Exchange Server 2007 számtalan apró elemből épül fel. Ezeket az apró funkcionális elemeket csoportokba rendezték – és a csoportokat szerepkörnek nevezték el

6. oldal

A megújult Exchange Management Console

(Subicz Péter)

Az új Exchange-et fejlesztő csapat nagy hangsúlyt helyezett az új Management Console kidolgozására is – az eredmény önmagáért beszél

12. oldal

Kitekintés az Outlook 2007-re

(Budai Péter)

Az e számunkat uraló Exchange Server 2007 mellett a Microsoft Office-ban található Outlook-kliens 2007-es változata is szolgál kellemes meglepetésekkel

15. oldal

Exchange Management Shell

(Szabó Dávid)

Az Exchange Server 2007 üzemeltetéssel kapcsolatos újdonságai közül talán a PowerShell-parancsgyűjtemény a legnagyobb áttörés az Exchange 2003-hoz képest

16. oldal

Tervezés és telepítés

(Petrényi József)

Alapos tervezésnek kell megelőznie a tényleges telepítést – ezzel később rengeteg időt és energiát spórolunk meg magunknak

18. oldal

Exchange Server 2007: magas rendelkezésre állás

(Ország Tamás)

Cikkünkben az Exchange Server 2007 hibatűrését fokozó – online és offline – megoldásokat tekintjük át

23. oldal

Mikor váltunk 64 bitre?

(Budai Péter)

Rengeteg a tévhit a 64 bites rendszerekkel kapcsolatban. Például ha ma valaki 64 bites hardvert vásárol, rögtön 64 bites operációs rendszert tesz rá. Biztosan jó döntés ez?

26. oldal

Biztonság

Tüzetes ellenőrzések

(Kelemen László)

Írásunkban az Exchange-nek azokat a biztonsági vonatkozásait tekintjük át, amelyek az internetes kellemetlenségek és fenyegetések, azaz a levélszemét és a kártevők elleni közvetlen védelmet szolgálják

30. oldal

Scripting

Powershell-scriptek a gyakorlatban

(Fóti Marcell)

Ezúttal picit behatóbban elemezzük a CommandLeteket, valamint megismerkedünk az objektumok feldolgozásának kilenc fontos parancsával

34. oldal

The diagram shows a workflow for Scorecard Reporting. It starts with 'Alkalmazásgazdák' (Application Owners) and 'Üzleti döntéshozók' (Business Decision Makers) using 'Office Scorecard Webpart' and 'Office Report Viewer Webpart'. These lead to 'Scorecard alkalmazás' (Scorecard Application) which connects to 'Levél-adatbázis' (Mail Database). The application also interacts with 'CRM adatbázis' (CRM Database), 'OLAP kockák' (OLAP Cubes), and 'IT adminisztráció' (IT Administration).

```

Machine: fdc | Scope: futureinc.com
[PS] C:\>import-csv users.csv | foreach {add-Distrib
júserok -member "$($_.Alias)"}
[PS] C:\>get-DistributionGroupMember -Identity ujüsero
Name RecipientType
-----
Kozlovicski Jeno UserMailbox
Borzenyak Tamas UserMailbox
Trap Istvan UserMailbox
  
```

Alkalmazásplatform

Készen kapott CRM

(Kovács László-Wentzel István)

Kinek és mire jó a Microsoft Dynamics CRM, és hogyan mérhető az üzleti teljesítmény az új, kiszolgáló alapú mutatószámrendszer és teljesítménymutató alkalmazás, a Business Scorecard Manager 2005 használatával?

37. oldal

LINQ

(Nagy Levente)

Ha valaki készített már objektumorientált alkalmazást, amelyik használt adatbázist, esetleg XML-dokumentumokat, az tudja, hogy mennyire különbözik az egyes területek hozzáállása az adatok tárolásához, eléréséhez, a vezérléshez

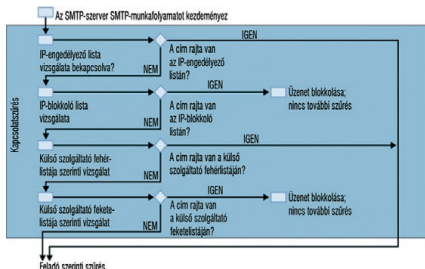
40. oldal

Konkurenciakezelés az SQL Server 2005-ben

(Kovács Zoltán)

Az SQL Server 2005 fejlesztői régi adósságot törlesztettek: megvalósították a versenytársaknál már több mint egy évtizede sikerrel alkalmazott optimista konkurenciakezelési modellt.

44. oldal



Közösség

A Microsoft Magyarország bemutatása

(Budai Péter)

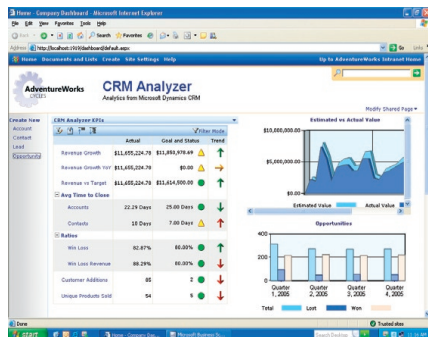
Bemutatkozik egy olyan vállalat, ahol sokan szeretnek dolgozni – és sokat is tesznek a sikerekért

49. oldal



```

Machine: fdc | Scope: futureinc.com
[PS] C:\Documents and Settings\Administrator\My Documents\PublicFolderClientPermission -I
-identity "Teszt Nyilvános Mappa\Alnappa" -user "Teszt user" -accessrights Create
Items
Identity User AccessRights
Teszt Nyilvános Mappa\... futureinc.com/Teszt user (CreateItems)
[PS] C:\Documents and Settings\Administrator\My Documents\PublicFolderClientPermission -I
-identity "Teszt Nyilvános Mappa\Alnappa"
Identity User AccessRights
Teszt Nyilvános Mappa\... Default (Author)
Teszt Nyilvános Mappa\... futureinc.com/Teszt user (CreateItems)
Teszt Nyilvános Mappa\... futureinc.com/Guests/Adm... (Come?)
Teszt Nyilvános Mappa\... Anonymous (CreateItems)
  
```



IT-BUSINESS TODAY goes mobile ...

Technológiai partnerünk: **hedz** mobile@info.com

Próbálja ki most!
Küldje el az ITBTODAY szót SMS-ben a +36 30 285 5441 számra és kövesse az instrukciókat!
WAP cím: wap.it-business.hu

Bővebb információ: www.it-business.hu

AZ EXCHANGE SERVER 2007 ARCHITEKTÚRÁJA

Lényeges változások történtek az Exchange Server 2007 felépítésében – ezektől azonban nem szabad megijedni, mind hasznunkra válik majd.

Az Exchange Server 2007 számtalan apró elemből épül fel. Ezeket az apró funkcionális elemeket csoportokba rendezték – és a csoportokat szerepkörnek nevezték el. Egy Exchange-szerveralkalmazás szerepkörökből áll össze, maga az Exchange-organizáció pedig az egyes szerepköröket megvalósító Exchange-szerverekből.

Ha az architektúrát akarjuk körbejárni, akkor először az egyes szerepköröket kell alaposan szemügyre vennünk, majd beszélnünk kell a köztük lévő kapcsolatokról.

Hogy maguk a szerepkörök ne csak úgy lebegjenek a levegőben, érdemes lesz rendszeresen vissza-visszatérni az 1. ábrához.

Itt látható vázlatosan, hogyan is épül fel egy teljes Exchange Server 2007-organizáció, milyen kapcsolatban állnak egymással az egyes szerepkörök.

De még mielőtt beleszarnánk a közepébe, nem árt letisztázni, milyen is a viszony a fizikai gép formájában megjelenő Exchange-szerver és az egyes szerepkörök között.

Vadregényes.

Az öt szerepkör közül négy minden további nélkül bezúfolható egy szerverbe. (Az 1. ábrán is látható, hogy az Edge Transport funkció tűzfal mögött van – ezt azért nem lenne egyszerű megvalósítani egy belső hálóra rakott mailbox-szerveren.) Mindemellert semmi akadálya nincs annak sem, hogy mindegyik funkció fizikailag is külön vasra kerüljön. Mindig az adott situáció szabja meg, hogy a megvalósítás milyen kompromisszumok szerint történjen meg.

Nézzük részletesen az egyes szerepköröket!

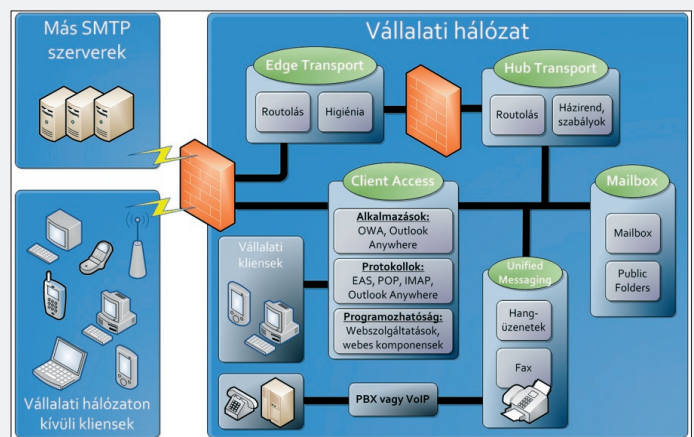
Az Edge Transport szerepkör

Bóklászik a levél az interneten, végül boldogan megtalálja a dzsungelben cégünk MX rekordját és fáradtan, de alapvetően elégedetten be is esik a megadott szerverre.

Valljuk be őszintén, ez manapság a legritkább esetben Exchange-szerver. Nem mindig volt ez így, de amióta annyira koszos lett az internet, egyre többen pakoltak az Exchange-organizá-

ciójuk elé valamilyen más gyártótól származó smarthostot. Ez elvégezte a piszkos munka nagy részét: szűrte a spameket, gyapálta a vírusokat és a spyware-eket.

Ebben a megközelítésben a fő szempont az volt, hogy az ember leválasztotta a mocskos



1. ábra. Az Exchange-organizáció architektúrája

munkát az éles organizációról. Sőt, magát a smarthostot is ki lehetett tenni az előszobába.

Tulajdonképpen ugyanezt a filozófiát valószínűleg meg az Edge Transport szerepkör. Például azt állítja magáról, hogy olyan Exchange-szerver, amelyik nem tagja az organizációnak. Sőt: nem tagja az Active Directorynek sem!

Azért ez így már elég érdekes. Gyakorlatilag kaptunk egy olyan Exchange-szerveret, amelynek semmi Exchange jellege sincs: működik rajta egy SMTP-motor, a motorra szorgos ügynökök serege kapcsolódik rá, mindegyik egyfajta szűrőként funkcionálva, teleaggathatjuk mindenféle szabályokkal – és ennyi. Mitől lesz ez mégis több, mint egy külső gyártó által megvalósított megoldás? Attól, hogy lesz rajta egy LDAP-szerver, amelyet össze tudunk

lőni a Microsoft Active Directoryjával – és innentől az Exchange-szerver teljesen úgy fog viselkedni, mintha benne lenne a címtárban, annak ellenére, hogy valójában nincs benne.

A Microsoft önállóan megvalósított LDAP-szervere az Active Directory Application Mode, röviden ADAM névre hallgat (amit épp most keresztelnek át Active Directory Lightweight Directory Services-re, vagyis AD/LDS-re).

Gyakorlatilag egy kicsi, önálló címtár, amelyik teljesen hasonlóan működik, mint a nagytestvére, ugyanúgy ADSIEdit-tel, LDAP-vel és ldidfe/csvde segédprogramokkal kezelhető, mint a nagy címtár – de nem olvad bele annak partícióiba, önálló, saját adatbázisa van. Mivel mindkét címtár Microsoft-fejlesztés, és működésüket tekintve nagyon is hasonlítanak egymáshoz, a szinkronizációjuk sem okozhat különösebb problémát. (Megjegyzés: akkor is ADAM-AD szinkronizálás történik, ha az Edge Transport számítógépet beléptetjük a tartományba.)

Magát a szinkronizációt az úgynevezett EdgeSync szolgáltatás fogja megvalósítani – ez a nevével ellentétben a Hub Transport szervereken fut.

Az Edge Transport szerepkör alapvetően három csoport tevékenységéből áll össze:

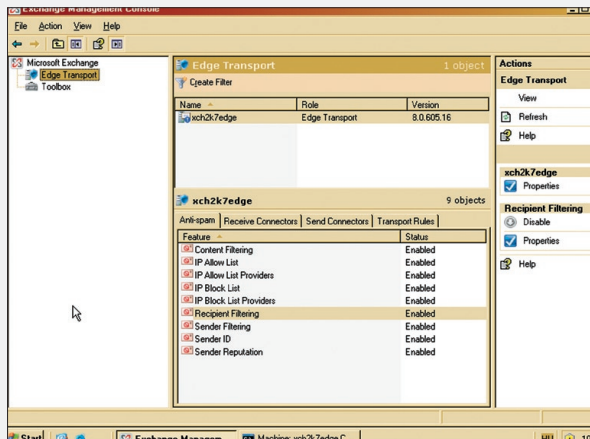
- az egyik a levéltovábbító mechanizmus, amely SMTP ugyan, de teljesen el van rejtve a rendszergazda elől;
- rendelkezésünkre áll egy sereg ügynök, ezek különböző apró feladatokra haragnak;
- végül kapunk egy szabályrendszert is.

A két utóbbira egyaránt vonatkozik, hogy egyszerre látnak el higiéniai jellegű, illetve levéltovábbítási funkciókat is.

Először vegyük szemügyre az ügynököket. Láthatjuk a grafikus felületen, van néhány. Aki azt hiszi, hogy ennyi a készlet, ezekkel kell dolgozni, az téved. Fura módon nem minden ügynököt vezettek ki a felületre.

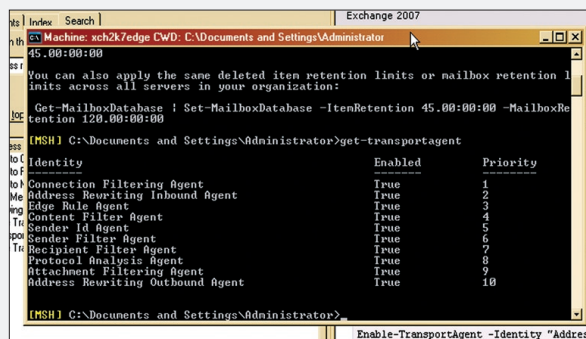
Ezeket látjuk (2. ábra) a grafikus konzolon.

És ezeket látjuk (3. ábra), ha a get-transport-agent paranccsal lekérjük az ügynöklistát az Exchange Management Shell és a Windows PowerShell segítségével.



2. ábra. Antispam-ügynökök a grafikus felületen

Ha a két képet összehasonlítjuk, feltűnhet, hogy a grafikus felületen akad néhány olyan elem, amelyik tulajdonképpen nem is ügynök (például IP Allow List), ezzel szemben alul találunk néhány olyan ügynököt, amelyet nem tudunk konfigurálni a konzolból (például Address rewriting-ügynökök). Mindez nem véletlen. Már a korábbi Exchange-verziókban is jelen volt az a tendencia, hogy az igazán hű-



3. ábra. Ügynökök PowerShell-commandletből

zós dolgokat elrejtették az 1.0 adminok elől. Itt is ugyanerről van szó: ha keményebb, veszélyesebb dolgokat akarunk csinálni, akkor vegyük elő a PowerShellt és világosan mondjuk meg, mit is szeretnénk. Csak úgy ne katingassunk bele a GUI-ba.

Természetesen nem feltétlenül kell ezt a szerepkört telepítenünk. A külső gyártók eszközei is megfelelőek lehetnek, de van egy olyan terület, ahol azért az Edge Transport szerver többet tud: ez a címtár-szinkronizáció. Habár ma már egyre több helyen konfigurálható LDAP-lekérdezés a címtár felé, de az ADAM-AD szinkronizáció emellé plusz összhangot is biztosít a smarthost és a címtár között.

Nemcsak a postafiókokhoz rendelt e-mail-címeket adja át, hanem például átküldi a címtár konfigurációs névtéréből az Exchange-topológiára vonatkozó információkat a Hub Transport szerverek elhelyezkedéséről.

A Hub Transport szerepkör

Képzeltbeli levelünk a Hub Transport szerepkört birtokló Exchange-szerverre került.

Amikor először találkozunk a névvel, egyértelműnek látszik: igen, ez foglalkozik a link state tábla kezelésével, azaz az e-mail routolásával. Tovább olvasva azonban nem kis meglepetéssel észlelhetjük, hogy az Exchange Server 2007-ben nem lesz link state tábla. Az egész, gyönyörűen felépített routolási rendszert a Routing Master funkcióval egyetemben kidobták a rendszerből.

Néha bizony úgy tűnik, az Exchange fejlődése nagyrészt kidobásból áll. (Persze nem egészen erről van szó, általában az Exchange-ben bevált technikák köszönnek vissza az operációs rendszer szintjén – és miatt lesz végül felesleges az Exchange-ben egyedileg megvalósított változat.) Most is hasonló folyamat játszódott le: volt egyfélő egy különálló, link state táblán alapuló rendszer, és volt egy Active Directory hierarchiakezelő rendszer – ez a Knowledge Consistency Checker (KCC) –, amely a telephelyen belüli, illetve telephelyek közötti replikáció topológiáját kezelte.

A fejlesztők észrevették a minták hasonlóságát, és azt mondták, miért ne bizzuk

az e-mail-routolást az Active Directory topológiakezelő metódusaira? Ezzel tulajdonképpen vége is lett a sunyításnak – eddig ugyanis megtehettük, hogy bár fizikailag egy telephelyen voltak az Exchange-szervereink, de mégis külön Routing Groupba kerültek. Ennek

mostantól vége. Az AD site lesz az Exchange Routing alapja is. (Ha belegondolunk, ilyen eset általában akkor fordult elő, ha mixed módú organizációban kellett különböző adminisztratív csoportokat létrehozni. Itt viszont az „adminisztratív csoport” fogalma is megszűnik.)

Próbáljunk hozzászokni a gondolathoz:

- Nem lesz Exchange Routing Master funkció. Helyette lesz Active Directory-KCC.
- Kicsit körülményes lesz a levelezési utak súlyozása, azt csak PowerShellben tudjuk beállítani (Set-adsitellink a parancs neve).
- Nem lesz RGC – azaz Routing Group Connector. Helyette lesz AD Site Link Connector.

Egy kis töprengést azért ez is megér. Alapvetően a site-site konnektor lehet IP/RPC vagy SMTP alapú. Alapvetően. De mi fog történni, ha egy SMTP site-site konnektort akarok használni egy SMTP levelezési rendszerben? Káros. Éppen ezért az Exchange 2007-ben nem támogatott az SMTP alapú site-site konnektor (ez újabb szög az SMTP site-site konnektor koporsójába – feltéve, hogy már korábban nem dobtuk ki korlátozott replikációs képességei miatt).

Milyen lesz az együttélés az Exchange 2003 szerverekkel?

Telepítéskor automatikusan jön létre egy Routing Group – a roppant hangzatos DWBGZMFD01QNBJR névre hallgatva. (Valószínűleg a fejlesztő a kutyájáról nevezte el a csoportot.) Ide kerül bele az összes Exchange 2007 szerver, és ez a Routing Group fog részt venni a kommunikációban – azaz a régi szerverek az új Exchange-szerverek ilyen arcát fogják látni. Amennyiben egy világméretű organizációban van szerencsénk dolgozni, akkor vélhetően nem fogja kielégíteni az igényeinket egy olyan megoldás, hogy a tokiói és az alaszakai Exchange 2007-szerverünk ugyanabban a Routing Groupban legyen, és a legelőször felhúzott szerver legyen közülük a bridgehead szerver. Rossz hír: az Exchange 2007-ben Routing Group Connectort nem tudunk grafikus felületről létrehozni. Jó hír, hogy Powershellből igen: a New-RoutingGroupConnector és a Set-RoutingGroupConnector commandlet lesz a barátunk.

Különálló Exchange SMTP-szerver

Tehát nem a Hub Transport szerver lesz a Routing Master az organizációban. Akkor mi

is lesz a szerepe? Nos, ez lesz az SMTP-szerver! Méghozzá egyedül ebben a szerepkörben lesz SMTP-motor (valamint az Edge Transport szerepkörben, amennyiben telepítjük.)

Kell egy kis idő mindezt megemészteni...

Nem lesz mindegyik Exchange-szerveren SMTP-motor. Ahol lesz, ott viszont egy megnedzselt kódban újraírt, teljesen új SMTP-motor fog dübörögni. Az IIS SMTP-szerver motorja megy a kukába. A szerepkörök közötti kommunikációban visszatérünk az RPC protokollra. Amennyiben egy Active Directory-site-on nem valósítjuk meg a Hub

transport lehet a leveleket. Természetesen ebbe a hálózatba bele tudunk avatkozni transport rule-ok segítségével – de a belső levelezésnél erre nem lesz szükség, mert a megfelelő implicit szabályok automatikusan generálódnak majd – beleértve ebbe a Hub Transport és Edge Transport szerepkörök közötti szabályokat is. Az Active Directory-site konnektorok fizikai kezdő és végpontjai az Active Directory-bridgehead szerverek, de az Exchange-organizációban a konkrét Active Directory-site-ok levelezésért felelős Hub Transport szerepkörök megvalósító szervereket fogjuk látni.

Ez annyira így van, hogy külön lesz egy Microsoft Exchange Active Directory Topology szolgáltatás. Ez folyamatosan karbantartja, hogy melyik Exchange-szerver melyik Active Directory-site-on van. A gyakorlatban ez úgy néz ki, hogy minden Exchange-szerverobjektumnak lesz egy olyan tulajdonsága, amelyiknek az értéke mutatja, hol van a konkrét szerver. (Ezzel a trükkel le-

het site-hoz rendelni az Edge Transport Server szerepkört megvalósító szervert is – annak ellenére, hogy bent sincs a címtárban.)

A transport rule-ok

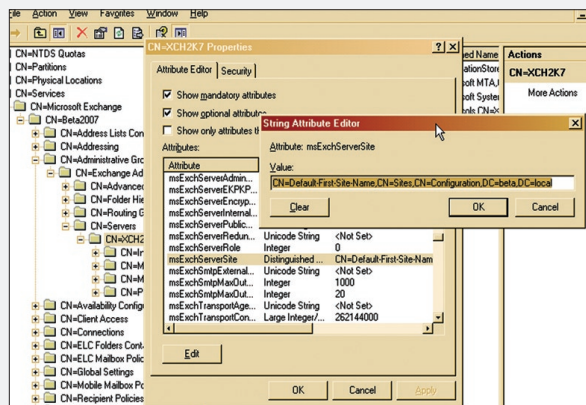
Ha már az érdekességeknél járunk, meg lehet majd tippelni, hol tárolódnak a transport rule-ok. Nem nehéz feladat, nyilván a címtárban. Ez viszont automatikusan azt is jelenti, hogy egy szabály, de hasonlóan egy send connector sem csak egy szerverre fog vonatkozni, hanem az organizáció összes Hub Transport-szerverén is meg fog jelenni.

Érdekes alaposan tanulmányozni az 5. ábrát. Ugye, látjuk? Az organizáció alatti Hub Transport szerepkör alatt fogjuk megtalálni az összes:

- transport rule-t;
- a journaling beállításokat;
- a send connectorokat és
- az összes e-mail address házirendet.

Házirendek

Térjünk vissza egy pillantás erejéig az 1. ábrára: láthatjuk, hogy a Hub Transport szerepkör felelősségi köre a routolás és a házirend.



4. ábra. Site-bejegyzés az Exchange Server-objektumon

Transport szerepkört, nem fogunk tudni kommunikálni a site-ban elhelyezkedő mailbox-szerverekkel. Kizárólag a Hub Transport szerepkörök és az Edge Transport szerepkörök fognak egymással SMTP-n kommunikálni. Csak ezeknek a szerepköröknek lesznek levelezési sorai (queue). Értelemszerűen csak ezeken a szerepkörökön lehet konfigurálni mindent, ami az SMTP-vel kapcsolatos: higiénia, routing, házirendek.

El lehet meditatálni rajta, mennyire kell majd megváltoztatnunk a gondolkodásunkat, a fejünkben lévő sémákat. Kezdvé olyan apróságokkal, hogy nem fogjuk megtalálni a szolgáltatások között a Simple Mail Transport Protocolt – helyette lesz olyan, hogy Microsoft Exchange Transport service. Bevégezve azzal, hogy ha a Hub-szerverek dolga lesz a routolás, akkor mi is van a nem sokkal ezelőtt említett AD alapú topológiával?

Az, hogy mind a kettő működik. A Hub-szerverek ismerik egymást, tudják, hogy melyik Active Directory-site-on ki – vagy kik – a felelősök a levelezésért – azaz tudják, hogy melyik az az SMTP-szerver, amelyeknek to-

A routolást megbeszeltük. Mit takar a házirend? Nos, egyfelől a már szintén említett transport rule-okat. Ezek ténylegesen levéltovábbításra vonatkozó házirendek – és mint ilyenek, abszolút újdonságnak számítanak.

Korábban bármit is akartunk röptében csinálni a levéllel, rögtön SMTP event sink-et kellett bepattintani az SMTP engine levélfeldolgozási mechanizmusába. Itt viszont beépítettek a termékbe egy jó nagy doboznyi építőkövet, amelyből az adminisztrátor meg lehetőségek sokféle levéltovábbítási előírást tud magának összerakni.

Szintén a házirend fogalmába tartozik a journaling funkció is. Igaz, ilyesmi volt az Exchange 2003-ban is, de az új termékben kibővült egy kicsit: a régiben csak egy postafiókot adhattunk meg, ahová másolatot kértünk az összes levélről – most tetszőlegesen szabályozhatjuk a másolandó levelek körét, célpontként pedig megadhatunk külső SMTP szervert, illetve Sharepoint site-ot.

Mailbox Server szerepök

Képzletbeli levelünk áthaladt a pályaudvar váltóin, lassan meg is érkezik végső helyére, a felhasználó postafiókjába. Nézzük, milyen architektúrális változásokat fog itt tapasztalni.

Nincs többé Recipient Update Service (RUS). Kész, megszűnt. Az eredeti szolgáltatás két modulból állt össze: az egyik kikalkulálta a felhasználó e-mail-címeit, a másik modul aszinkron szolgáltatásként időnként lefutott, detektálta, hogy új postafiók került a rendszerbe, és stempliként rányomta az előző modul által kikalkulált e-mail-címegyűttest.

Az Exchange 2007 alatt a második modul szűnt meg – és vele együtt ment a levesbe az aszinkron stemplizés is. Amikor létrehozunk egy új postafiókot, a Recipient Management commandlet rögtön rá is teszi az első modul által kikalkulált címetek.

Persze, a világ nem ennyire egyszerű. A RUS-nak nem csak ez az egy funkciója létezett, maga az aszinkron mechanizmus más feladatok elvégzésére is lehetőséget adott (például címlisták generálására). Az Exchange 2007 fejlesztői átvágták a gordiuszi csomót: ezeket a funkciókat vagy megszüntették, vagy annyira átalakították, hogy ne kelljen aszinkron módosításokra hagyatkozniuk.

(Természetesen amennyiben van Exchange 2003-as mailbox-szerver az organizációban, illetve van olyan alkalmazásunk, amely RUS-t

igényel – LoadSim –, akkor nem tudunk szabadulni tőle. Ilyenkor az Exchange 2003-konzolban lehet RUS-t létrehozni, menedzselni. Alternatív megoldás: használjuk az Update-EmailAddressPolicy, illetve Update-AddressList commandleteket.)

Az **Administrative Groups** sincs többé. Elméletileg az egész organizáció összes szervezete támadható, ha valakinek sürgős adminisztrátorkodhatnéja van. De csak elméletileg. Gyakorlatilag az organizáció szintjén delegálható jogosultságok léteznek, ezek csoprognak le az egyes szervezetre – így tudjuk szeparálni egymástól a különböző szervek adminisztrációs feladatait.

Mondjuk, ha jobban megpiszkáljuk a címtárat, láthatjuk, hogy mégis létezik Administrative Group. Méghozzá az a neve, hogy FYDIBOHF23SPDLT. (Mindez arra utal, hogy a fejlesztőnek valószínűleg két kutyája van.)

A név alapján már sejthető, hogy ez az Administrative Group szintén a visszafelé irányuló kompatibilitás miatt jön létre: amennyiben van az organizációban Exchange

Menedzselte folderok. Amikor a levél már elért a felhasználó postafiókjába, akkor már kikerült a céges házirend által erőltetett szabályok hatóköréből. Gondolja az egyszerű felhasználó. Valójában az Exchange 2007-ben érkezik az úgynevezett Managed Folder fogalom: be lehet állítani, hogy a felhasználó egy csoportjánál mindig jelenjen meg egy bizonyos folder – és éljenek az ehhez a folderhez rendelt szabályok. Nincs többé me nekvés.

Keresés. Feljavított Search-funkcionalitást kapunk. (Search2.0>3.0; ugyanezt használja az SQL2005 is.) Ez annyival jobb lett az elődjénél, hogy míg az Exchange 2003-ban alapértelmezetten ki volt kapcsolva, itt pont fordított a helyzet: telepítés után már működik is.

Client Access Server (CAS) szerepök

Képzletbeli levelünk eljutott egy külső feladótól a megfelelő postafiókba. Az egyik lehetséges útvonalon.

De nem csak ez az egy módszer létezik arra, hogy egy kliens – azaz egy felhasználó

– elérje valahogy a nálunk lévő postafiókot. Az egyik lehetséges módszer az internetes protokollokon keresztüli elérés.

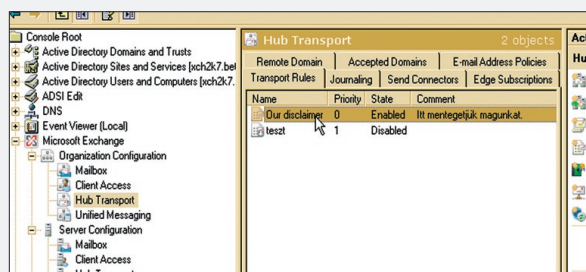
A Client Access Server szerepök hasonlít a legjobban a régi Exchange Frontend funkcióhoz: adatbázis-kezelés nincs, mindenféle hozzáférés kezelése

viszont igen. Mit is takar ez a pongyola „mindenféle” szó? Gyakorlatilag HTTP, HTTPS-, POP3-, IMAP4-eléréseket. (Az előző kettő tulajdonképpen az OWA, illetve az ActiveSync kapcsolódásokat takarja.)

Nézzük, hogyan is illeszkedik ez a szerepök a szervezet architektúrájába.

Amikor egy külső kliens a fenti protokollok bármelyikén keresztül próbál meg elérni egy benti mailbox-szerveren lévő postafiókot, közvetlenül ezt nem fogja tudni megoldani – mindenképpen kell az adott Active Directory-site-ra egy Client Access Server szerepököt ellátó Exchange-szerver.

Amennyiben olyan CAS-szerveren landoltunk, amely másik Active Directory-site-on van, mint amelyiken az elérni szándékozott postafiók, akkor a CAS-szerver meg sem



5. ábra. Házirend az organizációban

2000/2003-szerver, akkor arról az összes Exchange 2007-szerver a fenti izgalmas nevű csoportban fog látszódni.

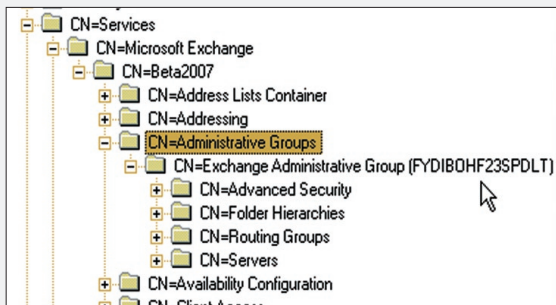
Adatbázis-határok. A jó hír, hogy az adatbázisoknak immár nincs felső mérethatára, egyik termékverzióval sem. A rossz hír, hogy a felhasználók egész biztosan ezt is megpróbálják majd átlépni.

Nézzük a konkrét számokat:

	Max Storage Group / server	Max. database / server	Max. database / Storage Group
Standard	5	5	5
Enterprise	50	50	5

Természetesen a fenti számokat a tényleges hardver is befolyásolja.

próbálkozik közvetlenül hozzáférni a távoli mailbox-szerverhez – ehelyett megpróbálja megkeresni az ott lévő CAS-szervert. Ergo,



6. ábra. A nem létező Administrative Group neve

ha egy adott site-on van mailbox-szerver, és szeretnénk, ha az azon lévő postafiókok a fenti protokollokon keresztül is elérhetők lennének, akkor feltétlenül szükségünk lesz a site-on egy CAS-szerepkörre is.

Így áll össze az a bizonyos „szentháromság”: ha egy adott Active Directory-site-on mailbox-szerverünk van, feltétlenül kell mellé egy Hub Transport szerepkör – hogy egyáltalán a mailbox-szerver számára legyen SMTP-szerver, azaz levélküldési lehetőség – és erősen ajánlott egy CAS-szerepkör is. Ezért nevezik ezt a szerepkörhármast „core roles”-nak, vagyis alapszerepköröknek.

De a CAS-szerepkörnek nem csak ez az egy funkciója van. Része emellett az úgynevezett Availability Service is. Ez az az újdonság, amely intenzíven veri a szöveget a Public Folderek koporsójába.

Arról van szó, hogy a CAS-szervereken fut ez a szolgáltatás, és rendszeresen begyűjti/frissíti a mailbox-szervereken található postafiókokból a felhasználók/erőforrások free/busy, illetve Offline Address Book-információit, majd ezeket az adatokat egy weboldalon keresztül publikálja. Az Active Directory-site-on tartózkodó modernebb (értsd 2007 jelzésű) kliensek (értsd Outlook) már ismerik ezt a trükköt, és a Public Folderek helyett itt fogják keresni az igényelt információt.

Ne gondoljuk azt, hogy a CAS-szerver allandóan felolvassa a postafiókok teljes tartalmát: a mailbox-szerver lelkesen aládolgozik. A szükséges információkat rendszeresen ki-pakolja egy hálózati megosztásba, XML formátumban. A CAS innen olvassa fel.

