

Információmenedzselés

III. rész: Az adatbázis elvi alapjai

Korszerű információs rendszer ma már elképzelhetetlen az adatbázis mint alap nélkül. Az adatbázis nem pusztán az ismeretek szervezett együttese, hanem egyben az ismeretekkel való gazdálkodás nélkülözhetetlen kelléke is. Az információs pokolba vezető út jószándékú, de az adatbázishoz nem értő fejlesztőkkel van kikövezve... Ezért nincs más megoldás: a vezetőnek jobban kell értenie az ismeretek célszerű elrendezéséhez (is), mint a beosztottjainak. A tengerentúlon ezt a témát a „business administration” fakultásokon kimerítően taglalják. Nekünk meg kell elégednünk azzal, hogy távirati stílusban összefoglaljuk az ismeretek célszerű és gazdaságos elrendezésének legfontosabb alapelveit.

Szemléletváltás szükséges

Az előző részben az adatot értelmezhető, de nem értelmezett ismeretként, az információt pedig értelmezett adatként – vagyis a személyes háttértudáshoz kapcsolódó új ismeretként – határoztam meg. Az informatikában minden az ismeret körül forog. Ebből logikusan az következne, hogy az információs rendszerek célszerű kialakításához mindenekelőtt az „ismeretek ismeretére” lenne szükség.

Tisztában kellene lennünk általában a valóságos tények (elsődleges valóság) és az azok tükrözésére szolgáló ismeretek (másodlagos valóság) viszonyával. Tudnunk kellene azt, hogy melyek az ismeretek alkotóelemei, összetevő tényezői. Át kellene látnunk ezek egymáshoz való viszonyait és a rájuk vonatkozó feltételeket. Mivel a valóság sokszínű, az azt tükröző ismeretek sem egyneműek, és ezért elkerülhetetlen, hogy az ismereteket osztályozzuk, illetve minősítsük. A valóság állandóan változik. Ebből pedig az következik, hogy az ismeretek előbbi statikus

aspektusai mellett tekintetbe kell vennünk a dinamikus szempontokat is.

Az ismerettel csak az tud bánni – a vele való gazdálkodásról nem is beszélve –, aki alaposan ismeri annak természetét. Ehhez képest eléggé meglepő, hogy az információs rendszerek fejlesztésének résztvevői általában meglehetősen járatlanok az ismeretek ismeretelméleti vonatkozásaiban, tehát végeredményben a szó valódi értelmében vett informatikában. Ennek a kitételnek az igazságát jól mutatja az a megközelítés, ahogyan az emberek manapság magát az adatbázist szemlélik.

- A fejlesztők nem információs rendszerben, nem alkalmazási feladatban, hanem az annak eszközeként szolgáló szoftverben gondolkodnak. Nem annyira a felhasználásokat kiszolgáló adatbázis – a cél – lebeg a szemük előtt, mint inkább az adatbázis-kezelő rendszer [DBMS – database management system], tehát az eszköz, ami egy bonyolult alapszoftver. Ahogyan ez a programcsomag változik, úgy módosul a szemléletük magáról az adatbázisról mint lényegről is. Számukra a „mit” és a „hogyan” az eszköz által meghatározott, és ezt a szemléletüket a fejlesztés többi résztvevőjére is rákényszerítik.

- A felhasználók nézete másként korlátos. Ők azt hiszik, hogy a számukra egyénileg fontos ismeretek halmaza az adatbázis. Ebből persze az következik,

hogy az X. felhasználónak is van egy úgynevezett „adatbázisa”, meg az Y. és a Z. alkalmazónak is van egy-egy „adatbázisa” – csak éppen a szervezet egészének nincsen olyan szervezett ismerethalmaza, amely nem redundáns, konzisztensen összefüggő – egyszóval: integrált – módon tartalmazná az adatokat.

- Végül a vezetők csodákat várnak az adatbázistól mint „új technológiától”. Nem fogják fel, hogy informatikai csodák nincsenek; az adatbázis mint olyan egyáltalán nem „új” dolog, hiszen ez az ismeretszervezési megoldás mintegy 30 éves; lényege nem a

“

Eléggé meglepő,
hogy az információs rendszerek
fejlesztésének
résztvevői általában
meglehetősen járatlanok az
ismeretek ismeretelméleti
vonatkozásaiban,
tehát végeredményben
a szó valódi értelmében vett
informatikában.

