

Adatbázisok elmélete, tervezése, és egy gyakorlati alkalmazás a B2C elektronikus kereskedelemről

1. Bevezetés, alapfogalmak

Adatbázison köznapi értelemben valamely rendezett, valamilyen szisztéma szerint tárolt adatokat értünk, melyek nem feltétlenül számítógépen kerülnek tárolásra.

Manapság nem elégszünk meg egy adatbázissal, mely az adatokat rendszerezve tárolja, hanem az adatok kezeléséhez szükséges eszközöket is az adatbázis mellé képzeljük. Az így kialakult programrendszerrel adatbázis kezelő rendszernek (DBMS Database Management System) nevezzük.

Az adatbáziskezelők fejlődése során többfajta logikai modell alakult ki, melyek főként az adatok közötti kapcsolatok tárolásában térnek el egymástól. A három alapvető modell a *hierarchikus*, a *háló* és a *relációs* modell. Ezek közül manapság a DOS/Windows 3.x/Windows 95/NT illetve UNIX operációs rendszerekben kizárólag a relációs modellre épülő adatbáziskezelőket használják, ezért itt csak a relációs modellel foglalkozunk.

A reláció nem más, mint egy kétdimenziós tábla, a tábla soraiban tárolt adatokkal együtt, a relációs adatbázis pedig ezen relációk összessége. A relációk tulajdonságai:

- minden relációnak egyedi neve van
- egy sor és oszlop kereszteződésében egyetlen érték szerepel
- minden sor egyedi, nincs két egyforma sor
- minden oszlopnak egyedi neve van a reláción belül, de más relációk tartalmazhatnak azonos nevű oszlopokat
- az oszlopok sorrendje lényegtelen
- a sorok sorrendje lényegtelen.

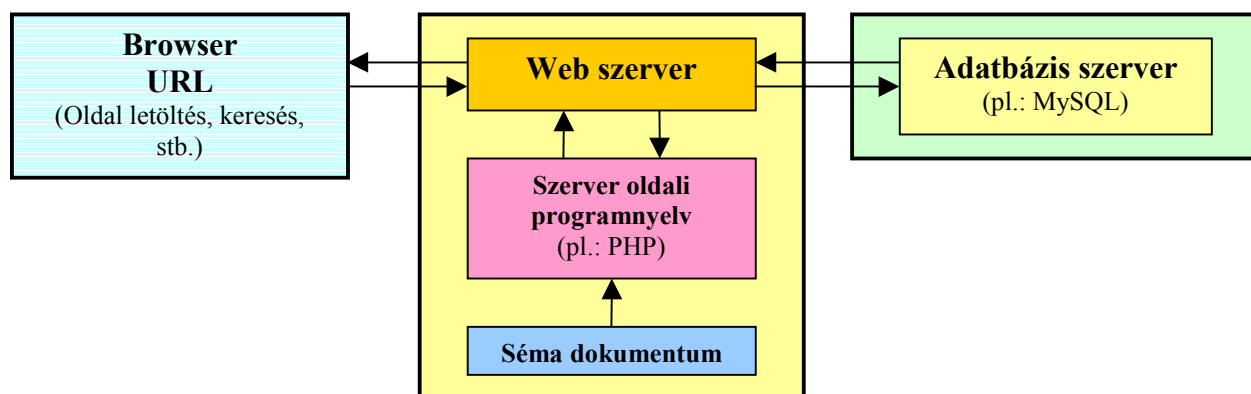
A relációk oszlopaiban azonos mennyiségre vonatkozó adatok jelennek meg. A reláció soraiban tároljuk a logikailag összetartozó adatokat. Egy reláció felépítését mutatja be az 1.1.-es ábra. A mezőkben oszloponként különböző típusú (numerikus, szöveges stb.) mennyiségek tárolhatók. A reláció helyett sokszor a tábla vagy táblázat, a sor helyett a rekord, az oszlop helyett pedig az attribútum elnevezés is használatos.