Egy másik hasznos szolgáltatása a CAS-szervernek az Autodiscover Service. Csak Outlook 2007-kliensek, illetve erre felk-

szített mobil eszközök képesek használni. Tulajdonképpen arról van szó, hogy e-mail cím és jelszó megadása után a CAS leküldi a kliensre a komplett MAPI-profilt, biztosítva ezzel, hogy kliensoldalon ne kelljen a felhasználóknak semmilyen konfigurációs varázslatokkal bábázkodniuk.

Van még egy érdekes apróság, amit úgy hívnak, hogy LinkAccess: ezzel tudjuk megoldani, hogy OWA-ból közvetlenül érthessük el az intranetünkön található, Exchange felé publikált fájlokat, illetve Sharepoint-site-okat.

Unified Messaging szerepkör

A másik lehetséges alternatív postafiók-elérési mód a szövegelés.

Képzeld el a következő telefonbeszélgetést:

– Start to read my emails.

...

– Stop. I want to hear this thread details.

...

– Stop. I want to answer to it.

...

– I want to call the email options. Set on the read receipt.

– OK.

– Send it.

És ez működik! Természetesen nem csak az Inboxra – például a Calendar kezelését már a kedves olvasó fantáziájára bizzuk.

A 7. ábrán jól látható, hogy a különböző telefonos trükközések után végül is a hang IP-csomagok formájában landol a Unified Messaging funkciót ellátó Exchange-szerveren. Ez a szerepkör látja el gyakorlatilag az Exchange-Voice illesztést.

Alapfunkciók:

Call answering. Üzenetrögzítő (testreszabható szöveg, rögzíti a beérkező üzenetet, hangfájl-levélbe csomagolva landol az Inboxban).

Fax receiving. Értelemszerűt (fax az inboxba).

Subscriber access. Outlook Voice Access (OVA), azaz Outlook-kezelés telefonon keresztül (hang- vagy nyomógomb-interfészen). Az IVR menü itt is testreszabható.

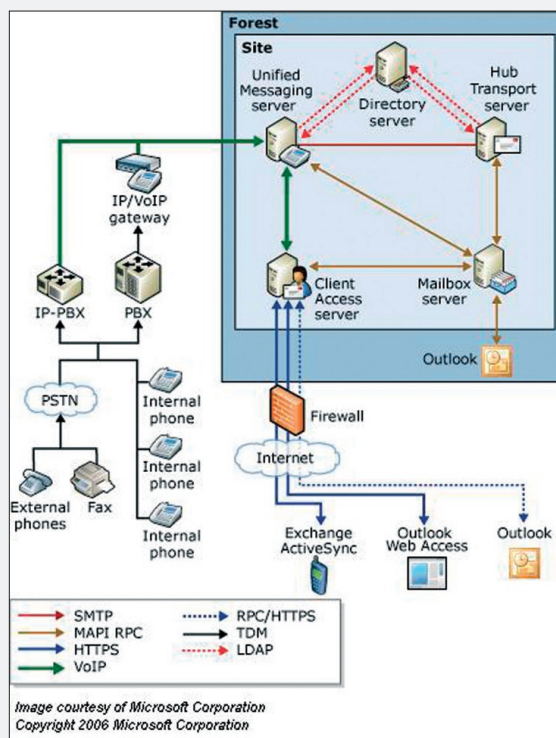
Auto attendant. Talán ez a legérdekesebb. A betelefonálót egy menü keresztül betelerjük az Exchange-organizációba, keresési lehetőséget biztosítunk neki a címtárban, majd amikor megtalálta a személyt, akit keresett, akkor az Exchange felhívja nekünk a személy adatlapján található telefonszámot.

Fontos észrevenni, hogy itt telefon-Inbox kapcsolat jön létre: azaz minden az Inboxba kerül, a voice mailek, a faxok kényelmesen elférnek a többi, hagyományos levél mellett.

Mit mondhatnánk még itt a végén? Mint látható, van újdonság bőven. Aki úgy tippel, hogy közülük egynéhányat bővebben is kifejtünk a közeljövőben, nem jár messze az igazságtól.

Petrényi József

(petrenyi.jozsef@sao.hu) SAO-Synergion, MCSE+M, MVP



7. ábra. Unified Messaging és Client Access Server architektúra

– Hi! I'll be your Exchange server today. Please type your email address and password.

– *****

– Welcome Mr. Bigfoot.

– Fine. Go to my Inbox.

– OK.

IT-SECURITY TODAY

INFORMATIKAI BIZTONSÁGI HAVILAP NAPI ONLINE TÁJÉKOZTATÓJA

- informatikai döntéshozóknak, technológiai szakembereknek
- az elmúlt 24 óra legfontosabb hazai és külföldi informatikai biztonság és információbiztonság hírei
- ingyenes napi online hírlevél

Regisztráljon!

www.it-business.hu/hirlevel



A MEGÚJULT EXCHANGE MANAGEMENT CONSOLE

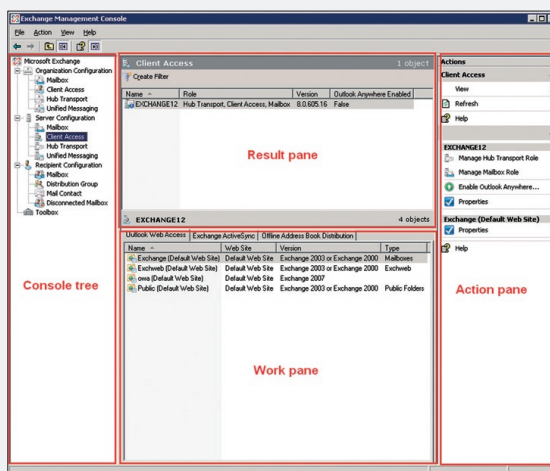
Az új Exchange-et fejlesztő csapat nagy hangsúlyt helyezett az új Management Console kidolgozására is – az eredmény önmagáért beszél.

Az Exchange Server 2003-ban található Exchange System Manager nem hozott jelentős változást a megelőző 2000-éhez képest, viszont a 2007-es Exchange-ben már egy tényleg felturbózott, vadonatúj koncepció és design köszön ránk. A „rég” felület elméletben áttekinthető struktúrára alapult ugyan, mégis sokszor körülményes és időigényes volt megtalálni egy megfelelően nagy környezetben a kívánt beállítást – jöhet csak egy kattintás kellene éppen, de az aztán jó mélyen el van rejtve a fastruktúrában.

Az új konzol tervezésénél az egyik fő szempont ennek az egyszerűsítése volt, valamint a gyakran használt funkciók minél gyorsabb elérése. Ebben a cikkben áttekintjük az új felület lehetőségeit, és rögtön nézzük is pár példát az Exchange Server 2007 képességeiből.

A megújult felület

Mint minden mostanában megjelent és megjelenő Microsoft-termék, amelyik MMC-konzolt használ, az Exchange Server 2007 is az új MMC 3.0-t ve-



1. ábra. A megújult felület

szí igénybe, ami a lelke mélyén Windows Powershell-cripteket futtat. Ez gyakorlatilag azt jelenti, hogy minden grafikus felületen végrehajtott módosításnak van egy parancs-sori, commandlet-megfelelője is.

Az egyszerűbb navigáció érdekében az organizáció nem szerverekre van lebontva, hanem szerepkörökre, erőforrásokra. A megadott szerepköröket/erőforrásokat kiválasztva látjuk az ehhez tartozó szerverek listáját és a rajtuk végezhető műveleteket.

A konzolt négy fő területre bontották. Bal oldalon található a Console tree, amely a telepített szerepkörök szerint csoportosított szervereket/erőforrásokat tartalmazza. Itt három egységet különböztethetünk meg. Az első az Organization Configuration, ahol a globális, egész organizációt érintő beállításokat módosíthatjuk. A második a Server Configuration, ez a szerverszintű adatokat tartalmazza szerepkörök bontásában. A harmadik egység a Recipient Configuration, itt a címzettek, terjesztési listák és külső kapcsolatok tulajdonságait menedzselhetjük.

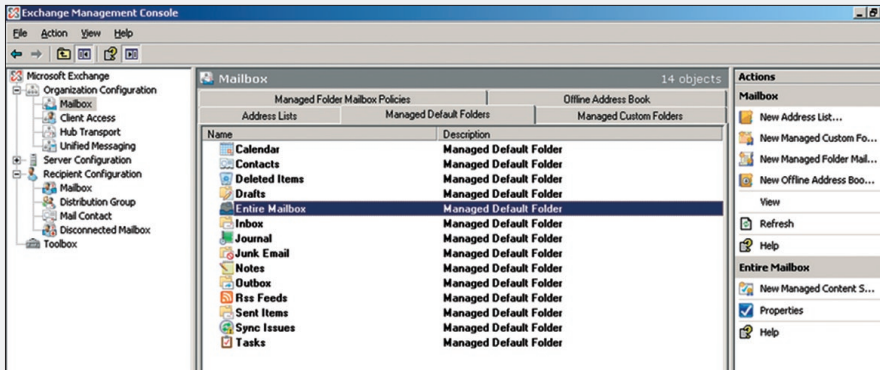
Középen felül a Result pane látható, itt a Console tree-ben kiválasztott erőforráshoz/szerepkörhöz, beállításához kapcsolódó elemeket módosíthatjuk. Lejjebb a Work pane helyezkedik el. Ez a terület csak akkor jelenik meg, ha a Console tree-ben a Server Configuration egy elemével dolgozunk. Jobb oldalon láthatjuk az Action pane-t, mely az előző 3 panelben kiválasztott objektumok függvényében különböző információkat, valamint az egyes funkciók gyors elérését teszi lehetővé. Most nézzük részletesen, hogy a Console tree elemei mire is használhatók.

Organization Configuration

Itt érhető el és módosíthatók az egész organizációt érintő információk, valamint itt lehet elvégezni a jogosultság-delegációt is. Az Organization Configuration menüpont alatt 4 további egységet találunk:

Mailbox. Itt hozhatunk létre és innen módosíthatjuk a címlistákat, offline címtárakat. Ezek már korábbról is ismert funkciók, de rögtön szembeötlik néhány érdekes újdonság. A Managed Default Folders fül alatt a felhasználók mailboxainak alapkönyvtárait találhatjuk, amit tetszés szerint bővíthetünk, módosíthatunk. Ennél talán sokkal fontosabb, hogy alapszabályokat rendelhetünk a folderekhez vagy akár az egész mailboxhoz.

Ezek után a felhasználó átrakhat egy beérkező levelet (vagy feladatot, naptárelemet) egy ilyen felügyelt mappába, amit a Managed Custom Folders alatt hozunk létre, és ekkor ezekre az elemekre érvényes lesz az általunk beállított szabály.



2. ábra. A Managed Default Folders elemei

Mire is jó ez? Például arra, ha szeretnénk, hogy szerződések vagy a munkamegbeszélések anyagai egy rendezett, felügyelhető helyen legyenek a felhasználók postafiókaiban, ezeket innen akár SharePointba vagy SQL-be is továbbíthatjuk. Ehhez kapcsolódik a Managed Folder Mailbox Policies is, ahol szabályokat definiálhatunk, majd azokat különböző felhasználók postafiókjára érvényesíthetjük.

Client Access. Ebben az egységben a globális ActiveSync-opciókat módosíthatjuk szabályokon keresztül.

Hub Transport. Ebben a szekcióban állíthatjuk be az összes belső kommunikációt, levéltovábbítást és kézbesítést érintő opciókat. A Remote Domain fül alatt egy régen várt opció érhető el: itt lehet akár teljes tartományként is szabályozni az üzenetek formátumát vagy az Out of Office üzenetek küldésének lehetőségét.

Az Accepted Domainben lehet megadni az összes általunk kezelt tartományt, legyen az akár az organizáción belül vagy kívül.

A Transport rule szekcióban olyan szabályokat hozhatunk létre, amelyek az üzenetek kézbesítését befolyásolják. Gyakorlatilag ez is egy nagyon régen áhitott opció, például ahhoz, hogy másolatot készítsünk az összes organizációt elhagyó e-mailről, vagy hogy a főnöknek címzett levelekből ki lehessen szűrni a fizetésemelést érintő leveleket, kivéve, ha valaki tudja a varázsszót.

A Journalingot választva különböző naplózási szabályokat hozhatunk létre, ezek segít-

ségével megfelellhetünk a különféle, törvényben lefektetett naplózási előírásoknak, valamint könnyen szétválaszthatjuk a külső és belső levélforgalom naplózását is. Az E-mail address policies-ban érhető el a régebbiről megszokott Recipient policies-t. Itt definiál-

hatjuk, milyen e-mail-címeket akarunk automatikusan generálni a postafiókjainkhoz. A varázslón keresztüli beállítás első látásra talán kicsit furcsa lehet, de nem nagyon különbözik az előző verziókban megszokottaktól.

A Send Connectors az előző verziókban megismert SMTP connectors funkcionalitásával egyezik meg. A különböző address space-ek függvényében különböző szerverekhez továbbíthatjuk akár DNS vagy smarthost segítségével az üzeneteinket. Az Edge Subscription segítségével pedig az Edge Transport szerepkörű szerveren telepített ADAM és a saját Active Directorynk közötti konfigurációs adatok szinkronizációját tudjuk biztosítani.

Az Exchange Server 2007-ben bemutatkozó újdonság a Unified Messaging (UM), ami hang/fax/e-mail alapú üzenetek közös store-ban tárolását teszi lehetővé. Az ugyanilyen nevű pont alatt állíthatjuk be az egész organizációt érintő szabályokat, tulajdonságokat. Gyakorlatilag itt egy leképezést végzünk a telefonrendszerünkről az Active Directoryba.

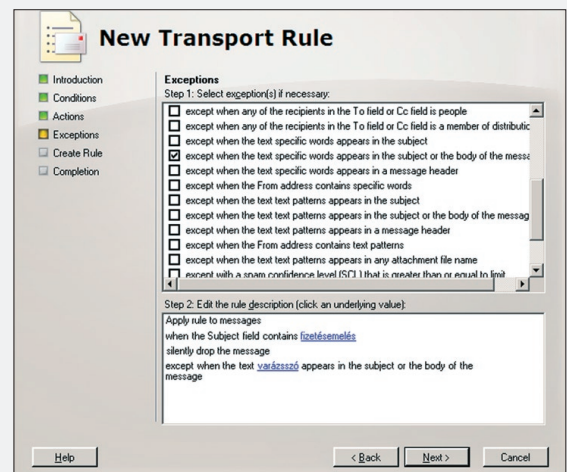
Server Configuration

A Server Configuration rész alatt találjuk organizációnk szervereit szerepkör szerinti bontásban:

Mailbox. Itt érhetjük el az adatbázisokat érintő beállításokat. A korábbi verziókhoz hasonlóan itt is Storage groupokkal szeparálhatjuk az adatbázisokat, és ugyanúgy tudjuk mountolni, dismountolni őket, mozgatni az adatállományokat. Érdemes megjegyezni, hogy most már a rendszer által létrehozott mailbox storage-en is van méretkorlátozás!

Client Access. Ebben a szekcióban lehet elérni a Microsoft Outlook Web Access, Exchange ActiveSync, offline címtár (OAB) beállításait. Valójában magát az IIS-t lehet itt menedzselni, megfűszerezve az Exchange saját opcióival. Az Outlook Web Access fül alatt ellenőrizhetjük az OWA alapkönyvtárainak tulajdonságait, többek között itt lehet bekapcsolni a Form Based Authenticationt is.

Hub Transport. Míg az előző részben a HTTP protokollt állíthatjuk be, itt az SMTP protokoll tulajdonságain módosíthatunk. Alapbeállításaként két connectort találunk, ezek: a Client (szervernév), valamint a Default (szervernév). Nyilván a Default connector a 25-ös porton fogadja a beérkező üzeneteket a nagyvilágból (fontos: ez nem azt jelenti, hogy az internetről – csakis szigorúan megbízható helyekről), míg a Client az 587-es porton várja a kliensüzeneteket.



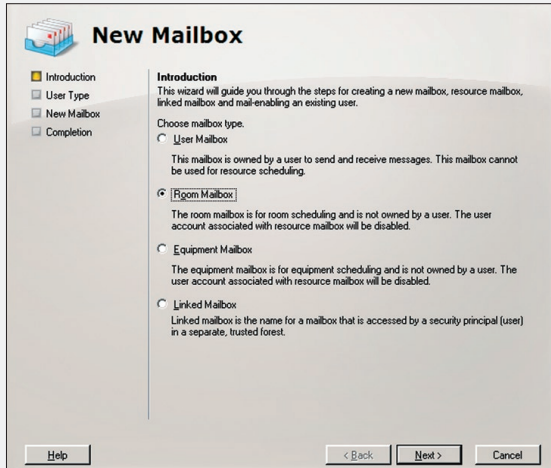
3. ábra. Szabályok végtelen variációja Transport Rule definiálásakor

Természetesen lehet állítani a használandó IP-kezt, portot, hogy milyen IP-kről érhető el a connector, és állíthatjuk az autentikáció típusát is (Basic, Intergrated, TLS stb.).

Edge Transport. Ezzel a ponttal csak akkor találkozunk, ha az organizációnkon belül van Edge Transport szerepkörű szerver. Ezek a szerverek felelősek az internet irányába küldött és onnan érkezett levelek tisztaságáért.

A Hub Transporthoz hasonlóan itt is egy SMTP-szerverről beszélünk, valamint az ehhez kapcsolódó Anti-Spam és Antivirus szolgáltatásokról. Itt is megtalálható a Receive

ki kell választani, hogy meglévő felhasználóhoz akarunk-e postafiókot rendelni, vagy új felhasználót hozunk létre. Ezek után még néhány általános tulajdonság következik, és



4. ábra. Mailbox-típusok

Connector, a Send Connector, a Transport Rules és az Accepted Domains. Ami ismeretlen lehet ebben a környezetben, az az Anti-Spam fül. Többek között az alábbi technikákat és technológiákat aktiválhatjuk itt (amik mind transport agentek amúgy, és akár Hub Transport szerepkörű szerverre is feltehetőek): Content filtering, IP Allow list, IP Allow List providers, IP Block list, IP Block List providers, Recipient filtering, Sender filtering, Sender ID, Sender reputation.

Unified Messaging. Ezen keresztül konfigurálhatjuk a hangüzenet, fax és e-mail-üzeneteket egy store-ba, amelyet a felhasználók elérhetnek számítógépen, de akár telefonon keresztül is.

Recipient Configuration

A korábbi verziókban a címzettek beállításait (postafiókok, címlisták, külső kapcsolatok) az Active Directory Users and Computers MMC segítségével lehetett elvégezni. A 2007-ben ezt is bevonták az Exchange Management Console-ba.

Mailbox. Valószínűleg az egyik legtöbbször használt rész lesz, ugyanis itt hozhatunk létre postafiókokat, és itt is távolíthatjuk el őket a rendszerből. Létrehozásnál meg kell határozni, milyen típusú postafiókot akarunk létrehozni: felhasználói postafiókot, erőforrás- (szoba- vagy eszköz-) postafiókot, illetve organizáción kívüli trusted domainben levő felhasználó postafiókját. A második lépésben

már kész is a postafiók. A postafiók tulajdonságait megnézve láthatjuk, hogy azok nagyban hasonlítanak az előző verziók ADUC füleire és opcióira.

Distribution Group. Itt menedzselhetjük terjesztési listáinkat és a hozzájuk kapcsolódó beállításokat. Új lehetőség a dinamikus terjesztési lista létrehozása, amelyben a csoporttagságot különböző felhasználói tulajdonságok alapján adhatjuk meg, nem kell egyesével belerakni csoportokba a felhasználókat.

Mail Contact. Felületet biztosít az organizáción kívüli személyek levelezési opcióinak beállításához.

Disconnected Mailbox. Itt láthatjuk az olyan mailboxok listáját a store-ban, amelyekhez nem tartozik felhasználó/erőforrás az Active Directoryban.

Toolbox

Szerszámládánkban (és végre valóban szerszámláda-ikonja van) jóval több hasznos eszközt találhatunk az elődökhöz képest.

Négy eszközcsoportot különböztethetünk meg: első a Configuration Management Tools, amelyben jelenleg egyetlen eszköz árválkodik, viszont ez az egyik leghasznosabb. Nevezetesen ez az Exchange Server Best Practices Analyzer.

A második csoport a Disaster Recovery Tools két hasznos eszközzel, nevezetesen a Database Recovery Management toolal és Database Troubleshooterrel. Mindkettő azonos külső programot használ, mégpedig a szintén az előbbi linken található Exchange Troubleshooting Assistantot. Mindkét eszköz hasznos lehet adatbázishibák felderítéséhez és kijavításához, storage groupok és postafiókok visszaállításához.

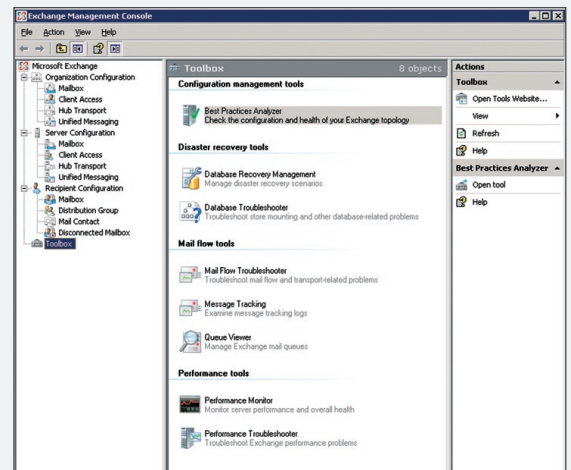
A harmadik csoportban (Mail Flow Tools) a levelek küldésével, fogadásával és feldol-

gozásával kapcsolatos eszközök állnak rendelkezésünkre. Három elemet találunk itt: Mail Flow Troubleshooter, Message Tracking tool, Queue Viewer. Érdekes újdonság, hogy mind a Message Tracking Tool, mind a Mail Flow Troubleshooter az előzőekben említett Exchange Troubleshooting Assistant eszközt használja. A Mail Flow Troubleshooter segítségével különböző tüneteket megadva próbálja meg a rendszer helyettünk megtalálni a probléma forrását.

Az utolsó csoportba kerültek a teljesítmény-méréssel és diagnosztikával foglalkozó eszközök. A Performance Monitor segítségével Exchange teljesítménymérő értékeket ellenőrizhetünk (RPC-kérések száma, helyi kézbesítési arány). A Performance Troubleshooter szintén az Exchange Troubleshooter hívja segítségül a teljesítményt érintő problémák megoldására.

Összegzés

Az új Exchange Management Console sok újdonságot hoz, mint ahogy maga az Exchange Server 2007 is. Ebbe a végeláthatatlan listába tartozik például a szerver föltelepítését követően az első indításkor látható Exchange Server Finalize Deployment varázsló is, amely segítséget nyújt abban, hogy milyen lépéseket is kell megtennünk egy jól működő és biztonságos levelezőszerver üzemelteté-



5. ábra. Szerszámláda

se érdekében. Ide tartozik szintén a rengeteg új varázsló, amelyek részletesen, ugyanakkor áttekinthetően segítenek bennünket feladataink elvégzésében.

Subicz Péter
Exchange Server MVP (splistas@stn.hu)

KITEKINTÉS AZ OUTLOOK 2007-RE

Az e számunkat uraló Exchange Server 2007 mellett a Microsoft Office-ban található Outlook-klieus 2007-es változata is szolgál kellemes meglepetésekkel.

Lássuk, mik a legérdekesebb újdonságok!

Automatikus konfiguráció. Az Exchange Autodiscover Web Service segítségével a már eleve domainbe léptetett Outlook 2007-klieus bármiféle felhasználói interakció nélkül is képesek kapcsolódni a rendszergazda által kijelölt Exchange Serverhez, illetve a már létrehozott mailboxhoz. Egyszerűen, már az Outlook 2007 első indításakor, egy gomb megnyomása után levelezhetnek felhasználóink, anélkül, hogy szervertíneket, e-mail-címeteket, jelzavakat, portokat kellene beírniuk.

Naptárkezelés. A naptárkezelés üzleti logikája teljesen átkerült az Exchange Server hatáskörébe, ezáltal végre nem fordulhatnak elő kavargások a megbeszélések státuszaival kapcsolatban.

Feladatok kezelése. A feladatok nagyobb szerepet kaptak, mint korábban. Nemcsak külön listaként láthatjuk, hanem naptárunkhoz is rendelhetjük őket: így láthatjuk, mikor kellene elvégeznünk az adott feladatot, valamint hogy végül mikor sikerült. A feladatokhoz hasonló módon kezelhetünk bármilyen más Outlook-elemet is, így akár leveleket, RSS-bejegyzéseket is megjelölhetünk feladatként, és ezeket akár a naptárunkba is ránthatjuk: megjelölendő, hogy mikor is szeretnénk foglalkozni vele.

RSS-támogatás. Az Outlook 2007 ugyanúgy kezeli az RSS-csatornákat, mint a hagyományos levelezést. A klieust használva archiválhatjuk, kereshetjük, rendezhetjük és kényelmesen olvashatjuk a számunkra érdekes bejegyzéseket, vagyis pontosan ugyanúgy használhatjuk az RSS-bejegyzéseket, mint a hagyományos leveleket.

Hangüzenetek kezelése. Nemcsak elektronikus dokumentumokat és írásokat, hanem hangüzeneteket is tárol az Outlook 2007 – már amennyiben használjuk az Exchange Unified Messaging szerepkörét. A nekünk hagyott hangüzenetünket meghallgathatjuk közvetlenül az Outlookból, ahol pontosan látszik az üzenet időpontja, feladója is.

Új Outlook Web Access. Ismét sokkal több funkció érhető el az Outlook webes felületéről, így most már egy szinte minden tekintetben teljes értékű klieust használhatunk hagyományos internetkapcsolat megléte esetén is, távol a belső hálózattól.

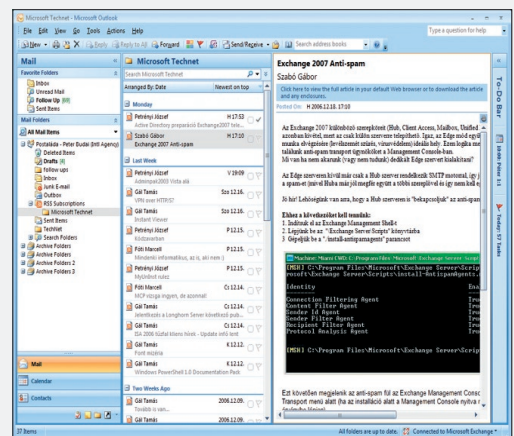
SharePoint-integráció. Mind az Outlook 2007, mind az új OWA képes az intraneten található SharePoint-webhelyek, listák, állományok szinkronizálására és publikálására, így közvetlenül az Outlookból érhetjük el a számunkra legfontosabb adatokat.

Keresés. Gyors és egységes keresőmotort kapunk, ami végre ugyanazt az algoritmust használja mind kapcsolódott, mind kapcsolat nélküli üzemmódban, akár van lokális cache-elésünk, akár nincs. Ha viszont éppen van aktív kapcsolatunk az Exchange Serverhez, akkor a keresést a szervertől is tudja végezni a klieus helyett.

Managed folders. Ezek olyan, rendszergazda által létrehozott könyvtárak, amelyek minden megadott felhasználónál megjelennek, és ha beletesz a felhasználó egy Outlook-elemet (levelet, feladatot – bármit), azon lefutnak azok a szabályok, amiket beállítunk. Ilyen módon nagyon kényelmesen lehet az egész cégre kiterjedő dokumentumkezelési szabályokat megvalósítani (például szerződések és üzleti levelezések egységes kezelése, archiválása, előírásoknak megfelelő automatikus kezelése).

Spam. A levélszemét kezelése is sokat változott mind a klieus, mind a szervertől oldalon, azonban ezzel részletesen foglalkozunk a Magazin egy másik cikkében.

Összességében elmondható, hogy az Outlook-klieus továbbra is az egyik legkényelmesebb felület a mindennapos irodai mun-



Beépített RSS-kezelés az új Outlookban

kához, és folyamatosan egyre több képesség épül bele a hatékonyabb csoportmunka támogatására is.

Budai Péter
(i-pbudai@microsoft.com) Microsoft Magyarország

EXCHANGE MANAGEMENT SHELL

Az Exchange Server 2007 üzemeltetéssel kapcsolatos újdonságai közül talán a PowerShell-parancsgyűjtemény a legnagyobb áttörés az Exchange 2003-hoz képest.

Az Exchange Server üzemeltethetőségének tervezésekor a cél a felhasználói felülettel rendelkező üzemeltetési eszközökkel egyenértékű támogatott és dokumentált script-parancsok fejlesztése volt. Ez össze is jött, és nem is akárhogyan!

Az Exchange Management Shell néhány parancsának gyakorlati bemutatásával az a célunk, hogy az olvasónak – ha még korábban nem jött volna meg – meg hozzuk a kedvét a használatához. A következő oldalakon a kételkedők is megbizonyosodhatnak arról, hogy milyen egyszerűen, milyen hatékony és összetett dolgokat lehet művelni a PowerShell használatával.

Már az Exchange Server 2007 architektúrális áttekintésekor is számtalanszor hivatkoztunk arra, hogy egyes feladatok elvégzése az Exchange Management Console segítségével nem, vagy csak limitáltan lehetséges. Emiatt a Windows PowerShell alapú, Exchange Management Shellben kiadott scriptek, utasítások nemcsak a meglévő képességek automatizálására kiválóak, hanem arra is, hogy a felületre ki nem vezetett funkciókat azok teljességében használatba vehessük.

Az Exchange Server 2007-tel közvetlen kapcsolatban álló parancsok listáját az Exchange Management Shell elindítása után a `get-excommand` parancs kiadásával érhetjük el. A parancsokról a Windows PowerShellben megszokott eszközökkel kaphatunk részletesebb segítséget, példákat is.

Nyilvános mappák kezelése, jogosultságállítás

A kínálat bemutatását a nyilvános mappákkal kezdjük. A telepítés után a Mailbox-szerverünkre nem települt Public Folder adatbázis (mivel nem pipáltuk be a telepítőben, hogy van Outlook 2007-nél régebbi kliensünk is – lásd a telepítéssel foglalkozó cikket), így a Windows PowerShell erejét kihasználva hoztunk létre egyet:

```
New-PublicFolderDatabase -Name "Public Folders" -StorageGroup "First Storage Group"
```

Hosszú, izgatott várakozással töltött másodpercek, és már el is készült! A következő parancsral le is kérdezhetjük a tartalmát:

```
Get-PublicFolder
```

A `Get-PublicFolder -Recurse` bejárja a teljes mappastruktúrát, míg a `-Recurse` paraméter nélkül csak a legfelsőbb szintű mappákat kapjuk vissza.

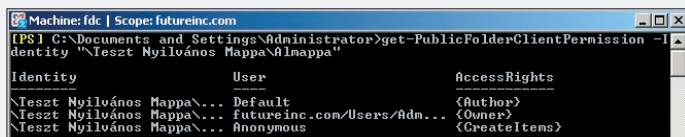
Kellene még pár mappa, amivel tesztelhetünk!

```
New-PublicFolder -Name "Teszt Nyilvános Mappa"
-Path "\"
```

Csináljunk most már valami izgalmasabbat is: módosítsuk az új mappánk kliensszintű (MAPI) jogosultságait. Az aktuális jogosultságok:

```
Get-PublicFolderClientPermission -Identity "\Teszt
Nyilvános Mappa\Almappa"
```

Szeretnénk egy új felhasználót hozzáadni az ACL (Access Control List)-hez, viszont



```
Machine: fdc | Scope: futureinc.com
[PS] C:\Documents and Settings\Administrator>get-PublicFolderClientPermission -I
dentity "\Teszt Nyilvános Mappa\Almappa"
Identity          User              AccessRights
-----
\Teszt Nyilvános Mappa\... Default          <Author>
\Teszt Nyilvános Mappa\... futureinc.com/Users/Adm... <Owner>
\Teszt Nyilvános Mappa\... Anonymous          <CreateItems>
```

a környezet annyira új, hogy még csak az Exchange Admin felhasználó létezik. Újra kihasználjuk a PowerShell erejét, és csinálunk egy új Exchange-mailboxot:

```
New-MailUser -Name "Teszt user" -ExternalEmailAddress
tesztuser@futureinc.com -UserPrincipalName teszt-
user@futureinc.com -OrganizationalUnit futureinc.com
```

Ezután szabad a pálya az ACL módosításához.

Ha ezeknél komplexebb műveleteket szeretnénk végezni, a C:\Program Files\Microsoft\Exchange Server\Scripts mappában egy szép

```
Machine: fdc | Scope: futureinc.com
[PS] C:\Documents and Settings\Administrator>add-PublicFolderClientPermission -Identity "Teszt Nyilvános Mappa\Almappa" -user "Teszt user" -accessRights CreateItems
Identity      User              AccessRights
-----
Teszt Nyilvános Mappa\... futureinc.com/Teszt user      <CreateItems>

[PS] C:\Documents and Settings\Administrator>get-PublicFolderClientPermission -Identity "Teszt Nyilvános Mappa\Almappa"
Identity      User              AccessRights
-----
Teszt Nyilvános Mappa\... Default              <Author>
Teszt Nyilvános Mappa\... futureinc.com/Teszt user      <CreateItems>
Teszt Nyilvános Mappa\... futureinc.com/Users/Adn... <Owner>
Teszt Nyilvános Mappa\... Anonymous            <CreateItems>
```

script-gyűjtemény áll rendelkezésünkre, többek között az AddUsersToPFRecursive.ps1 script, amelynek használatával egy egész nyilvános mappa fa-csomópontjainak ACL-jeit gazdagíthatjuk a felhasználókkal. Lépjünk be a mappaiba, és futtassuk a (PowerShell) scriptet:

```
AddUsersToPFRecursive.ps1 -TopPublicFolder ""\Teszt Nyilvános Mappa"" -User "Teszt user" -Permission CreateItems
```

A -TopPublicFolder paraméter után a mappa nevét három idézőjel közé raktuk, hogy a script megkapjon egy idézőjelet (dupla idézőjel idézőjelben). Külön érdekesség, hogy ha nem adunk meg "dupla" idézőjelet, akkor a script nem működik olyan "root" mappákkal, amelyeknek szökőz van a nevében.

