

Már alkalmazott technológiák és új üzleti intelligencia megoldások összehangolása

GALLI RICHÁRD

Széchenyi István Egyetem, Műszaki Tudományi Kar, Informatika Tanszék
richard@sze.hu

Kulcsszavak: üzleti intelligencia, döntéstámogatás, integráció, vállalati rendszerintegráció

Nagyvállalati környezetben az üzleti intelligencia, annak bevezetése és a benne rejlő lehetőségek meglehetősen kurrens témának számítanak. Azonban egy ilyen rendszer adoptálása során számos nehézséggel szembesülhet a fejlesztő csapat, az egyik ilyen a már meglévő rendszereknek a bevezetésre kerülő megoldással való illesztése, operációs rendszer és adatbázis-szinten. A cikk arra keres választ, vajon milyen esély van összeférhetlenségre.

1. Bevezetés

Az üzleti intelligencia szó először 1989-ben került használatba. Legfőbb oka, hogy a szoftvercégek reklámszakemberei jobban eladhatónak érezhették ezt a kifejezést, az addig használatos döntéstámogató rendszer helyett. A két kifejezés között gyakorlati határvonal nincs. Minimális különbségként talán az alkalmazhatóságot lehet felhozni: kellően strukturált-e a megoldandó probléma? Régebben annak kellett lennie, ha döntéstámogató rendszert akart használni a vállalat, mára ez már nem feltétlen követelmény. Üzleti intelligencia rendszereknél több út is járható: a középpontban továbbra is az elemzés áll, de a technológia fejlődése lehetővé tesz komoly automatizálást, tömeges és részletesebb döntéstámogatást.

Számos esetben fordul elő az alábbi szituáció: már meglévő céges infrastruktúrához kell új üzleti intelligencia rendszert kapcsolni. Ideális lenne persze mindezt úgy megoldani, hogy a meglévő rendszereket és üzleti folyamatokat a lehető legkisebb mértékben (vagy egyáltalán ne) érintse a bevezetés.

A cikk második szakaszában kerül kifejtésre, hogy miért is okoz alkalmanként problémát az üzleti intelligencia rendszerek bevezetése erőforrás szempontból, továbbá bemutatásra kerülnek azok a pontok, ahol az üzleti intelligencia rendszerek illesztése kritikus lehet. A harmadik szakasz tartalmazza a vizsgálat eredményeit, górcső alá véve az elérhető üzleti intelligencia rendszereket és azok működési környezetét. A negyedik szakasz egy kitekintést tartalmaz egy új megoldás (irányzat), a valós idejű üzleti intelligencia rendszerek felé.

Vizsgáljuk meg, lehetséges-e a legújabb üzleti intelligencia megoldások adaptálása olyan környezetbe, ahol már meglévő hardver- és szoftverinfrastruktúra üzemel.

2. Üzleti intelligencia rendszerek és már meglévő környezetük

A feltételezés azért aktuális, mivel számos tanulmány lát napvilágot olyan témákkal, hogy hogyan célszerű,

hogyan költséghatékony egy vállalati informatikai rendszer felépítése, komoly üzleti intelligencia (BI, Business Intelligence) megoldások alkalmazása mellett. Ugyanakkor a tanulmányok nagy része figyelmen kívül hagyja azokat a tényezőket, melyek igencsak jellemzőek a vállalati rétegre. Az első tényező, miszerint a vállalat már régóta üzemel, és az üzleti intelligencia bevezetése alatt is üzemelnie kell – általában egyáltalán nem, vagy csak rendkívül korlátozott mértékű leállás engedhető meg. A második tényező az, hogy a vállalati rendszerek meglehetősen heterogének hardver és szoftver megoldások terén, azaz sokféle hardver dolgozik együtt és sokfajta szoftver van használatban. Ennek oka általában a vállalatok evolúciós fejlődésében található, ugyanis sok cégnél megtartják a régi rendszereket – vagy mert muszáj, annak célja miatt, vagy mert nincs pénz illetve erőforrás annak kiváltására.