”

„technológiában”, hanem éppen az adatok megfelelő menedzselésben keresendő; ha pedig csak technológiaként alkalmazzák, akkor az adatbázis minden többleteredmény nélkül a többszörösébe fog kerülni, mint a „hagyományos” adatfeldolgozás. Éppen ezért szinte minden adatbázis-szakkönyvben az egyik legfontosabb kitétel így hangzik:



Nincs értelme adatbázis-fejlesztésbe fogni a vezetés értő (!) elkötelezettsége és aktív (!) közreműködése nélkül.



„Nincs értelme adatbázis-fejlesztésbe fogni a vezetés értő (!) elkötelezettsége és aktív (!) közreműködése nélkül!”

Mármost a vezetés elkötelezettségi szándékát sokszor tapasztaltam. Aktív közreműködését szinte sohasem, aminek a hozzáértés hiányából fakadó ódzkodás és a rejtett félelem az oka. Ezért nem meglepő, hogy Magyarországon eddig még sehol sem találkoztam valóban jól megépített adatbázissal. Mivel pedig a jól megépített adatbázis az információmenedzselés „sine qua non”-ja, az olvasó levonhatja a végső következtetést...

Ezen a mai helyzeten csak úgy lehet változtatni, ha a menedzserek szemléletet váltanak és arra kényszerítik a fejlesztések többi résztvevőjét is. Ehhez pedig az szükséges, hogy a vezető gyakorlati és valódi informatikusként próbálja meg szemlélni a szervezet adaterőforrásának a lényegét. A vezetőnek két dolgot kellene megértenie. Az egyik az, hogy az adatbázis-kezelő rendszert mint szoftvert nem szükséges ismernie ahhoz, hogy jól megépített adatbázist készíttessen a fejlesztőkkel, és ezért nincs semmi oka a tartózkodásra. A másik az, hogy az ismeretekhez viszont jobban kell értenie, mint akár magának a fejlesztőnek, és ezért jogosak az aggályai, ha járatlan azoknak az általános elvi elrendezésében.

Ennek és a következő cikk részletnek az a célja, hogy megpróbálja egyszerűen összefoglalni az adatbázis amúgy bonyolult lényegét, és ezzel eloszlassa az említett aggályokat.

Egyed és tulajdonság

Most egy rövidke „nyelvtanóra” következik. Ezen nem kell meglepődni. Az ismeretek közlésének a legfontosabb eszköze a természetes – írott és/vagy beszélt – nyelv. Ezért a nyelvtnak nagyon is sok köze van az adatbázisokhoz, amint az hamarosan kiderül.

(Csak érdekességképpen említem, hogy a hetvenes években az ismeretek alapvető tárolási egységét – a rekordot – egyes környezetekben „mondhatnak”, magát az elemi tárolt ismeretet – amit az előző részben adattételnek tituláltam – „szónak” nevezték. Tehát a vevőre vonatkozó ismeretek teljes sora egy „mondatt”; a vevőkód, vevőnév stb. adattétel pedig egy-egy „szó”.)

Annak szemléltetéséhez, hogy nyelvtani alapokon miként érthető meg az adatbázis lényege, vegyük példának a következő két hétköznapi mondatot:

A BMZ 873 rendszámú gépkocsi / típusa Lada.

Az FGS 802 rendszámú gépkocsi / típusa Ford.

A kijelentő tőmondatokban van alany és állítmány. Mindkettő további két részre oszlik. A „BMZ 873 rendszámú gépkocsi” alanyban az ún. legközelebbi nem [latinul: genus proximum] a gépkocsi, a megkülönböztető jegy [latinul: differentia specifica] a BMZ 873 rendszámú. A mondatokban nem könyvekről, személyekről, számlákról esik szó, hanem gépkocsikról. Nem tetszőlegesekről, hanem a megadott rendszámok által meghatározottakról.