A fenti commandleteken kívül még számos parancs rendelkezésünkre áll nyilvános mappák kezeléséhez: például replikáció, a "mail-enable" állítása. Végezetül nézzünk egy példát az utóbbira:

```
Enable-MailPublicFolder -Identity "\Teszt Nyilvános Mappa\Almappa"
```

Az összes mail-enabled nyilvános mappát (a mi esetünkben csak egy van) pedig a következő paranccsal tudjuk lekérdezni:

```
Get-MailPublicFolder | Format-List
```

A kimenetet átirányítottuk a Format-List parancsnak, amely megformázza a kimenetet. Ha az eredményhalmaz 1000 elemnél többet tartalmaz, akkor csak az első 1000 elemet adja vissza.

E-mail-felhasználók és disztribúciós listák kezelése

A következőkben az e-mail-felhasználók és disztribúciós listák körüli műveletekkel fog-

lalkozunk. Csapjunk rögtön a közepébe! Az alábbi példák azt mutatják, hogy egy kicsit összetettebb, de még mindig egyetlen sorból álló scripttel milyen egyszerű dolgokat végezhetünk el. A feladat egyszerű: importáljunk be egy .CSV fájlban tárolt felhasználókat az Active Directoryba, és csináljunk nekik mailboxokat. A .CSV fájl első sorában definiáltuk az oszlopok neveit (Alias,Name), utána pedig a felhasználók aliasait, hosszú neves formában felsorolva.

A script első felében beolvassuk a bemenetről (a billentyűzetről) a feladat elvégzésé-

```
Machine: fdc | Scope: futureinc.com
[PS] C:\>$password = Read-Host "Password" -AsSecureString; import-csv users.csv | Foreach {New-Mailbox -alias "$($_.Alias)" -Name "$($_.Name)" -Org futureinc.com/alias -UsePrincipalName "$($_.Alias)" -Database "Mailbox Database" -Password $password}
Password: *****

Name      Alias              ServerName  ProhibitSendQuo
-----
Korolovszki Jeno      kjeno        fdc         unlimited
Borzenyak Tamas      btamas      fdc         unlimited
Trap Istvan          trapi        fdc         unlimited
```

hez szükséges jelszót, és nem adjuk azt meg szöveggént a script részeként.

Ezután csinálunk egy disztribúciós listát, és tegyük bele a felhasználóinkat. Disztribúciós listát a felhasználókhöz hasonlóan, egyszerűen a következő paranccsal készíthetünk:

```
New-DistributionGroup -Alias ujUserok -Name "Új userok" -Type Distribution -Org futureinc.com/Users -SamAccountName usUserok
```

Csoporttagságokat pedig az előbb már megismert import-csv commandlet használatával, variálva az add-DistributionGroup-Member paranccsal vihetünk fel:

```
Import-csv users.csv | foreach {Add-DistributionGroup-Member -Identity ujUserok -Member "$($_.Alias)"}
```

Mailbox-jogok beállítása

A következő parancsok a mailbox jogosultságainak beállítására szolgálnak. Az első a set-Mailbox, amellyel a mailbox különböző tulajdonságai módosíthatók, például különböző user-kvóták, a maximálisan fogadható e-mail-méret, a felhasználó teljes neve stb. Ugyanezzel a paranccsal ki-
osztható a „send on behalf” jogosultság is:

```
set-Mailbox administrator -GrantSendOnBehalfTo kjeno
```

Egy másik, a mailbox-jogosultság beállítására szolgáló parancs az add-AdPermission, amely az Active Directory-beli user-objektum ACL-jét módosítja. A következő példában ezzel a paranccsal „Send-as” jogot adunk az egyik teszt-userünknek:

```
Add-AdPermission administrator -ExtendedRights Send-As -User kjeno
```

A „send on behalf” és „send as” jogosultságok két különböző „megszemélyesítést” (vagy helyettesítést) jelentenek: míg az előbbinél a címzett látja a levél valódi küldőjének e-mail-címét, addig az utóbbinál úgy tűnik számára, mintha valóban a helyettesített személy írta volna a levelet.

Végül a mailbox ACL-jét kibővíthetjük egy full mailbox access jogosultsággal:

```
Add-MailboxPermission administrator -AccessRights FullAccess -User trapi
```

Összefoglalás

Fontos még egyszer megjegyezni, hogy az Exchange körüli valamennyi karbantartási és üzemeltetési feladat megoldható PowerShell-parancsokkal, és olyan dolgokat is elvégezhetünk vele, amelyeket más (grafikus) eszközökkel nem tudnánk.

Korábban a hasonlóan komplex feladatok elvégzésére egyedül a fejlesztők voltak képesek a számukra kialakított API-k és osztálykönyvtárak segítségével – azonban most már ezt bármelyik rendszergazda megteheti néhány egyszerű script írásával.

Remélhetőleg sikerült felkelteni az olvasó érdeklődését a PowerShell alkalmazása iránt – akinek viszont még nem jött volna meg a kedve, annak akkor fog, amikor először auto-

matizálni szeretné valamely Exchange körüli feladatát.

Szabó Dávid
(dszabo@microsoft.com) Microsoft Magyarország

TERVEZÉS ÉS TELEPÍTÉS

Alapos tervezésnek kell megelőznie a tényleges telepítést – ezzel később rengeteg időt és energiát spórolunk meg magunknak.

Itt van a kezünkben az Exchange Server 2007 telepítő-cédéje, nosza dugjuk be a gépbe, és telepítsünk! Amennyiben így jártunk volna el, garmadával követtük volna el a hibákat. Vagyis tegyük szépen vissza a cédét a fiókba, emeljük le a polcra az internetet... vagy ezt az újságot (legjobb, ha mindkettőt), és gondoljuk végig, mit is szeretnénk, hogyan is szeretnénk.

Szerepek

Az új Exchange-verzióban az elemi funkciókat szétválogatták, és csoportokba rendezték. Ezeket a csoportokat nevezzük szerepköröknek.

Ha visszagondolunk az elődre (Exchange Server 2003), abban tulajdonképpen sok szerepkör nem volt, csak a background és frontend szerver. A background szerveren futott egy adatbázis-motor (ESE, leánykori nevén JET), ez kezelte a levelezési adatokat: a leveleket, a naptárbejegyzéseket, kontaktokat, taszkokat, amelyek tartozhattak felhasználóhoz, illetve nyilvános mappákhoz. A felhasználók elérhették az adatokat MAPI felületen (a jó öreg RPC), illetve internetes protokollokon (POP3, IMAP4, HTTP) keresztül. A frontend szerver ettől annyiban különbözött, hogy lemondott az adatbázis kezeléséről, cserébe egyfajta előtét szerepet vállalt fel az internet felőli elérésekhez: kezelte azokat a weblapokat, amelyeken keresztül a HTTP/HTTPS-kliensek (OWA, OMA, ActiveSync) bedőngettek a szervezetbe. Mindemellert célszerű volt vagy erre a szerverre, vagy egy külön beiktatott smarthostra telepíteni valamilyen levelezési higiénia biztosító funkciót is (itt a vírus és spam elleni védelemre kell gondolni).

Lássuk, tervezési és telepítési szempontból mit kell figyelembe vennünk a szerepkörökkel kapcsolatban.

Edge Transport szerepkör. Kezdjük rögtön a különccel:

- Nem kötelező a szerepkör megléte. Tulajdonképpen nagyon jól elvan nélküle az organizáció, a feladatát – persze jócskán leegyszerűsítve – el tudja látni a Hub Transport szerver is.
- Nem kötelezően tartományi tag. Ez merőben új dolog, mert eddig először be kellett léptetni a gépet a címtárba, hogy utána felugorhasson rá egy Exchange-szerveralkalmazás, amely már tagja lehetett az organizációnak.

Ha röviden akarnánk jellemezni ezt a funkciót, akkor azt mondhatnánk, hogy ez az Exchange-szerver látja el mindazokat a funkciókat, amelyekre eddig jellemzően nem Exchange-szervereket szoktak használni. Ez a smarthost, amelyik a levelezési higiéniaért felelős.

Hub Transport szerepkör. Mindent tud az organizációról. Tudja, hogy mikor hová kell mennie egy levélnek. Nála van a Nagy Szabálykönyv, amely kegyetlenül meghatározza, mi történjen a mezei levelekkel.

Client Access Server (CAS) szerepkör. Mint a neve is mutatja, ez kezeli a kliensoldali hozzáféréseket. Ez a munka áll legközelebb a régi frontend szerepkörhöz: rajta keresztül megy az OWA, az ActiveSync – és ide került a POP3-, illetve az IMAP4-hozzáférések kezelése is, illetve ezen keresztül érhetik el az egyedi webes fejlesztések az organizációt. Azaz minden itt tobzódik, ami nem MAPI. (Ha valaki esetleg hiányolná az OMA-t, nem kell végleg elsratnia: funkcionalitása beleolvadt az ActiveSync-be.)

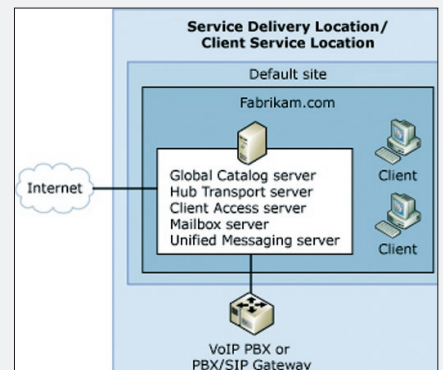
Unified Messaging szerepkör. Ez a funkció abszolút új. Célja a telefonközpontok és az Exchange-kapcsolódás megvalósítása mind hang-, mind faxvonalon.

Mailbox Server szerepkör. A legkonzervatívabb funkció a családban. Gyakorlatilag fogadja a MAPI-n keresztül érkező igényeket, kezeli az adatbázisokat – ez maga a tulajdonképpeni klasszikus levelezőszerver.

A szerepek lehetséges megosztásai

Hogyan érdemes elosztani az egyes funkciókat a rendelkezésre álló szerverek között?

Rögtön az 1. ábránál szembesülünk azzal, hogy még nem is beszéltünk minden szerepkörrel. Ott van ugyanis a Global Catalog Server funkció. Bár ez szigorúan véve nem az Exchange része, de azért tudjuk, hogy Active Directory nélkül nincs Exchange sem. Márpedig a címtár – legalábbis a címtár tartományi partícióit – a globális katalóguson keresztül használja az Exchange.



1. ábra. A legegyszerűbb szerepkörkiosztás

Nilván nem mondunk nagy újdonságot azzal, hogy Global Catalog szerver önállóan nem létezik, a funkciót csak tartományvezérlőre lehet telepíteni.

Mire is jó a Global Catalog?

Egy kis kitérőt érdemes itt tennünk, ugyanis a tapasztalatok szerint a Global Catalog

funkció megértése általában problémát szorított okozni.

Az Active Directory partíciókból áll, ezek: séma, konfiguráció- és tartományi partíció. Sémából és konfigurációból is csak egy létezik erdőszinten, míg tartományi partícióból annyi van, ahány tartomány az erdőben. Miket tartalmaz ezekből az adatokból egy tartományvezérlő? Attól függ. Mindegyik DC-n ott van az erdő teljes konfigurációs partíciója,

kát annak nyakába varrunk: ez a tartományvezérlő, a globális katalógus, az Exchange-szerver, sőt, délutánoként ez takarítja ki a szerverszobát... Értelemszerűen az összes Exchange 2007-szerepkört is ez látja el.

Mi is ez az összes? Figyelmesebb szemlélőnek például feltűnhet, hogy nem látja az ábrán az Edge Transport szerepkört. Nem is. Egyfelől ez nem kötelező, másfelől pedig nem telepíthető semelyik másik funkció mellé.

(Emlékeztünk rá, hogy ez a funkció tulajdonképpen a smarthost – így azért már sokkal érthetőbb ez a megközelítés.)

Reálisabb szerepkörkiosztás

A 3. ábra szerinti már egy fokkal szerencsésebb – és a realitáshoz jobban közelítő – felállás: habár itt is tartományvezérlőn van az Exchange-szerver, de legalább már van mellette másik, dedikált tartományvezérlő is. És igen, jól látjuk, már telepítettünk külön gépre egy Edge Transport funkciót is. Véleményem szerint ez a minimális architektúra ahhoz, hogy egy rendszergazda felelősen üzemeltethessen egy Exchange-rendszert.

(Pontosítva: a Unified Messaging rendszer természetesen nem része a minimális architektúrának, és nem is kötelező telepíteni.)

Végül vizsgáljuk meg egy kicsit nagyobb szervezet felépítését!

Itt (4. ábra) már van minden, mi szem-szájnak ingere: redundáns Edge Transport szerverek a DMZ-ben, dedikált Exchange- és DC-számítógépek, több Mailbox Server funkciót ellátó számítógép, Client Access Server és Hub Transport funkciókkal, egyik szerveren mindez kiegészítve Unified Messaging szerepkörrel is. Látható, hogy itt már több telephelyünk van, de az is látható, hogy ha tervezünk az Active Directory-site-ra Mailbox Servert, akkor kötelezően tennünk kell mellé Hub Transport szervert és erősen ajánlott a Client Access szerepkör is.

Megjegyzés: az Edge-szerverek redundánsak ugyan, de ez nem valamilyen cluster-megoldással érhető el, hanem csak DNS round robinnal.

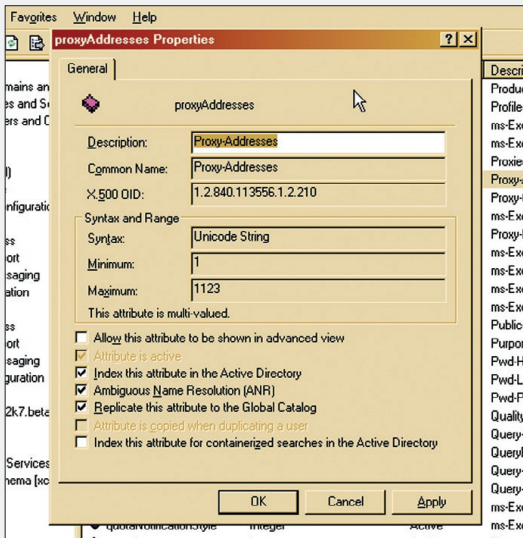
Persze nem merítettük még ki az összes lehetőséget. Létezhetnek nagyobb szervezetek, ahol érdemes lehet egy szerverre még keve-

sebb funkciót telepíteni. Sőt, amennyiben gyors gépeink vannak gyors hálózati kapcsolattal, akár azt is megtehetjük, hogy minden egyes funkciót külön szerverre telepítünk. A teljesen szétszórt rendszernek meglesz az az előnye, hogy egy-egy gép meghibásodása csak egy funkciót fog érinteni – cserébe persze a korábbi gépen belüli adatforgalmat kivisszük a gépek közötti térbe.

Gyakorlati tanácsok a szerepekhez

Az elméleti megfontolások után nézzünk néhány gyakorlati előírást:

- A Unified Messaging funkció nem létezhet önállóan, bármennyire is csak erre a funkcióra lenne szükségünk. Az organizációnak tartalmaznia kell mellette az Edge Transport szerepkör kivételével mindegyiket. (Viszont a Unified Messaging szerepkör bátran elhagyható, ha nincs rá szükségünk.)

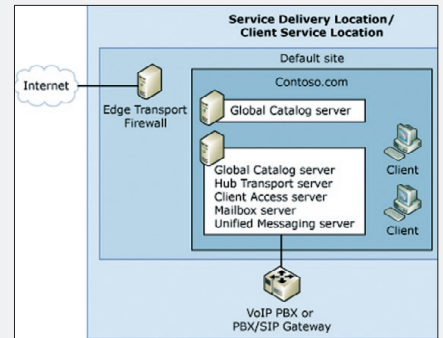


2. ábra. Mely tulajdonságok kerüljenek be a GC adatbázisába?

írható változatban. Ugyanúgy ott van a teljes sémapartíció, de ez csak a Schema Master szerepet ellátó gépen írható. És minden tartományvezérlőn megtalálható a saját tartományának domain-partíciója – felhasználónevek, gépek stb... Minden adat, amelyet az Active Directory-UC-ban szerkeszthetünk, de csak a saját tartományából.

Amennyiben bekattintjuk, hogy a tartományvezérlő legyen GC is egyben, akkor tulajdonképpen a konkrét gépen található saját tartományi partíciója mellé lekerülnek rá a többi tartomány domain-partíciós adatai is, pontosabban azoknak csak egy részhalmaza. Értelemszerűen ez a részhalmaza a legfontosabb tulajdonságokat jelenti. A tulajdonságok körét a Schema Admin konzolban lehet beállítani – de nem igazán ajánlott, tekintve, hogy minden bővítés növeli a replikációs forgalmat, a szűkítések meg előre nem látható problémákat okozhatnak.

Térjünk vissza az 1. ábrára. Nos, ez a felállítás az, amelyet röviden csak a „szegény ember vízzel főz” mondással jellemezhetünk. Egy szerverünk van összesen, minden mun-



3. ábra. Reális szerepkörkiosztás

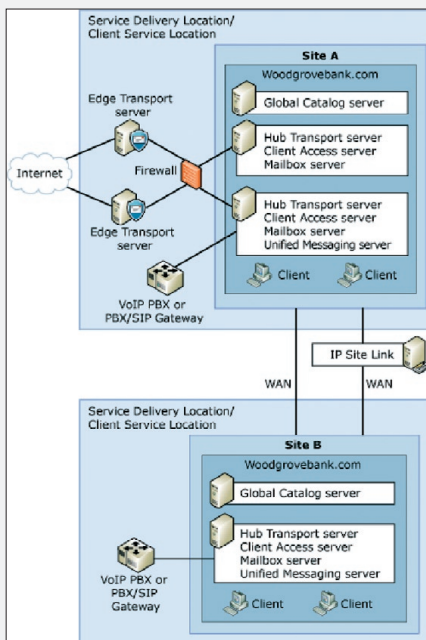
- A Unified Messaging szerepkör virtuális környezetbe nem telepíthető.
- Clusterbe kötött Exchange 2007-szervereken csak Mailbox Server szerepkör lehet – minden mást ledobnak magukról.
- Amennyiben külön szerverekre telepítjük az egyes funkciókat, a következő telepítési sorrend az ajánlott:
 1. Client Access Server szerepkör,
 2. Hub Transport Server szerepkör,
 3. Mailbox Server szerepkör,
 4. Unified Messaging Server szerepkör.

Az Edge Transport szerepkört ellátó gép független a tartománytól, gyakorlatilag bármelyik fázisban üzembe helyezhető.

Telepítés – második nekifutás

Upgrade vagy tiszta telepítés? Első ránézésre az upgrade tűnik sokkal bonyolultabbnak, de igazából nem annyira az: nem létezik

ugyanis olyan út, hogy in-place upgrade. Az egyedüli lehetséges mód az organizáció átállítására az, hogy betesszünk egy új vasat – meghozzá kizárólag 64 biteset –, majd erre felpalcoljuk az összes megkívánt funkciót, átmig-



4. ábra. Több telephelyes organizáció szerepkörkiosztása

ráljuk rá a postafiókokat, majd eltávolítjuk a régi Exchange-szervereket. Értelem szerűen úgy, hogy helyükre már az új, 64 bites vasakat tesszük be.

Létezik még egy mód, ez pedig a migráció: létrehozunk egy szűz címtárat, benne egy szűz Exchange 2007-organizációval, majd átmigrálunk minden felhasználót, minden erőforrást, illetve minden alkalmazást ebbe az erdőbe. (Esetleg bonyolultabb lelkű egyének megtehetik azt is, hogy csak a felhasználókat és a postafiókokat migrálják az új erdőbe, majd a két erdő között alakítanak ki interorg kapcsolatot. Hajrá!)

Számomra az első – átszisszanásos – módszer a szimpatikusabb, de nem mindig szubjektív szempontok alapján kell döntenünk: vannak esetek, amikor csak migrációval jutunk egyről a kettőre. Két ilyen eset van:

- Exchange 5.5 rendszerről szeretnénk Exchange 2007 rendszerig eljutni: ekkor először létre kell hoznunk egy Exchange 2000/2003-organizációt, ebbe átmigrálni az Exchange 5.5-adatokat, majd az első módszerrel továbblépni az Exchange 2007-organizáció felé.

- Nem Exchange-rendszerről (például Notes, Groupwise) szeretnénk átállni Exchange 2007-organizációra.

A meglévő organizáció átalakításához képest a szűz Exchange-organizáció kialakítása már nem annyira nagy kaland. De azért unatkozni sem fogunk....

(Gyors megjegyzés: azzal azért legyünk tisztában, hogy ha új Exchange 2007-organizációt hozunk létre, akkor abba utólag már nem léptethetünk be Exchange 2003-szervereket.)

Előfeltételek. Mit szeretne a címtár és a hálózat? Alapvetően Windows Server 2003 SP1-et. Konkrétabban:

- A Schema Master szerepet hurcoló tartományvezérlőnek minimum Windows Server 2003 SP1 szervernek kell lennie.
- Minden Active Directory-site-on, ahová Exchange-szervert tervezünk, lennie kell egy Global Catalog funkciónak. Ennek a tartományvezérlőnek is legalább Windows Server 2003 SP1 szervernek kell lennie.
- Abban a tartományban, ahol majd kiadjuk a setup /prepareLegacyExchangePermissions parancsot (és ki fogjuk adni, ezt megígérhetjük), feltétlenül lennie kell egy olyan tartományvezérlőnek, amelyik legalább Windows Server 2003 SP1.

Érdekes lehet, hogy miért ragaszkodik annyira a címtár ehhez a verzióhoz. Nem titok: a Windows Server 2003 SP1 szerverben van egy olyan funkció, hogy Service Notification. Ennek az a szerepe, hogy amikor valami változás történik a címtárban, erről értesítést küld az egyes szolgáltatásoknak. A legtöbb Exchange 2007-szolgáltatás intenzíven ki is használja ezt a lehetőséget.

Persze nem csak ennyi feltételtől van szó. Jöjjenek a többiek!

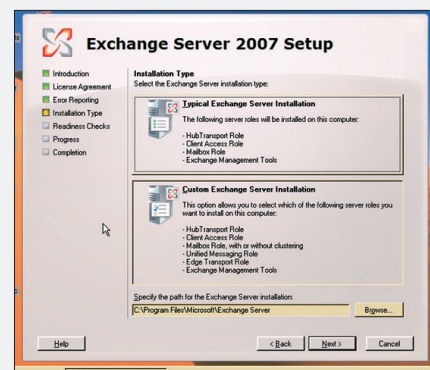
- Amennyiben nem angol nyelvű tartományvezérlőt használunk, és szeretnénk, ha lenne OWA-nk az organizációban, akkor telepíteni kell a KB919166 cikkben említett foltot.
- Az összes tartományban, amelyik részt vesz az organizációban, a tartományi funkcionális szintnek minimum Windows 2000 natívnak kell lennie.
- Amennyiben különböző Active Directory-erdők – azaz organizációk – között szeretnénk bizonyos szintű Exchange-kapcsolatokat, -átjárásokat létrehozni, akkor természetesen biztosítani kell az erdők megfelelő összekötését. Ez gyakorlatilag a

megkívánt forest-forest trustok beállítását jelenti. (Esetleg érdekelhet valakit, mik lehetnek ezek a kapcsolatok. Nos, lehetőség van hozzáférni – organizációk között – egymás free/busy információihoz, képesek lehetünk feladatköröket delegálni, hozzáféréseket engedélyezni a másik erdő erőforrásaihoz – feltéve persze, hogy a trust ezt megengedi.)

- Upgrade esetén nem lehet Exchange 5.5 szerver az organizációban. Ez gyakorlatilag azt jelenti, hogy az organizációnak natív módban kell üzemelnie.
- A DNS olyan precízen működjön, mint egy svájci óra.
- Végül mielőtt bármi komoly munkának nekiállnánk, ki kell preparálni a címtárat. Kipreparálni... könnyű ezt leírni, de a címtár azért csak nem vaddisznó – a preparálás itt egy kicsit komplikáltabb. Hogy pontosan mi az ehhez szükséges négy lépés, az a TechNet Portálon olvasható magyarul a <http://tinyurl.com/wyjid> linken található cikkben.

Hardverkövetelmények

A termék csak 64 bites processzort tartalmazó vason fut – és azok közül sem mindegyiken, csak az x64 alapú rendszereken (Intel EM64T, AMD64). Az Intel Itanium IA64 platformja nem támogatott.



5. ábra. Telepítés – az installálás típusa

A minimális memória mennyisége 2 gigabájt, az optimális memória mennyisége pedig további 5 megabájt mailboxonként. (Érdemes azonban figyelembe venni, hogy a memória-előírások szerepkörönként változnak.)

Ezenkívül legalább 1,2 gigabájt szabad területre lesz szükségünk azon a partición, ahová az alkalmazást telepítjük, valamint további 200 megabájtra a rendszerpartíción. A Unified Messaging funkció telepítésekor

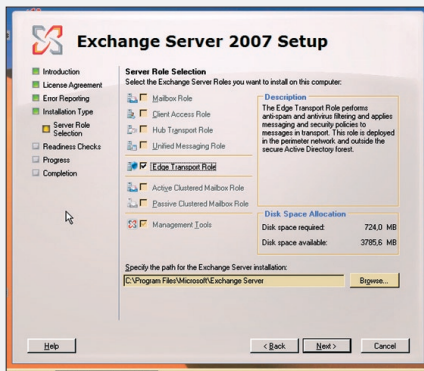
további 500 megabájtot foglal minden egyes nyelvű csomag felrakása.

A telepítéshez használt partíciók csak NTFS formátumúak lehetnek.

Szoftverkövetelmények

Legalább Windows Server 2003 SP1-re lesz szükségünk, amin telepíteni kell a .Net keretrendszer 2.0-t, a Windows PowerShell, valamint a Microsoft Management Console (MMC) 3.0-t.

Fontos még hangsúlyozni, hogy a szerepörök nagyon nem szeretik, ha egy szerveren vannak Network News Transfer Protocol (NNTP) vagy Simple Mail Transport Protocol (SMTP) szolgáltatással – ezeket el kell távolítanunk a telepítés megkezdése előtt.



6. ábra. Telepítés – szerepkörválasztás

Kíváncsiságból kipróbáltam, mi történik, ha minden előkészítés nélkül végignyomkodjuk a telepítőt. Jó hír, hogy másodkora sikerült is a teljes telepítés – érte ezen egy olyan teljesen szűz organizáció létrehozását, amelyben egy szerveren volt a szükséges három szerepkör.

Azért másodkora, mert a telepítési folyamatban létezik egy ellenőrzési fázis, az pontosan kidobta, mi minden nem stimmel. Ezeket leírtam egy cetlire, majd korrigáltam – és innentől hasítottak is a bitek. Szintén jó hír, hogy semmit sem kellett foglalkozni az Active Directory preparálásával: a telepítő észlelte, hogy teljesen nyers címtár van alatta, és szó nélkül lezavarta mind a négy preparációs folyamatot. (Természetesen ehhez az is kellett, hogy maga a címtár nagyon egyszerű legyen: 1 erdő, 1 fa, 1 tartomány, 1 DC – mi pedig legyünk tagjai az Atyauristens csoportnak.)

Éles környezetben azért nem javasolnám ezt a kamikáze-módszert.

Telepítés, végre. Feltehetően most már mindenkinek erősen viszket a keze, és motorikus mozdulatokkal tekergeti tokjában a telepítő-cédét. Igen, mostanra jutottunk el végre oda, hogy betegyük a cédét a meghajtóba, és beírjuk a bűvös szót: setup. Csak így, mindenféle kapcsoló nélkül.

Első körben megjelenik néhány nem túl izgalmas képernyő: a telepítő észreveszi, hogy már van MMC3, .Net keretrendszer és PowerShell a gépen, feldobja a szokásosan gyorsolvasással kivégzett licenccserelezést, megkérdezi, részt szeretnénk-e venni a hibabejelentési programban. Végül megkapjuk az első komolyabban kinéző képernyőt (5. ábra).

Itt dönthetjük el, hogy tipikus telepítést kérünk-e vagy testreszabottat. Az utóbbit a következő esetekben érdemes választani:

- Ha magunk akarjuk meghatározni, hogy milyen szerepkört szeretnénk felrakni a szerverre.
- Ha clusterbe kötött Exchange-szervert szeretnénk telepíteni.
- Ha csak az adminisztrációs konzolra van szükségünk.
- Mindig. Igazi rendszergazda soha nem választ tipikus telepítést.

A 6. ábrán azt láthatjuk, mit is takar a custom opció.

Jelen esetben épp egy Edge Transport szervert fűztünk az organizációhoz. Látható az is, hogy a telepítő okos, nem hagyja olyan mixek kikeverését, amelyek egyébként sem élnének meg. (Például clusterbe kötni csak mailbox-szervereket lehet – tehát itt szűrke a checkbox. Ugyanúgy szűrke a menedzsmeszközök telepítése is, mert ez az elem mindenképpen települ, ha akár csak egy szerepkört is kiválasztunk.)

Ha kikevertük az izlésünknek megfelelő kombinációt, mehetünk tovább. Amennyiben szerepelt a kottélban a Mailbox Server szerepkör, és ez az első szerver az organizációban, akkor először is a telepítő bekéri az organizáció nevét, majd következik egy roppant trükkös kérdés.

Van-e Outlook 2007-nél korábbi kliens a hálózatban? Elsőre nem tűnik olyan sarkalatosnak a probléma, pedig... Ezen a válaszon múlik az, hogy kell-e Public Folder a szerverre vagy sem. Az Outlook 2007 előtti kliensek ugyanis a free/busy és az Offline Address Book információkat a hagyományos módon

próbálják leszedni – márpedig ezek rendszerfolderként a Public Folder adatbázisban vannak.

Ha azt mondjuk, hogy csak Outlook 2007 vagy annál frissebb kliensprogramunk van, akkor nem kapunk Public Folder adatbázist. De nem csak nyilvános mappánk nem lesz – akkor is hoppon maradunk, ha MAPI-n keresztül szeretnénk elérni a mailbox-szervert. A szerver ugyanis blokkolja a MAPI-elérést, amire egyébként is csak a régi klienseknek lenne igényük.

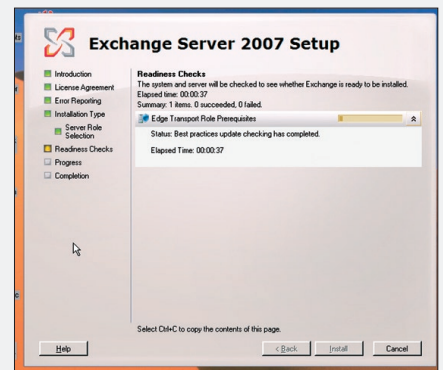
Amennyiben meggondolnánk magunkat, és mégis szeretnénk MAPI hozzáférést, telepíteni kell a későbbiekben egy Public Folder adatbázist – ez oldja a blokkolást is.

Térjünk vissza a telepítési folyamathoz. Jelenleg ott járunk, hogy a telepítő éppen húzza a csikot: ellenőrzi, hogy minden telepítési feltétel teljesült-e.

Amennyiben valami nem stimmel, meglehetősen részletezve közli a program, hogy mit kell pótolnunk.

Szerencsére itt minden rendben volt. (Mondjuk, egy Edge Transport szervernél ez nem egy nagy mutatvány.) Mint a 7. ábrán látható, a 32 bites operációs rendszert már csak hivatalból is megszólta, elvégre ez egy nem támogatott konfiguráció.

Majd amikor nyugtáztuk – mi is, ő is –, hogy minden rendben, akkor indul el a telepítés maga.



7. ábra. Telepítés – az előkészítési fázis ellenőrzése

És itt van. Igen, megcsináltuk!

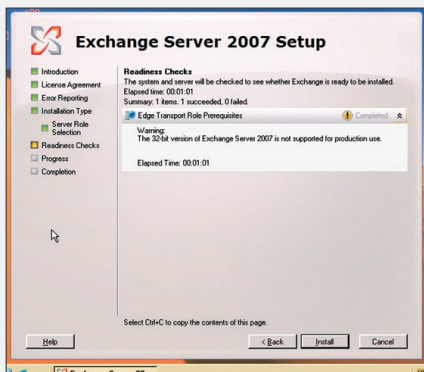
Persze butaság lenne azt hinni, hogy ezzel vége is a telepítési folyamatnak.

A telepítés ellenőrzése. A telepítés végén célszerű végignézni a folyamat során keletkező logfájlokat. Ez jelenti egyfelől a klasszikus event logokat, de az Exchange emellett gyárt két, kifejezetten bőbeszédű szöveges fájlt is:

```
<system drive>\ExchangeSetupLogs\ExchangeSetup.log,
<system drive>\ExchangeSetupLogs\ExchangeSetup.
msilog.
```

Az utóbbi a telepítő fájlok kitömörítésének folyamatát rögzíti, míg az előbbi a teljes telepítést.

Volt-e már az olvasók között olyan személy, aki valaha is rá volt kényszerítve, hogy végigolvasson egy ilyen logot? Esméletlen munka. Nem véletlen, hogy az Exchange mellé csomagoltak egy kis szkriptet is (<system drive>\Program Files\Microsoft\Exchange\Scripts\Get-SetupLog.ps1).



8. ábra. Telepítés – az ellenőrzés eredménye

A következő kombináció faszervezetben kelistázza az összes hibaüzenetet a logfájlból:

```
Get-SetupLog c:\exchangelogs\exchangelogs\exchangesetup.log -
error -free
```

Ha eddig nem tettük, ellenőrizzük, hogy a korábban részletezett változások tényleg megtörténtek-e a címtárban.

A telepítés végén akár 20(!) Exchange-szolgáltatás is felkerülhet a gépre a szerepektől függően. (Elég jól sikerült elszakadni az Exchange 4.0 négy darab alapszolgáltatásától.) Érdekes ezeket végignézni, hogy amelyek „automatic”-ra van állítva, tényleg elindult-e.