	Oszlop			
Sor	Érték			

1.1. ábra Reláció felépítése

A relációs adatmodellben a relációkból történő információ kinyerése is biztos elméleti alapokon nyugszik. A Codd által definiált adatlekérdező műveletcsoportot összefoglalóan relációs algebra-nak nevezik. A relációs algebra az alapja a ma már szabványként elfogadott és leginkább elterjedt adatlekérdező relációs parancsnyelvnek, az SQL-nek is. Az SQL nyelv egyik fő jellemzője, hogy hiányoznak belőle a procedurális elemek. Ennek következtében nem lehet pusztán SQL utasításokra építve komplett alkalmazásokat készíteni, hiszen az SQL nem tartalmaz elágazási, ciklusvezérlési, vagy éppen terminál felület működését leíró nyelvi

elemeket. Így az SQL nem tekinthető egy alkalmazásfejlesztő nyelvnek. Az SQL kizárólagos célja az adatbázissal történő adatforgalom biztosítása. Ebből az is következik, hogy a WEB-re történő fejlesztésekhez egy más jellegű procedurális, szerveroldali nyelvre lesz szükségünk. Napjainkban számos ilyen fejlesztő nyelv áll rendelkezésre. Megemlíthetjük a PHP-t, amely nyílt forráskódú, az ASP-t amely a Microsoft terméke, vagy a JAVA alapú nyelveket, amelyek a platformfüggetlenségük miatt használatosak, többek között ezek a legelterjedtebb webes alkalmazás-fejlesztő programnyelvek. Az utóbbi időkben nemcsak az Internetre fejlesztenek ezekkel a programokkal, hanem úgynevezett Intranetes, vállalati alkalmazásokat is. A kéréseket itt is egy webservert szolgálja ki és a kliens oldalról csak egy böngésző installálása szükséges. A rendszer felépítését az 1.2-es ábra mutatja be. Különböző nyilvántartó, projekt-támogató, információs, hibabejelentő, és még számos egyéb alkalmazást lehet a dolgozók munkájának segítésére, támogatására fejleszteni és bevezetni.



1.2. ábra Adatbázis-szerver és Web-szerver kapcsolata Internetes felhasználás esetén

2. Adatbázis tervezés

2.1. Adatbázis tervezés elmélete

Az adatbázis tervezés egy folyamat, mely több lépésből tevődik össze. Először az adatbázisban leképezendő *rendszer elemzésnek* vetjük alá és meghatározzuk a tárolandó adatok körét, és az adatbázissal szemben felmerülő igényeket. Ezután következik a *rendszertervezés*, ezalatt megvizsgáljuk a relációk közötti kapcsolatokat, ennek eredménye a rendszer specifikáció, vagy logikai modell. Végül *fizikai szinten* képezzük le a logikai adatbázis modellt a felhasználható szoftver és hardver függvényében, e folyamatok egymásra épülését mutatja be a 2.1. ábra.

A tényleges tervezés ismertetése előtt néhány újabb fogalmat kell bevezetni - funkcionális függőség, reláció kulcs, redundancia, - melyek segítségével a tervezés módszertana egyszerűbben magyarázható.

Funkcionális függőség:

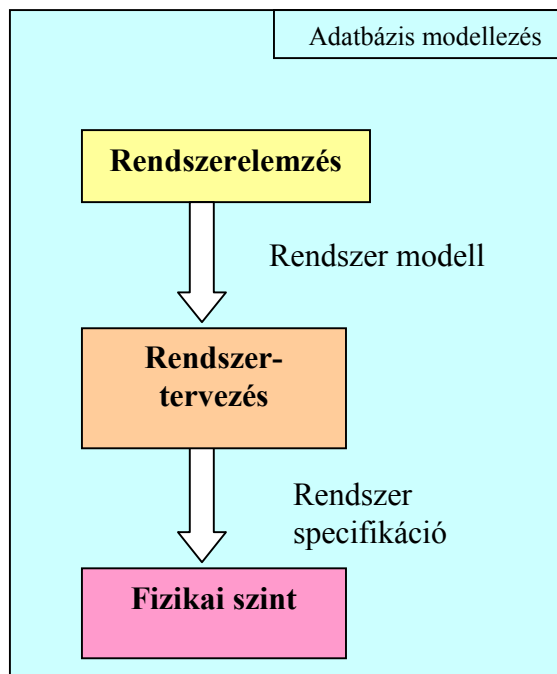
Adatok között akkor áll fenn funkcionális kapcsolat, ha egy vagy több adat konkrét értékéből más adatok egyértelműen következnek. Például a személyi szám és a név között funkcionális kapcsolat áll fenn, mivel minden embernek különböző személyi száma van.