Az üzleti intelligencia bevezetési projektek tervezésekor ezeknek a tényezőknek külön figyelmet kell szentelni – mely a szoftvermérnököket gyakran válaszüti elé állítja. A jóval költségesebb megoldás az, ha a bevezetés egyben bizonyos szintű átszervezést és modernizálást is jelent, ekkor az alkalmazott megoldások átgondolása is szerepet kap [1]. A költségkímélőbb megoldás az, ha megvizsgáljuk, hogy a már meglévő infrastruktúra képes-e ellátni a neki szánt többletfeladatot és ha igen, akkor ráültethető az üzleti intelligencia megoldás. Ám ha nem, akkor meg kell vizsgálni, milyen lehetőségek vannak világszerte, melyeket segítségül lehetne hívni.

Ahhoz persze, hogy a tervező csapat számolni tudjon az infrastruktúrával szembeni elvárásokkal, már a tervezés korai szakaszában ismerniük kell azokat a követelményeket, melyeknek a készülő üzleti intelligencia rendszer meg kell feleljen. A megvalósítandó funkciók (elvárások) ismerete esetén számolni lehet azok erőforrásigényével. Általánosan elmondható, hogy erőforrásigény alapján a funkciók a következőképpen csoportosíthatóak:

Alacsony erőforrás-igényű funkciók: Ide tartoznak azok a funkciók, melyek időszakosan működnek (tehát nem folyamatosan, így megfelelő ütemezéssel egymáshoz

igazíthatóak a terhelési hullámok). Ilyenek a hagyományos statisztikai módszereken alapuló számítások.

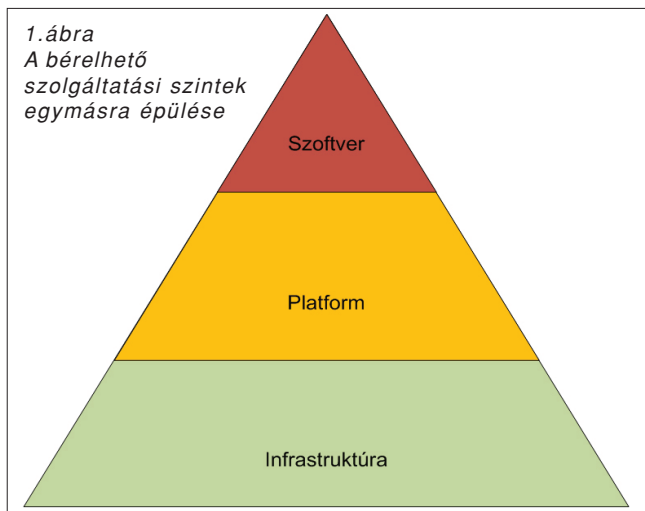
Nagy erőforrás-igényű funkciók: Azok a funkciók, melyek vagy folyamatosan működnek, vagy valós idejű működési feltételeknek kell eleget tenniük. Tipikusan nagy erőforrás-igényű a valós idejű üzleti intelligencia, a valós idejű CRM, valamint minden olyan döntési alkalmazás, melyben az épp adódó döntési helyzet kiértékeléséhez az összes eddigi döntési helyzet vizsgálata szükséges. A feldolgozandó/megvizsgálendő adatmennyiséggel arányosan nő a számítási igény is.

Amennyiben a bevezetéssel kapcsolatos követelmények közt nem szerepelnek nagy erőforrás-igényű funkciók, akkor tapasztalati úton megállapítható, hogy az üzleti intelligenciával kapcsolatos infrastruktúra-igény fedezhető a már meglévő infrastruktúra használatával. Ehhez persze szükséges a bevezetés előtti terhelési hullámok ismerete (ez felderíthető a bevezetési projekt helyzet-elemzési fázisában), majd ennek tudatában a rendszeres funkció-végrehajtás időzíthető az addigi hullámvölgyekre, egy olyan állapotot elérve, hogy ne legyenek túlterheléses időszakok (mivel akkor a végrehajtás sokszorosan lassabb lesz).