Mármost azt az állapotaitól függetlenül létező dolgot, izét, valamit, amit ismeretekkel akar valaki leírni, az informatikában egyednek nevezik. A konkrét, a valós egyedekben közös vonásokat fedezünk fel, ezért azokat egyed típusokba soroljuk. Ilyen egyed típus a gépkocsi, de más példa esetén ilyen lenne a könyv, a személy, a számla is. A besorolás után az adott típusba tartozó jelenséget annak egyed-előfordulásának hívjuk. A BMZ 873 rendszámú dolog a gépkocsi egyed típus egyéki előfordulása. Adott környezetben és adott időpontban az egyed típus az előfordulások meghatározott halmazával rendelkezik. Például az X. biztosítónál más ez a halmaz, mint az Y. biztosítónál, és mindkettőnél naponta változik. (NB.: A fentebbi „állapotaitól függetlenül” kitétel arra vonatkozik, hogy a gépkocsi mint egyed lényege nem módosul attól, hogy átfestik, kárt szenvedett, tulajdonosa megváltozik stb.) Vegyük észre, hogy a mondatok alanya eleve kijelöli az egyed típust (legközelebbi nem) és annak vonatkozó előfordulását (megkülönböztető jegy), tehát azt az ismeretekkel leírt objektumot, amiről szó van. Ezért úgy gondolom, hogy az egyed fogalom megértése nem okozhat nehézséget.

Az állítmány analóg módon szintén két részből áll. Az első mondatban a típusa a legközelebbi nem, a Lada pedig a megkülönböztető jegy. A példamondatokban nem a kocsi színéről, súlyáról stb., hanem a típusáról van szó. Az elsőben ez a típus Lada – és nem Opel.

Az informatikában tulajdonságnak hívjuk azt a sajátosságot, vonást, jellemzőt, amivel az egyedeket le akarjuk írni. A típus egy tulajdonságtípus, a Lada egy tulajdonság-előfordulás, amit tulajdonságértéknek hívunk. A tulajdonságtípus adott környezetben és időpontban az értékek meghatározott halmazával rendelkezik. A mondat állítmánya utal a tulajdonságtípusra (legközelebbi nem) és annak vonatkozó értéké-

re (megkülönböztető jegy). Úgy vélem, hogy ebben az első megközelítésben a tulajdonság megértése sem jelenthet problémát. Az egyeddel és a tulajdonsággal kapcsolatos tényezők együttesét az 1. ábra mutatja.

1. ábra

Az ismeret négy dimenziója

Gépkocsi

Rendszám	Típus
BMZ 873	Lada
FGS 802	Ford

Az egyedtípus/-előfordulás, tulajdonságtípus/-érték az ismeret úgynevezett négy alapidimenziója, ami táblázatos formában is szemléltethető. A táblázat neve utal az egyedtípusra. A tábla sorai az egyed-előfordulások, oszlopai a tulajdonságtípusok. A sorok és az oszlopok találkozásánál találhatók a tulajdonságértékek. Az első dimenzió – az egyedtípus – akkor mutatkozna meg pregnánsabban, ha lenne egy másik – például TULAJDONOS – táblázatunk is.

Mármost mindennapi közléseink nem csak tömondatok formáját öltik. Lehet bennük több állítmány illetve tárgy, jelző is. A következő két mondat tartalma lényegileg azonos:

A BMZ 873 rendszámú gépkocsi / típusa Lada és színe bordó.

A BMZ 973 rendszámú / bordó gépkocsi típusa Lada.

Az állítmány, a jelző stb. mind egy-egy tulajdonságra utal. A fenti közléseket formálisan a 2. ábra mintájára tükrözhetjük.

2. ábra

Egyedek és tulajdonságok

Gépkocsi

Rendszám	Típus	Szín	...
BMZ 873	Lada	Bordó	...
FGS 802	Ford	Bordó	...