Reláció kulcs:

A reláció kulcs a reláció egy sorát azonosítja egyértelműen. A reláció - definíció szerint- nem tartalmazhat két azonos sort, ezért minden relációban létezik kulcs. A reláció kulcsnak a következő feltételeket kell teljesítenie:

- az attribútumok egy olyan csoportja, melyek csak egy sort azonosítanak (egyértelműség)
- a kulcsban szereplő attribútumok egyetlen részhalmaza sem alkot kulcsot
- a kulcsban szereplő attribútumok értéke nem lehet definiálatlan (NULL).

A logikai adatbázis tervezés egyik fő célja a redundanciák megszüntetése.



2.1. ábra A tervezés lépései

Redundanciáról akkor beszélünk, ha valamely tényt vagy a többi adatból levezethető mennyiséget ismételten (többszörösen) tároljuk az adatbázisban. A redundancia, a szükségtelen tároló terület lefoglalása mellett, komplikált adatbázis frissítési és karbantartási műveletekhez vezet, melyek könnyen az adatbázis inkonzisztenciáját okozhatják. Az egyszerű tervezési metodika, az adatbázis tervezése jól definiált, egyértelmű elméleti alapokon nyugszik. A tervezés elméleti alapjai az úgynevezett normalizálási szabályokon alapulnak, amely a relációs modell strukturális lehetőségeit és a modellezendő fogalmak közötti függőségi kapcsolatokat veszi figyelembe. A normalizáció végigkövetésével egy hatékony, áttekinthető adatmodellt kaphatunk.

A relációs adatbázisok esetében a logikai tervezés során a relációk már elnyerhetik végleges alakjukat, melyeket egyszerűen leképezhetünk az adatbáziskezelőben.

Indexek fogalma és felépítése:

Az indexek logikailag egy rendezett listaként foghatóak fel, fizikailag a rendezett sorrendet táblába rendezett mutatók biztosítják. A relációkban tárolt információk visszakeresését az indexek nagymértékben meggyorsíthatják, így a tervezés során nagy hangsúlyt kell fektetni a helyes indexek kiválasztására, szem előtt tartva azt is, hogy az indexek számának növelésével az adatok beviteléhez illetve módosításához szükséges idő megnövekszik az indexek frissítése miatt. A relációkhoz kapcsolt indexek segítségével az index kulcs ismeretében közvetlenül megkaphatjuk a kulcsot tartalmazó sor fizikai helyét az adatbázisban.

2.2. Rendszeranalízis

Az adatbázisban leképezendő rendszerünk jelen esetben egy Internetes áruház teljes adat és információ igényének tárolása és az áruház üzemeltetése.

Az Internetes áruházakhoz szükséges adatokat többféleképpen csoportosíthatjuk. Ilyen csoportosítás lehet, ha az adatokat a felhasználók számára elérhetőekre (publikusra), és csak az adminisztrátorok számára látható adminisztrátori adatokra osztjuk. Természetesen bizonyos átfedések vannak a két csoport között. Az adminisztrátori adatok felosztása során különböző jogosultsági és hozzáférési szinteket határozhatunk meg.

Egy másik felosztás lehet, ha adatállományok szerint bontjuk az adatokat, áru-adatállomány, megrendelés-adatállomány, háttér-adatállományokra, stb. Mielőtt meghatározzuk a tárolandó adatokat, célszerű egy ilyen csoportosítást elvégeznünk, és ezt kifejtve továbbhaladni az elemzésben. Ez megkönnyítheti munkát, és így kisebb a valószínűsége, hogy valami elkerüli az elemzők figyelmét.