Abban a valószínű esetben viszont, ha igenis van igény nagy erőforrás igényű feladatok végrehajtására, át kell gondolni, hogy milyen lehetőségek elérhetőek [2]:

- Infrastruktúrát bérelni szolgáltatásként: az IT infrastruktúra teljes kiszervezettsége mellett annak bérletét jelenti.
- Platformot bérelni szolgáltatásként: a fejlesztői platform bérletét jelenti (az infrastruktúrával együtt).
- Szoftvert szolgáltatásként igénybe venni: a szoftver szolgáltatásként történő igénybevételét jelenti, egyfajta használatarányos bérleti modellel egybekötve. (Hozzá tartozik az infrastruktúra szolgáltatása is.)

A három szolgáltatási szint felfogható piramisként is (1. ábra), ahol a legrészletesebb szolgáltatási szintet a szoftvert szolgáltatásként modell képviseli, míg a legalacsonyabb szintűt az infrastruktúra szolgáltatásként való igénybevétele/nyújtása jelenti.



Gyakorlatilag a tervezési fázisban kell eldönteni, hogy melyik modellt lenne célszerű igénybe venni – jó hozzáállás lehet a három szolgáltatási szintet a Kesselring-algoritmus bemeneteként kezelni.

Ma 50-60 körülre tehető azon szoftverrendszerek száma, melyek üzleti intelligencia szolgáltatást kínálnak. (Megjegyzendő persze, hogy vannak összetettebb, sokrétű szolgáltatást nyújtó rendszerek, illetve egyszerűbb, csak egy-egy részfeladat elvégzésére képes szoftverek – ilyen részfeladat lehet az adatok összegyűjtése, migrálása, elemzések készítése, a riportkészítés, az adatvizualizáció.) Bár nyilvánvalóan az összetettebb rendszerek vannak jelen kisebb számban, a két szintet elválasztó vonal meghatározása igen nehéz feladat, mivel a piac sokszereplős, sok megoldással tarkítva.

A már meglévő/kifejlődött vállalatirányítási rendszerhez való illesztés két ponton lehet kritikus:

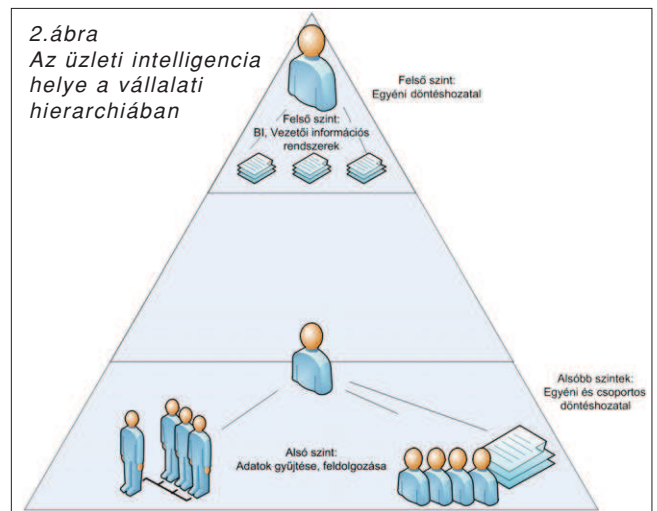
- az egyik a már működő operációs rendszer (megoldja az alatta fekvő hardverrel való kommunikációt),
- a másik a meglévő adatbázisokhoz való illeszkedés.

Így a meglévő vállalatirányítási vagy más rendszerek mellé (helyesebben inkább fölé – 2. ábra) történő illesztés előtt figyelembe kell venni, hogy melyik rendszerek képesek a meglévő operációs rendszeren működni, valamint képesek a már meglévő adatbázisokból adatok kinyerésére. Ezek után kell megvizsgálni, hogy a szűrés után fent maradt megoldások közül melyek képesek a funkcionális igények kielégítésére.