Nem nagyon téved az, aki a 2. ábrán egy olyan mini adatállományt vél felfedezni, aminek két rekordja (sora) és három adattétele (oszlopa) van. (NB.: A „...” azt mutatja, hogy van több is, de azok egyelőre nem érdekelnek bennünket.) A részleges tévedés abban rejlik, hogy amennyiben a hétköznapi mondatok és a rekordok közötti összefüggések ennyire egyszerűek lennének, akkor nem volna szükség adatbázistervezésre. Ez az ábra még nem tudja jelezni, hogy mely

pontokon lépnek be a tudatos megfontolásra készítő tényezők. Az esetleges problémákra majd a következő passzusban és azt követően térek ki.

Azonosító

Minden ismeret – és így szükségszerűen minden adatbázis – az általunk elképzelt egyedek és tulajdonságok együttesén alapul. Az egyed és a tulajdonság két értelemben is relatív fogalom. Egyrészt rajtunk múlik, hogy egy jelenséget egyedként és/vagy tulajdonságként akarunk-e szemlélni. Például a „szín” lehet a gépkocsi egyik tulajdonsága, de lehet olyan önálló jelenség – vagyis egyed – is, amit saját ismeretekkel akar valaki leírni. Mondjuk a biztosítónál azért, hogy a színnek a balesetekre gyakorolt hatását is elemezhessek. Másrészt az egyedekről nem tudunk anélkül beszélni, hogy ne azok tulajdonságára hivatkoznánk. Ezért minden egyednek kell, hogy legyen egy (vagy több) olyan tulajdonsága, amellyel a jelenséget kölcsönösen és egyértelműen be tudjuk határolni. Vagyis amely minden egyed-előfordulásra más-más értéket vesz fel. Példánkban ilyen jellemző volt a rendszám, amely a gépkocsiknak a helyettesítőjeként, reprezentánsaként szolgált.

Az egyed-előfordulások egyedi behatárolására alkalmas tulajdonságtípust azonosítónak nevezzük. A logika törvényei szerint az azonosító kétféle lehet: nominatív (név szerinti) és deskriptív (leírás alapján).

- *Nominatív* az azonosítás akkor, ha a dolgot „néven nevezzük”, ahol a „név” alatt nem mindig tulajdonnevet kell érteni. A logika szabályai szerint például a rendelésszám is név, hiszen ha megadjuk annak értékét, akkor ki tudjuk választani a megfelelő rendeltést. Ilyen értelemben név volt a személyi szám is, ami tökéletesen alkalmas volt a személyek egyedi behatárolására. Velük szemben nem tekinthető jó „névnek”, vagyis nominatív azonosítónak a rendszám. Azért nem, mert annak értékei csak korlátozottan – Magyarországon és adott időpontban – egyediek. Például az FGS 802 rendszám némi készpénzért átörökíthető lenne a következő kocsimra is, és ez komoly zavarokat okozna az azonosításban, mondjuk a régi kocsim után fennmaradó tartozásaim rendezésében.

- Ha a nominatív azonosítás lehetetlen, akkor két utat választhatunk. Az egyik az, hogy „névként” felveszünk egy új mesterséges tulajdonságot. A személyek vonatkozásában ilyen manapság a TAJ-szám, aminek a használata viszont korlátos. A másik az, hogy deskriptív azonosításhoz folyamodunk. A személyi szám és a TAJ-szám hiányában a személyeket a leíró tulajdonságok sorával – név, születési dátum, hely stb. – próbáljuk beazonosítani. Ez sem biztonságos megoldás, jóllehet ma már a névváltoztatás nem gyakori.

A megfelelő azonosító kiválasztása nem könnyű feladat, mivel nemcsak az egyértelműség, hanem a

kényelem is követelmény. Például a gépkocsinál az alvázsám megfelel az első igénynek, de a 24 karakterből álló jelsorozat kezelése minden, csak nem kényelmes. Ráadásul az azonosítóra az egyedi behatároláson kívül más szerep is hárul (lásd alább a kapcsolatot) és így szerintem a célszerű azonosító megtalálása az egyik legkomolyabb informatikai kihívás. Az azonosítónak ugyanis térben és időben stabilnak kell lennie, aminek az ellenkezőjére csak egy példát említek.



A kérdéses azonosító nem jól szolgálta a célját, mert állandó változása miatt a valós jelenséget nehéz volt nyomon követni.