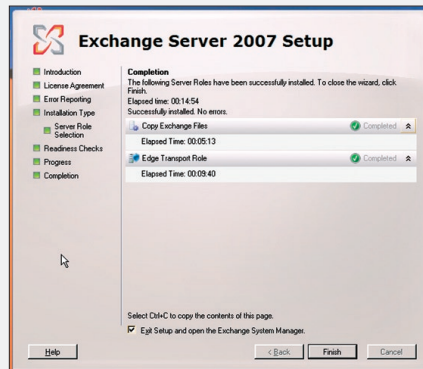
Edge/Hub transport szerverek esetén úgynevezett ügynökök (agentek) települnek a szerverre. Célszerű ezeket is ellenőrizni, hogy rendben létrejöttek-e.

Ezek az ügynökök gondoskodnak egyfelől a higiéniai (antispam, antivírus) szabályok végrehajtásáról, illetve a különböző házirendek érvényesítéséről.

Legvégül érdemes lefuttatni az Exchange Best Practice Analyzert is, ellenőrizendő,

hogy valóban minden rendben működik-e. (A program elérhető az adminisztrációs konzolból is.)

Utómunkálatok. Nem, nem ez nem mániakusság – egyszerűen csak képtelenség még elszakadni a telepítés folyamatától. Ugyanis attól, hogy fellapátoltuk a binárisokat, még nem lesz működőképes a rendszerünk.

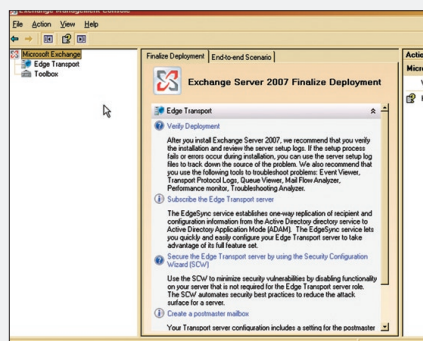


9. ábra. Telepítés – záró képernyő

Ilyenkor még célszerű lefuttatni az úgynevezett Security Configuration varázslót, valamint mindegyik szervernél érdemes beállítani a különböző jogosultság-delegációkat – azaz, hogy mely szerepkörök mely felhasználói csoportok menedzselhetik.

További teendők szerepkörönként:

- Hub Transport Server esetén: hozzuk létre és állítsuk be megfelelően a transzport jellegű szabályokat, illetve a házirendet.
- Elemezzük ki a levélfolyam útvonalát a Mail Flow Analyzer segédprogrammal. Gyártunk egy külön postafiókot a karantén számára, majd rendeljük ehhez egy felhasználót.



10. ábra. Telepítés utáni munkálatok

- Client Access Server esetén: engedélyezzük az IMAP4, POP3 protokollokat, amennyiben szükségünk lesz rájuk. Érvényesítjük a külsős kapcsolatokra vonatkozó biztonsá-

gi megfontolásainkat (HTTPS, RPC over HTTP, autentikációk stb.).

- Mailbox Server esetén: hozzuk létre a megálmodott adatbázis-szerkezetet, és kezdjük el felvenni/migrálni a felhasználói postafiókokat. Ezek után gondoljuk végig a backup/restore folyamatokat.
- Edge Transport Server esetén: konfiguráljuk be az ADAM-et.

Látható, van utómunkálatból is bőven. Nem is várja el senki, hogy fejből emlékezzünk mindegyikre: ha a konzolban a „Microsoft Exchange” folderre állunk, akkor a középső területen a „Finalize Deployment” fül alatt látható, hogy az adott szerveren milyen munkálatok vannak még hátra.

Szerepkörök módosítása

Hozzáadás. Van egy működő rendszerünk. Egy idő után észrevesszük, hogy a szerepkörök nem igazán optimális módon osztottuk szét a szerverek között – szeretnénk egy kicsit átrendezni az organizációt. Berakjuk az első számítógépre a telepítő-cédét, hogy bővítsük a szerveren a szerepek körét... és nem történnik semmi.

Ez bizony kellemetlen. Az Exchange-szerveren a szerepkörök utólag már nem lehet bővíteni. Legalábbis a grafikus felületről nem. De szerencsére a setup programnak van néhány kapcsolója, ezekkel elég sok mindenre rá lehet venni ezt a kedves binárist, amit a /? kapcsoló segítségével tudunk áttanulmányozni.

Eltávolítás. Igen, bármennyire is hihetetlen, de előfordulhat olyan eset, hogy akár egy szerepkört, akár egy komplett szervert el szeretnénk távolítani. A szerepkört eltávolíthatjuk egyfelől a fenti setup parancs segítségével – de itt már használhatjuk a grafikus felületet is, a következőképpen:

Control Panel/Add-remove Programs/Microsoft Exchange Server/Remove – ekkor bejön a telepítő grafikus felülete. Kiválasztjuk az Exchange Maintenance módot, majd bejelöljük az eltávolítandó szerepköröket.

Amennyiben az egész szerver eltávolítása a cél, akkor a fenti munkamenetet kell végigcsinálni, csak most az összes szerepkört – és az admin konzolt – kell kijelölni eltávolításra. Szerepkör nélkül pedig szerver sincs többé.

Petrényi József

(petrenyi.jozsef@sao.hu)

SAO-Synergion, MCSE+M, MVP

EXCHANGE SERVER 2007: MAGAS RENDELKEZÉSRE ÁLLÁS

Bizonyos első találkozásokra egész életünkben emlékszünk – hasonlóképp izgalmas dolog egy új szerverszoftverrel való ismerkedés is –; többek között az előző verziókkal kapcsolatos tapasztalatok alapján várunk és – jó esetben – kapunk újdonságokat.

A kommunikáció különösen érdekes téma, manapság már majd' mindennel összefonódik, és megkockáztatható, hogy a helyzet még tovább fog bonyolódni. Az Exchange-szerver összetett adattovábbító, -kezelő- és -tároló rendszerként ezen a területen teszi a dolgát jó ideje, verzióváltásnál viszont automatikusan eszünkbe jut, lehet-e még tartalmasan újítani/bővíteni rajta valamit.

A *TechNet Magazin* többi cikkét átböngészve hamar rájöhettünk, hogy igenis lehet.

E rövid fejezetben az Exchange Server 2007 magas rendelkezésre állási, a hibátűrést fokozó – online és offline – megoldásait tekintjük át; és ezen a speciális területen is találkozhatunk figyelemre méltó érdekességekkel.

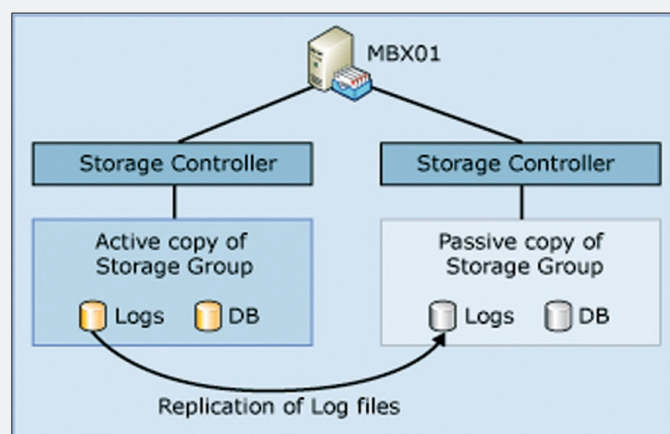
Magas rendelkezésre állás esetén az ember azonnal a különféle fűrtözési megoldásokra asszociál. Ilyenkor nagy költségvetésű konfigurációkat és bonyolult installációkat vizionálunk, az egész ügy egy picit misztikus ködbe is burkolódik; százalékok, kilencesek, fura szavak és rövidítések (downtime, SLA, SAN stb.) kerülnek elő. Remélhetőleg e cikk arra is jó lesz, hogy kiderüljön: ha ismerjük a szoftvereink adta lehetőségeket, egy átgondoltan kialakított rendszerrel is rengeteg lehetőség nyílik a rendelkezésre állási szint emelésére, anélkül, hogy az IT-büdzsé jelentős részét elvinné az extra hardver és support. Erősen leegyszerűsítve a kérdést, két dologra kell figyelni: 1. Ne romoljon el, ami fontos, 2. Ha mégis, nagyon gyorsan helyre lehessen állítani.

Lássuk akkor, mire is képes az új Exchange!

Folytonos helyi replikáció (Local Continuous Replication, LCR)

Érdekes az LCR-rel kezdenünk áttekintésünket, mivel ehhez a megoldáshoz nincs szükség fűrtözésre. Exchange-szerverünk mostantól hajlandó lesz másolatot készíteni – és fenntartani – egy tárolócsoportról vagy adatbázisról ugyanannak a kiszolgálónak egy másik diszk(tömb)-

jére a tranzakciónaplók továbbításával. Az LCR elszállítja és újrajátssza – a mostantól 1 megabájtra lefogyott – logokat a tartal-



A Local Continuous Replication működése

lék adatbázison, így előállít egy teljes értékű másolatot a produktív adatokról, amire egy adminisztratív lépéssel bármikor át lehet kapcsolni. Mindezek mellett az LCR a hagyományos backup-megoldásokra is jótékony hatással van, mert csökkenthető a napi teljes mentések száma, másrésztől a mentőszoftver dolgozhat a másolattal, meghagyva az értékes diszk I/O-t az éles adatbázis alatt. Ettől füg-

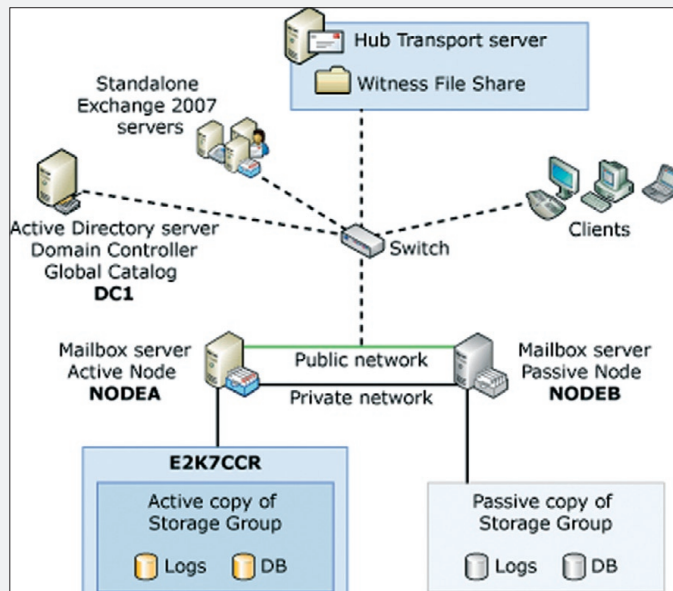
getlenül a már megszokott offline mentéseinkre is szükség lehet, mivel az LCR nem biztosít 24x7-es rendelkezésre állást (például szerverhiba előfordulásakor). Ezekre az esetekre nézünk megoldásokat a továbbiakban.

Fürt közös tárolóval (Single Copy Cluster, SCC)

Az SCC egy fürtözött mailbox-szerver közös diszkrendszerrel. Ez tulajdonképpen az Exchange előző változataiban már használható shared-nothing modell, amelyben a fürt bármelyik tagja hozzáférhet a tárolócsoporthoz, de egyszerre csak az egyikük. Változások azért ebben az üzemmódban is vannak, például az Exchange Server 2007 csak az aktív/passzív SCC-összeállítást támogatja, hozzátevé, hogy az előző verziókban is ez volt az ajánlott konfiguráció. Az SCC mód Exchange-szerverünk szolgáltatásainak közel folyamatos rendelkezésre állását biztosítja, de továbbra is érzékeny marad a közös tároló hibáira. Így a fürt tagjainak számától függetlenül mindig lehet SPOF (Single Point of Failure) a rendszerben, ellentétben a következő pontban tárgyalt CCR-megoldással.

Folytonos fürtbeli replikáció (Cluster Continuous Replication, CCR)

A CCR kombinálja az Exchange Server 2007 naplózási és -újrajátszási lehetőségét az



A Cluster Continuous Replication minden ellen véd

MSCluster-szerviz hibatűrési képességeivel. Folyamatos rendelkezésre állást biztosít adatokban és szolgáltatásokban SPOF nélkül, mivel nincs egy közös diszkrendszer a fürt tagjai között (mint a hagyományos failover fürtökben, lásd SCC).

A CCR az MSCluster-szerviz Majority Node Set (MNS) módját használja, mivel így lehetséges közös diszkrendszer nélküli, különálló szerverekből álló fürtöt kialakítani. Tehát részben az operációs rendszer nyújt segítséget, hogy a quorum elérhető legyen a fürt tagjain, emellett az Exchange replikációs szervi-

vagy több ügyis elő kell venni a mentéseinket. A már ismert és bevált lehetőségeket fus-

parált, katasztrófatűrő konfigurációk, egyszerűbb telepítés és az LCR-nél már említett mentéssel kapcsolatos pozitív hatások.

Befejezésül nézzük át, hogy mit nem tehet, aki a CCR mellett dönt: egy tárolócsoporthoz csak egy adatbázis lehet (az Exchange Server 2007 maximum 50 tárolócsoporthoz kezel), nem futhat ugyanebben a fürtben sem az Exchange előző verziója, sem SQL-szerver és nyilvánosmappa-adatbázis is csak korlátozások mellett lehet CCR-fürtben tartani.

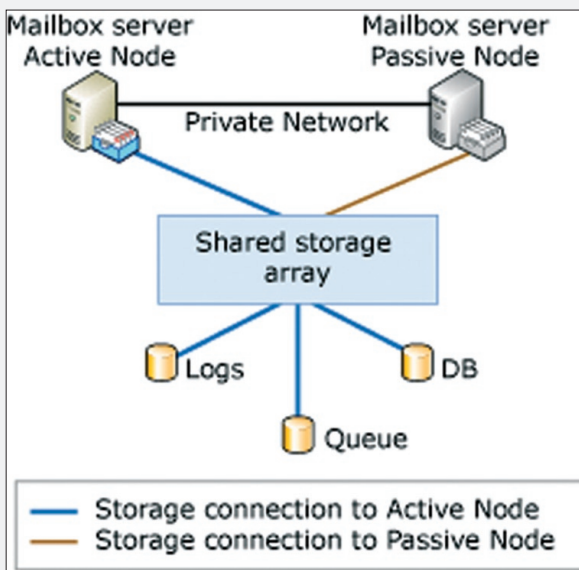
Mentés

Ákár fürt, akár nem, előbb

Availability – A Holistic Approach (cont)

- Mailbox Server
 - Cluster Continuous Replication (CCR)
 - Offers Data Availability
 - Non-Shared Storage (mitigation of single point failure)
 - Multiple Data Storage Options
 - Based on Windows Clustering (auto-failover)
 - 2 Physical Node Limitation
 - Local Continuous Replication (LCR)
 - Data availability for Standalone Servers
 - Non-shared storage
 - Multiple Data Storage Options
 - Single Copy Cluster (SCC)
 - Shared-Storage
 - Based on Windows Clustering
 - Multi-Node possibilities

Microsoft



A Single Copy Cluster megoldás

Rendelkezésre állási lehetőségek az Exchange 2007-ben

suk át gyorsan: hagyományos online mentés és a Volume Shadow Copy Service.

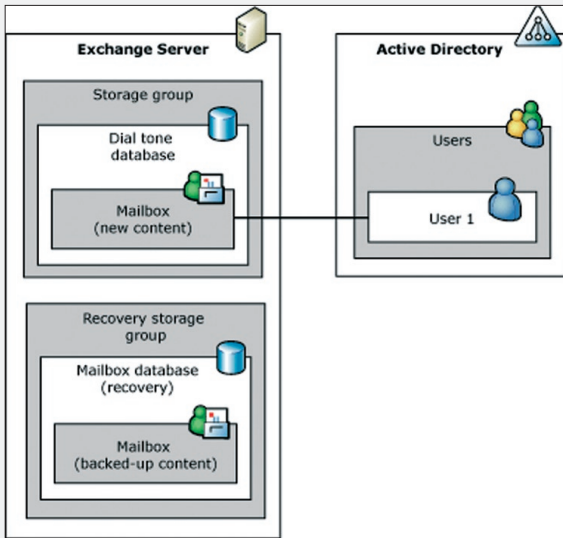
Az első az Exchange ESE Backup API-ján keresztül, a második az operációs rendszer VSS-szervizével érhető el. Mindkettőhöz használható a Windows Server 2003 beépített mentőszoftvere, hozzátevé, hogy „igazi” Exchange-VSS-mentéshez más gyártó megoldásainak alkalmazására lesz szükség.

A mentési típusok nem változtak: Full, Copy, Incremental, Differential. A választás alapja lehet, hogy mentsünk-e mindent vagy csak a változásokat, illetve, hogy a mentés vagy a visszaállítás folyamata legyen-e gyorsabb.

Ugyanakkor az is igaz, hogy teljes adatvesztés ritkán fordul elő, inkább elemi adatokat kell helyreállítani, és ezt az Exchange

eddig is támogatta megfelelő beállítások mellett: a törölt elemek/mailboxok visszatartása mostantól alapértelmezés szerint 14/30 na-

Exchange-változatokban ez egy recovery storage group, egy, az eredetivel megegyező nevű vagy egy azonos adminisztratív csoportba



A Storage Groupok helyreállítási lehetőségei

– vagy tovább is, ha szükséges –, figyelembe véve, hogy az adatbázisok mérete ezek az elemek is beleszámítanak.

Összefoglalva, ami fontos a mentésekkel kapcsolatban: mennyi helyi és/vagy hálózati erőforrást visz el, mennyi ideig tart, milyen gyakran kell ismételni. Abból kiindulva, hogy folyamatos üzem tervünk, optimális megoldásként a hagyományos mentési eljárások kombinálhatóak az előző bekezdésekben tárgyalt kétféle folytonos replikáció valamelyikével.

Bár mindenkinek a saját, személyes információinak elvesztése a legkritikusabb, a teljes rendszer fenntartásának szempontjából fontos a komplett adat-helyreállítási procedúrákat is rögzíteni a katasztrófa-elhárítási tervben. Az Exchange Server 2007 ezen a területen is nyújt újat.

Hordozhatóság

Új lehetőség az adatbázisok hordozhatósága, amely lehetővé teszi, hogy azokat egy másik szerveren el tudjuk indítani. Az előző

csatlakozásukkor – a mail-profiljuk módosításával – az új helyre továbbítja az Exchange Autodiscover szervize. Régebbi Outlook-verziók esetén manuális módosítás szükséges, ha a szerver neve megváltozott. Jó tudni, hogy a hordozhatóság csak a mailbox-adat-

bazisokra vonatkozik, mivel a public folder adatbázisok hozzáféréseik kezelésében, replikációjuk során közvetlenül kötődnek egy bizonyos Exchange-szerverhez. Szorosan kapcsolódik a hordozhatósághoz az úgynevezett dial-tone helyreállítási módszer

– a mail-profiljuk módosításával – az új helyre továbbítja az Exchange Autodiscover szervize. Régebbi Outlook-verziók esetén manuális módosítás szükséges, ha a szerver neve megváltozott. Jó tudni, hogy a hordozhatóság csak a mailbox-adat-



Már telepítéskor is kiválaszthatjuk a clusterezést

– a mail-profiljuk módosításával – az új helyre továbbítja az Exchange Autodiscover szervize. Régebbi Outlook-verziók esetén manuális módosítás szükséges, ha a szerver neve megváltozott. Jó tudni, hogy a hordozhatóság csak a mailbox-adat-

bazisokra vonatkozik, mivel a public folder adatbázisok hozzáféréseik kezelésében, replikációjuk során közvetlenül kötődnek egy bizonyos Exchange-szerverhez. Szorosan kapcsolódik a hordozhatósághoz az úgynevezett dial-tone helyreállítási módszer

Gyors ismétlés: Majority Node Set (MNS)

A W2K3-ban megjelent új fűrtözési típus, amiben nincs szükség egy közös diszkrendszerre a tagok között. Mindenki maga intézi a quorum adatainak tárolását, a Cluster-szerviz pedig lerendezi annak replikációját. Így aztán földrajzilag szétszórt, illetve. olcsóbb szerverek is fűrtözhetők. (Az alkalmazás adatait viszont nem replikálja az Microsoft Cluster Service, azt oldja meg valaki más.) Eredetileg páratlan számú taggal lehetett kialakítani, hogy egy esetleges tudathasadásnál az egyik tagscsoport nagyobb legyen (lehesse majority). Windows Server 2003 SP1-re, és R2-re van egy 921181-es számú hotfix, ami után már lehet kéttagú MNS-fűrtöt csinálni (például Exchange Server 2007 alá), ekkor kell viszont egy File Share Witness (harmadik, független szerver), amelyik szükség esetén voksol az aktív tag mellett.

is. Tegyük fel, hogy bekövetkezett, amire senki sem vágyott: leállt a levelezőszerver. Ilyenkor a rendszergazdáknak legkevésbé sem hiányoznak a felhasználók (plusz ingerült főnökök) folyamatos jelzései, miszerint úgy tűnik, nem megy a levelezés. Tehát két fontos feladat van: lehesse küldeni/fogadni üzeneteket, ha a régebbi adatok átmenetileg nem is érhetőek el a szerveren, és emellett a lehető leggyorsabban álljon helyre megszokott környezet.

A hordozhatósággal itt is javulnak az esélyek: készíthetünk a felhasználóknak egy új, üres mailbox-adatbázist, így folytathatják külső-belső levelezésüket (cache-elt módú Outlookkal még a régebbi üzeneteiket is elérjük), amíg az informatikai csapat nagy sebességgel visszaállítja a régi adatokat egy recovery storage group segítségével, amit utolsó felvonásként összefesülünk a menet közben keletkezett új adatokkal.

A procedúra alatt a felhasználók átirányítása az előző bekezdésben vázolt módon természetesen itt is megtörténik, mi pedig reménykedünk, hogy ez a kis kaland nem érinti a negyedéves bónuszunkat.

A felvetett témákkal kapcsolatos további érdekes részletek találhatóak a <http://www.microsoft.com/technet/prodtechnol/exchange/e2k7help> linken az „Operations – High Availability” bekezdésre kattintva.

Ország Tamás

MCSE+S+M, MCT (tamas@edupro.hu) Számlak

MIKOR VÁLTSUNK 64 BITRE?

Rengeteg a tévhit a 64 bites rendszerekkel kapcsolatban. Például ha ma valaki 64 bites hardvert vásárol, rögtön 64 bites operációs rendszert tesz rá. Biztosan jó döntés ez?

A hogy folyamatosan nő az igény az egyre gyorsabb és nagyobb tárkapacitással rendelkező számítógépek iránt, újabbnál újabb ötletekkel állnak elő a hardvergyártók. A 64 bites technológiák már igencsak régen megjelentek a nagyobb szerverhardverek piacán: nagyobb mennyiségű megcímezhető memóriaterületet kínáltak, valamint a sebességet is növelték azáltal, hogy egy utasítással több beérkező adaton végeztek el műveleteket. Azonban ezek többsége a mai hardverekkel és szoftverekkel nem volt kompatibilis, mivel teljesen újszerű utasításkészletet és módszereket használtak. Az egyik ilyen versenyző az Intel Itaniumja.

Eleinte nem örvendett hatalmas népszerűségnek ez az új architektúra, hiszen minden alkalmazást teljesen újra kellett hozzá írni (beleértve az operációs rendszereket is), azonban ma már megtalálta a saját közönségét, és sokan választják ezt a legnagyobb teljesítményt igénylő alkalmazásaihoz.

Ahhoz, hogy megértsük a 64 bit másik fejlődési irányát, fontos kiemelni: ekkor következett be az a pont a hardverfejlődésben, amikor rádöbent a teljes iparág, hogy nem lehet tovább növelni a processzorok frekvenciáját a sebesség további növelése érdekében. Ekkor az AMD előállt saját 64 bites architektúrájával, az x64-gyel – ami kompatibilitást ígért a korábbi hardverekkel és szoftverekkel is.

x64

Gondolkodjunk el egy kicsit, mit is jelent ez! Nyilvánvalóan nem használ olyan újszerű utasításkészletet, mint az Itanium. Viszont 64 bites. Biztosan sokan emlékeznek még arra, milyen óriási jelentőséggel bírt a váltás a 8-ról 16-ra, vagy a 16-ról 32 bitre. De miért is? Mert több címezhető memóriára és tárhelyre volt szükségünk. Volt ezen kívül más újdonság is a korábbi

átállásokkor? Igen, optimalizálták a processzor belső felépítését. Egyet elárulhatunk: ez utóbbi most, a 64 bit esetében nem sikerült számottevően.

Nem kell sokat gondolkodni azon, hogy akkor mégis miért volt ennyire fontos az x64 megjelentetése, ha tulajdonképpen az átlagfelhasználók számára csak annyit ígért, hogy 4 gigabájtól több memóriát is használhatnak. Ekkor már eltelt néhány év, hogy igazán új, mindennél jobb vagy gyorsabb processzor nem jelent meg a piacon, és az emberek nem tudták eldönteni új gép vásárlásakor, hogy melyiket vegyék meg a közel egyforma árú és teljesítményű versenyzők közül.

Viszont a számokat mindenki ismeri. „A 64 több, mint a 32, biztosan jobb, ráadásul kompatibilis...” – gondolhatták sokan, és elkezdtek vásárolni az AMD processzorait. Nem meglepő módon az AMD ekkor jelentős növekedésnek indult.

Az sem meglepő, hogy az Intel is kihozta ezek után EM64T processzorait (gyakorlatilag kénytelen volt), ezek ugyancsak az AMD-féle 64 bites megoldást használták. Az AMD64 és az Intel EM64T chipjeit hívjuk azóta gyűjtőnéven x64-nek, hiszen ugyanarra az alapra épülnek mind a ketten. Az x64 gyakorlatilag időt nyert a hardvergyártóknak, hogy kidolgozzák az új típusú, többmagos processzorokat, de addig is újat tudjanak adni a vásárlóknak, még ha nem is kaptak sokkal többet a pénzükért. A marketing bevált.

Arról van tehát szó, hogy talán kicsit korán erőltették rá a piacra azt, amire még valójában nem volt szüksége. Viszont mivel hamarosan tényleg szükség lesz a 64 bites váltásra, már nem volt értelme a szoftvergyártóknak sem visszakozniuk, hiszen pár éven belül tényleg minden otthoni felhasználó és szerver eléri a 4 gigabájtos határt.

Tudva a technológia gyengéit, próbáljuk meg mégis kihozni belőle azt, amit lehet!

Mire is jó az x64?

Korábban, ha 2–3 gigabájtól több memóriát szeretnénk volna használni egy x86-os, windowsos gépen, már trükkökhöz kellett nyúlnunk (például a PAE és/vagy a /3GB kapcsolóhoz), de még így sem mehettünk 64 gigabájt fölé, és ez a megoldás sem volt tökéletes. Az x64 segítségével elhárul az akadály, a határ a csillagos ég (konkrétabban: most 8196 gigabájt, de ez még növelhető lesz). Ez



A 64 bites Intel Itanium

nagy adatbázisok, nagy terheléssel futó web-farmok és lekérdezéseket kiszolgáló rendszerek esetében jelentős segítség, hiszen minden maradhat a memóriában, több lekérdezést és felhasználót tudunk egyszerre kiszolgálni – vagyis tudunk fölfelé skálázódni egy számítógépen belül is.

A sebesség is nőhet bizonyos esetekben, mivel a 64 bites rendszereken a processzor regiszterei is 64 bitesek, ami például a nagy pontosságot igénylő lebegőpontos számításokon sokat tud gyorsítani. Mikor jó ez nekünk? A mérnöki munkát segítő CAD/CAM-alkalmazások, a 3D-renderelés, valamint képfeldolgozó és néhány tömörítőprogram, valamint egyes szerversoftverek képesek ezt kihasználni, de a szoftvernek külön támogatnia kell a 64 bites natív működést. Ez azt jelenti, hogy a szoftver bináris állományainak natív 64 bitre fordítva kell lenniük (nem is indulnak el 32 bites rendszeren). A hagyományos 32 bites alkalmazások azonban nem lesznek gyorsabbak akkor sem, ha alattuk tényleg ott egy 64 bites operációs rendszer és maga a 64 bites hardver.

A szakmai teljesség kedvéért azt azonban meg kell említenünk, hogy a 32 bites alkalmazások egy speciális esetben tudnak profitálni a 64 bites rendszer nyújtotta előnyből még abban az esetben is, ha az alkalmazás nincs optimalizálva az új futtatókörnyezetre. Ez akkor következhet be, ha párhuzamosan kell futtatnunk két vagy több olyan 32 bites alkalmazást, amelyek eleve 2–2 gigabájt memóriára van szüksége. Mivel a 64 bites architektúrán minden 32 bites alkalmazás önállóan lát egy legfeljebb 4 gigabájtos memóriaterületet, így ha hardverünk képességei azt megengedik, akkor akár két 32 bites alkalmazás párhuzamosan futva, használhat 3–3 gigabájt memóriát. Ez elképzelhetetlen egy 32 bites rendszerben, még trükközésekkel is.

Ha már új architektúránk van, amire úgyszólván újra kell fordítani az alkalmazásokat (ha újrainni nem is kell), miért ne követelhetnénk meg néhány újdonságot alapkövetelményként? Elvégre van egy-két hasznos új biztonsági technológia, például a DEP (Data Execution Prevention) és az, hogy 64 bites Windowsokkal csak aláírt drivereket tölthetünk be Kernel módba. Ha már úgyszólván újra fogják fordítani a szoftver- és driverfejlesztők a kódjukat, legalább teszteljék le azt is, hogy ezeknek az igényeknek is jól megfelelnek-e!

Ezzel véget ér a 64 biten a DEP-inkompatibilis és a hibás kernel módú driverek korszaka, ugyanis a 64 bites Windows Vista esetében mind a DEP, mind a driver-aláírás kötelező érvényű.

Mi a baj veled?

Az imént láthattuk, hogy csak speciális alkalmazások és terhelések nyerne a 64 bit használatával. A 32 bites szoftverek pedig a 64 bites Windows WoW (Windows on Windows) alrendszerének köszönhetően működnek. Ez az emulálóréteg fordítja át a 32 bites utasításokat és címzéseket olyanná, hogy 64 biten is megfelelően működjenek. Ez azonban akár mennyire kicsit is (egy százalékban belül mozog az esetek többségében), de lassít a 32 bites kód végrehajtásán, de messze nem annyira, mint az Itanium alapú 64 bites rendszerek 32 bites „kompatibilitási” rétege. Külön érdekesség, hogy a 32 bites Windows XP is használt egy másik, régebbi WoW-ot 16 bites alkalmazások futtatására a 32 bites rendsze-

kompatibilis – megvalósítani már sokkal nehezebb. A 32 bites driverek, illetve a kernel-szinten futó kódok esetében a WoW ugyanis nem működik. Ez azt is jelenti egyben, hogy azok a hardvereink, amelyekhez csak 32 bites driver érhető el, nem fognak működni 64 bites operációs rendszeren (gondoljunk itt a régi nyomtatókra, scannerekre). Sajnos szám-talan olyan alkalmazás is van, ami hardverközeleli hívásokat használ, és emiatt működésük nem emulálható ezekkel az eszközökkel. Összefoglalva: igaz, hogy maga a processzor tökéletesen futtatja a 32 bites alkalmazásokat – vagyis 64 bites vasra telepíthetünk 32 bites operációs rendszert, és nem lesz semmi kompatibilitási gondunk, bár 64 bites alkalmazások sem fognak menni, és a 64 bit semmi előnyt nem tudunk így kihasználni. Az már sajnos nem igaz, hogy 64 biten futó operációs rendszeren gond nélkül elférnek egymás mellett a 32 és 64 bites driverek.

Az utolsó nagyobb probléma ennek a következménye, de a fejlesztőket sújtja: újra kell írni minden alkalmazást – vagy legalább fordítani 64 bitre is –, és a meglévő 32 bites alkalmazásokat is tesztelni kell a WoW-ban. Erre jó példa a Symantec víruskeresője, amelyik 8-as és 9-es 32 bites szériája nem fut megfelelően a WoW-ban. A Symantec ezt követően nem az elegáns utat választotta – hogy készít egy natív 64 bites víruskeresőt –, hanem a meglévő 32 bites víruskeresőjét készítette fel a WoW-ban való futásra.

Vagyis minden szoftvert tesztelni kell 32 és 64 biten is, ráadásul mindkét verzióhoz támogatást is kell adni. Ez jelentős pluszmunkát jelent, és még sokan nem szánták rá a szükséges energiát.

Valljuk be, ezek azért nem akora meglepetések, ehhez hason-

lót majdnem minden generációváltáskor átélünk már. A lényeg, hogy mi magunk alaposan és nagyon körültekintően járjunk el, ha generációváltásra adjuk a fejünket, meg kell találnunk azt az időpontot, amikor érdemes váltanunk saját gépünkkel vagy éppen az IT-infrastruktúránkkal.



Az Intel legújabb, kétmagos, 64 bites processzora, a Core 2 Duo

reken. A lényeg: aki azt várja, hogy meglévő 32 bites alkalmazásai gyorsabbak lesznek 64 bites operációs rendszeren és hardveren, az lehet, hogy pont az ellenkezőjét fogja tapasztalni, még ha csak kis mértékben is.

A nagyobb gond azonban az, hogy túlságosan könnyű azt mondani valamire, hogy

Fontos megjegyezni azonban azt is, hogy nemcsak a túl korai, de a túl késői átállás is sok kényelmetlenséggel jár, ezért nagyon fontos a megfelelő időben lépni.

Ha 64 bites hardvert vettem...

Mikor telepítsünk 64 bites operációs rendszert egy számítógépre? Kizárólag akkor, ha az általunk futtatandó alkalmazások vagy szerverszoftverek mindegyikére igaz az alábbiak állítások valamelyike:

- 32 bites szoftver, aminek nincs natív 64 bites verziója, de gond nélkül működik 64 biten is a WoW-on keresztül, és az esetleges minimális teljesítménycsökkenés sem jelent problémát számunkra.
- Natív 64 bites alkalmazás, és vagy nincs belőle 32 bites változat, vagy szükségünk van több memória hatékonyabb használatára (például nagy adatbázisra), és szükségünk van a lebegőpontos műveletek gyorsítására.