Annak az állításnak az eldöntésére, miszerint az üzleti intelligencia rendszerek tetszőleges konfiguráció (már-mint a meglévő platform és a szükséges adatbázis kapcsolat) mellé telepíthetőek, meg kell vizsgálni a piacon elérhető rendszereket. A vizsgálat 54 üzleti intelligencia funkcionális rendszerre terjedt ki. A vizsgált rendszerek nagy része kereskedelmi forgalomban kapható, de a valóságnak megfelelően tartalmaz a vizsgálat ingyenes, illetve nyílt forráskódú termékeket.

3. A vizsgálat eredményei

A vizsgálat egyfelől bebizonyította, hogy ebben a szektorban is folyamatban van a klasszikus kliens-szerver



architektúra elhagyása és webes architektúrára áttérés. Azok a rendszerek, amelyek vagy a klasszikus klienszerver architektúrára, vagy desktop környezetre épülnek, (kliensoldalt tekintve) leginkább az elterjedt keretrendszereket alkalmazzák, amit a JAVA technológia vagy a .NET keretrendszer nyújt. A megállapítás pozitív oldala az, hogy mivel mindkét keretrendszer elérhető az elterjedt operációs rendszereken, így ezen az oldalon nem jelentkezhethet probléma a használatot illetően.

A meglévő szervereken futó operációs rendszerek tekintetében (amennyiben új hardver illetve operációs rendszer is kerül beszerzésre az üzleti intelligencia rendszer bevezetésekor, az nagyobb árkategóriába sorolja ugyan a beszerzést, ám nem is jelentkezik a megkötés a rendszer architektúrájának tervezése során) a termékek rugalmassága más képet fest. Ebben a tekintetben könnyedén osztható két részre a „mezőny”, a csoportosításnak jó alap az, hogy szerveroldalon több operációs rendszer támogatott-e.

A nagyobb, összetettebb rendszerek (főként azok, melyek tekintélyes több évtizedes múltra tekintenek vissza), evolúciójuknak köszönhetően általában támogatják az iparban elterjedt szerver operációs rendszereket.

Ezek a következők:

- **z/OS:** Az IBM mainframe gépeinek operációs rendszere. Nagy előnye az elsőrendű visszafelé kompatibilitás, a nagyméretű memóriák és a mainframe technológiák támogatása. A rendszer a legendás OS/390 követője.
- **Unix/Linux:** Az 1969-es Unix kifejlesztése óta – mely az első hordozható operációs rendszer volt – számos Unix-szerű operációs rendszer látott napvilágot (BSD, Linux). A mindenki számára elérhető, ingyenes, nyílt forráskódú változat a Linux, persze a kategórián belül is találhatóak kereskedelmi forgalomban lévő változatok.
- **Solaris:** A Sun Microsystems által fejlesztett SPARC és x86 támogatással rendelkező Unix-szerű operációs rendszer szerverekhez és munkaállomásokhoz egyaránt.
- **Suse Linux:** A legrégebb óta létező nyílt forrású Linux-változat. Elérhető a támogatott, vállalati változat is.
- **Red Hat Enterprise Linux:** A vállalati környezetben megtalálható, másik jelentős részesedéssel bíró változat.
- **Windows:** A Microsoft cég szerveroldali operációs rendszeréből három változat terjedt el ebben a szegmensben nagyobb mértékben: A Windows NT, mely megalapozta a cég szerveroldali térnyerését, a Windows Server 2003, illetve a Windows Server 2008 – utóbbiból elérhető az R2 változat is. Az olyan webes BI-rendszerek tipikus szerveroldali operációs rendszere, melyek a .NET keretrendszer igény miatt IIS web szerveret igényelnek.