Egy Internetes áruház elemzése során meg kell határoznunk, a kínált áruk nyilvántartásához szükséges adatok körét, ezek nagyban hasonlítanak a hagyományos raktárakban tárolt információkra. Ilyen adatok például áru megnevezés, mennyiség, minőség, fényképek, termékleírás, stb. Ez azért is lehet így, mert az Internetes áruházak legtöbbször „hagyományos” darabárus raktárak szolgálják ki. Nagyon fontos továbbá az árucikkek típus, fajta szerinti besorolhatósága, hiszen a vásárlók ezen információk alapján szűkítik a keresendő árucikkek körét, és ezért ennek lehetőségét az adatbázis kialakításánál figyelembe kell venni.

Az áruk, készletek nyilvántartása mellett, meghatározó terület a felhasználók adatainak tárolása. Az áruházakban csak regisztrált felhasználók vásárolhatnak. Az Ő személyes, elérhetőségi és egyéb adataik mellett rögzíteni kell a vásárlásuk időpontját, mit vásároltak és mekkora összegben. Egyre gyakoribb, hogy a regisztrációs folyamatban rögzítésre kerül az iskolai végzettség, érdeklődési kör, fizetés nagysága, stb. is. Ezekből az információkból különböző statisztikákat készíthetünk, vásárlási szokásokra következtethetünk, meghatározhatjuk, melyik korosztály, milyen időközönként látogatja áruházunkat, hírlevelet, termékkatalógust küldhetünk, kiemelt ügyfeleinket is „díjazhatjuk”, stb.

A megrendeléseket a feldolgozás fázisainak megfelelően, különböző státuszokkal láthatjuk el, például „feldolgozás alatt”, „kiszállítás alatt”, „teljesített”, ez a megrendelések pontos követése miatt elengedhetetlen.

A háttér-és egyéb információk adatállományába nagyon széles körben kerülhetnek adatok, a szállítási információk, termék-ismertető, az áruház használati útmutatója, stb.

Az Internetes áruházak csak akkor lehetnek hatékonyak, és jelenthetnek alternatívát a hagyományos kereskedelmi formákkal szemben, ha a vásárlók egyszerűen, gyorsan megtalálják a keresett termékeket, és emellé egyszerű vásárlási folyamat társul. Ha mindazok az előnyök, amiket az Interneten történő vásárlás nyújthat a vásárlóknak, kiemelésre kerülnek, és ezekről megfelelően tájékoztatják is az ügyfeleket. Ehhez elengedhetetlen az adattáblák közötti kapcsolatok megfelelő tervezése. De nemcsak a felhasználók számára fontos, hogy összetett kereséseket hajthassanak végre, hanem az adminisztrátoroknak is fontos, hogy a háttéroldalakon könnyen tudják aktualizálni az áruházak tartalmát. Ugyancsak jelentős szerepe van az adattáblák közötti kapcsolatoknak a forgalmi adatok, statisztikák, elemzések készítésénél is, hiszen gyakran kell a megrendelések, áruk, felhasználók, státuszok adattábláit összekapcsolni egy-egy bonyolultabb lekérdezésnél. Ezeket a lekérdezéseket természetesen nem a felhasználóknak kell leprogramozni, hanem a háttérprogramokat kell erre felkészíteni.