A fent említetteken kívül még két szempont segít dönteni:

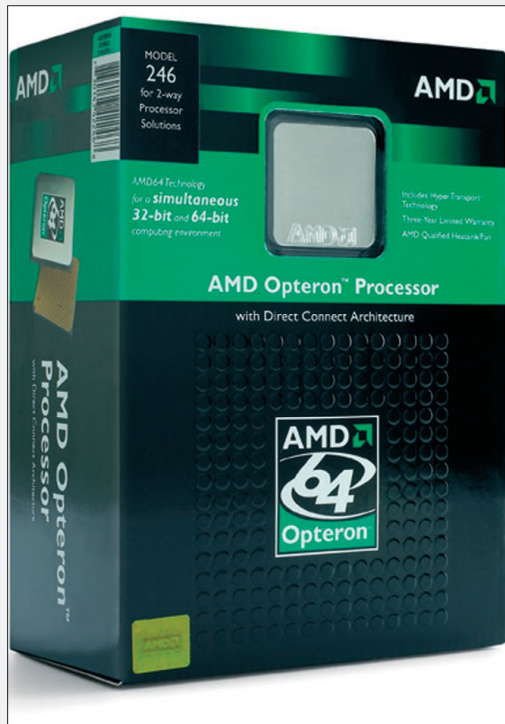
- Minden hardverkomponenshez kell lennie 64 bites drivernek, különben nem tudjuk használni. (Tapasztalatunk az, hogy az újonnan szállított brand szerverekhez a gyártó minden, a dobozban levő hardverelemhez biztosítja a 64 bites drivert. Meglepetés ettől még érhet minket, tapasztalataink szerint a felügyeleti programokat a gyártók nem, vagy csak korlátozott funkcionalitással készítik el.)
- Bármilyen, nem natív 64 bites (vagyis 32 bites) szerverszoftver telepítése és üzemeltetése gyakorlatilag tilos és életveszélyes. A javasolt alternatíva ilyenkor, hogy a 64 bites operációs rendszerrel rendelkező virtuális gépet üzemeltessünk, amin már nyugodtan futhat a 32 bites szerverünk (abban az esetben, ha a szerverszoftverünk virtualizációja támogatott).

Jó tanácsok

Ha otthonra vettünk számítógépet, és nem vagyunk CAD/CAM-mérnökök, véletlenül se tegyünk fel 64 bites kliens-operációs rendszert még, hacsak nem vagyunk biztosak

abban, hogy a számunkra kritikus alkalmazások (esetleg játékok) és driverek megfelelően működnek. Ha biztosak vagyunk abban, hogy minden számunkra fontos szoftver megy 64 biten, akkor aggodalomra nincs okunk, de teljesítményjavulást ne várjunk, nem lesz, vagy ha igen, minimális, és csak néhány alkalmazásra lesz jellemző.

Ha x64 alapú szervert veszünk, gondoljuk át, milyen szerverszoftvereket szeretnénk futtatni rajta. Ha a szerverszoftverből nincs natív 64 bites változat, ne tegyük fel a 64 bites operációs rendszert se, inkább maradjunk a



Az AMD 64 bites Opteronjának doboza

32 bitesnél egyelőre, és várjuk meg, míg lesz natív 64 bites verzió belőle. Ha van 64 bites változat, és szeretnénk kihasználni az előnyeit (több memóriára lesz szükségünk, illetve szeretnénk kapni némi teljesítménytöbbletet), akkor tegyük fel a 64 bites operációs rendszert (a fentebbi szempontok figyelembevételével). Természetesen jó, ha a telepítés előtt tájékozódunk arról, hogy a jelenlegi teljesítményproblémákat valóban a 32 bites platform korlátjai jelentik-e? Mert, ha nem a platform korlátait értük el, hanem egyéb szoftver- vagy lekérdezés-optimalizációs problémával állunk szemben, abban az esetben tényleg ne várjuk azt, hogy a 64 bites platform megoldja a problémánkat. Ha csak 64

bites verzió létezik a szerverszoftverből, akkor kénytelenek vagyunk feltenni alá a 64 bites operációs rendszert is – azonban ez már egyáltalán nem rossz kombináció.

A választék

Az operációs rendszerek közül a következők érhetőek el 64 bites változatban:

- Windows XP (x64)
- Windows Vista (x64)
- Windows Server 2003 (R2 – Itanium és x64)

Lássuk, milyen fontosabb Microsoft alkalmazásokból áll rendelkezésünkre 32 bites és natív 64 bites változat egyaránt. Ezeket már nyugodt szívvel feltehetjük 64 bites operációs rendszerre is:

- Office 2007 (kliens- és szerveralkalmazások is);
- SQL Server 2005, BizTalk Server 2006;
- Virtual Server 2005 R2 (fut natív 64 biten, de csak 32 bites virtuális gépet tud futtatni);
- a későbbi System Center termékek (tavasszal érkeznek);
- a „Longhorn” Server (2007 vége).

Ehhez kapcsolódik még a Windows Server 2003-ban (az R2 funkcióival egyetemben) található valamennyi képesség, ezek mind elérhetőek natív 64 biten is, beleértve az Internet Information Servert, a .Net keretrendszer 2.0-s változatát is, vagy akár az Active Directoryt. A .Net és az IIS képes profitálni a 64 bit előnyeiből, és így a rajta futtatott alkalmazások is.

A következő szerverekből például nem érhető el jelenleg natív 64 bites változat, ezeket most még ne akarjuk 64 bites operációs rendszeren futtatni:

- ISA Server 2004/2006;
- Microsoft Operations Manager 2005, Systems Management Server 2003;
- Exchange Server 2003.

Ezek a szoftverek pedig már csak natív 64 bites változatban érkeznek:

- Exchange Server 2007;
- SQL Server vNext (Katmai);
- Small Business Server vNext (Cougar);
- „Longhorn” Server R2.

Magyarázat az Exchange Server 2007-hez

Az Exchange Server 2007 – mint az első, kizárólag natív 64 biten támogatott Microsoft

szerveralkalmazás – némi külön magyarázatra szorul. Az első érdekesség vele kapcsolatban, hogy létezik belőle 32 bites változat – csak nem támogatott és kereskedelmi forgalomban sem kapható. De akkor mégis miért van 32 bites változat?

Ennek több oka is van: egyrészt jelenleg nincs olyan virtualizációs technológiánk, ami képes 64 bites operációs rendszerű virtuális gépet futtatni. A Virtual Server 2005 R2 létezik ugyan natív 64 bites verzióban, de ettől még az sem tud 64 bites virtuális gépet emulálni nekünk. Márpedig azok, akik ki szeretnék próbálni az Exchange Server 2007-et – vagy éppen a korábbi bétatesztetek –, szeretnek virtuális környezetben dolgozni vele, hogy felfedezhessék a rendszer képességeit. Éppen ezért amikor megjelent az Exchange Server 2007 beta 1, már volt is belőle 32 bites változat. Erre a problémára amúgy a Windows Virtualization ad majd megoldást, amint megjelenik a „Longhorn” Server után nem sokkal, ugyanis az már képes lesz futtatni 64 bites virtuális gépeket is.

Külön érdekesség, hogy a végleges Exchange Server 2007-ből is létezik 32 bites vál-

tozat, ami tesztelésre bármikor használható, csak éles környezetben nem lehet sem licencelni, sem használni, valamint ebből következően támogatást sem kapunk hozzá. De ennek mégis mi az oka? Az, hogy az Exchange Server 2007 valóban lényegesen jobban teljesít 64 bites környezetben, mint 32 biten, valamint a legtöbb nagy hasznát veszi a 4 gigabájtól több memóriának is – emiatt már a kezdetektől fogva erre fókuszáltak a fejlesztők és a tesztelők a Microsoftnál.

Mit hoz a jövő?

Minden Microsoft-szerverszoftverből lesz 64 bites változat a közeli jövőben, sőt 2008-tól már kizárólag 64 bites szerverszoftverek jelennek meg.

Az első, kizárólag 64 bites változatban elérhető szerverszoftver a decemberben elkészült Exchange Server 2007. A Windows Vista és a „Longhorn” Server pedig nem más, mint a Microsoft két utolsó, 32 bitesen is elérhető operációs rendszere.

A készülő „Longhorn” Serverre épülő új Small Business Server, valamint a „Longhorn” Server R2-es változata viszont

már csak 64 biten lesz elérhető. Várhatóan az SQL Server soron következő változata is csak 64 bites verzióban érkezik és a következő ISA Server is.

Szerencsére jól látható, hogy a Microsoft azokból a szoftverekből, amelyeknél nem mindenki számára egyértelmű a 64 bit használata, elkészíti a 32 bites változatot is. Azoknál a szoftvereknél azonban, amelyek egyértelműen és lényegesen gyorsabbak 64 biten, a jövőben csak 64 bites verziókat szállít és támogat majd (ilyen például az Exchange Server 2007).

Ez a hozzáállás azonban csak időszakos, mint minden generációváltáskor, és csak addig tart, amíg a legtöbben végre átállnak 64 bitre. A trend megállíthatatlan, ugyanis a szerverszoftverek adatbázisainak mérete és teljesítményigénye már egyre több helyen igényli ténylegesen a 64 bit adta előnyöket, valamint végre a drivergyártók és a kisebb szoftverfejlesztők is elkezdik komolyan venni a 64 bitre történő átállás és fejlesztés szükségességét.

Budai Péter

(i-pbudai@microsoft.com) Microsoft Magyarország

IT-BUSINESS PRODUCTS

INFOKOMMUNIKÁCIÓS TERMÉKEK ÜZLETI DÖNTÉSHOZÓKNAK

- informatikai döntéshozóknak és technológiai szakembereknek
- az elmúlt 168 óra legfontosabb hazai és külföldi, ICT-piaccal kapcsolatos termékbejelentések, hardvereszközök és termékújdonságok
- heti ingyenes elektronikus hírlevél

Regisztráljon!

www.it-business.hu/hirlevel



TÜZETES ELLENŐRZÉSEK

Spam- és vírusszűrés az Exchange Server 2007-ben.

A Microsoft utóbbi 4-5 évének történései nagyrészt a platformok és az alkalmazások biztonságának jegyében teltek. Ez világosan érezhető a 2006-os év triászán, azaz a Vista, az Office 2007 és az Exchange Server 2007 (a továbbiakban egyszerűen Exchange) együttesén is. A törekvéssel már a telepítéskor találkozunk, hiszen ezek a rendszerek és köztük is főként az Exchange – a biztonsági alrendszerrel együtt – olyan technológiákkal települnek, amelyek a legtöbb felhasználó és szervezet védelmi igényeit kapásból kielégítik.

Írásunkban az Exchange-nek azokat a biztonsági vonatkozásait tekintjük át, amelyek az internetes kellemetlenségek és fenyegetések, azaz a levélszemét és a kártevők elleni közvetlen védelmet szolgáltatják.

Réteges védelem

A kéréstlenlevél- és kártevőküldők számos technikát használnak küldeményeik célba juttatására, ezért elég nehéz lenne egyetlen olyan eszközt megjelölni, amelyet tökéletes védelmi megoldásnak lehetne kikiáltani. Az Exchange olyan, a korábbi változatok képességeire épülő védelmet használ, amelyben a biztonsági technológiák többszintű védelmet biztosítanak, és amelyben a spam- és a vírusvédelem szorosan kapcsolódik egymáshoz. Ennek közismert oka: a rosszindulatú kódok terjedése közvetlenül összefügg a levélszeméttel, és ha utóbbit sikerül eliminálni, akkor az előbbi veszélye is megszűnik, de legalábbis csökken.

Minél korábban, azaz a szervezet hálózatba lépési pontjához minél közelebb történik a szűrés, annál kisebb az infrastruktúra belső erőforrásainak a sávszélesség- és erőforrás-terhelése, ezért a spam- és vírusvédelmi funkciók alapbeállítás szerint az Edge Transport szerepkörbe vannak beágyazva. Ettől függetlenül az alább ismertetett folyamatokat a Hub Transport szerepkörön is engedélyezhetjük kézi konfigurálással (PowerShelllel kell elvégezni ezt a műveletet, az install-AntispamAgents.msh scripttel), de általában érdemesebb ezeket a képességeket az Edge-en megvalósítani, elvégre az teljesen elszeparáltan helyezkedik el még a címtárunktól is.

A réteges védelem az Exchange esetében azt jelenti, hogy a beérkező küldeményeket vizsgáló technológiák meghatározott sorrendben végzik a dolgukat; minden szűrőmódszer az üzenetek meghatározott tulajdonságait vizsgálja az 1. ábrán bemutatott struktúrának megfelelő



1. ábra. A spam- és vírusszűrés egymásra épülő rendszere az Exchange 2007 rendszerben

módon. A továbbiakban a séma különböző részfeladatait tekintjük át.

Kapcsolatszűrés

Az Exchange-ben a kapcsolatszűrő a 2. ábrán bemutatott módon vizsgálja az SMTP-szerver által kezdeményezett kapcsolatot a következő feltételek szerint:

1. A szűrőágens megvizsgálja, hogy az SMTP-szerver címe rajta van-e az adminisztrátorok által definiált IP-engedélyezési listán. Ha igen, akkor az üzenet továbbmegy a feladatszűrőhöz.

2. Ha az SMTP-szerver címe rajta van a tiltandó IP-címek listáján, akkor az üzenetnek elutasítás a sorsa, és nincs további szűrés.

3. A kapcsolatszűrő megnézi, hogy külső szolgáltatótól származó engedélyezési listán szerepel-e a küldő IP-címe. Ha igen, akkor az üzenet továbbmegy a feladatszűrőhöz.

4. Ha a küldő címe szerepel az ugyancsak külső szolgáltatótól származó RBL-en (valós idejű blokkolólista), akkor a küldemény blokkolódik, és nincs további szűrés.

Nem mellékesen a küldő IP-címét a rendszer két helyen is figyeli: az egyik forrás maga az engedélyezett kapcsolat, a másik pedig a levelek fejléce, de ezt már a feladó szerinti szűrést végző egység figyeli.

Feladó

A kapcsolatszűrés után a 3. ábra szerint történik a feladó e-mail-címének vizsgálata. Ez egyszerűen annak ellenőrzését jelenti, hogy a levél fejlécében szereplő küldési cím szerepel-e az adminisztrátorok által karbantartott blokkolandók listáján. Ha igen, akkor a küldemény protokollszinten blokkolódik, és nincs további szűrés.

Ha a feladó a felhasználók Outlookjaiban szerepel is a biztonságos küldők listáján, de a feladó szerinti szűrés során „könnyűnek találtatott”, akkor természetesen blokkolódik.

Címzett szerinti szűrés

A 4. ábra címzett szerinti szűrést ábrázoló sé májából nem tűnik ki, de ezt a részfeladatot megelőzi egy ismételt kapcsolatszűrés.

A címzett szerint szűrő ágens azt vizsgálja, hogy a küldemény címzettje szerepel-e a blokkolandó fogadók listáján. Pozitív esetben a kézbesítés leáll, és nincs további szűrés sem. Ugyanez történik, ha a rendszerben nem létezik a címzett e-mail-címe. Ha a levélnek több

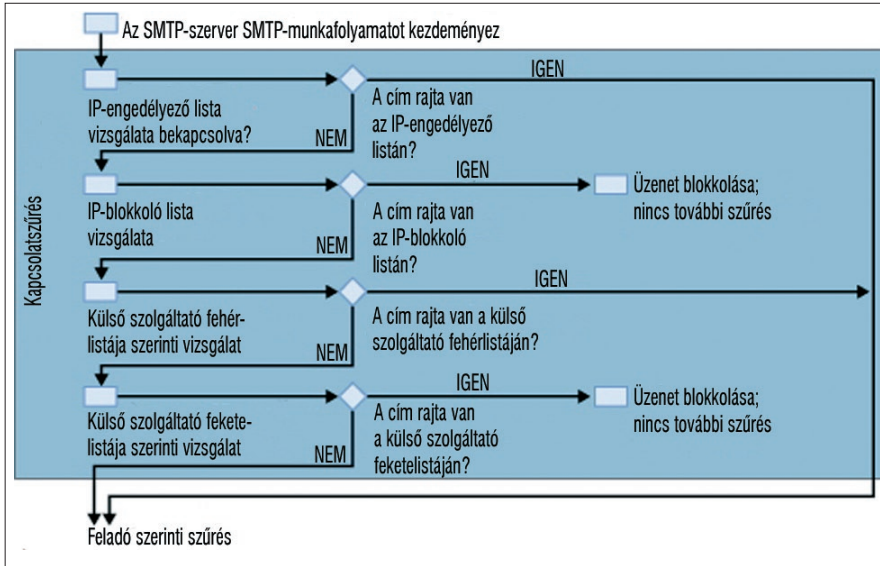
címzettje is van, akkor a küldemény csak akkor blokkolódik, ha egyik cím sem létezik, vagy az összes fogadó rajta van a blokkolandók listájában. Egyébként csak a blokkoltak felé

nél – komoly szempont az elutasított küldemények mennyisége.

A „Sender ID” vagy „küldőazonosító” keretrendszerrel az 5. ábrán bemutatott mó-

IP-címeket, amelyek az adott tartományból leveleket jogosultak feladni. Ha a megcímezett rendszer lekérdezi az SPF-bejegyzést, a visszatérő „oké” jelzést úgy kell értelmezni, hogy a kérdéses küldemény nagy valószínűséggel legitim feladótól származik. Természetesen az Exchange a speciális esetekre – például DNS-problémákra – is felkészíthető.

A küldőazonosító ágens először egy speciális, az RFC 4407-es szabványban leírt algoritmussal meghatározza a feladó SMTP-címét, ami mondjuk így néz ki: „valaki@pelda.hu”. Ezután végrehajt a cím doménszegmensére – „pelda.hu” – egy DNS-lekérdezést (lookup). Ha az adott internetes tartomány közöttett egy SPF-bejegyzést, akkor az ágens erről kap egy pozitív eredményt, amit „rábélyegez” a küldeményre.

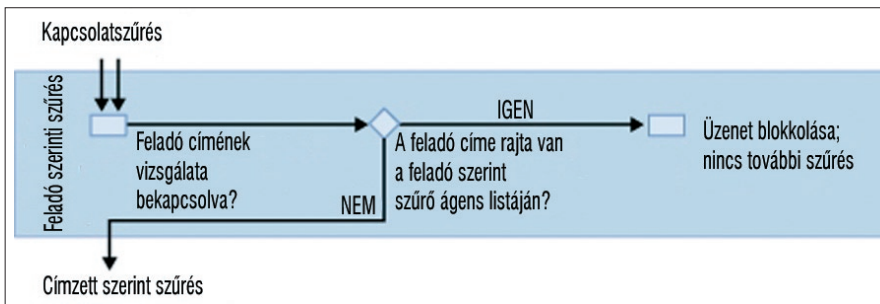


2. ábra. A kapcsolatszűrés

áll le a továbbítás olyan módon, hogy a szűrő-ágens ezeket a fogadókat törli a fejlécből.

Itt jegyezzük meg, hogy az elutasított küldeményekről a feladók rendszerei értesítést

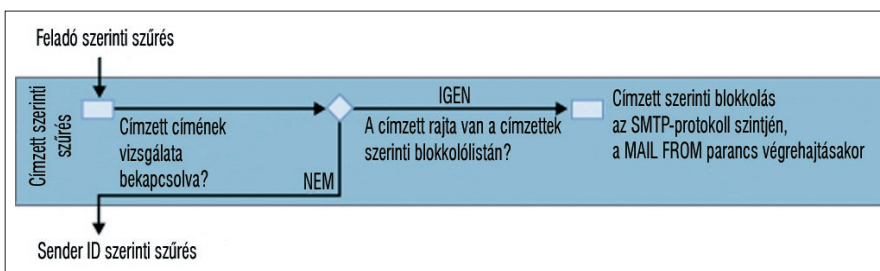
don annak a vizsgálatra megy végbe, hogy a küldemények olyan internetes doménről érkeznek-e, amelynek IP-cím tulajdonságaihoz illeszkedik a levél küldőjének az IP-címe. A



3. ábra. Feladó szerinti szűrés

kapnak. Ugyancsak itt fontos szólni a „küldő reputációjáról”, amely egy diszkrét számmal jelölt paraméter, és a kiszámításánál – azaz a küldő megbízhatóságának számszerűsítésé-

Sender ID nem más, mint a küldő nyilvános DNS-kiszolgálójának SPF-bejegyzése (Sender Policy Framework, körülbelül: „küldő szabályrendszer”). Az SPF tartalmazza azokat az



4. ábra. Címzett szerinti szűrés

Peremszinkron

Az Exchange védelmi képességei olyan transzport-szintű ágenseket (is) alkalmazhatnak, amelyek az Exchange Management Shell segítségével felügyelhetők és szkriptelhetők.

A felügyeletet segíti még az EdgeSync, amely szervesen kapcsolódik a transzportágensekhez, pontosabban a beállításaihoz és azok érvényesítéséhez: a szolgáltatóval könnyen szinkronizálhatók az ágensek konfigurációs információi azokon a kiszolgálókon, amelyeken az Edge Transport szerver szerepkört telepítették.

Az EdgeSync olyan folyamatok gyűjteménye, amelyek a címezett- és beállítási adatokat replikálják egyirányú üzemmódban az Active Directory felől az ETS alkalmazás módú (ADAM) címírára felé. Az EdgeSync csak azokat az adatokat másolja át, amelyek az ETS spamszűrő és üzenetbiztonsági konfigurációs feladataihoz kellene, illetve a küldemények kifelé – az internet felé – irányuló zavartalan forgalmazásához szükségesek egy vagy több ETS-en keresztül. Az EdgeSync szolgáltatás időzített frissítései gondoskodnak róla, hogy az ADAM beállításai a teljes organizációban frissítve legyenek.

Ha az adott domén nem tett közzé SPF-bejegyzést, akkor is lebélyezi, de egy „semmi” pecséttel.

- A küldemények további lehetséges sorsa:
- Ha a küldőazonosító elutasításra van beállítva, akkor a levél blokkolódik, és erről a küldő SMTP-szervere is értesül.
 - Ha a beállított válaszlépés a törlés, akkor

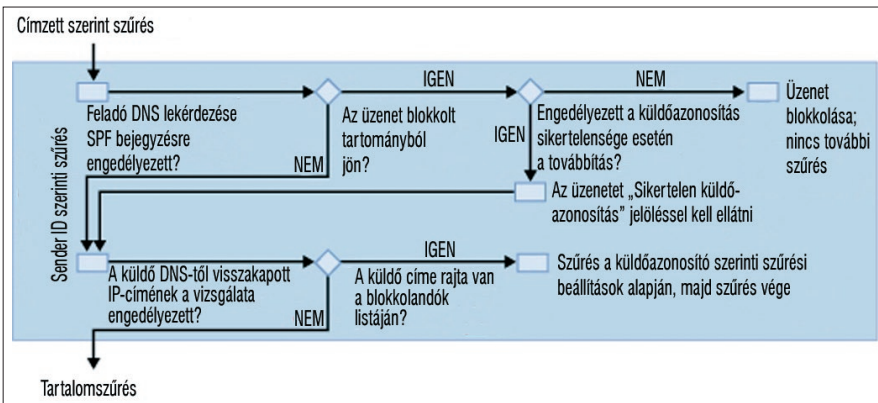
az Exchange törli a levelet és nem informálja a küldőt – pontosabban küld neki egy hamis „oké” jelzést, és szívfájdalom nélkül töröl, hogy a küldő szerver ne bajlódjon az ismételt küldésekkel.

zö üzeneteket valószínűségi paraméterekkel látja el annak megfelelően, hogy a vizsgálat alapján mennyire látszanak elfogadható vagy káros tartalmat hordozó küldeménynek, illetve spamnek.

- nincs küldés;
- az üzenetet fogadja a szerver, de a továbbítás helyett nemes egyszerűséggel törli.

Az említett „spam-karantén” egy ideiglenes szervezeti tár a levélszemétként azonosított küldeményeknek. A funkció a tartalomszűrés szerves részeként van implementálva. Az ide kerülő levelek természetesen nem jutnak el a felhasználóhoz, de az adminisztrátorok hozzájuk férnek, és törölhetik őket, illetve a tévesen spamként azonosított – fals pozitív – küldeményeket továbbíthatják a felhasználóhoz.

Maga a tulajdonság kétlépcsős mechanizmussal működik. Az első lépcsőben a rendszergazdák kapcsolódnak a periméteren elhelyezkedő spam-karanténhoz. Az elérés egyszerűen az Outlookkal történik. Az üzeneteket vizsgálat után egyszerűbb esetben tovább lehet küldeni a címzetthez, vagy ki lehet törölni. A második lépcső az adminisztrátorok által kérdésesnek minősített SCL-értékű, azaz „határeset-küldemények” szintje. Ezek először a járulékos védelem okán szöveges formátumra konvertálódnak, majd a felhasználói levélszemét-fiókba továbbítódnak – hadd győzködjön velük ő maga, pontosabban a kliensprogramja.

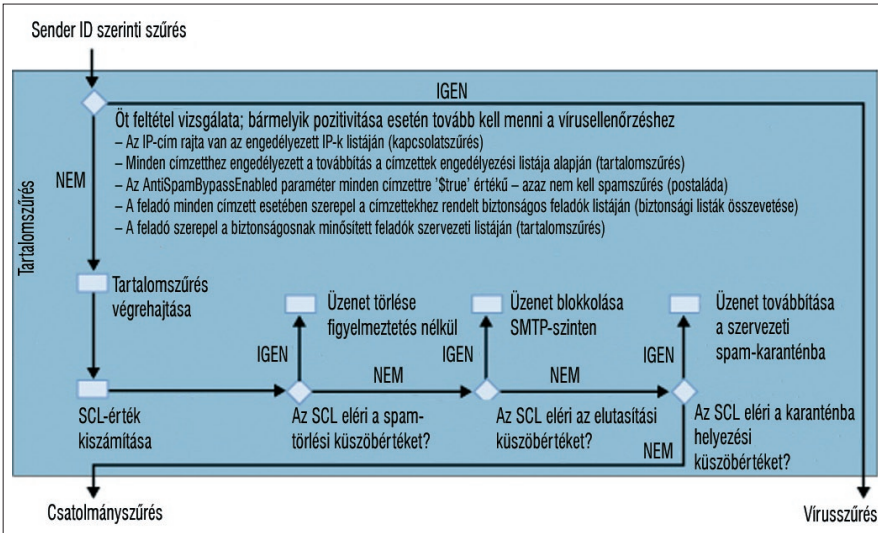


5. ábra. Szűrés küldőazonosító szerint

- Ha a küldőazonosító szerint szűrő ágens továbbengedésre van programozva, akkor az ágens a vizsgálat eredményéről készült pecséttel lebélyegzi a küldeményt, és annak tovább folyik a vizsgálata. Ez az eredmény a továbbiakban a küldő megbizha-

A tartalomszűrés fontos előszobája az ábra öt feltételének ellenőrzése. Ezek bármelyikének pozitívítása esetén a levél egyenesen a víruskeresőhöz kerül.

Az elemzéshez az EIMF összesíti a kapcsolatot, küldő-, fogadó-, reputációs, Sender ID



6. ábra. Tartalomszűrés

tóságának és a következő szakaszban részletezett SCL-értéknek a kiszámításában is fontos faktor lesz.

Tartalomszűrés

A 6. ábra szerint működő folyamat alapja a SmartScreen „önképző” eljárásal működő Exchange Intelligent Message Filter (EIMF) technológia. Ez a szűrőmódszer a beérke-

ellenőrző adatokat is – sőt, figyelembe veszi az Outlook 2007 bélyeghitelesítési információit –, és ezek alapján határozza meg az adott üzenet megbízhatósági szintjét (spam confidence level, SCL). Az SCL különböző szintjeihez további akciók rendelhetők:

- kézbesítés Outlook-postafiókba vagy a felhasználó levélszemétgyűjtőjébe;
- küldés levélszemét-karanténba;

A csatolmányok ellenőrzése

A 7. ábra szerinti csatolmányvizsgálat több szempont szerint – így például MIME-tartalomtípus, fájlnev vagy kiterjesztés, méret – történhet.

Az eredménytől és a beállításoktól függően a következők történhetnek az üzenettel:

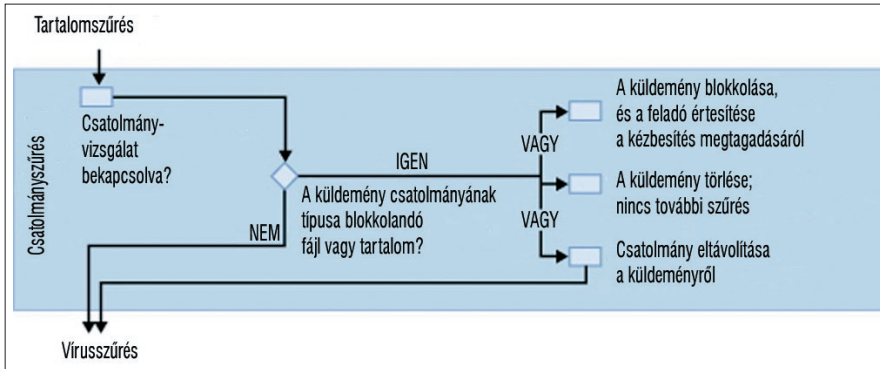
- Ha elutasításra van programozva az ágens, akkor sem a levél, sem a csatolmánya nem kézbesítődik, és a küldő kap egy DNS-hibáról szóló értesítést, amely egyébként tetzés szerint alakítható.
- A „csendes törlésnél” is blokkolódik a küldemény, de a küldő rendszer nem kap értesítést.
- A „lehámozásnál” az e-mail a csatolmánytól megfosztva továbbítódik úgy, hogy a csatolmány eltávolításáról szóló megjegyzéssel egészül ki a szöveges tartalma.

Többszintű vírusvédelem

Az Exchange 8. ábra szerint működő vírusvédelméhez több újítás is kapcsolódik. A korábbi változatok víruskereső programozására szolgáló felülete (VSAPI) itt is megvan,

de a Microsoft – szándékai szerint hatékonyabb és ugyancsak programozható – pártázóeszközöket valósított meg transzportszinten is. Érdeemes kiemelni, hogy a Microsoft

Tartalmazza annak eredményét, és így kiküszöböli a többszörös – és felesleges – vírusellenőrzéseket, illetve jelzi, ha további kontrollra van szükség.

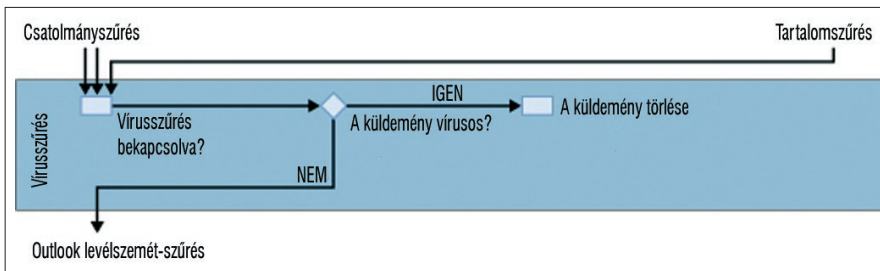


7. ábra. Csatolmányok szűrése

ForeFront vírusellenőrző megoldások képesek futni a Mailbox szerepkörű szervereken is, ott a mailbox tartalmát vizsgálják időről időre a VSAPI-n keresztül.

A ForeFront szoftverek és az Exchange 2007 azonban most már sokkal inkább részé-

Az Exchange transzportszintű ügynökei meglehetősen hasonlítanak a korábbi változatok event sink-jeire. A transzportágensek segítségével külső fejlesztők is írhatnak olyan komponenseket, amelyekkel további, trükkös kórokozók ellen védő megoldásokat le-

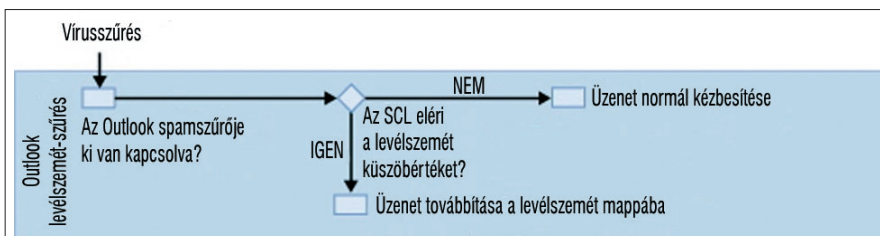


8. ábra. Kórokozók vizsgálata

vé válik a tényleges levélforgalom vírusszűrésének is. Ennek letéteményesei a Magazin más cikkében tárgyalt transzportágensek, melyek az alkalmazások eseményeihez beállított válaszlépéseket fogyanatosítanak.

Az Exchange-ben megjelent az „antivírus-

het implementálni a rendszerbe, vagy akár a teljes levélforgalmat is lehet ezekkel vizsgálni és manipulálni. Ezeknek a használatát az Exchange stabil és az évek során meglehetősen élesre – és megbízhatóra – kőszőrült MIME-kezelő motorja segíti.



9. ábra. Az Outlook levélszemét-kezelése

peccsét” is. Ez egy olyan, az üzenetekre kerülő jelzés, amely a szervezeten belül igazolja a már lezajlott kórokozó-ellenőrzést.

Itt utalunk még az előző számban tárgyalt Forefront Security for Exchange Serverre, amely a beépített vírusvédelem kiegészítője-

ként további robusztus kártevővédelmet kínál az Exchange-rendszerekhez – ez is itt kapcsolódik be ugyanis a folyamatba.

Végállomás

Ezen a szinten a küldemények a szervezet levelezőrendszerében a rendeltetési helyükhöz érnek. Itt már csak a címzettek postafiókjai-ban van hátra némi munka.

Az Exchange teljes rendszerének szerves részét képező Outlook-kliensek az Exchange

Néhány további mozzanat

Szignatúrák. Az Exchange járulékos lehetőségei gondoskodnak a spamszűrők aktualizálásáról:

- az Exchange Standard Anti-spam Filter Update kérhető,
- a Forefront Security for Exchange 2007 24 óránként frissít.

EdgeSync. A címzett szerinti szűrésnél a lapunk más írásában tárgyalt EdgeSync gondoskodik a címzett szerint szűrő ágenseknek szóló adatok ADAM-szintű aktualizálásáról az Edge Transport Server(ek)en. A transzportszintű ügynökprogramok egyszerűen le tudják kérdezni, hogy a beérkező küldeménynek van-e valós vagy legális címzettje a szervezetben.