A másik csoportba a nem összetett, funkcionálisukat tekintve szegényesebb szoftverrendszerek tartoznak, melyek csak egy-egy részfeladat megoldására alkalmasak és nem rendelkeznek ekkora rugalmassággal sem

telepíthetőség tekintetében. Ezek a kisebb rendszerek legtöbb esetben nem rendelkeznek szerveroldali telepítési és számítási lehetőségekkel, így desktop szoftvereknek minősülnek. Mivel ezeket a rendszereket célszerűbb és gyorsabb valamilyen magas szintű nyelven implementálni, kivitelezésük során a JAVA nyelv és a .NET keretrendszert támogató nyelvek jöhetnek szóba. Az ilyen megvalósítás előnye, hogy a későbbiekben a szoftverek működtetéséhez elegendő a számítógépre a JAVA futtatókörnyezet, vagy a .NET keretrendszer előzetes telepítése. A legelterjedtebb operációs rendszereken ezek elérhetőek.

A vizsgált megoldások 25,9%-a íródott JAVA nyelven, míg 48,1%-uk a Windows-os környezetre készült. A két halmaz közt itt átfedés nincs, de megemlítendő, hogy az ingyenes és nyílt forráskódú rendszerek mindegyike JAVA platformra készült.

A JAVA futtatókörnyezet elérhetőségét az 1. táblázat mutatja.

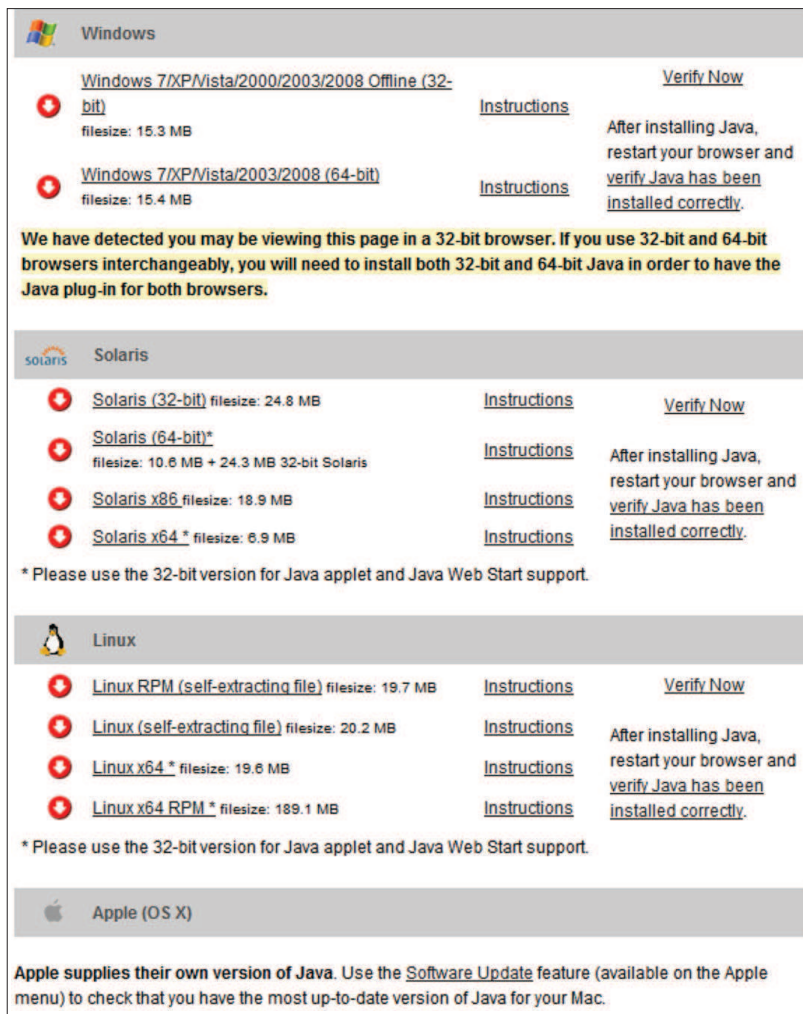
	32-bites változat	64-bites változat
Microsoft Windows 7	•	•
Microsoft Windows XP	•	•
Microsoft Windows Vista	•	•
Microsoft Windows Server 2003	•	•
Microsoft Windows Server 2008	•	•
Microsoft Windows 2000	•	
Sun Solaris	•	•
Linux	•	•
Apple OS X	van, saját verzió	

1. táblázat Java elérhetőség

A .NET keretrendszer a Microsoft operációs rendszerein elérhető (adott változatokban előre telepítve megtalálható bizonyos verzió, a felett tetszőlegesen frissíthető), Linux rendszerekre pedig a MONO Projekt keretében fejlesztik a keretrendszert (Mac OS X és Solaris operációs rendszerekre is elérhető). Általában a legfrissebbnél eggyel régebbi változat elérhető a MONO keretben belül.