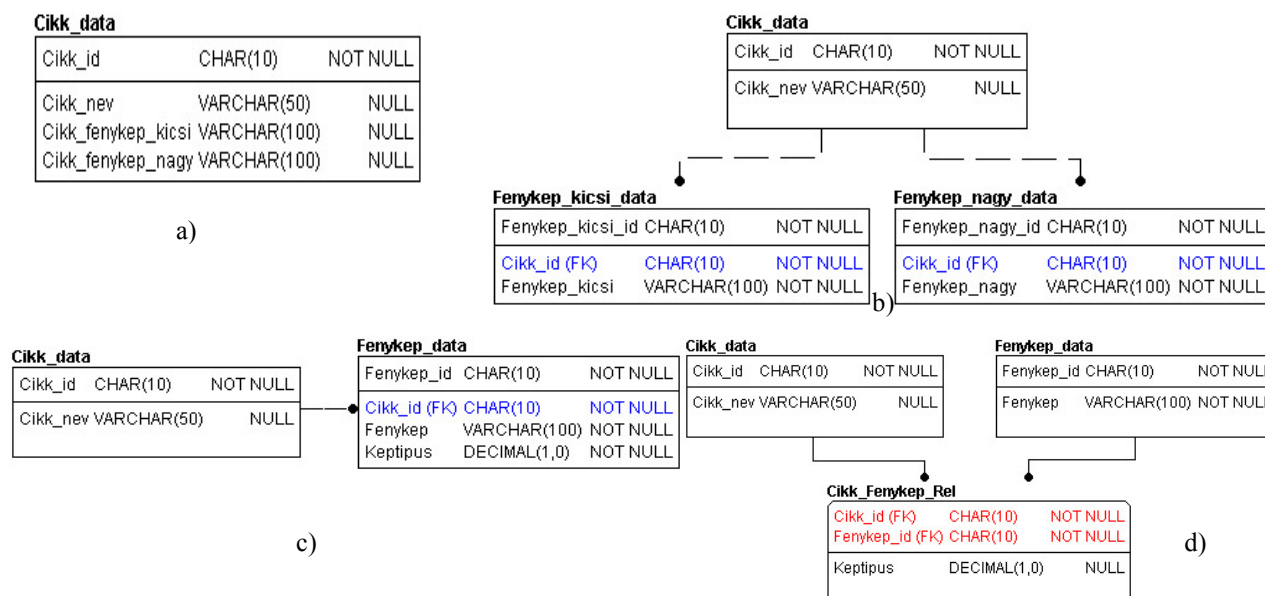
2.3. Rendszertervezés

A rendszertervezés során a rendszerelemzés alatt összegyűjtött információk alapján egy kész logikai modellt kell előállítani. A logikai modell közvetlenül felhasználható az adatbázis kialakításakor. A logikai szint megalkotásakor már figyelembe veszünk olyan szempontokat, amelyek már a konkrét használattal kapcsolatosak.

A tervezési folyamatok során mérlegelnünk kell, hogy a táblák mezőinek számát növeljük meg, vagy több táblában helyezük el az információkat, és így a táblák közötti kapcsolatrendszerre helyezük át a hangsúlyt. Különösen akkor jelentős ez a kérdés, és akkor tudunk keresési időt, helyigényt megtakarítani, ha olyan mezőket helyezünk el új táblákba, ahol a rekordok egyébként csak kis százalékban tartalmaznak információt, így csak azokhoz a rekordokhoz tartozik egy másik tábla kapcsolódó rekordja, ha annak tartalma is van.

Szintén a táblák kialakításának fontos szempontja és befolyásoló tényezője, hogy előre rögzítsünk tulajdonságokat az információ mennyiségét, vagy hagyjuk meg a lehetőséget a későbbi bővítésre. Példaként említhetjük termékekhez tartozó fényképek nyilvántartását. A fényképek elérési útvonalát tartalmazhatja az árucikkek törzsadat tábla is, de akkor csak fix számú fénykép tartozhat egy-egy termékhez (célszerű legalább két fényképet letárolni, egy kisképet a találati oldalra és egy nagyobb fényképet a terméklapra). Másik megoldásként

választhatjuk azt, hogy kapcsoló mezővel, egy külön táblában tároljuk le a fényképek útvonalát, és így tetszőleges számú fényképet tudunk eltárolni egy termékről, így akár egy későbbi időpontban is helyezhetünk el újabb fotókat a termékről. Erre az esetre mutat be több tábla-struktúra felépítést a 3.2-es ábra.