Reputáció. A küldő reputációjának, azaz a feladó elfogadhatóságának az adatai nem kőbe vésett paraméterek, hanem több információ – SMTP-munkafolyamatok, tartalomszűrés, Sender ID-ellenőrzés és a feladó általános viselkedésére és „előéletére” jellemző profilok elemzéséből származnak. A reputációs kalkuláció ezekből dinamikusan számítja ki, hogy „fekete-” vagy „fehérlistára” tegye-e a feladót, és blokkolja-e a küldeményeit.

IP-reputáció. Ez a vadonatúj technológia, a küldő IP-címe elfogadhatóságának vagy megbízhatóságának elemzését és a kapott számszerű értéket használata jelent. Ilyen jelenleg csak az Exchange-ben létezik, és a többi valós idejű blokkolólehetőséghez hasonlóan ki-be kapcsolható.

Nyelvfüggetlenség. A spamvédelem további fontos jellemzője, hogy az alkalmazott biztonsági technológiák nagy része nyelvfüggetlen.

minősítésének – elsősorban az SCL-értéknek – a figyelembevételével dönti el, hogy értékes küldeményként a postaládába vagy spamsként a levélszemét-mappába sorolja be a küldeményeket.

Kelemen László
(kelemen@hungary.com)

POWERSHELL-SCRIPTEK A GYAKORLATBAN

Az előző részben megismerkedtünk a „fejjel a falnak” módszer hatékonyságával, és bizony elég jelentős haladást értünk el a PowerShell-rendszer megismerésében. Ebben a részben picit behatóbban elemezzük a CommandLeteket, valamint megismerkedünk az objektumok feldolgozásának kilenc fontos parancsával.

Az előző cikk óta volt szerencsém részt venni Barcelonában, a Tech•Ed: IT Forum konferencián a PowerShell atyjának, Jeffrey Snovernek az előadásán, aki azt mondta: a PowerShell-felhasználónak négy barátja van:

1. HELP
2. ALIAS
3. GET-MEMBER
4. GET-PSDRIVE

A rendszer felfedezése

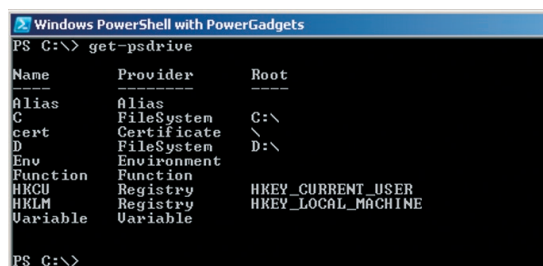
Míndez magyarul azt jelenti, hogy erre a cikkre sincs semmi szükség, minden benne van a helpben. Csak hát angolul. Jeffreynek viszont mégiscsak teljesen igaza van, mert ha valaki egy adott parancs részleteire kíváncsi, akkora helpet kap, hogy arról kódul. Külön ki kell emelni az -example parancsot, amivel minimum három kiváló példát kapunk az adott CmdLet használatára. Például ezt találjuk a HELP DIR -EXAMPLE outputjában:

```
get-childitem registry::hklm\software
```

Lám-lám, a DIR parancsral (ami melleleg csak egy alias a get-childitem CmdLetre) ki lehet listázni a registry tartalmát! Innentől beindul a fantáziánk: és még merre csatangolhatunk? A rendszergazda negyedik barátja, a GET-PSDRIVE (magyarul: kérem a PowerShell-meghajtótokat!) szerint még jó pár helyre vándorolhatunk a sima CD és DIR parancsokkal.

Változók

Itt vannak például a környezeti változók (ENV:). Milyen jó



```
Windows PowerShell with PowerGadgets
PS C:\> get-psdrive
Name      Provider      Root
-----
Alias     Alias
C         FileSystem    C:\
cert      Certificate    \
D         FileSystem    D:\
Env       Environment
Function  Function
HKCU     Registry      HKEY_CURRENT_USER
HKLM     Registry      HKEY_LOCAL_MACHINE
Variable  Variable
```

A PowerShell „meghajtókkal” bejárhatjuk az operációs rendszert

is lenne kihalászni ebből a listából mondjuk a számítógép nevét! Így a script pontosan tudná, hogy hol fut. Vagy esetleg a LOGONSERVERT értékét, mert akkor pontosan tudhatnánk, hogy melyik tartományvezérlő jelentkeztetett be minket. Hát lássuk!

```
dir env: | where {$_.Name -eq „COMPUTERNAME”}
```

Magyarázat: a dir env: (valójában get-childitem env:) előállítja az összes környezeti változó értékét egy .net objektumkupaiban (collection). Ezt a pipe (függőleges vonal, |) karakter segítségével „belecsövezük” egy lekérdezésbe, amelyik csak azokat a sorokat engedi át, ahol az aktuális objektum neve „COMPUTERNAME”. Így egyetlen érték marad, pontosan az, amelyikre vadászunk. A PowerShell mindösszesen kilenc objektumkezelő CmdLetet tartalmaz.

Ismerkedjünk meg velük!

Objektumkezelés

A kilenc parancs mindegyike valamilyen apró műtétet hajt végre az objektumkupaiban. Ezek közül jó pár az SQL-nyelv elemeire „hajaz” (SELECT * FROM TÁBLA), vagyis az objektumhalmazt horizontálisan (where-object) vagy vertikálisan (select-object) ritkítja. A teljes készlet a következő (HELP OBJECT):

ForEach-Object, Where-Object, Compare-Object, Measure-Object, Tee-Object, New-Object, Select-Object, Group-Object és Sort-Object. A help ebben a sorrendben adogolja őket, mi inkább fontossági sorrendben foglalkozunk velük!

ForEach-Object (ALIAS: foreach)

Az objektumkúpac elemenkénti feldolgozására való (kurzor). Végigszalad az összes elemén, és az utasítászárójelben {} megadott feladatot elvégzi rajtuk.

Ha ez e-mail-címek listája, akkor például minden címzettnek egyedi levelet küldhetünk a segítségével. Az előző cikkben (egyéb-ként téves algoritmussal) körök kerületét számoltuk ki vele, ezért erre most külön példát nem érdemes hozni.

Select-Object (ALIAS: select)

Amikor valaki lekérdez néhány objektumot (például: DIR), talán nem is tudja, hogy mennyi, de mennyi adattal rendelkezik az adott objektum! A képernyőkép (output) csálóka: van néhány olyan tulajdonság, amelyik megjelenik, a többi rejtve, settenkedve nyomakodik át az objektumfeldolgozási csövön. Tehát egy sima parancs a „SELECT *” brutalitásának felel meg: ide nekem az oroslánt is! Ez még egy-két objektumnál nem jelent gondot, de pár tizezernél már igen. Van ekkora objektumlista? Persze, például az eseménynapló! Csak hasonlítsuk össze az alábbi két parancs kimenetét:

```
dir | get-member
```

és

```
dir | select name | get-member
```

A get-member – mint a bevezetőben láttuk – a rendszergazda egyik legjobb barátja. Arra való, hogy megmutassa, egy adott objektum mit tud, milyen adatokkal és metódusokkal rendelkezik.

Nos, a sima DIR legalább huszonharminc tulajdonságot tárol minden fájlról és könyvtárról. A második, szűrt változat viszont csak egyet. Hát szűrjünk!

(Egészen pontosan 28 property lesz a szűrőmentes lista minden elemén. Kis előtekintés, így számoltuk meg: dir | get-member | where

```
Windows PowerShell with PowerGadgets
PS C:\> compare-object (get-content boot.ini) (get-content autoexec.bat)

InputObject                                     SideIndicator
-----
Echo off                                         =>
[boot loader]                                   <=
timeout=30                                      <=
default=multi(0)disk(0)rdisk(0)parti...       <=
[operating systems]                             <=
multi(0)disk(0)rdisk(0)partition(1)\...      <=
```

Fájlok tartalmának összehasonlítása compare-objecttel

```
{$_ .membertype -eq „property”} | measure-object -property membertype. Türelem, még két oldal, és ez az egész parancs érthető lesz...)
```

Where-Object (ALIAS: where)

Sorok ritkítása. Egy get-process vagy egy get-eventlog az összes processzt, illetve az eseménynapló mind a százezer elemét lazán az objektumkúpacba tolja, csak győzzük feldolgozni. A where-object abban segít, hogy ne az menjen át a következő feldolgozóegységbe, ami a csövön (pipe) kifér, hanem csak azok az objektumok, amelyek valóban feldolgozásra kerülnek. Például a 20 megabájtnál nagyobb memória-munkaterületet (working set) használó folyamatok:

```
get-process | where {$_.WorkingSet -gt 20MB}
```

Ez ennyi, nem több. A Where utasítászárójelében lévő feltétel magyarázata: a dollár speciális karakter, amellyel változókat jelölünk (mint ahogy például SQL-ben a kukaccal: @ változó). A dollár-aláhúzás pedig egy speciális változó, az éppen aktuális, feldolgozás alatt álló objektumot jelöli. A -gt pedig nem Gál Tamás monogramja (bár az is lehetne), hanem Greater Than, azaz nagyobb, mint. Miért nem úgy jelölik, ahogy „normális” nyelvekben szokás (>)? Azért, mert egy parancssorban állunk, ahol a kacsacsőrök (< és >) teljesen mást, nevezetesen átirányítást jelentenek. Ez a furcsa „műveleti jel” a gordiuszi csomó átvágásával egyenértékű professzionális megoldás.

Compare-Object

Tárgyak és tárgylisták összehasonlítása. A PowerShell objektumorientált megközelítése miatt ez a parancs bármilyen objektumtípus összehasonlítására alkalmas: fájlok tartalmára éppúgy, mint processzlistákra és registry-kulcsokra. Vessük össze például a boot.ini és az autoexec.bat tartalmát! Az alábbi példa kilistázza a két fájl közötti eltérést.

Az eredmény önmagáért beszél, bal oldalon látjuk az eltérő sorokat, jobbra pedig a kis nyílcska azt jelzi, hogy ez a bal vagy a jobb oldalon megadott fájlban van-e. Az @ echo off például a jobb oldaliban, tehát az autoexec.bat-ban van.

Érdekes elemzésre adhat lehetőséget, ha a compare-objectet előtte-utána üzemmódban használjuk. Egy hosszabb folyamat előtt is készítünk egy pillanatfelvételt a rendszerrel (például processzlista, könyvtárlista stb.), meg utána is, a compare-object pedig kimutatja a változást.

Ehhez a listatartalmakat változóba kell gyömöszölni, ami nem bonyolult dolog: kitálunk egy változónevet (dollárral az elején), és elkezdjük használni. Íme:

```
$Setotte=get-process
```

Most jön egy hosszabb folyamat...

```
$Sutana= get-process
compare-obejct $Setotte $Sutana
```

Ez már gyakorlatilag script, hisz egynél több sorból áll. Mentsük el .PS1 kiterjesztéssel, és valóban PS Scriptet kapunk.

Measure-Object

Ez a parancs összegzéseket végez az objektumok általunk megadott propertyjein. Ha szövegfájltra engedjük rá, megszámlolja a szavakat, sorokat stb. Ha objektumkúpacra, akkor pedig bármit megszámlol, összegez, átlagol stb. Nézzünk először példát a szövegfájlkezelésre:

```
get-content boot.ini | measure-object -line -word
```

A fenti parancs összeszámlolja a boot.ini sorait és szavait. Most számoljuk meg, mennyi helyet foglalnak a merevlemezen a PPT-k:

```
dir -r -filter *.ppt | measure-object -property Length
-sum
```

Látható, a fájlok darabszámát önszorgalomból megadta (count), az összegzésen felül pedig minimum-, maximum- és átlagszámításra képes.

Tee-Object

Ez a parancs a csövezeték megcsapolására való. Amikor „összecsövezünk” egy pár parancsot (régipélda: DIR | SORT | MORE), az adatok csak a cső végén pottyannak ki. A DOS alkalmazásokor menet közben nem fértünk hozzájuk, és ennek megfelelően nem tudtunk például részeredményeket fájlba menteni.

```
Windows PowerShell with PowerGadgets
PS C:\> get-eventlog application | group -property source -noelement | sort -property count -desc
```

Count	Name
4796	.NET Runtime Optimizat...
2172	MSSQLSERVER
1373	MSSQL\$SQLEXPRESS
197	LoadPerf
192	MsIInstaller
133	Gi
84	SQLISService

A legszószátyárabb alkalmazás díját a .net runtime kapja

```
Windows PowerShell with PowerGadgets
PS C:\> dir -r -filter *.ppt | measure-object -property Length -sum
```

```
Count       : 22
Average     :
Sum         : 9694720
Maximum     :
Minimum     :
Property    : Length
```

Számoljuk ki, mennyit esznek a PPT-k!

A tee-object megoldja ezt a gondot. Csak bele kell tenni a csőbe, és „ereszt”, mégpedig fájlba vagy változóba.

Group-Object (ALIAS: group) és Sort-Object (ALIAS: sort)

Objektumok csoportosítása és sorba rendezése valamelyik tulajdonságuk alapján. Például szá-

moljuk meg az application logban, hogy melyik alkalmazás hány bejegyzést tett, és ebből készítsünk egy sorba rendezett listát. Ehhez az eseménynapló bejegyzéseit a Source (forrás) tulajdonság alapján csoportosítjuk, majd a kész listát a Count alapján csökkenő sorrendbe tesszük. A -noelement arra szolgál, hogy dobálja el a konkrét eventlog-bejegyzéseket a feldolgozás során, mert arra nem lesz szükség.

New-Object

A következő cikk témája a COM, WMI és .NET-objektumok kezelése lesz, ezekhez a new-object parancsra lesz szükségünk.

Fóti Marcell

MCSE, MCDBA, MVP, MZ/X, MCT since 1995

(marcellf@netacademia.net)

**TRÓJAI FALOVAK
VISSZAFORDÍTVA
A KAPUNÁL**

MICROSOFT.HU/BIZTONSAG
Microsoft

Ismerje meg az üzleti biztonsági alkalmazások legújabb generációját, a betolakodók elleni küzdelem hatékony fegyverét, a microsoft.hu/biztonsag oldalon!

- A Microsoft Forefront**
 A Microsoft Forefronttal olyan üzleti biztonsági termékcsaládhoz juthat, amely a korábbiaknál átfogóbb, magasabb fokú védelmet biztosít, és tágabb szabályozási lehetőségeket kínál. Az ügyfélgépek, a kiszolgálói alkalmazások és a hálózat pereme számára egyaránt képes védelmet nyújtani.
- Teljes körű szolgáltatás**
 A Microsoft Forefront a teljes operációs rendszerre, minden alkalmazásra és kiszolgálóra kiterjedő védelmet és hozzáférés-szabályozást biztosít az Ön információi számára, így azok biztonságban lehetnek a folyamatosan változó fenyegetésektől.
- Integrált**
 Több területen fokozhatja a hálózata biztonsága feletti ellenőrzést, mivel a termékcsalád biztonsági képességeinek integrálása a Microsoft kiszolgálói alkalmazásaival és a meglévő informatikai infrastruktúrával jóval nagyobb hatékonyságot nyújt.
- Egyszerű**
 A biztonsági termékek felügyeletének, telepítésének és használatának egyszerűbbé tétele nagyban hozzájárul a szervezet biztonságának növeléséhez, és így Ön is egyszerűen bizonyosodhat meg arról, hogy folyamatosan a megfelelő védelemben részesül.

© 2006 Microsoft Corporation. Minden jog fenntartva. A Microsoft, az Antigen és a Windows Server logó a Microsoft Corporation bejegyzett védjegyei az Egyesült Államokban és/vagy más országokban.

Microsoft

KÉSZEN KAPOTT CRM

Kinek és mire jó a Microsoft Dynamics CRM, és hogyan mérhető az üzleti teljesítmény az új, kiszolgáló alapú mutatószámrendszer és teljesítménymutató alkalmazás, a Business Scorecard Manager 2005 használatával?

Sokan kérdezték tőlünk korábbi, Office témájú TechNet-eseményeken, hogy milyen segítség áll rendelkezésre saját CRM-alkalmazások készítésére az Office platformra. Létezik ugyan válasz erre is, de ismerte a fejlesztési projektek költségeit és kockázatait, elsőként érdemes megnézni, van-e már eleve olyan kész szoftver, ami megfelel igényeinknek.

A Microsoft Dynamics termékcsaládja olyan informatikai megoldásokat foglal magában, amelyek a vállalatok mindennapi üzletmenetét, működését teszik hatékonyabbá. Segítségükkel az informatika és maga az informatikus is közvetlenül tud részt venni abban, hogy eredményesebbé tegye az érintett céget, nemcsak azáltal, hogy infrastruktúrát üzemeltet, hanem azáltal is, hogy a döntéshozók és az értékesítők munkáját támogatja.

A Dynamics szoftverek hasonlóképpen állíthatók üzembe, mint bármely másik Microsoft-szerveralkalmazás vagy -megoldás. A legtöbb képességet a rendszerbevezetéssel foglalkozó informatikai szakemberek a megfelelő dokumentáció átböngészésével könnyen bekonfigurálhatják, hogy a szoftver tényleg az adott vállalat igényeinek feleljen meg. Ha mégis hiányzik valamilyen képesség, azt pedig könnyen hozzá lehet fejleszteni – bár ilyenkor érdemes lehet elsőként a rendelkezésre álló partnermegoldásokat megvizsgálni.

A Dynamics CRM a termékcsalád legkisebb tagja, ezt kifejezetten kisebb vállalatok teljeskörű kiszolgálására vagy nagyobb vállalatok front office jellegű teendőinek ellátására találták ki. Az 5–10 felhasználóval dolgozó kisvállalatok ezt a terméket készen, dobozosan, előrekonfiguráltan érhetik el, a nagyobb vállalatoknak (akár 3000 felhasználóval is) viszont érdemes testre szabniuk.

Ügyfélinformáció: csak a fejekben?

A CRM lényege egyébként, hogy az ügyfelekkel kapcsolatos elektronikus interakciókat és tranzakciókat is a mindennapi munka szerves részévé tegye, és olyan kényelemmel tudjunk ezekkel az új, elektronikus eszközökkel is bánni, mint ahogyan hagyományos, papír alapú leveleinkkel és névjegyeinkkel tennénk. Emiatt már egyre több cég veszi igénybe ügyfélkapcsolatainak, ügyfélszolgálatának, illetve az ügyfelekkel kapcsolatban álló munkatársak munkájának összehangolására.

A cégek egy része már eddig is sokat tett annak érdekében, hogy ügyfeleit kellő módon ki-

szolgálja, de még mindig nem tud eleget saját ügyfeleiről. Még ha meg is van a szükséges tudás a szervezetben belül, az sokszor csak a fejekben, szétszórtan található meg – vagyis az információ nem minden érdekelt számára hozzáférhető módon áll rendelkezésre. A Microsoft Dynamics CRM-alkalmazásának az a feladata, hogy a technológiát hívja segítségül e probléma orvosolására.

Felhasználási lehetőségek, folyamatok

Vállalati értékesítési folyamat automatizálása. Az érdeklődők, ügyfelek, kapcsolattartók nyilvántartása, megosztása és ripolthatósága révén gyorsabb értékesítési ciklusok és jól mérhető értékesítési hálózat működtetését teszi lehetővé. Lehetővé válik a potenciális lehetőségek felfedése, az értékesítési folyamatok testreszabása, valamint a hozzáférés az ügyfelek vásárlási és kapcsolatfelvételi előzményeihez – bármely böngészőből, online vagy kapcsolat nélkül.

Kampánymenedzsment és célcsoportok kezelése. Marketinglisták, kampányok, célcsoportok, vásárolt adatbázisok, de akár már a vállalat karácsonyi rendezvényére meghívottak listájának összeállítására is kiváló eszköz. Kampányok feladatainak kiosztása, azok követése, a kampányban szereplő résztvevők üzleti teljesítményének mérésével magát a kampányokat is pontosan mérhetővé teszi.

IT-ügyfélszolgálat: jobb kiszolgálás, kevesebb ráfordítás. A Microsoft CRM 3.0 segítségével az informatikusok célirányos információkkal láthatják el a felhasználókat, hogy az általuk használt alkalmazásokkal kapcsolatos problémáikra mielőbb megoldást kapjanak. A Microsoft CRM beépített tudásbázisában a kézikönyvek, a gyakran ismételt kérdések és a hibaelhárítási tippek azonnal elérhetőek, akár az egyes telephelyekért felelős munkatársak számára is.

A beérkező problémák rögzítésére, automatikus továbbítására és a várólisták kezelésére szolgáló funkciók rögtön az illetékesekhez irányítják a kéréseket. Az eszköz támogatást nyújt az ügyfélszolgálati vagy helpdesk-munkatársak ütemezéséhez és irányításához is, hogy az adott feladat elvégzésére alkalmas szakembert minél egyszerűbben meg lehetne találni.

Vállalatirányítási rendszerek előtét-alkalmazása. Az Outlook-integráció miatt számos

cég vállalatirányítási rendszereinek előtét-alkalmazásaként is használja a CRM-et, amely a BizTalk adapterek segítségével számos – akár Microsoft, akár más – gyártó vállalatirányítási rendszerével képes szinkronizálni az ügyfelek törzsadatait és az üzleti tranzakciókat.

A Dynamics CRM rövid technikai áttekintése

A Microsoft Dynamics CRM 3.0 olyan jól bevált technológiákra épül, mint a .Net keretrendszer és az SQL Server, valamint aktívan használja a lelke mélyén az Exchange Servert és az IIS-t is. Könnyen összeköthető a BizTalk Serverrel, és ha további képességeket szeretnénk implementálni hozzá, használhatjuk a Visual Studiót is erre a célra.

A CRM szorosan együttműködik az Outlookkal, valamint a Microsoft Office részét képező Word, Excel, PowerPoint és más alkalmazásokkal, így a felhasználó megszokott, kényelmes Outlook-környezetben véggezheti munkáját, integrálva az ügykezelési mappákat a levelezőrendszer kezelőfelületébe. Ez az Office-integráció azért nagyon fontos, mert a felhasználók ezáltal hamar megbarátkoznak a CRM-mel, ha már amúgy is ezeket az alkalmazásokat használják a mindennapi munkájukhoz.

Tudom, mit vettél tavaly nyáron!

A CRM részeként a 3.0-s verzió megjelenésével új üzletiintelligencia-szolgáltatásokhoz is hozzájuthatunk, ha segítségül hívjuk a Business Scorecard Managert is. Ezeknek az analitikus eszközöknek a megértéséhez vegyünk rögtön egy példát: egy, az ügyfeleit jól kezelő utazási iroda azt is meg tudja jósolni, idén mire vágyunk. Pihenésre, egzotikus útra? Amennyiben emlékszük, hol nyaraltunk tavaly, mely ajánlatokat néztük meg alapsabban, akkor ajánlatai bizonyára célba érnek nálunk.

A Microsoft Dynamics CRM analitikus eszközeinek célja az ügyfél jellemzőinek meghatározása. Megismerhetők a fogyasztási szokásai, a fogyasztási szokásaihoz kapcsolódó információigénye. A Microsoft Dynamics CRM a megértett ügyfelekkel való törődés mindennapjait automatizálja, akár értékesítésről, ügyfélszolgálatról, termékfejlesztésről vagy marketingaktivitásokról van szó.

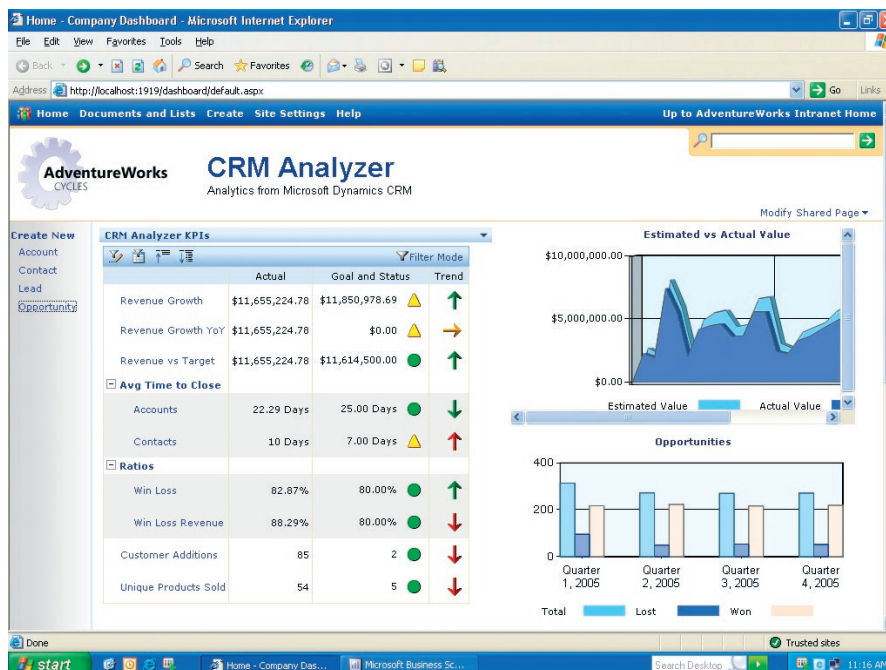
Nézzük meg szakmai szemmel, hogyan is működik ez a gyakorlatban!

Üzleti intelligencia a Microsoft Dynamics CRM-ben. Az üzletiintelligencia-rendszerekről, adatbányászatról sokat hallhatunk manapság, elsősorban az ERP-rendszerekkel (Dynamics NAV, AX) kapcsolatban. A Dynamics CRM mint a vállalat „kommunikációs központja” természetesen szintén alkalmas arra, hogy adatforrásként szolgáljon különféle üzleti elemzésekhez. Eddig – megfelelő eszköz hiányában – nem hoztuk összefüggésbe a CRM-alkalmazást a BI-megoldásokkal. A megfelelő eszköz – a Microsoft Office Business Scorecard Manager 2005 – azonban nemrégiben megjelent, új lehetőségeket nyitva mind az üzleti döntéshozóknak, mind pedig a CRM-bevezetést végző partnereknek.

Kulcsfontosságú információk döntéshozóknak. Az ügyviteli rendszerekhez hasonlóan a CRM is tartalmaz olyan mérhető üzleti adatokat, amelyeket felhasználhatunk

rú lehet itt is egy mutatószámrendszerekre és teljesítménymutatókra épülő megoldást használni, amely gyors, könnyen áttekinthető információkkal látja el az üzleti döntéshozókat, ugyanakkor lehetővé teszi a „lefúrást” a strukturálatlan adatok szintjéig. Ez az eszköz a Business Scorecard Manager.

A scorecardok használatához és készítéséhez szükséges első lépéseket legkönnyebben úgy sajátíthatjuk el, hogy letöltjük a www.microsoft.com/dynamics/crm/using/downloads oldalról a mintaadatbázist a scorecard template-ekkel együtt, és megfigyeljük azok működését. Az 1. ábrán látható egy, a CRM-ben tárolt üzleti lehetőségekből (opportunity) BSM segítségével készített árbevételi tény-terv összehasonlító jelentés. A képernyő bal oldalán a lehetőségek listánézet mellett láthatók a színes ikonokkal jelölt teljesítménymutatók (Key Performance Indicator, KPI), illetve az aktuális trendek.



1. ábra. A scorecardok használata

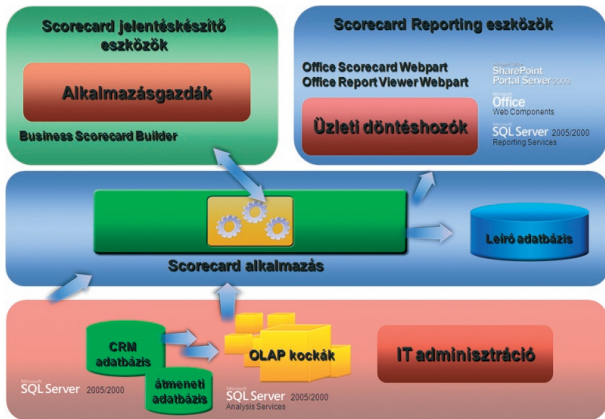
üzleti döntéstámogatáshoz, hiszen a vállalat kereskedelmi folyamatának jelentős részét a CRM-alkalmazásban követjük nyomon: javarészt itt készül az értékesítési ciklus dokumentálása, az ügyfelek visszajelzései pedig olyan információk, amelyek az ERP-rendszerekben esetleg egyáltalán nem is jelennek meg. A hagyományos jelentéskészítő eszközök (SQL Reporting Services, Excel Pivot-táblák) használata mellett ezért célsze-

jobb oldalon a két Excel Pivot Chart a lehetőségek időbeli eloszlását mutatja.

Ezekből a kimutatásokból jól látható, hogy mely lehetőségek alakulnak a tervek szerint és melyek nem. Amennyiben a részletekre is kíváncsiak vagyunk, a listaelemekre kattintva megnyithatjuk a szóban forgó CRM-rekordot, tehát nem kell azt egy másik alkalmazásban visszakeresni. Lehetőség van arra is, hogy az elemek SQL Reporting

Services- jelentésekre, kapcsolódó listákra vagy bármilyen más információs weboldalra hivatkozzanak.

Természetesen a hivatkozott jelentés származhat más rendszerekből – például a Dynamics AX vagy NAV adatbázisaiból – is.



2. ábra. Architektúra és technikai háttér

A 2. ábrán látható a BSM logikai architektúrája, illetve a rendszerelemekkel kapcsolatos üzleti és informatikai szerepkörök.

Az adatbázisrétegben kap helyet a CRM adatbázisa és szükség szerint egy átmeneti adatbázis, amely nem elengedhetetlen ugyan, de számos érv szól használatára mellett.

Miért szükséges az átmeneti adatbázis? Az alkalmazások – így a CRM is – általában a napi munkavégzés támogatására, az adatbevitelre és módosításra vannak optimalizálva, nem lekérdezésekre és számítások végzésére. Az összetett, kalkulációkat tartalmazó jelentések készítése erőforrás-igényes művelet, átmeneti adatbázis használata nélkül jelentősen csökkentheti a rendszer teljesítményét, azaz akadályozhatja a napi munkavégzést.

Az átmeneti adatbázisba történő ütemezett adatátvitel – amely történhet az SQL Server SSIS vagy a DTS technológiáinak használatával – biztosítja továbbá, hogy csak olyan üzleti információkból készüljenek jelentések, amelyek valóban értelmezhetők is; ne szerepeljenek az adatok között még nyitott ügyletek; például időbeli változások nyomán követése esetén nem lezárt időszakok. Ezzel a módszerrel kiszűrhetők a jelentéseket torzító adatok. Az átmeneti adatbázis szolgál az OLAP-kockák alapjául, tehát struktúráját már ennek a célnak megfelelően lehet kialakítani, azaz lekérdezésekre optimalizálni.

Az OLAP-kockák aggregált adatokat tartalmaznak, a jelentések készítésekor így jóval kisebb adathalmazból kell a rendszernek lekérdeznie és kalkulációkat végeznie. Az átmeneti adatbázis és az OLAP-kockák készítése, dimenziók (dimension), perspektívák (perspective) és mérőszámok (measure) meghatározása az adatbázis-adminisztrátorok és a fejlesztők feladata.

Az alkalmazásrétegben a CRM-webalkalmazás és a BSM scorecard-alkalmazás „végzi a munkáját”, az adminisztrációs feladatokhoz szükséges eszközök és a felhasználói felület már a megjelenítési rétegben kaptak helyet. A felhasználók – üzleti döntéshozók – SharePoint

Portal oldalakon érhetik el a scorecardokat. Amennyiben pivot-táblákat és chartokat is szeretnének használni, azt az Office-webösszetevők (Office Web Components, OWC) segítségével tehetjük meg.

Az alkalmazásgazdák egy jól áttekinthető, könnyen használható grafikus eszközzel – a Business Scorecard Builderrel – készíthetik el a jelentéseket.

Fontos újdonság, hogy a KPI-definíciók is ezen a felületen hozhatók létre, lényegesen egyszerűbben, mintha azt a hagyományos módon, SQL Server 2005 MDX (Multi Dimensional eXpression) lekérdezések írásával tennék meg, utóbbihoz ugyanis fejlesztői ismeretek és eszközök (Visual Studio) szükségesek, SQL 2000-ben pedig még nem is létezett a KPI-k létrehozásához szükséges felület.

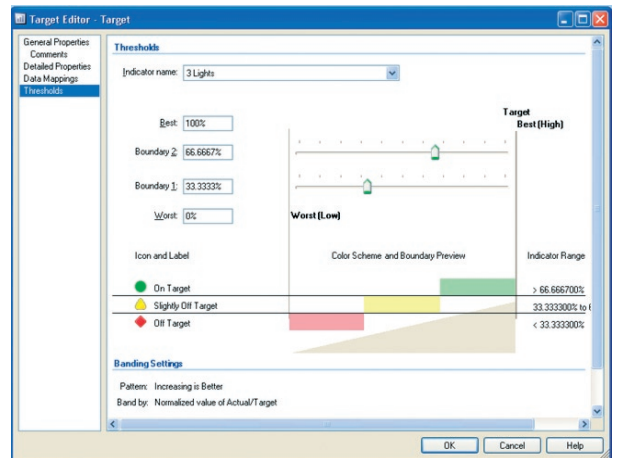
Példa az indikátorok meghatározására

Nézzük meg egy egyszerű példán keresztül, hogyan definiálhatjuk a teljesítménymutatókat a BSM-en keresztül. A vizsgált üzletilehetőség-elemzést (mintaadatok, OLAP-kockák,

scorecardok) a letölthető template tartalmazza. A 3. ábrán a Business Scorecard Builder Target Editor látható – itt határozzuk meg a terv-tény adatok összehasonlítására szolgáló indikátorok viselkedését. Esetünkben a tényadat (actual) a valós bevétel (actual revenue), a tervadat (target) a tervezett bevétel (estimated revenue). A Target Editorban a terv-tény teljesítések százalékos értékét három szakaszra osztjuk, majd meghatározzuk, hogy az egyes szakaszok milyen üzleti jelentést hordoznak:

- jó eredmény (On Target): jelen esetben a terv 66–100 százalékig történő teljesítése, ezt a könnyebb áttekinthetőség érdekében zöld színű ikonnal jelöljük;
- még megfelelő érték (Slightly Off Target): a terv 33–66 százalékig történő teljesítése, ezt sárga ikonnal jelöljük;
- nem megfelelő (Off Target): 33 százalék alatti teljesítés, piros színű ikon.