Szakmai (és online) fórumokon gyakran heves vitát vált ki, hogy melyik elgondolás a jobb befektetés: az ingyenes operációs rendszerek és szoftverek használata, melyek működtetéséhez (általában) drágább szakemberek kellene, vagy célszerű inkább drágább, de egyszerűbben használható szoftvereket alkalmazni, melyek nem rendelkeznek akkora szakemberigénnyel.

Az adatbázisokhoz való kapcsolódás a másik fontos technológiai tényező. Itt azt kell megvizsgálni, hogy a piacon fellelhető üzleti intelligencia megoldások milyen adatbázis típusokat képesek adatforrásként használni, illetve azt, hogy ezek a lehetőségek milyen mértékben garantálják a széleskörű kompatibilitást.



3. ábra Java keretrendszer elérhetőségei (www.java.com)

A vizsgálat során a következő adatbázis-kapcsolati lehetőségeket találtuk:

Gyártóspecifikus adatforrások:

Azon termékekre jellemző, amik olyan nagy gyártótól érkeznek, akik képesek ügyfeleik számára a teljes technológiai stack nyújtására (Microsoft, Oracle) és a saját adatbázis kezelő termékek kiemelt támogatása és integrációja révén próbálják ügyfeleiket arra ösztönözni, hogy a szükséges komponenseket is tőlük szerezzék be.

A vizsgált rendszerek 44,4%-a képes valamilyen gyártóspecifikus adatforrás hasznosítására.

SQL (Simple Query Language):

A szabványosított lekérdező nyelv, melyet szinte minden adatbázis kezelő motor ismer [5]. Gyártóspecifikus eltérések persze vannak az alap szabványtól, mint a Microsoft által használt MS-SQL, vagy az Oracle által használt PL/SQL.

Az összes vizsgált megoldás képes az SQL-utasítások feldolgozására, azonban megjegyzendő, hogy bizonyos esetekben (saját programozási nyelv miatt) az SQL-kódrészleteket be kell burkolni.

XML:

A W3C szöveges formátuma, melyet kifejezetten az internetes anyagok terjesztésére lett specifikálva. Tá-

mogatja az unicode karaktereket, így minden nyelven használható, és jól használható webszervízek készítésénél, illetve adatforrások előállításánál [6]. Használatához speciális API szükséges, ám ez már minden programozási nyelven rendelkezésre áll.

A vizsgált rendszerek 72%-a képes XML forrás hasznosítására.

ODBC (Open DataBase Connectivity):

Az adatbázisoktól, programozási nyelvektől és operációs rendszerektől függetlenek fejlesztett interfész [7]. Segítségével a szoftvereknek elegendő az ODBC által definiált szintaxis ismerete, onnantól a driver feladata ezen szintaxis konkrét adatbázis kezelő rendszer számára értelmezhető nyelvre fordítása [8].

A vizsgált megoldások 54%-a képes ODBC-adatforrások hasznosítására.

JDBC (Java DataBase Connectivity):

A Java programozási nyelvhez tartozó, relációs adatbázisok eléréséhez szükséges API [9]. A JDBC-ODBC híd segítségével Java nyelvből is elérhető az ODBC funkcionalitása. A .NET keretrendszerben a megfelelője az ADO.NET [10].

Minden vizsgált JAVA-alapú rendszer képes JDBC-adatforrások alkalmazására.