3.2. Termékekhez tartozó fényképek tárolási módozatai (a) egy termékhez fix számú fénykép tartozik (b) egy termékhez korlátlan számú kis és nagyméretű fénykép tartozik (c) egy termékhez több fénykép egy kapcsolódó táblában (d) egy termékhez több fénykép és egy fényképhez több termék tárolása

2.4. Fizikai szint tervezésének szempontjai

A fizikai tervezés során inkább arra koncentrálunk, hogy a logikai szerkezet mennyire felel meg a hatékony végrehajtás feltételeinek, illetve milyen indexeket rendelünk az egyes relációkhoz. A relációkon végrehajtott művelet együttest tranzakciónak nevezzük és általában a tranzakciók gyors végrehajtását kívánjuk elérni.

A fizikai tervezés során előfordulhat, hogy a relációkba szándékosan redundanciákat építünk a hatékonyabb tranzakció kezelés érdekében. Ez visszalépésnek tűnhet a logikai tervezés során követett következetes redundancia megszüntető manipulációkhoz képest. A lényeges különbség viszont az, hogy itt a redundancia ellenőrzött módon kerül be a relációba, nem csak véletlenül maradt ott a hiányos tervezés miatt. Gyakran előfordul például az, hogy a sűrűn együtt szükséges adatokat egy relációban tároljuk a lehető leggyorsabb visszakeresés, és a bonyolult kapcsolatok elkerülése érdekében.

A tervezéskor figyelembe kell vennünk a táblákban történő tranzakciók gyakoriságát, hiszen nem mindegy, hogy egy háttér információkat tartalmazó tábláról van szó, amit néhány adminisztrátor használ, vagy pedig az árucikkeket tartalmazó tábláról, amiben naponta akár több ezer keresést is indíthatnak az ügyfelek.

Az index állományok segítségével gyorsabban kereshetünk a tábláink között, azonban egy új rekord beszúrása több időt vesz igénybe. Célszerű tehát az indexelésnél ezt figyelembe venni, és azokra a mezőkre elkészíteni, amikben várhatólag nagyszámú keresés fog történni.

A végleges adatstruktúra kialakításkor fontos szempont ugyanazon adatok különböző módokon történő rögzítése, ami által ismét redundancia lép fel, de ami ennél is nagyobb probléma, hogy ez igen nagymértékben megnehezíti adatok feldolgozását. Ezen problémák elkerülése érdekében, ahol csak lehet, meg kell szüntetni a szöveges adatrögzítési feladatokat, ezt checkbox (többválasztós), radio (egyválasztós), vagy legördülő mező alkalmazásával lehet

elkerülni. Erre az adatbázis struktúra kialakításra a 3.2-es ábra b, c, és d módozatai mutatnak példát.

3. Internetes áruház adatállományai

Az elektronikus kereskedelem (e-commerce) egyik meghatározása az üzleti tranzakcióknak minden olyan formája, melyek során a felek inkább elektronikus, mint fizikai úton, vagy közvetlenül érintkeznek. Az Internetes kereskedelem egyik alapvető formája a Business-to-consumer (B2C), amely a kereskedő és vásárló közötti kereskedelem.

A fenti tervezési lépéseken végighaladva felépíthetjük a teljes Internetes áruház adatállományát. Minden egyes állomány tervezése során figyelembe kell venni, az egyes adattáblák tervezési alapelveit, de fontos az adatállományok közötti egyértelmű kapcsolhatóság, lehetővé téve a be-és kitárolási műveletek követését: az áruk készlet-, tárolóhely-, rendelés-, forgalmi, pénzügyi stb. adatainak folyamatos rendelkezésreállítását.

Az Internetes áruházak a következő kereskedelmi és logisztikai adatállományokból épülnek fel:

Kereskedelmi adatállomány:

- felhasználó adatállomány
- megrendelők háttér adatállománya
- megrendelések adatállomány
- egyéb kereskedelmi adatállományok.

Logisztikai adatállomány:

- tárolóhely adatállomány
- árukészlet adatállomány
- járművek és anyagmozgató gépek adatállomány
- egyéb logisztikai adatállományok.