Ezt a problémát nagyon komolyan kell venni, mert a rossz ügyintézésnek sok esetben a rosszul megválasztott azonosító az oka!



Régebben a biztosítási szerződéseim számában mint azonosítóban szerepelt a kötvényemet kezelő fiók megjelölése is. Mármost fiókokat összevonnak, megbontanak; ha elköltözöm, a kötvényemet fiókok között átadják stb. Tehát a kérdéses azonosító nem jól szolgálta a célját, mert állandó változása miatt a valós jelenséget – a kötvényt – nehéz volt nyomon követni. Ezt a problémát nagyon komolyan kell venni, mert a rossz ügyintézésnek – általában az ismeretek el- és összekeveredésének – igen sok esetben a rosszul megválasztott azonosító az oka!

Az első absztrakció gondjai

Az egyedabsztrakció nem könnyű informatikai feladat. Az informatikában extenzióknak nevezik az egyedtípus fogalmi kiterjedését, ami parlagian szólva azt jelenti, hogy mi mindent óhajtunk az egyedtípus körébe tartozónak tekinteni. Tegyük fel, hogy járműbiztosításokról van szó. E biztosítás szempontjából bizonyára nem tekinthető járműnek – holott a valóságban az – mondjuk az önjáró cséplőgép. Ezért tehát el kell könyvelnünk, hogy a hétköznapi és az informatikai fogalmaink nem azonosak.

Ugyanakkor a kismotor biztosítható ezzel a módozattal, tehát az gépjármű, ámde olyan, aminek nincs se rendszáma, se alvázszáma. Mivel fogjuk a kismotorokat reprezentálni, vagyis azonosítani? Rájuk nézve mesterséges azonosítóértékeket fogunk kiadni, tehát a valóságtól el fogunk rugaszkodni, és az ilyen természetellenes kényszerű megoldás mindig bajokat

okoz. A vezetőnek ügyelnie kellene arra, hogy a fejlesztők akkor is hajlamosak az örült absztrakciókra – a mesterséges adatok tucatjainak a bevezetésére –, amikor arra nincs is valós kényszer. Ezzel a rendszert bonyolítják; a felhasználó számára érthetlenné tesszik; végeredményben pedig az adatbázis sokkal „drágább” lesz az indokoltnál.

A tulajdonságabsztrakció is komoly informatikai feladat. Az informatikában intencionálnak nevezik a tulajdonságtípusok fogalmi kiterjedését, ami egyszerűen szólva azt jelenti, hogy a jelenségek milyen tulajdonságait tartjuk fontosaknak. Például a gépkocsit jellemzi annak hossza, de ez az adat nem szerepel az én biztosításomban, amiből arra következtetek, hogy a biztosító ez az ismeret nem érdekli. Az pusztán a felhasználó magánügye, hogy milyen tulajdonságokat köt az egyedekhez. Az viszont már az informatikus dolga, hogy törődjön a stabilitással és az egyértelműséggel.

Ellenpéldaként említhetem magát a rendszámot. Az olvasó most már nyilván tudja, hogy a példamondatokban magán-személygépkocsiról volt szó. Ám a rendszámok önmagukban ezt azért nem árulták el, mert nem kötöttem ki, hogy mai és magyar kocsiról beszélek! Nálunk a rendszám értékei hol utalnak implicit módon a jármű valamilyen sajátosságára, hol nem. Tehát informatikai szempontból a rendszám tökéletesen kaotikus – annyi más hétköznapi adatunkhoz hasonlóan. Persze az nagyon praktikus, ha a rendszámból felismerhető a mentő, a busz, a stb. kocs, mert különben valaki összetévesztené azokat mondjuk egy vízibiciklivel...