A végeredmény a 1. ábrán látható: egy egyszerű, jól áttekinthető felületen megjelennek az összefoglaló adatok, listánézetben a teljesítménymutatókkal, amely a döntéshozókat azonnal tájékoztatja az ügyletek és tervek teljesítéséről.



3. ábra. Egy KPI tulajdonságainak beállítása

Akinek felkeltettük az érdeklődését, nétan szakmai kérdése merülne fel a Dynamics CRM-mel kapcsolatban, szívesen látjuk a TechNetKlub.hu újonnan nyílt CRM-levelezőlistáján, ahol igyekszünk mielőbb válaszolni bármilyen kérdésre.

Kovács László,
(kovacs@sb.hu) System Builders
Wentzel István
(Istvan.Wentzel@microsoft.com) Microsoft Magyarország

LINQ

Ha valaki készített már objektumorientált alkalmazást, amelyik használt adatbázist, esetleg XML-dokumentumokat, az tudja, hogy mennyire különbözik az egyes területek hozzáállása az adatok tárolásához, eléréséhez, a vezérléshez.

Nem véletlen, hogy minden programozásról szóló könyv általában külön fejezetet szentel az SQL vagy az XML tárgyalásának, és egész tudományág van kialakulóban az objektum-relációs leképezés körül. Anders Hejlsberg – a C# nyelv tervezője – már régóta szeretett volna erre valamilyen elegáns választ adni.

A C# 3.0-val eljött az idő, most lehet olyan nyelvi újításokat bevezetni, amelyek segítenek áthidalni ezt a gondolkodásbeli különbséget. Ne ijedjünk meg! A C# 3.0 még nem jelent meg,

kedvenc nyelvük 9.0-s változatában szintén elérhetők lesznek ezek az újdonságok.

LINQ-áttekintés

A C# 3.0 (és a VB 9.0) része lesz az úgynevezett LINQ technológia. Ez a Language Integrated Query (framework) – tehát a nyelvbe ágyazott lekérdező keretrendszer – kifejezés rövidítése. Talál az elnevezés, mert amellett, hogy a LINQ segít a lekérdezésekben, kapcsolatot is teremt a korábban már említett világok között (a „link” szó angolul kapcsolatot is jelent). A LINQ keretrendszer több részre oszlik aszerint, hogy mely objektumokkal dolgozik. Az alapszintű LINQ-rendszer lehetővé teszi, hogy a szokásos nyelvi formulák között megfogalmazzunk lekérdező jellegű programrészeket. Az adatbázisokkal való kapcsolat megteremtése a LINQ to SQL feladata, míg az XML-fájlok és formátumok világába a LINQ to XML vezet el minket.

A LINQ telepítése

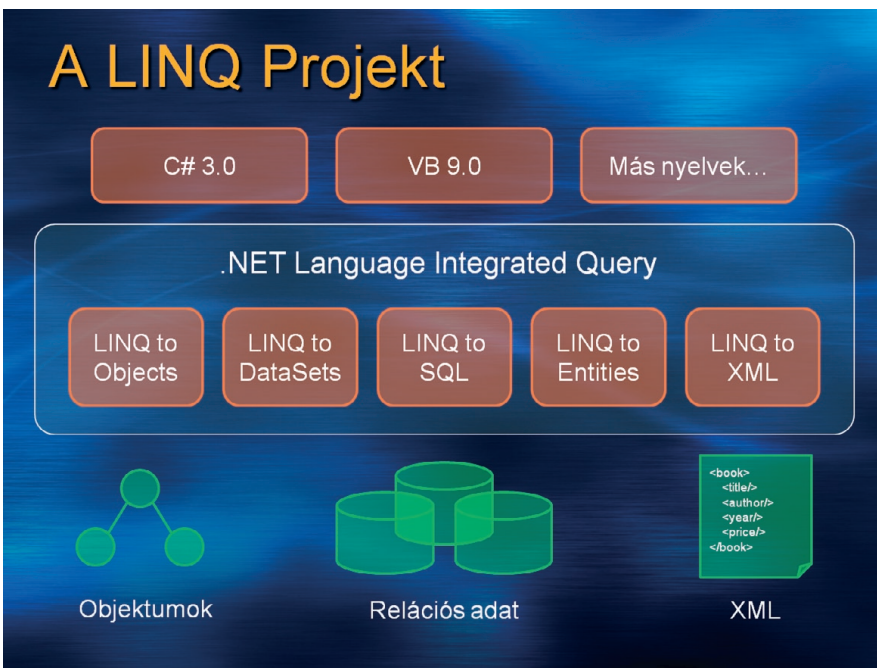
Jelenleg a LINQ megjelenés előtti, CTP állapotban érhető el, ez a cikk a 2006. májusi CTP-változat alapján készül. A LINQ kipróbálásához szükség van a Visual Studio valamelyik változatára (akár az ingyenes Express is megteszi), és ha arra telepítjük, akkor kapjuk meg az új C#, vagy VB fordítót. Ha szeretnénk használni a LINQ to SQL designert, akkor nem elég az Express változat. A telepítés után nincs is más teendőnk, mint kinyitni kedvenc Visual Studio-változatunkat és máris láthatunk egy új projektípust, valamint a start menüben található a LINQ-dokumentációra és a mintaprogramokra mutató hivatkozásokat.

Egy egyszerű LINQ-példa

Nézzünk egy nagyon egyszerű LINQ-példát! Itt most szándékosan nem használjuk még sem a LINQ to SQL-t, sem a LINQ to XML-t, hogy először megérthessük a lekérdezések szintaktikáját. Ha készítünk egy új LINQ-konzolalkalmazást, akkor kapunk egy klasszikus program.cs fájlt. Szükségünk lesz egy lekérdező forrásra, ehhez használjuk a System.Diagnostics névtér Process osztályát (ne feledjük a forrásfájl elején a using-ok közé felvenni a névtérrel, ha az alábbi, rövid hivatkozást szeretnénk használni!)

Ennek GetProcesses metódusa visszaadja nekünk az éppen futó folyamatok gyűjte-

A LINQ Projekt



A LINQ technológiák körképe

de Community Technology Preview változatban elérhető már, így megkezdhetjük a hozzászólást ezekhez a formulákhoz és gondolatokhoz.

Fontos megemlíteni azt is, hogy a Visual Basic kedvelőinek sem kell csüggedniük, ugyanis

ményét. Írjuk be az alábbi kódot a program Main metódusába:

```
var expr = from s in Process.GetProcesses()
           select s.ProcessName;
foreach (string i in expr) Console.WriteLine(i);
Console.ReadLine();
```

Ha lefuttatjuk ezt a programot, és van jogunk a listázáshoz, akkor a konzolon láthatjuk az éppen futó folyamatok listáját. Elemezzük egy kicsit ezt a programot!

A lekérdezés elemei

Egy pillanatra tegyük félre a szokatlan „var” kifejezést a bal oldalon, és nézzük a jobb oldalt. Igencsak ismerős kifejezésekkel van dolgunk, hiszen ezeket már az SQL nyelvből ismerhetjük. Mégis láthatjuk, hogy objektumorientált kóddal van dolgunk, hiszen a kiválasztott elemek típusát mi határozzuk meg a select után, amikor a listában szereplő Process típusú objektumokról a nevüket vesszük be a lekérdezésbe. A LINQ lehetővé teszi, hogy ilyen és ehhez hasonló lekérdezéseket intézzünk minden olyan osztályra, ame-

További információk

Don Box, Anders Hejlsberg: The Linq project: <http://msdn.microsoft.com/library/en-us/dndotnet/html/linqprojectovw.asp>
 Linq May 2006 Preview: <http://www.microsoft.com/downloads/details.aspx?familyid=1E902C21-340C-4D13-9F04-70EB5E3DCEEA&displaylang=en>
 The ADO.NET Entity Framework Overview: <http://msdn.microsoft.com/library/en-us/dnvs05/html/ADONETEnFrmOvw.asp>
 Blinq preview: http://www.asp.net/sandbox/app_blinq.aspx

lyik implementálja az IEnumerable interfejszt. Ilyen például a Process.GetProcesses() által visszaadott tömb, de sok ilyen objektumot ismerünk még, illetve magunk is könnyen elkészíthetjük őket.

A kötelező from és select mellett többek között van lehetőségünk szűrésre (where), rendezésre (orderby) vagy akár csoportosításra is. Számos más lehetőségünk is van, ezeket részletesen a LINQ-kel együtt települő „Standard Query Operators.doc” dokumen-

tum taglalja. Aki dolgozott már SQL-lel, az hamar otthon fogja magát érezni, de örülni fog az objektumorientált lehetőségeknek. Egy ilyen lehetőséget láthatunk az alábbi lekérdezésben is:

```
var expr = from s in Process.GetProcesses()
           where s.ProcessName.Length == 5
           orderby s.WorkingSet descending
           select s.ProcessName;
```

Ezzel kiválaszthatjuk az öt karakter hosszúságú névvel rendelkező processzeket a working set mérete szerint csökkenő sorba rendezve.

És mi történik a típusokkal?

Aki szereti az erős típusosságot, de dolgozott már gyengén típusos környezetben, az bizonyára kellő döbbenettel konstataulta a var kulcsszó megjelenését, ami sok nyelvben a meghatározatlan és megváltoztatható típusok jelzése (gondoljunk csak például a JavaScriptre). A C# 3.0-ban a var annyit jelent, hogy „vedd a kifejezés jobb oldalán előálló típust, és legyen ennek a változónak az az erős típusa”. Ezzel egyszerre megtarthatjuk az erős típusosságot; amellett, hogy nem kell azon gondolkodnunk egyfolytában, hogy a jobb oldalon található, akár egészen bonyolult lekérdezés végül milyen típust eredményez.

A LINQ alternatív megfogalmazása

A korábban bemutatott lekérdezésforma a legrövidebb, legkönnyebben olvasható forma. Ha szeretnénk tudni, hogy mi zajlik a háttérben, akkor cseréljük le az alábbira:

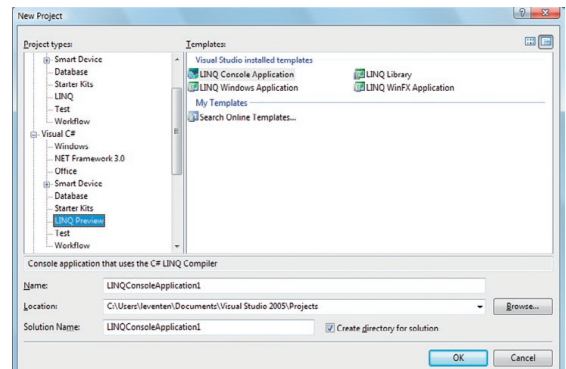
```
var expr = Process.GetProcesses()
           .Where(s => s.ProcessName.Length == 5)
           .OrderBy(s => s.ProcessName)
           .Select(s => s.ProcessName);
```

Itt azt láthatjuk, hogy a korábbi lekérdezés típusú kódból valójában a fordító mit készít, és ezzel betekintést nyerhetünk a LINQ működésébe. A zárójelben található kifejezések az úgynevezett Lambda-kifejezések, ezek hasonlítanak a névtelen metódusokra abban, hogy valamilyen művelet elvégzésével adha-

tunk meg feltételeket. Ezek teszik lehetővé, hogy a feltételektől függetlenül implementáljuk az egyes kulcsszavaknak megfelelő metódusokat, majd azokat a nyelv részeként egymás után fűzzük. De ne is merüljünk mélyre! Aki szeretne, utána tud olvasni a LINQ mélyebb lelkivilágának és kiterjeszhetőségének az MSDN oldalain.

LINQ to SQL

Most, hogy már van egy hasznos és rugalmas lekérdezőeszközünk, jó lenne minél többre használni. Az egyik kézenfekvő lehetőség a relációs adatbázisok lekérdezése. Ma ez egyfajta skizofrén tudatállapotot igényel, és folyamatosan váltanunk kell fejben az adatbázis funkcionális és a C# objektumos megközelítése között. Nem is beszélve arról, hogy mennyi alacsony szintű, üzemeltetés



A LINQ projekt létrehozása Visual Studióval

jellegű kódot kell írunk (például connection-objektumok, lekérdezések összefűzése stb). Ehelyett a LINQ to SQL egy egyszerű szintaktikával kínál meg minket (a minta a LINQ to SQL dokumentációból származik):

```
[Table(Name="Customers")]
public class Customer
{
    [Column(Id=true)]
    public string CustomerID;
    [Column]
    public string City;
}
//Ez már valahol a program részben van
DataContext db = new DataContext(connectionString);
Table<Customer> Customers = db.GetTable<Customer>();
var q =
    from c in Customers
    where c.City == "London"
    select c;
foreach (var cust in q) Console.WriteLine("id = {0}, City = {1}", cust.CustomerID, cust.City);
```

A végén található két sor már ismerős lehet, ami pedig előtte történik, az nagyon egyszerű: először a LINQ to SQL által definiált attribútumok segítségével készítünk egy osztályt, amit hozzárendelünk egy, az adatbázisban tárolt táblához, illetve az osztály mezőit a tábla mezőihöz. Ezután már tudjuk is használni, a DataContext segítségével megadjuk, hogy melyik adatbázist használjuk egy connection stringgel, majd pedig a GetTable hívásával kaphatunk egy hivatkozást a valódi táblára.

Ezután a megszokott módon folytathatjuk a lekérdezéseket objektumorientált módon.

Linq to XML

A másik, nem kifejezetten objektumorientált adatforma az XML. Ez a dokumentum alapú adatforma többféleképpen is feldolgozható. Csak a .Net XML névterei legalább két lehetőséget adnak, amelyek közül az egyik inkább a dokumentum jellegű szekvenciális feldolgozás (a readerek világa), a másik pedig a teljes struktúrát a memóriába töltő DOM (Document Object Model) lehetősége. Most ezek helyett, illetve mellett kaphatunk egy közvetlenül a programozási nyelvbe integrált XML nyelvű definíciós és lekérdezőformát.

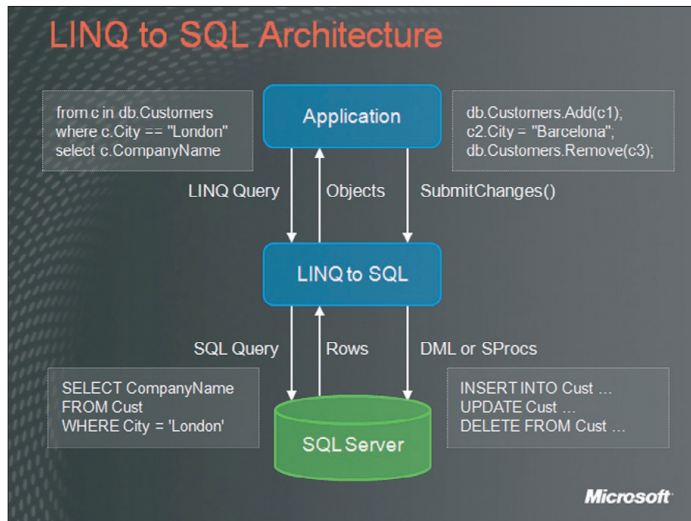
Lássunk először egy példát a minták közül arra, hogy hogyan lehet egy új XML-elemet definiálni:

```
var e = new XElement("Ember",
    new XElement("Programozó", true),
    new XElement("Név", "Kóder Károly"),
    new XElement("Kor", 31));

var s = e.ToString();
Console.WriteLine(s);
Console.ReadLine();
```

Az eredményt megjelenítve láthatjuk, hogy milyen egyszerű ez.

Ha az IntelliSense segítségével végignézzük, hogy az objektumnak milyen tulajdonságai és metódusai vannak, akkor azonnal láthatjuk azt is, hogy egyrészt miként tudjuk objektumorientált eszközökkel manipulálni ezt az XML-modellt, másrészt, hogy miként



A LINQ to SQL architektúrája

lesz alkalmazható erre az XML-moddellel a LINQ. Ezzel akár komplex lekérdezéseket is alkothatunk, amelyek összekötnek a memóriában található adatforrásokat a relációs és az XML-adatokkal.

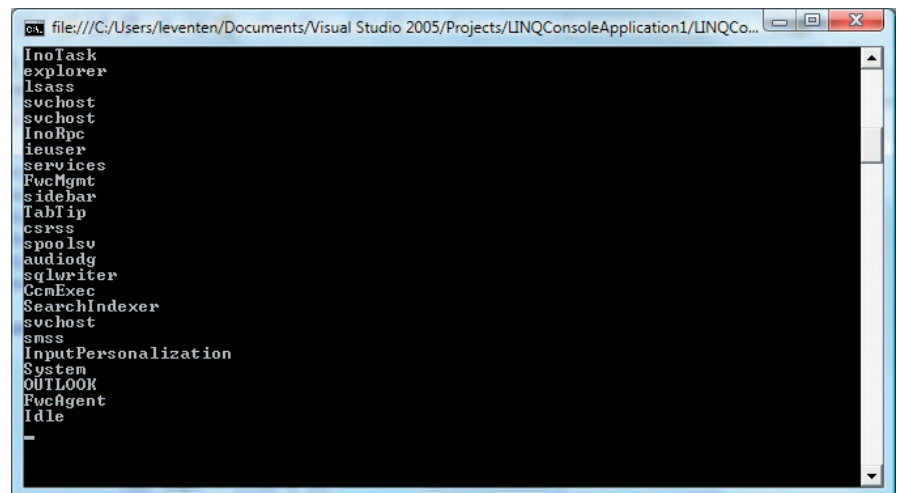
Oldalak egy szempillantás alatt: Blinq

Adatbázis alapú alkalmazásoknál számos unalmas órát (napot) okoznak azok a műveletek, amikor az egyszerűbb alapműveleteket végző felületeket készítjük el (például update, delete stb.). Rutinosabb versenyzők

legenerálja az adatbázisunk webes felületét. Ebben lehet majd alapműveleteket végezni (create, read, update, delete), illetve lehet navigálni a sémában; ismeri a kulcsok mentén az összefüggéseket, és az adatok lapozását is lehetővé teszi. Ezek az oldalak jól használhatók az alkalmazásunk adminisztrációs felületein vagy akár magában az alkalmazásban is. A Blinq a LINQ-re épül, jelenlegi változata a májusi CTP-t használja. Az még nem látszik teljesen, hogyan fog beilleszkedni a nagy egészbe, de mindenesetre ígéretesnek tűnik. Mostani állapotában megtekinthető a http://www.asp.net/sandbox/app_blinq.aspx címen.

Az adatelérési réteg további fejlesztései

Természetesen egy ilyen rövid cikkben épp-hogy csak a felszínt lehet megkapargatni, a LINQ mögött sokkal több van. Komplex és kibővíthető keresések, másfajta adatforrások – például a LINQ to DataSet vagy az új ADO.NET Entity Framework, ami az objektumrelációs leképezés teljes problémakörére próbál átfogó választ adni. Akinek megtet-



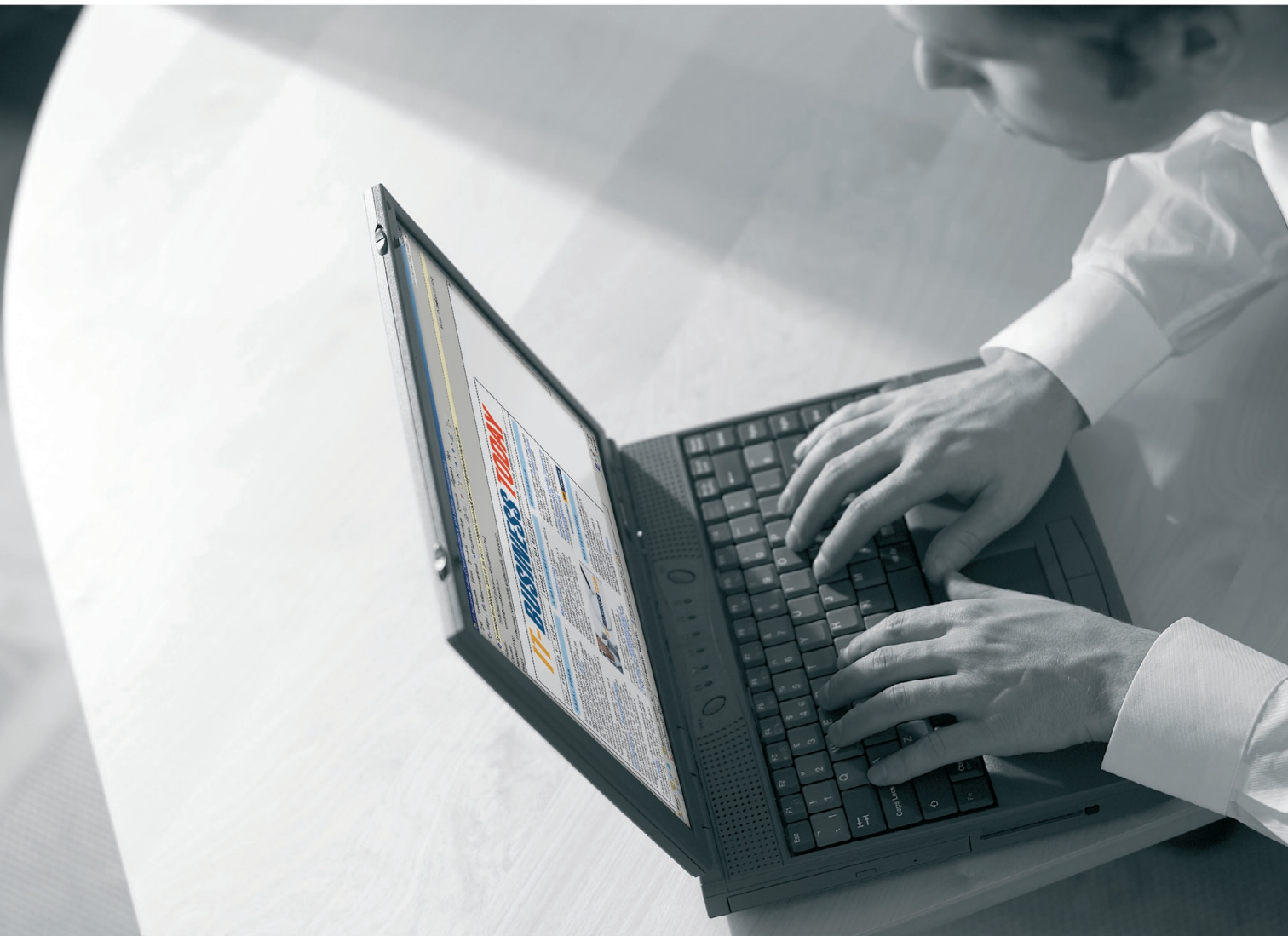
Az éppen futó folyamatok listája a konzolon

persze már generálják ezt a kódot. Ebben a generálásban nyújt segítséget az ASP.NET-fejlesztőknek a Blinq. Ez egy parancssori eszköz, és ha megadjuk neki egy adatbázist, egy mappát és egy virtuális mappát, akkor

szett a LINQ, javasoljuk, hogy próbálja ki, illetve a keretes cikkben felsorolt oldalakon olvasson utána a többi újdonságnak is!

Nagy Levente
(nagy.levente@microsoft.com) Microsoft Magyarország

IT-BUSINESS TODAY



- felsővezetőknek, döntéshozóknak
- az elmúlt 24 óra legfontosabb két-három magyar és nemzetközi ICT híre
- ingyenes napi hírlevél

Regisztráljon!

www.it-business.hu/hirlevel

KONKURENCIAKEZELÉS AZ SQL SERVER 2005-BEN

Az SQL Server 2005 fejlesztői régi adósságot törlesztettek: megvalósították a versenytársaknál már több mint egy évtizede sikerrel alkalmazott optimista konkurenciakezelési modellt.

Az adatbázis-kezelő rendszerekkel szembeni egyik legkomolyabb kihívás, hogy egyszerre biztosítsák az adatok gyors, konzisztens és konkurens kezelését. A megfelelő teljesítmény alapkövetelmény: senki nem szeret perceket várni egy-egy számlatétel rögzítésére, és a gyártósor sem állhat addig, amíg az adatkezelőnk feldolgozza az egyes munkafázisok adatait a következő előkészítéséhez. Minden alkalmazás valamilyen szabályrendszer alapján kezeli az adatokat: legyenek ezek nagyon vagy kevésbé szigorúak, az adatbázis-kezelőnek biztosítania kell, hogy az adatok megfeleljenek ennek a szabályrendszernek, azaz konzisztensek legyenek. Az alkalmazásokat általában egyszerre több, akár több ezer felhasználó is használja. Az adatbázis-kezelőnek megfelelő teljesítmény mellett, az adatbázis konzisztens állapotát folyamatosan fenntartva kell biztosítania a konkurens felhasználást.

Teljesítmény, konzisztencia és konkurencia

Nagyon egyszerűen belátható, hogy e három tényezőt nem kezelhetjük egymástól függetlenül, azaz minél erősebb adatkonzisztenciára törekszünk, annál nehezebb a konkurens hozzáférések koordinálása, és ez bizony csak a teljesítmény csökkenésének árán valósítható meg. Amikor tehát a konkurens és konzisztens adatkezelés szabályozásáról beszélünk, mindig tekintettel kell lennünk arra, hogy minél konzisztensebb, pontosabb adatkezelést szeretnénk megvalósítani, annál nagyobb árat kell ezért fizetnünk.

Tranzakciók

Az adatbázist egy adott időpontban konzisztensnek nevezzük, ha az adatbázisban tárolt adataink megfelelnek egy bizonyos, rögzített szabályrendszernek. A szabályok sokféleképpen lehetnek, az adatok egységességétől kezdve egészen az olyan összetett szabályokig terjedhetnek, hogy például a zárás után a főkönyvi számla egyenlege nulla legyen.

Az adatbázistól általában nem várhatjuk el, hogy minden időpillanatban konzisztens legyen, mivel lehetnek olyan összetett műveleteink, amelyek végrehajtása közben az adatbázis egyes részei óhatatlanul inkonzisztens állapotba kerülnek. Gondoljunk például arra az esetre, hogy az egyik számláról már levontuk az összeget, de a másik számlához még nem adtuk hozzá, vagy hozzáadtunk egy tételt a számlához, de a számla egyenlegét még nem módosítottuk.

Az átmeneti inkonzisztens helyzet a műveletet éppen végrehajtó folyamat szempontjából nem okoz semmilyen problémát, ha az összetartozó műveletek mind végrehajthatók, és a módosítások sikeresen be is kerülnek az adatbázisba. A helyzet azonban teljesen más az adatbázisban műveleteket végző konkurens folyamatok szempontjából vizsgálva. Ha a konkurens folyamat beleolvás az átmenetileg inkonzisztens adatokba, és azokat felhasználva módosításokat végez az adatbázison, az rövid idő alatt akár az adatbázis teljes inkonzisztenciáját is eredményezheti. Még súlyosabb lehet a helyzet, ha két folyamat egymás tudta nélkül egyszerre módosítja ugyanazokat az adatokat.

A konzisztens és konkurens adatmódosítás megoldására a tranzakciók szolgálnak. A tranzakciók biztosítják az összetartozó műveletek egyszerre történő érvényesítését („atomiság”), a tranzakciók végrehajtása utáni adatkonzisztenciát („konzisztencia”), az egyes folyamatok átmeneti inkonzisztens állapotának más folyamatok előli elzárását („izoláció”) és a műveletek eredményének tartósságát („tartósság”). Ezt a négy tulajdonságot jelöljük az ACID betűszóval az angol elnevezések (Atomicity, Consistency, Isolation, Durability) kezdőbetűi alapján.

Egy rendszert tranzakció-kezelő rendszernek nevezünk, ha biztosítja a tranzakciók ACID szerinti kezelését. Az ACID-tulajdonságok közül az atomiság és a tartósság egyszerűen definiálható követelmények, azaz vagy a tranzakció minden művelete érvényesül és tárolódik az adatbázisban, vagy egyik sem, tehát az adatbázis a tranzakciót megelőző állapotba kerül vissza. A konzisztencia és az izoláció azonban sokkal bonyolultabb fogalmak, és a gyakorlatban csak együttesen értelmezhetők.

Ma már minden komoly adatbázis-kezelő rendszer megfelelően kezeli a tranzakciókat, azonban mindegyik más-más módon valósítja meg azokat. Az SQL Server 2005 a tranzakciók kezelését több együttműködő komponens segítségével oldja meg. Mivel az összes komponens tárgyalása meghaladja e cikk terjedelmi lehetőségeit, a továbbiakban csak a tranzakció-kezelés konkurenciaproblémáira (konzisztencia, izoláció), és azok megoldására koncentrálnunk.

Konkurenciakezelési modellek

A konkurenciakezelés alapvető kihívása a közös erőforrásokhoz való hozzáférés szabályozása. Egy relációs adatbázis-kezelő esetében közös erőforrás az adatbázis, a tábla, az adatlap, a sor, a memória, a lekérdezéseket futtató végrehajtási szál, és minden egyéb olyan rendszerelem, amelyen a felhasználói folyamatoknak osztozniuk kell. Az SQL Server 2005 a felhasználói folyamatok által közvetlenül kezelhető erőforrások (adatbázis, tábla, adatlap, adat vagy indexsor) konkurens hozzáféréseinek szabályozását teszi lehetővé, míg a belső erőforrásokhoz (memória, végrehajtási szálak, belső adatstruktúrák) történő hozzáférést automatikusan szabályozza.

Az adatbázis-kezelők két alapvető konkurenciakezelési modellt alkalmaznak: a pesszimista és az optimista modellt. Az SQL Server 2005 mindkét modellt megfelelően támogatja.

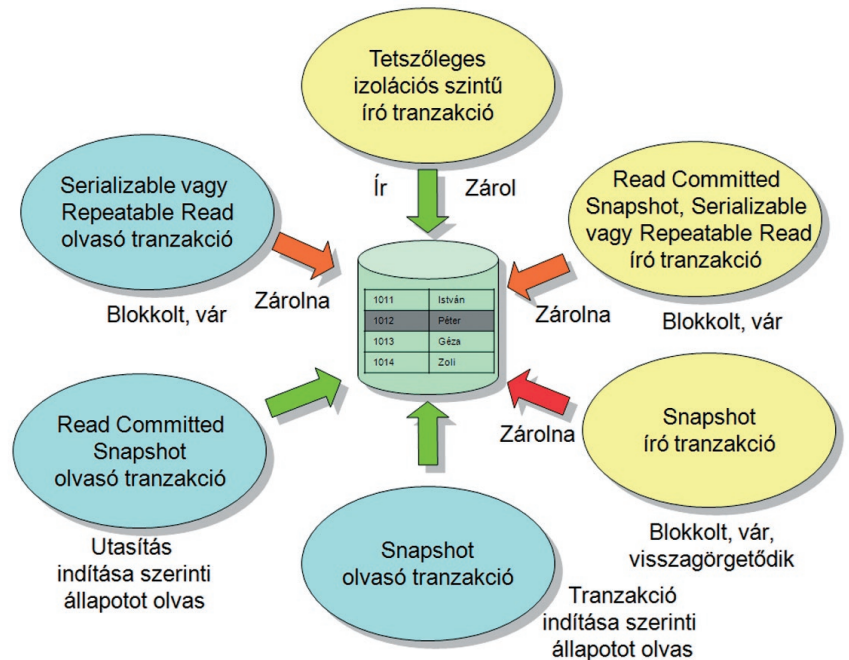
Pesszimista konkurenciakezelés

A pesszimista konkurenciakezelés lényege, hogy a használandó erőforrásokat lefoglaljuk, és mindaddig nem szabadítjuk fel, amíg azokkal műveleteket végzünk. A megközelítés azért pesszimista, mivel arra számít, hogy az erőforrást más is le akarja foglalni, és ezt igyekszik önző módon megakadályozni, az

az megszerezni és megtartani az erőforrást, ameddig csak szükség lehet rá.

Ez a gyakorlatban azt jelenti, hogy az adat-olvasó tranzakciók nem engedik más tranzakcióknak írni az adatokat mindaddig, amíg be nem fejezik az olvasást, az adatokat író tranzakciók pedig minden más műveletet

nagyobb konkurenciát tesz lehetővé, mivel a legtöbb megvalósításban az olvasók nem blokkolják az írókat, és az írók nem blokkolják az olvasókat. Az optimista megközelítés velejárója, hogy ugyanazt az adatot több tranzakció is megpróbálhatja módosítani. Ezt azonban ebben az esetben sem engedhet-



Az SQL Server 2005 tranzakciós lehetőségeinek áttekintése

(írást és olvasást) megakadályoznak, a tranzakció-kezelés terminológiájával elve „blokkolják” a konkurens folyamatokat a tranzakció befejezéséig.

A pesszimista konkurenciakezelés általában magas szintű konzisztenciát, de kisebb konkurenciát tesz lehetővé. A folyamatos blokkolás az erősen konkurens rendszerekben komoly teljesítményproblémákhoz és számos esetben a folyamatos munkát nehezítő feloldhatatlan blokkoláshoz, holtponthoz (angolul „dead-lock”) vezet.

Optimista konkurenciakezelés

Az optimista konkurenciakezelés alapelve az, hogy az erőforrásokat csak a szükséges mértékben foglaljuk le, és a műveletek elvégzése után gyorsan felszabadítjuk, hogy más tranzakciók rendelkezésére álljanak. Ez optimista hozzáállás, mert arra számítunk, hogy az általunk használni tervezett erőforrás rendelkezésre áll, amikor szükségünk van rá.

Az optimista konkurenciakezelés általában alacsonyabb szintű konzisztenciát, de

jük meg (belátható, hogy ez a teljes káoszhoz vezetne), ilyenkor az optimista tranzakciónak is várnia kell, amíg az erőforrás felszabadul, vagy vissza kell azt görgetnünk, és újra meg kell próbálnunk a tranzakció végrehajtását.