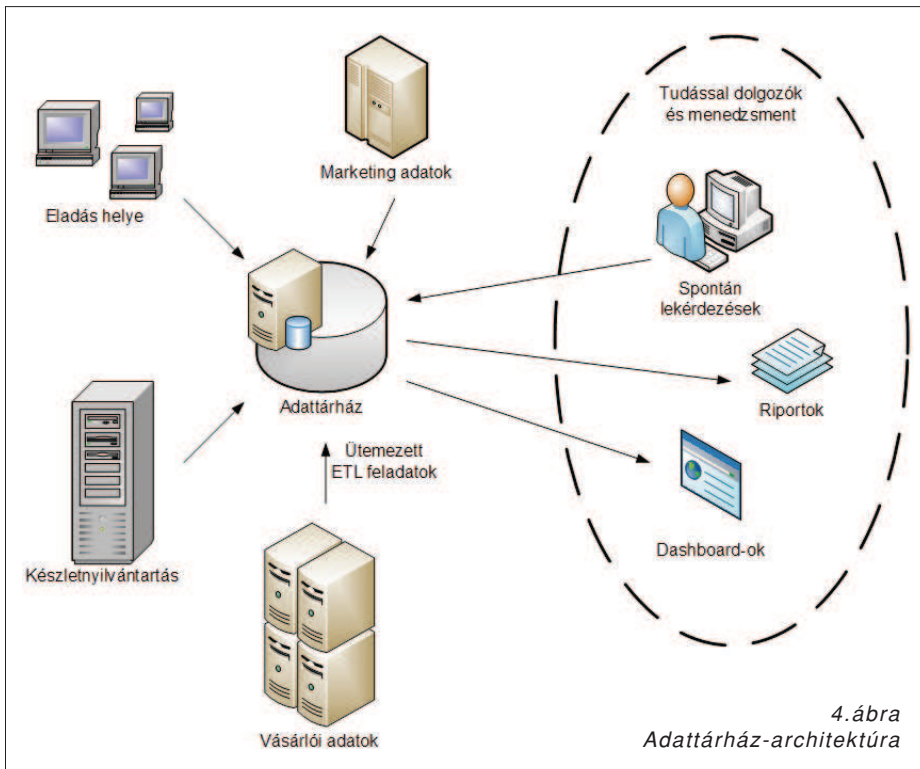
Az üzleti intelligencia rendszerek fejlesztése közben a fenti szabványos technológiák alkalmazásának köszönhetően

(persze részben köszönhető ez a fejlesztéshez alkalmazott nyelvnek) a vizsgált szoftverek képesek tetszőleges adatbázissal való munkára. Kivételek persze adódhatnak, ezekhez a speciális esetekhez célszerű más kapcsolati megoldás után nézni – ha bizonyítottan szükséges.

4. Más kapcsolati megoldások és valós idejű üzleti intelligencia rendszerek

A mai versengő gazdasági környezetben – ahol igen magasak a vásárlói elvárások – a döntések legfrissebb adatokra alapozása tovább javíthatja a vásárlói kapcsolattartást, növelheti a bevételeket, és az operatív hatékonyságot is maximalizálhatja. A technológia fejlődése és az adatfeldolgozási sebesség jelentős növekedése lehetővé teszi a klasszikus, jól bevált adattárház rendszerek valós idejűvé tételét. Ennek eredménye a valós idejű üzleti intelligencia (RTBI, Real Time Business Intelligence).

Minden üzleti tranzakció már létrejövetelkor bekerül abba a valós idejű rendszerbe, mely a vállalat állapotáért felel. Így az RTBI rendszer nem kizárólag a mára klasszikusnak mondható stratégiai funkciókat valósítja meg (mint az adattárházazás vagy információk és tudás előállítás a múlt adataiból), hanem valós idejű



4. ábra
Adattárház-architektúra

Az adattárház-koncepció (4. ábra) egy olyan informatikai rendszert jelent, mely a vállalat minden adatának repozitóriuma olyan formában [4], hogy azon elemzéseket lehessen készíteni, riportokat lehessen generálni a menedzsment és más tudással dolgozók számára. Ha ez a cél adott, számos kihívással kell szembe nézni:

- Az adatokat számos egymással nem kompatibilis rendszerből kell összegyűjteni.
- Ugyanaz az információ különböző rendszerekben, különböző formában lehet tárolva, ráadásul el is térhetnek egymástól. Meg kell határozni, hogy az adott adat melyik és milyen formában helyes.
- Meg kell határozni, hogy a folyamatosan változó adatokat milyen gyakran frissítsék az adattárházakban.

taktikai támogatást is nyújt, aminek eredményeként a vállalat azonnal képes lesz reagálni az eseményekre. Jól használható tehát a klasszikus adattárház rendszerek kiváltására és a vállalati rendszerek integrációjára (EAI, Enterprise Application Integration) is.

Az RTBI másik neve eseményvezérelt üzleti intelligencia. Ahhoz ugyanis, hogy valós időben lehessen reagálni, az események bekövetkezésekor azonnali beavatkozásra van szükség, itt nem engedhetők meg óras vagy perces késések sem. Az RTBI-megközelítés használatával a vállalat olyan hosszú távú stratégiát valósíthat meg, mely segíti a műveletek optimalizálását és egyidejűleg képes lesz az eseményekre azonnal reagálni [3].

Valós idejű rendszerektől (mint minden más kontextusban, itt is) megkövetelt valamilyen szigorú válaszidő betartása.

Tíz éve még elegendő információnak számított az, ha felismerhető volt, hogy melyik termék iránti kereslet a legnagyobb, így abból kellő mennyiség volt folyamatosan raktározható. A mai agresszív marketing környezetben ez már nem elegendő. Ma inkább az lehet a követendő példa, ha a vásárlók korábbi vásárlásai alapján kiegészítő terméket ajánl az üzlet, esetleg árengedménnyel, így egyszerre valósulhat meg a nagyobb haszon, a vásárlási kedv növekedése és a vásárlási élmény javulása. Ebben a komoly elvárásokkal teli, versengő környezetben még a műveleti hatékonyság is javítható.

Az információtechnológia fejlődésével a vállalatok egyre több rendszerüket automatizálták. Azóta ezekben a rendszerekben nagy mennyiségű kihasználatlan adat van. Az eladási, könyvelési, termelési, HR-rendszereken kívül ugyancsak sok szignifikáns történeti, aktuális vagy prediktív adat lelhető fel.

- Meg kell határozni, hogy a hatalmas adatmennyiség hogyan reprezentálható használhatóan és egyszerűen.

Ahhoz, hogy ezek az igények kielégíthetőek legyenek, számos támogató alkalmazás kifejlesztésére volt szükség, ezek a következők:

- ETL-folyamatokat megvalósító alkalmazások, melyek az adatok mozgatásáért felelősek (az adattárházba).
- Adatbányászati és elemző szoftverek, melyekben lehet egyedi módszereket definiálni és ezek elvégezhetőek az adattárház tartalmán.
- Egyszerű, de lényegre törő megjelenítést támogató riportkészítő eszközök.

5. Összefoglalás

Végezetül általános megállapításként elmondható – az üzleti intelligencia rendszerek hatékonyságának és szükségességének megkérdőjelezése nélkül – hogy egy meglévő, üzemelő vállalatnál komoly szoftverfejlesztési kihívás a már meglévő rendszerekhez történő üzleti intelligencia rendszer illesztése.

Ennek a kihívásnak pedig nem egyszerű megfelelni: ugyanúgy készülni kell rá, mint minden más szoftvercélú projektre (létezik hozzá ajánlható módszertan is, hogy szigorú lépésekre bontja a folyamatot) és legalább ugyanakkora – ha nem nagyobb figyelemmel kell eljárni.

Az egyik kezdeti lépés a később alkalmazandó termék kiválasztása: itt fontos szempont az, hogy képes-e a választandó rendszer együttműködni a már meglévő (üzemelő) rendszerekkel, adatbázisokkal és adatforrásokkal.