Az ilyen és hasonló „zseniális” és „konceptiózus” adatokkal sok bajom van. Egyrészt az, hogy a rossz absztrakció bonyolult – tehát hibáknak kitétt – programokhoz vezet. Ha a rendszám V-vel kezdődik, nem más betűvel folytatódik és magyar, akkor..., viszont ha nem magyar... Másrészt az adatok redundánsak lesznek. Például a „BU” rendszám utal arra, hogy a gépkocsi a BKV birtokában van, ami azért fontos, mert a tulajdonos adat ugyanezt teszi. Harmadrészt nem is a távoli jövőben majd jön az újabb „konceptió”, amikor az összes régebbi programot át kell írni a korábbi feltételezések megváltozása miatt. Az implicit ismeretekből fakadó csacsiságokat jól szemlélteti az én esetem. Alig másfél éve történt, hogy igazoltattam a rendőr. Rendszámtáblámra nézett – „BM...!” – és megkérdezte, hogy kolléga vagyok-e. Csak azután firtatta, hogy ittam-e...

Az adatbázis-építés egyik legnagyobb problémája, hogy a fejlesztők szolgálaián átveszik az adatbázisba a mai papírok – bizonylatok, nyilvántartások, kimutatók – adatait anélkül, hogy az azok által elrejtett valós információk szándékot, mélyebb ismereti lényegét elemeznék. Erre a jelenségre azért érdemes figyelni, mert papír adathordozóink informatikai szempontból iszonyatosan rosszak, tehát... Egy mulatságos és egy szomorú eset szemlélteti a gondokat.

Egy pályázati űrlapon 5-10%, 10-15% stb. „dobozkát” kellett kiiktatni, ahol a százalék valamilyen

saját részarányt mutatott. Akadt, aki több dobzott is kiíkszelt; más nem tudta, hogy hová tegyen ikszet a 10% esetében; a fejlesztő pedig felvett Részarány-, Részarány- stb. nevű hat darab (!) adattételt, holott itt nyilván egyetlen adat különböző értékeiről van szó.

Egy szöröncsétlen kollégámat kipenderítették egy több ezer kocsit üzemeltető intézménytől, mert az ikszedik nekifutás után is rossz típus szerinti kimutatást készített a kocsikról. Hát persze! Ő a könyvelési papírokból vette át a „típus” adatot, ami a főmérnöknek természetesen nem tetszett, mert a főkönyvelő és a főmérnök számára más informatikai lényeg volt a „típus”.

Az adatbázis építése nem ott kezdődik, hogy a papírok adatait átmásoljuk a számítógépre, hanem ott, hogy feltárjuk a fogalmakat és azoknak a valósággal való összefüggéseit. Például az olvasó nyugodtan a szememre vehetné, hogy a Lada nem típus, hanem márka, gyártmány („make”), modell stb. Közléseink semmit sem érnek, félrevezetőek, ha nem egyezünk meg a fogalmak alapvető jelentésében. Amint az előző példa mutatta, ez a megegyezés nem is olyan egyszerű. Azért sem az, mert maguk az ismeretek végső felhasználói is képtelenek tiszta és stabil ismereti elrendezésekben gondolkodni. Emiatt hétköznapi adataink egy sor rendszerezetlen, nem eléggé átgondolt, implicit és éppen emiatt menedzselhetetlen ismeretet hordoznak. Például a biztosítás adatai között szerepelt egy „terület” nevű valami. Ez hol egy települést, hol egy ügynököt, hol egy bejárési körzetet jelentett, olykor pedig egy sorszámmal töltötték fel csak azért, hogy ne legyen üres...!

Szakmai körökben a számítógépet „Gigo”-nak – garbage in, garbage out – becézik. Ennek irodalmi fordítása: „szemét be, szemét ki”, de mi nem ezt alkalmazzuk, bár az „sz” betű a mi szóhasználatunkból sem marad el... Az eleve rosszul megfogalmazott – mondjuk csak ki bátran: összetákolt – adatbázisból sok jóra nem számíthatunk. Ezért sóhajtott fel így az anonim szerző: „Adatunk már van rengeteg, de a tudásunk mit sem gyarapodik.”

Kapcsolat

Az adatbázis leginkább abban különbözik a másféle ismeretszervezési módoktól, hogy abban lehetőség van az egyedek közötti viszonyok meghatározására is. Vegyük csak alapul a következő két példamondatot:

A BMZ 873 rendszámú gépkocsi / tulajdonosa H. B., aki Budapesten lakik.

Az FGS 802 rendszámú gépkocsi / tulajdonosa H. B. H. B. Budapesten lakik.