Fontos hangsúlyoznunk, hogy a pesszimista szónak semmilyen leminősítő jellege nincs, nem arról van szó, hogy a pesszimista konkurenciakezelési modell rosszabb vagy kevésbé hatékony, mint az optimista. A két modellt önállóan vagy egymás mellett alkalmazva különböző feladatok megoldására tudjuk használni.

Zárolás és verziókezelés

Az SQL Server 2005 a konkurens hozzáférés szabályozására a zárolást és a verziókezelést alkalmazza. A zárolás a pesszimista konkurenciakezelés legfontosabb eszköze. Az SQL Server az erőforrásokra elhelyezett zárok segítségével szabályozza, hogy melyik tranzakció milyen módon férhet hozzá az adott erőforráshoz. A verziókezelés az optimista konkurenciakezelést támogatja azért, hogy

a módosított adatok módosítás előtti állapotát egy verziótárban tárolja, és a többi tranzakció rendelkezésére bocsátja.

Zárolás

Az SQL Server az adatbázisokon, táblákon, adatlapokon, index- és adatsorokon helyez el különböző zárat a hozzáférés szabályozására. A zárolás automatikus, és legtöbbször a sor szintjén történik, azaz a szerver adat- vagy indexsorokat zárol. A rendszer által használt két legfontosabb zár az olvasásokhoz használt osztott zár („shared lock” – „S”), és az adatmódosításhoz használt kizárólagos zár („exclusive lock” – „X”). Az SQL Server számos egyéb zártípust is használ – mint például a módosítást és törlést megelőző módosítási zár („update lock” – „U”) vagy a zárolási szándékot jelző szándék zárok („intent lock” – „IX”, „IS”) –, amelyek táblára vagy adatlapra alkalmazva az alacsonyabb szintű erőforrások zárolását jelzik.

Zárat az erőforrásokra zártulajdonosok („lock owner”) helyezhetnek el. Zártulajdonos lehet egy munkamenet, egy tranzakció vagy egy kurzor. A továbbiakban csak tranzakciókkal foglalkozunk.

Egy erőforrásra több tranzakció is helyezhet zárat, amennyiben a zárok egymással kompatibilisak. A pesszimista konkurenciakezelés esetén az, hogy egy tranzakció hozzáférése valamilyen erőforráshoz olvasási vagy írási céllal, azon múlik, hogy az erőforráson milyen zárok vannak. Amennyiben

a tranzakció által kért zár nem kompatibilis a már meglévő zárokkal, a tranzakciónak várnia kell a meglévő zárok feloldására, azaz blokkolt állapotba kerül. Az osztott zárok kompatibilisak egymással, azaz több tranzakció olvashatja ugyanazt az adatot. A kizárólagos zárok egyik-másik zártípussal sem kompatibilisek, tehát az írók blokkolják az írókat és az olvasókat is. Az egyes zárok egymással való kompatibilitását az SQL Server egy kompatibilitási mátrix alapján ellenőrzi.

Megjegyezzük, hogy a blokkolás önmagában nem probléma, hanem a tranzakció-kezelés természetes következménye.

Azt, hogy az SQL Server milyen zárat helyez el az erőforrásokra, és azokat mennyi ideig tartja fenn, a tranzakció-izolációs szintek segítségével szabályozhatjuk.

Verziókezelés

A verziókezelés alapértelmezetten kikapcsolt állapotban van, és az „ALLOW_SNAPSHOT_ISOLATION” vagy a „READ_COMMITTED_SNAPSHOT” adatbázis-opciók bekapcsolásával aktiválhatjuk. A két opcióra az izolációs szintek tárgyalásánál visszatérünk. Amennyiben az opciók valamelyike bekapcsolt állapotban van, az SQL Server automatikusan elkezd lementeni az adatmódosító műveletek során megváltozott sorok módosítást megelőző állapotát a tempdb-be.

```
--1. kapcsolat: az ALLOW_SNAPSHOT_ISOLATION
--opció bekapcsolása
```

```
USE master
GO
```

```
ALTER DATABASE PeldaDB SET ALLOW_SNAPSHOT_ISOLATION ON
GO
```

```
--1. kapcsolat: tranzakció indítás.
--Az Adat mező tartalma 'AAA'.
```

```
|
USE PeldaDB
GO
```

```
BEGIN TRAN
UPDATE T1 SET Adat = 'XXX' WHERE ID=1
```

```
--2. kapcsolat: a SNAPSHOT tranzakció
--indítása és sikeres konkurens olvasás.
--Az eredmény 'AAA'.
```

```
SET TRANSACTION ISOLATION LEVEL SNAPSHOT
GO
```

```
BEGIN TRAN
SELECT Adat FROM T1 WHERE ID=1
```

A Snapshot tranzakció működése

Amennyiben egy sort több, párhuzamosan futó tranzakció is módosít, az SQL Server a változatokat egy láncolt listában őrzi meg, amely az újabb verzióktól a régebbi verziók felé halad. A verziókat az SQL Server mindaddig megőrzi, amíg azokra szükség lehet, majd automatikusan kitakarítja a verziótárból. Mivel ezáltal az adatsorok módosítások előtti minden konzisztens állapota megőrződik, az optimista konkurenciakezelést alkalmazó tranzakciók mindig hozzáférnek

azokhoz, így az írók nem akadályozzák az olvasókat.

A verziókezelés módját és a verziók megőrzésének időtartamát a tranzakció-izolációs szintek határozzák meg.

Tranzakció-izolációs szintek

A tranzakció-izolációs szintek azt határozzák meg, hogy a konkurens tranzakciók adatmódosító műveletének hatása milyen mértékben van egymástól elzárva, izolálva. A tranzakció-izolációs szint mindig az azt alkalmazó tranzakció szempontjából határozza meg az izoláció módját. Az SQL Server 2005 négyféle pesszimista (Read Uncommitted, Read Committed, Repeatable Read, Serializable) és kétféle optimista (Read Committed Snapshot, Snapshot) tranzakció-izolációs szintet különböztet meg.

A „Read Uncommitted” izolációs szint

A „Read Uncommitted” a leggyengébb izolációs szint. Ezen a szinten az olvasók nem blokkolják az írókat, és az írók nem blokkolják az olvasókat, mivel az olvasók nem alkalmaznak osztott zárat, és figyelmen kívül hagyják a kizárólagos zárat. Az SQL Server 2005 semmilyen szinten nem engedi meg ugyanannak az erőforrásnak több tranzakció általi írását, így ezen a szinten sem, azaz az írók mindig blokkolják az írókat. A kizárólagos zárok tehát az adatmódosítástól kezdve a tranzakció befejezéséig fennállnak. (Ez minden izolációs szintre igaz, úgyhogy a továbbiakban ezzel nem foglalkozunk.)

Ha egy tranzakció „Read Uncommitted” izolációs szinten van, akkor beleolvashat más tranzakciók nem véglegesített adataiba. Ez inkonzisztens vagy más szóval „piszkos” olvasáshoz vezet, és az az eredménye, hogy nem konzisztens vagy később érvénytelenített adatot olvasunk, tehát az eredmény nem mindig lesz pontos. Hangsúlyozzuk, hogy ez nem feltétlenül rossz viselkedés, az esetek jelentős részében elfogadható, tehát mindig az adott situációnak megfelelően kell értékeljük: ez a viselkedés nemkívánatos egy bankszámla nyilvántartásban, de teljesen elfogadható lehet egy kereskedelmi rendszer trendanalízis-jelentésében.

Bár a „Read Uncommitted” modell kevesebb zárat alkalmaz, mégis pesszimista abban az értelemben, hogy azért nem zárol, és azért nem veszi figyelembe a zárat, mert

azt feltételezi, hogy mások zárolnak, és csak mások „piszkos” adataihoz férhet hozzá.

A „Read Committed” izolációs szint

A „Read Committed” izolációs (rövidítve RCI) szint az SQL Server 2005 alapértelmezett tranzakció-izolációs szintje. Az RCI csak a tranzakciók által véglegesített adatok olvasását teszi lehetővé, nem engedi meg a piszkos olvasást. Ezt azért éri el, hogy az olvasáshoz osztott zárat alkalmaz az erőforrásokon. Mivel az osztott zár nem kompatibilis az írók által alkalmazott kizárólagos zárral, az RCI tranzakciónak mindaddig várnia kell, amíg a kizárólagos zárok feloldódnak. Az RCI tranzakciók adatolvasás közben blokkolják az írókat, mivel az olvasás idejére osztott zárat alkalmaznak. A blokkolás tehát kétirányú, az írók blokkolják az olvasókat, és az olvasók blokkolják az írókat. Ez alapvetően nem probléma, természetes viselkedés mindaddig, amíg nem okoz túlzott várakozási időket és kezeletlen holtpontokat.

Az RCI tranzakciók tervezése komoly elemző munkát kíván: tisztában kell lennünk azzal, hogy milyen tranzakciók fognak futni a rendszerben, azok pontosan milyen erőforrásokat, milyen módon és meddig zárolnak. A leggyakoribb megoldandó feladat a hosszabban futó lekérdezések által okozott írási problémák kezelése, mivel a hosszan futó olvasási műveletek számos írási műveletet blokkolhatnak, amelyek aztán más folyamatokat blokkolnak. Ez gyakran nem várt és nem kezelt holtpontokhoz vezet – ugyanakkor a problémák megfelelő tervezéssel és alkalmazásfejlesztési szabályok kialakításával elkerülhetők, illetve minimalizálhatók.

A „Repeatable Read” izolációs szint

A „Repeatable Read” izolációs szint biztosítja az elolvasott adatok tranzakción belüli változatlanágát. Ez magában foglalja az RCI szintet, és megszigorítja az osztott zárok kezelését azért, hogy az osztott zárat nemcsak az olvasási művelet végéig, hanem a tranzakció befejezéséig fenntartja. Ezáltal a már egyszer elolvasott adatot senki sem módosíthatja a tranzakció végéig. Nyilvánvaló, hogy a „Repeatable Read” izolációs szinten futó tranzakció egyidejűleg több adatot zárol, és a zárok tovább állnak fenn, mint az RCI tranzakciók esetében. Ez hosszabb blokkolásokat, több holtpontot és ezáltal csökkenő

konkurenciát eredményez. Sok konkurens felhasználó esetén ez a rendszer komoly teljesítménycsökkenéséhez vezethet.

A „Repeatable Read” izolációs szintet gyakran használjuk arra, hogy bonyolultabb lekérdezéseket kisebb lépésekre bontva fut-

adathalmazokon semmilyen módosítást nem engedhetünk meg a tranzakció végéig. Ilyen tranzakció lehet például egy bankszámlamozgás-művelet, amellyel pénzt emelünk le a számláról, majd a számla egyenlegét módosítjuk, naplózzuk és könyveljük. A folyamat

KOVACSZ.Pelda...QLQuery1.sql*

```
SELECT *
FROM sys.dm_tran_version_store
WHERE database_id = DB_ID('PeldaDB')
```

Results Messages

	transaction_sequence_num	version_sequence_num	database_id	rowset_id	status
1	4930	1	17	72057594038452224	0

Sorverziók lekérézése

tassunk le, miközben a lekérdezések között fenntartjuk a zárat a konzisztens eredmény elérése érdekében. Tipikus példa erre például egy eredménykimutatás készítése, amely általában számos lekérdezés futtatását teszi szükségessé úgy, hogy közben az adatok ne változzanak. Ilyen esetben a „Repeatable Read” kiváltható az új „Snapshot” izolációs szinttel, ahogyan azt később látni fogjuk.

A „Serializable” izolációs szint

A „Serializable” a legszigorúbb tranzakciós izolációs szint. Azt biztosítja, hogy a konkurens tranzakciók úgy fussanak le, mintha egymás után hajtánánk őket végre. Ez azt feltételezi, hogy a tranzakció az adatbázist minden pillanatban aktuális és konzisztens állapotban látja, az általa egyszer már elolvasott adatok nem változnak, és új adatok sem keletkeznek. A „Serializable” izolációs szinten futó tranzakció az olvasási zárat a tranzakció végéig fenntartja, és azt is megakadályozza, hogy az általa futtatott lekérdezések eredményhalmazába új adat kerüljön be. Ezt a lekérdezések feltételeiben szereplő mezőkön lévő indexek kulcstartományainak zárolásával – vagy megfelelő index híján, a táblák zárolásával – éri el.

A „Serializable” nagyon magas szintű konzisztenciát biztosít. Olyan esetekben használjuk, amikor az egyszer már elolvasott

során nem engedünk új számlamozgást rögzíteni, mert a további lépések végrehajtása során eltérő összesítéseket kapnánk a kiinduló állapothoz képest.

A „Serializable” izolációs szinttel kell a legóvatosabban bánnunk. Alapszabály, hogy csak akkor alkalmazzuk, ha feltétlenül indokolt, és ebben az esetben minden lekérdezés minden feltételére megfelelő indexszel kell rendelkezniük a táblázatok megakadályozása érdekében.

A „Read Committed Snapshot” izolációs szint

A „Read Committed Snapshot” izolációs (rövidítve RCSI) szinten futó tranzakciók az adatok olvasásakor mindig az olvasási művelet indítása előtti utolsó konzisztens állapotot látják. Az RCSI-t akkor használhatjuk, ha bekapcsoltuk a READ_COMMITTED_SNAPSHOT adatbázis-opciót. Ebben az esetben a „Read Committed” izolációs szint „Read Committed Snapshot” szintre változik. (Minden RCI tranzakció RCSI-ként működik.) Az SQL Server a tranzakciókhoz ilyenkor egy folyamatosan növekvő tranzakció-azonosítót (XSN) rendel, és az egyes sorok adatbázisba, illetve verziótárba írásakor a sort módosító tranzakció XSN-jét is kiírja. Ez nemcsak az RCSI tranzakciókra igaz, hanem minden adatmódosító tranzakcióra is.

Ha az RCSI tranzakció olyan adatot olvasna, amelynek XSN-je nagyobb vagy egyenlő, mint az utasítás végrehajtásának indításakor aktív tranzakciók legkisebb XSN-je (azaz a futó tranzakciók valamelyike módosította az adatot), akkor megkeresi a sornak azt a legfrissebb változatát a verziótárban, amelynek az XSN-je kisebb ennél, és érvényesített tranzakcióhoz tartozik, azaz az utasítás végrehajtása előtti legfiatalabb, még konzisztensnek tekinthető verziót. A nem RCSI tranzakciók (a „Snapshot” izolációs szintek kivételével) nem használják a verziótárat az adateléshez.

Ha az RCSI tranzakció olyan adatot akar módosítani, amelyet egy másik futó tranzakció módosít, az RCSI tranzakciónak várnia kell a másik tranzakció befejeződésére.

A `READ_COMMITTED_SNAPSHOT` adatbázis-opció bekapcsolásával tehát utasításszintű adatolvasási konzisztenciát érhetünk el anélkül, hogy más folyamatokat blokkolnánk. Mivel az alkalmazások semmilyen módosítást nem igényelnek az RCSI használatához, és az RCSI nem változtatja meg az egyidejű adatírás tiltására vonatkozó szigorú konzisztenciaszabályt sem, az alkalmazások túlnyomó többsége gond nélkül futtatható a `READ_COMMITTED_SNAPSHOT` adatbázis-tulajdonság beállításával.

Mit nyerünk és mit veszítünk az RCSI használatával? A válasz egyszerű: nagyobb konkurenciát, kevesebb holtpontra kapunk, a teljesítmény csökkenése mellett.

Az RCSI által okozott teljesítménycsökkenés viszont sokkal jobban mérhető (általában néhány százalékos), kiszámítható, és könnyebben orvosolható, mint a túlzott blokkolás és a nehezen elemezhető, ad hoc módon jelentkező holtpontok által okozott teljesítménycsökkenés.

A „Snapshot” izolációs szint

A „Snapshot” izolációs (rövidítve SI) szinten futó tranzakció az adatbázist mindig az elindulása előtti utolsó konzisztens állapotban látja. SI szintű tranzakciót csak akkor indíthatunk, ha bekapcsoltuk az „`ALLOW_SNAPSHOT_ISOLATION`” adatbázis-opciót minden olyan adatbázisban, amelyet a tranzakciónk használ. Az SQL Server ekkor az RCSI szint tárgyalásánál megismert azonosítóval (XSN) látja el a tranzakciókat. Ha az SI tranzakció olyan adatot olvasna, amelynek XSN-je nagyobb vagy egyenlő, mint az

SI tranzakció indításakor aktív tranzakciók legkisebb XSN-je, akkor megkeresi a sornak azt a legelső változatát a verziótárban, amelynek az XSN-je kisebb ennél, és érvényesített tranzakcióhoz tartozik, azaz a tranzakció indulása előtti legfiatalabb, még konzisztensnek tekinthető verziót. A nem SI tranzakciók (az RCSI tranzakciók kivételével) nem használják a verziótárat az adateléshez.

A „Snapshot” izolációs szint alkalmazásakor, ha a tranzakció egy másik tranzakció által módosított adatot próbál írni, konfliktus keletkezik, mivel az SI tranzakció nem látja a sor aktuális állapotát, tehát feltehetőleg inkonzisztens állapot állna elő a módosításkor. Ezért az SI tranzakció ilyen esetben csak addig vár, amíg a másik tranzakció véglegesítődik, majd hibával leáll, és visszagörgetődik. Ha a másik tranzakció visszagörgetődik, az SI tranzakció sikeresen végrehajthatja a módosítást. Az SI tranzakció tehát rosszul viseli a konkurens írást, így abban az esetben, ha ennek nagy az esélye, szerencsésebb az RCSI vagy valamelyik pesszimista izolációs szint használata.

Az SI tranzakció nem módosíthat táblát, indexet, adatbázist, és nem hozhat létre új indexet sem, továbbá osztott tranzakciókezelésre sem használhatjuk.

Kiválóan alkalmas viszont a pesszimista izolációs szintekben sok blokkolást okozó jelentéskészítő, lekérdező funkciók megvalósítására, mivel biztosítja az adat-konzisztenciát, és nem blokkolja az írási műveleteket.

Milyen konkurenciakézelési modellt és milyen izolációs szintet válasszunk?

Általánosságban azt mondhatjuk erre, hogy az online tranzakció-feldolgozó rendszerek (OLTP) esetében leginkább az RCI vagy az RCSI izolációs szintek használata javasolt, míg az SI-t legjobban a lekérdező rendszerekben tudjuk kihasználni. A „Repeatable Read” és a „Serializable” izolációs szintek használatát csak olyan tranzakciók esetén javasoljuk, ahol erősebb konzisztenciára van szükség. (A „Repeatable Read” gyakran kiváltható SI-vel, azonban vigyázzunk: az SI nem garantálja azt, hogy más tranzakciók nem írják felül az adatokat, és nem tolerálja a konkurens adatmódosítást sem!)

Amennyiben a rendszerünk jól működik a jelenleg alkalmazott pesszimista izolációs

szintekkel, és csak időnként jelentkeznek hosszasan várakozások, holtpontok, akkor célszerűbb a problémák okát megkeresni és kijavítani, mint vállalni az optimista RCSI vagy az SI által okozott teljesítménycsökkenést és az esetleges írási konfliktusok kezelését. Példa lehet erre egy jól megírt ügyviteli alkalmazás, ahol az egymással konkuráló folyamatok gyakran okoznak rövid ideig fennálló blokkolást, de nem gyakran vezetnek holtpontokra.

Az RCSI előnyét leginkább a nagy komplexitású, hosszasan tartó tranzakciókat futtató vagy a konkurens hozzáférés szempontjából nem kellően átgondolt OLTP-rendszereknél lehet jól kihasználni. Mivel drasztikusan tudjuk csökkenteni a zárok számát, és a zárolt adatokat is el tudjuk olvasni, az RCSI ebben az esetben ideális választás lehet. Az SI izolációs szint pedig legjobban azoknál az olvasásintenzív rendszereknél fizetődik ki, ahol az adatírások nem kerülnek egymással konfliktusba, vagy az SI tranzakció egyáltalán nem ír adatot.

Ilyen példa a lehet a folyamatosan frissített, online lekérdezéseket támogató adattárház. Az adattárház folyamatos frissítése az SI alkalmazásával nem okoz gondot, mivel az SI lehetővé teszi az íráskor melletti konzisztens adatolvasást.

Az OLTP-rendszerekben az SI alkalmazására a legjobb példa a jelentések futtatása. Ha a jelentést SI izolációs szinten indítjuk, akkor a jelentés futtatásának idejére nem blokkoljuk a felhasználókat, és mégis konzisztens adathalmazhoz jutunk.

Az SI használatánál ne feledkezzünk meg az SI alkalmazási korlátairól sem: csak lokális tranzakciókban használhatjuk, bizonyos DDL-műveletek nem engedélyezettek, és az alkalmazásnak fel kell készülnie az írási konfliktusok kezelésére is.

Tökéletes receptet nem tudunk adni arra vonatkozóan, hogy mikor melyik izolációs szintet érdemes alkalmazni, célszerű azonban itt is azt az elvet követni, hogy a lehetséges megoldások közül általában a legegyszerűbb a legjobb: használjuk az RCI-t vagy az RCSI-t, és csak akkor használjuk a többi izolációs szintet, ha feltétlenül szükséges.

A cikkhez kapcsolódó példák a <http://www.szamalk.hu/tisza/sql2005> címről tölthetők le.

Kovács Zoltán

(kovacs@szamalk.hu) vezető oktató, Számalk

A MICROSOFT MAGYARORSZÁG BEMUTATÁSA

Mit is csinál pontosan a magyarországi leányvállalat?

A Microsoft Magyarország Kft. a Microsoft Corporation magyarországi képviselőjeként 1993-ban alakult két fővel. Napjainkban a vállalat dolgozóinak létszáma eléri a 150 főt. A magyar leányvállalat ügyvezető igazgatói tisztségét 2000 decembere óta *Vityi Péter* tölti be.

A cég elsősorban kereskedelmi képviselőként működik, vagyis a Microsoft termékeit értékesíti hazai disztribútorokon és viszonteladó partnerein keresztül, valamint ehhez hasonlóan partnercégeket von be szolgáltatásainak kiszélesítésére is.

Ismét a legjobb hazai munkahely

A Microsoft Magyarország nemcsak az informatikai élet területén tölt be vezető szerepet, hanem ugyanilyen kiemelkedő helyet foglal el a munkaadók között is. A Hewitt Inside kutató-cég és a Figyelő című gazdasági hetilap a 2003-as év „Legjobb munkahely”-évé nyilvánította a Microsoft hazai leányvállalatát. 2006-ban ismét részt vettünk az ezúttal a Hewitt és a HVG által közösen meghirdetett felmérésben, és ismét első helyet értünk el.

Részlegek és szakmai kihívások

Joggal merülhet fel a kérdés az olvasóban, hogy informatikai szakemberként érdekes munkahely lehet-e a Microsoft Magyarország, vagy csak a konkrét termékfejlesztéssel foglalkozó központokban (például Redmondban) érdemes rövid vagy hosszú távon gondolkodni. Ahhoz, hogy bemutassuk, milyen szakmai kihívásokat kínálunk a hazai szakembereknek, tekintünk át a cég felépítését. Jóllehet az egyes szervezeti egységekben más-más szakmai munka folyik, ami közös: az itt dolgozó szakemberek egy folyamatosan megújuló, innovatív cég tagjaként nagy önállóság mellett, izgalmas feladatokat végeznek. Ehhez minden munkaeszköz, képzés, és információ rendelkezésükre áll. Az itt dolgozó szakemberek éves munkaidejük 10 százalékát képzéseken töltik, ami évi 20–25 nap hazai és nemzetközi tréninget jelent.

Nagyvállalati szolgáltatások üzletág – MCS (Microsoft Consulting Services). Ez az üzletág a kiemelt ügyfeleknek segít a megvásárolt termékek bevezetésében, támogatásában, ezen belül

a konzulensek magas színvonalú informatikai tanácsadással járulnak hozzá a projektek sikereihez. Az MCS-en belül egyaránt dolgoznak az IT-infrastruktúra kiépítésével, valamint szoftverfejlesztéssel és projektmenedzsmenttel foglalkozó szakemberek is.

Ők az adott vállalat informatikai vezetésével szorosan együttműködve biztosítják, hogy a már meglévő Microsoft-szoftverek a lehető legjobban szolgálják az ügyfél konkrét informatikai igényeit. Munkájuk során természetesen gyakran kapcsolatban állnak a szoftverek tényleges fejlesztőivel is: egyfelől visszajelzéseikkel segítik a következő termékverziók minőségének és képességeinek további javítását, másrészt lehetőségük van mindenkinél korábban megismerni a még csak készülő, új szoftverek terveit, már működő verzióit.

Innen már több kolléga is erősíti a redmondi konzulensi csapatot, szívesen látják a külföldi csapatok a magyar tehetségeket.

Nemzetközi terméktámogató központ – GTSC (Global Technical Support Center). A Microsoft európai támogatóközpontjának szakemberei nagyvállalatok és Microsoft-partnerek számára nyújtanak szakmai segítséget felmerülő problémáik orvoslására mind az üzemeltetés, mind a fejlesztés terén. A GTSC munkatársai nemcsak a magyar, hanem szinte valamennyi európai ügyféllel foglalkoznak, és munkájuk részeként – a konzulensekhez hasonlóan – napi kapcsolatban állnak a szoftverek fejlesztőivel is.



A díj

Fejlesztői platform üzletág – DPE (Developer & Platform Evangelism). Ez az üzletág felelős a fejlesztőeszközökkel és fejlesztőkkel kapcsolatos kommunikációért, valamint szintén az ő feladatuk a vállalatokkal szoros kapcsolatot ápoló, nagyszámú független szoftverfejlesztő segítése, ösztönzése. Az itt dolgozó szakemberek elsődleges feladata, hogy a vállalatok és a fejlesztői közösség számára megmutassák a Microsoft fejlesztőeszközeinek és teljes platformjának lehetőségeit, tényleges értékét, és ezzel együtt segítsék a fejlesztők mindennapi munkáját. Ennek részeként – többek között – hozzájuk tartozik a teljes MSDN programsorozat is.

Marketing – CMO (Central Marketing Organization). A marketing feladata az értékesítés támogatása egyfelől azáltal, hogy kialakítja a megfelelő kommunikációs csatornákat, másfelől, hogy figyeli és elemzi a piaci és értékesítési trendeket, és ez alapján hangolja újra a későbbi kommunikációt. Ide tartozik az IT-szakemberekkel folytatott szakmai kommunikáció is (ez a TechNet), aminek elsődleges célja az IT-szakemberek elégedettségének és szakmai tudásának növelése.

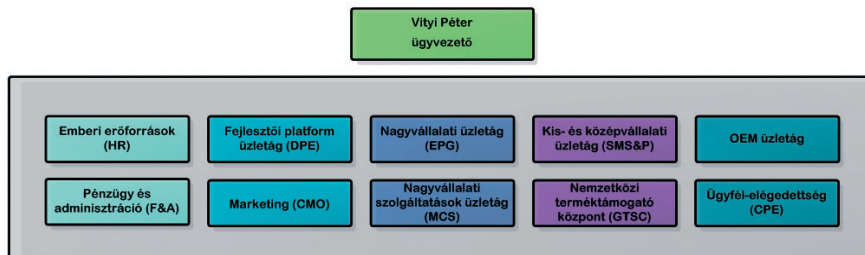
Nagyvállalati üzletág – EPG (Enterprise & Partner Group). A nagyvállalati üzletág foglalkozik a nagyobb méretű magyarországi vállalatokkal, beleértve a közületi felhasználókat, közintézményeket, oktatási intézményeket és kormányzati hivatalokat is. Az ügyfelekkel dedikált ügyfélmenedzserek

működik az a csapat, amelyik kifejezetten nagyvállalatok számára mutatja be a legújabb technológiákat, és segít elkészíteni a nagyobb informatikai projektek rendszerterveit, pilotjait. Hosszú évek óta az ő nevükhöz kötődnek a TechNet-események talán legjobb szakmai előadásai is (a teljesség igénye nélkül, de itt dolgozik többek között *König Tibor*, *Szalontay Zoltán* és *Koszó Károly* is).

Kis- és középvállalati üzletág – SMS&P (Small and Midsize Solutions & Partners).

mutató: Microsoft – 90 százalék, magyar átlag – 48 százalék), és munkatársaink a legtöbbet hozzák ki magukból. Ehhez a HR-es stratégia 5 fő pilléren nyugszik: a teljesítménymenedzsment, a bérezés és kiemelkedő juttatások, a szakmai fejlődés/karrier támogatása (hazai és nemzetközi tréningekkel), a jó vezetők és a kiváló munkakörülmények biztosítása.

Pénzügy és adminisztráció – F&A (Finance & Administration). A jól működő



A Microsoft Magyarország szervezeti felépítése (nagy vonalakban)

A kis- és középvállalkozásokra fókuszáló kereskedelmi egység; itt azonban az ügyfelek nagy száma miatt már nyilván nincs minden egyes ügyfélhez saját ügyfélmenedzser. Viszont sokkal nagyobb hangsúlyt kap a régió szerinti gondolkodás, valamint az értékesítési, megoldásszállító és oktatási partnerek támogatása, fejlesztése. Itt is fokozatosan épül és növekszik egy, az EPG-nél ismertetthez nagyban hasonlító szakmai csapat, ők elsősorban a partnerek szakmai továbbképzésében vesznek részt.

OEM-üzletág (Original Equipment Manufacturer). Nemrég vált önálló üzletággá. Ez a csapat kifejezetten az OEM-rendszerépítőkre, valamint az áruházakra és a dobozos szoftverek értékesítésére fókuszál.

Emberi erőforrások – HR (Human Resources). A HR szerepe nagy jelentőségű egy olyan cég életében, ahol ennyi különböző szakterülettel és tapasztalattal rendelkező

cégnek ez a szervezet is fontos része. Szakmai szemmel itt az lehet érdekes, hogy ide tartozik az a két-három fős IT-csapat is, akik a Microsoft Magyarország munkatársai számára biztosítják a napi munkavégzéshez szükséges informatikát.

Ügyfél-elégedettség – CPE (Customer and Partner Experience). A Microsoft kiemelt figyelmet szentel ügyfelei és partnerei elégedettségének javítására, véleményük meghallgatására. A CPE részleg elsősorban az ezzel kapcsolatos feladatokat látja el, így többek között ők felügyelik és üzemeltetik a telefonos és e-mailes ügyfélszolgálatot is.

Érdeemes lehet körülnézni

Akit mélyebben érdekel a munkánk, vagy szeretne többet tudni a cégről, nézzen körül blogjainkon. Egyre többen blogolunk, és a szakmai újdonságok mellett igyekszünk információval szolgálni mindennapi munkánkkal kapcsolatban is. Az egyik, amit érdemes lehet megnézni, az a nemrégiben hozzánk csatlakozott EPG-rendszermérnök, *Lepénye Tamás* webnaplója: a <http://lepenyet.spaces.live.com> címen érhető el.

Akit pedig esetleg állásajánlatok érdekelnének a fent ismertetett részlegek bármelyikében, nézzen körül a <http://www.microsoft.hu/allasajanlatok> oldalon.

Budai Péter

(i-pbudai@microsoft.com) Microsoft Magyarország



Az év munkahelye – dolgozói party

foglalkoznak, ők segítenek a vállalatoknak a számukra legmegfelelőbb informatikai megoldások megtalálásában.

Ide tartozik még a nagyvállalati partnerek menedzsmentje is. Azonban ez sem csupán értékesítési tevékenység. Az üzletág részeként

ember dolgozik közös célok érdekében. A HR azon kívül, hogy folyamatosan kutat a lehető legjobb, leendő Microsoft-munkatársak után, azon fáradozik, hogy az itt dolgozók elkötelezettsége folyamatosan magas legyen (a Hewitt szerinti 2006-os elkötelezettségi



BI

MI FELTESSZÜK
A BI-RE A PONTOT.

BUSINESS INTELLIGENCE (BI) TANFOLYAMOK A NETACADEMIÁNÁL!

A sikeresen működő vállalatok versenyképességének egyik titka, hogy tudatosan használják, elemzik és értelmezik a rendelkezésükre álló adatokat.

Az ehhez szükséges üzleti intelligencia (BI) technikai háttere minden SQL Server vásárlónál elérhető, bevetésre készen áll, hisz benne van a termékben.

Itt az ideje, hogy használjuk is!

BI-képzéseinken az alábbi kérdésekre adunk választ:

- ◆ Mire használható egy többdimenziós OLAP kocka?
- ◆ Miként használható jól a Reporting Services 2005?
- ◆ Mik az újdonságai az Analysis Services 2005-nek?
- ◆ Mit csinál az Integration Services 2005?

A képzések már elkezdődtek!

2007-ben a NetAcademia szolgáltatja a hiányzó dimenziót.

További információk:

[HTTPS://WWW.NETACADEMIA.NET/BI](https://www.netacademia.net/bi)

LEGYEN ÖNNEK IS OLCSÓBB!

Képzéscsomagjainkat 2007-ben 20%-kal olcsóbban **keretszerződésben** is megrendelheti.

Keretszerződéses partnereink számára további érték növelt szolgáltatásokat nyújtunk:

- ◆ Ingyenes tanfolyami utókövetés
- ◆ Ingyenes szaktanácsadás, egyedi konzultáció
- ◆ Ingyenes vizsgafelkészítő kit
- ◆ Ingyenes EU-pályázatfigyelés és tanácsadás

A keretszerződéssel kapcsolatban keresse
Szántó Zoltánt

E-mail:

SZANTO.ZOLTAN@NETACADEMIA.NET



NetACADEMIA

A LEGJOBBAKAT TANÍTJUK.

Cím: 1062 BUDAPEST, ANDRÁSSY ÚT 62.

TELEFON: (06 1) 472-1214
IRODAI MOBIL: (06 20) 369-6947
FAX: (06 1) 472-1215

INTERNET: WWW.NETACADEMIA.NET
E-MAIL: INFO@NETACADEMIA.NET

Microsoft®

Új korszak kezdődik. Készen állsz?



 Microsoft®
Office

 Windows Vista™

Microsoft®
Exchange Server 2007

A szoftvekről szakmai információk a **TechNet Portálon**, a www.microsoft.hu/technet/launch oldalon, valamint a Microsoft weboldalon, a www.microsoft.hu/windowsvista és www.microsoft.hu/office2007 oldalakon található.