Hétköznapi közléseinkben összetett mondatokat is használunk, illetve a gondolatainkat egymásba fűz-

zük, azaz egyik mondatunk a másik folytatásaként szolgál. Az első mondatban továbbra is egy gépkocsiról van szó úgy, hogy az állítmány utal annak sajátosságaira. Viszont a második mondat már jól mutatja, hogy valójában nem egy, hanem két jelenségről folyik a diskurzus: egy kocsiról és egy emberről mint tulajdonosról. Az adatbázisokhoz nem igazán értők ezt a különbséget nem veszik észre, és a 3. ábra szerint gondolkodnak.

3. ábra

Téves szemléletmód

Gépkocsi

Rendszám	Típus	Szín	Tulaj	Lakik	...
BMZ 873	Lada	Bordó	H. B.	Budapest	...
FGS 802	Ford	Bordó	H. B.	Budapest	...

Ez a felépítés azért téves szemléletű, mert a gépkocsi és a tulajdonos két különböző valós jelenség, és ezért azokat az adatbázisban két eltérő egyedtypusként illenek tükrözni. Nem a „szépségről” van itt szó, hanem nagyon is gyakorlatias megfontolásról, méghozzá a pénzről. Ha egy vállalatnak ezer kocsija van, akkor ezerszer fogják tárolni a nevét és a címét. Ha a tárolás többszörös, akkor az adatrögzítés, az adatbevitel és az adatkezelés is az. Mindez pedig sokszoros kiadást jelent az egyszeres helyett. Am nemcsak az ilyen közvetlen költségekre kell figyelni, hanem a közvetettékre is.

Nem tudom, hogy a MATÁV-nál rólam hány kimutatást vezetnek, de kettőt legalább. Érdekes módon a számlát mindig megkapom a lakcímemen, viszont a tudakozó még ma is azt a címet adja meg az érdeklődőknek a jó helyett, aminek a korrekcióját már hússzor kértem. Kérésemnek lett is foganatja: ma már a telefonszámom is rosszul szerepel a telefonkönyvben. Vajon milyen „adatbázisa” lehet a MATÁV-nak és mennyibe kerül az – nekem?

Visszatérve a fenti példára, ha például elköltözöm Pestről és több kocsim is van (ami persze nem igaz), akkor több helyen kellene átírni a lakcímemet, de arról természetesen el fognak feledkezni. Az adatbázis a többszörösség miatt inkonzisztens állapotba fog kerülni. Ezért senki se csodálkozzék azon, ha helyes és téves címzésű leveleket is kap. Az alapvető ok igen egyszerű: a szoftverzszenik a fenti módon tervezték meg az „adatbázist”. Kérdezzék csak meg a pénzügyi zeteinket, hogy hány függő pénzügyi tételük van az ügyfelek hibás adatai miatt!

Az adatbázis szíve, lelke, veleje a kapcsolat, azaz két egyedtypus logikai/tartalmi viszonya. Ennek a tényezőnek is két szintje van. A tulajdonos és a gépkocsi között kijelölhető a „tulajdonosa” nevű kapcsolat-típus, ami a kapcsolat-előfordulások időben és térben

adott halmazával rendelkeznek. Példánk esetében a „BMZ 873 tulajdonosa H. B.” és az „FGS 802 tulajdonosa H. B.” a két kapcsolat-előfordulás. A hétköznapi mondatokban a kapcsolatok úgy érthetők tetten, hogy az egyik mondat(rész) állítmányát a másik mondat(rész) alanya. Mindez pedig a 4. ábra módjára tükrözhető.

4. ábra

Helyes szemléletmód				
Gépkocsi				
Rendszám	Típus	Szín	Tulaj	...
BMZ 873	Lada	Bordó	H. B.	...
FGS 802	Ford	Bordó	H. B.	...

Tulajdonos		
Tulaj	Lakik	...
H. B.	Budapest	...
T. GY.	Győr	...

Vegyünk észre két dolgot. Az egyik az, hogy a 4. ábra két állománya nem tartalmaz már redundanciát, mert elég egyszer tárolni a tulajdonosok lakcímét. A két tábla között az egyik azonosítójaként szolgáló tulajdonság (Tulaj) teremt kapcsolatot két irányban úgy, hogy ki lehet keresni a kocsi tulajdonosát és meg lehet mondani, hogy a tulajdonos melyik kocsikat birtokolja. A másik az, hogy ez a megoldás több-letinformációk tárolását és kezelését teszi lehetővé. A T. GY. tulajdonos – mint ügyfél – adatai akkor is tárolhatók és kezelhetők, ha az adott időpontban még vagy már nincs kocsija. A 3. ábra megoldása ezt nem teszi lehetővé, mert a tulajdonos adatai csak a kocsi által létezhetnek. Ha nincs kocsi, nincs ügyféladat sem.

Mindezek után a fentebbi kitételeim egyikét így kell pontosítanom: Minden ismeret – és így szükség-szerűen minden adatbázis – az általunk elképzelt egyedek és tulajdonságok együttesen alapul úgy, hogy egyes tulajdonságok két egyed között teremtenek kapcsolatot. Ezek szerint – első megközelítésben – az adatbázis nem más, mint egyedeknek, azok tulajdonságainak, illetve kapcsolatainak a szervezett együttese. Szervezett – azaz tudatosan, komoly elemzések után kialakított, alaposan átgondolt rendszer. Az adatbázis-tervezés – amiről lesz még szó a sorozat további részeiben – egy igen komoly tudományág az informatikai diszciplínán belül, amihez sajnos még a szakemberek sem nagyon értenek. Ami azért roppant nagy baj, mert az adatbázis-tervezés az információkkal való gazdálkodás alfája és ómegája.

Mondanivalómat azzal kezdtem, hogy az információs rendszerek célszerű kialakításához mindenekelőtt az „ismeretek ismeretére” lenne szükség. A vezetőnek nincs szüksége a szoftver, az adatbázis-kezelő rendszer ismeretére. Viszont szerintem elengedhetetlen, hogy értsen magához az adatbázishoz. Legalábbis

olyan szintig, hogy meg tudja állapítani, vajon az általa alkalmazott informatikai fejlesztők értenek-e az adatbázishoz. Merthogy azt nem teszik.

Nemrégiben voltam tanúja egy szomorú esetnek. A vezetés adatbázist akart, és annak elkészítésére felbérelt három programozót. Többek között ügyfelekről, bankszámlákról, devizaáramlásokról stb. volt szó.

Megnéztem – csak úgy, szívességből – az adatbázistervet. A fejlesztők még azt sem tudták, hogy egy cégnek több bankszámlája lehet; a devizának nem fix az ára; a fordulónapról még hírből sem hallottak stb. Mire mindez kiderült, a cég már elköltött pár mil-

liót az „adatbázisra”, és megkezdték a programozást... Vagyis a végletesen és végzetesen rosszul meghatározott ismeretek bedrótozását a mindennapi eljárásokba.

Más. Legnagyobb légiforgalmi vállalatunk „informatikai” vezetője – ifjú zseléhajú, ahogyan azt a számítástechnikus öregarcik mondják – szemöldökét ráncolta, amikor valaki beajánlott neki adatbázis-tervezési célokra. Két hónap (!) múlva kaptam tőle egy udvariatlan levelet. Ha valamit akarok – sic! (én nem akartam semmit, engem szoktak megkérni dolgokra) –, akkor közöljem velem a programterveimet és adjam át a forráskódokat egy héten belül. Az ifjú titán hallatlanul érthetett az információs rendszerhez, ha egy adatbázis elvi tervét összetévesztette pár konkrét programsorral. Hát körülbelül itt állunk az információmenedzselés terén...

Szerzőnk a közgazdaság- és informatikai tudományok kandidátusa

HALÁSZ

BIZTONSÁGTECHNIKAI ÉS KERESKEDELMI KFT.

1143 Budapest, Gizella út 42-44.

Telefon: 163-3023, 183-2933. Fax: 183-2933

ÜZLET ÉS BEMUTATÓTEREM

1055 Budapest, Bajcsy-Zsilinszky út 78.

Telefon: 112-7480/114-es mellék