Kiegészítő adatállomány:

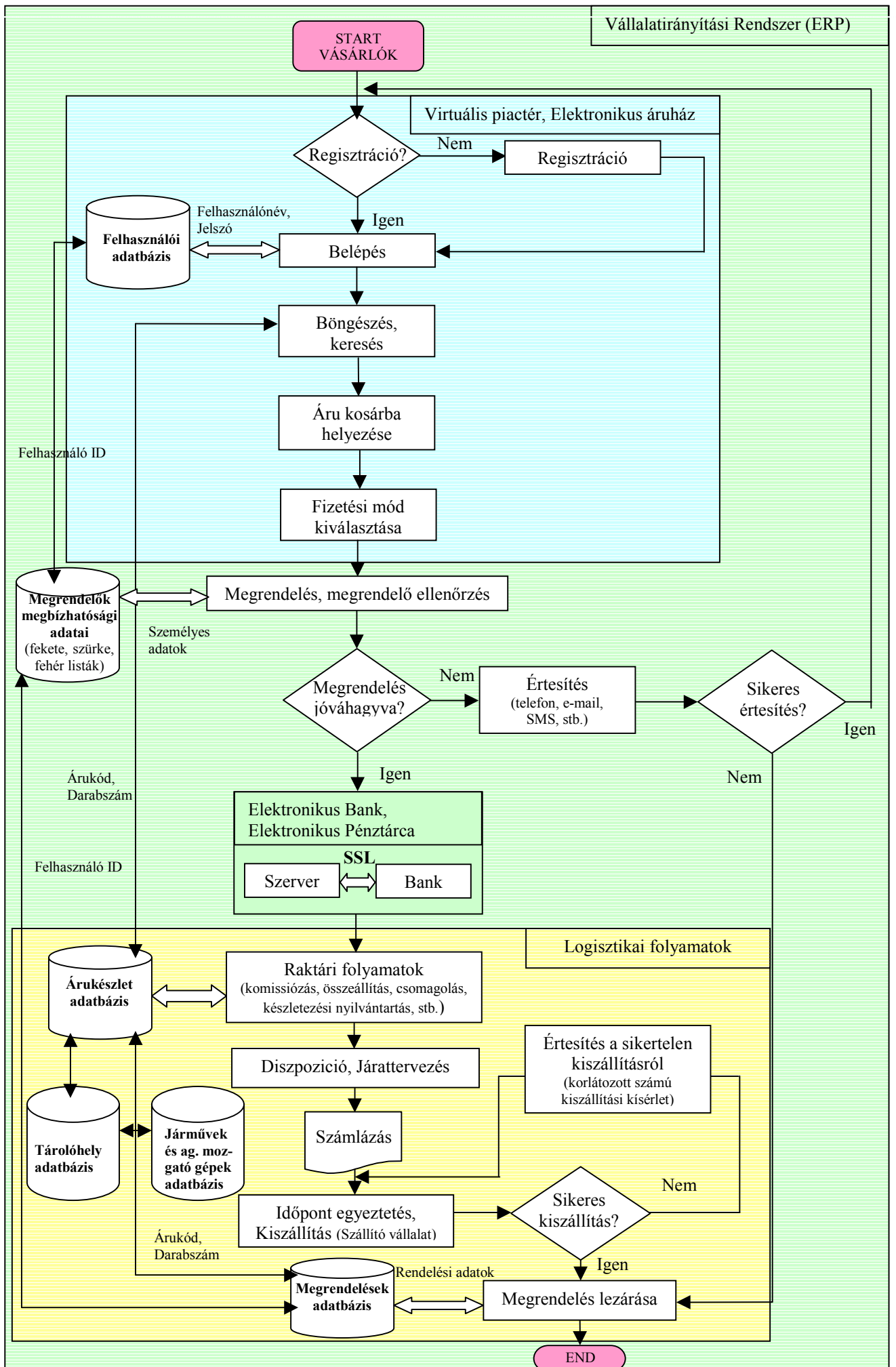
- háttér-és egyéb információk adatállomány.

A felhasználói adatállomány legfontosabb szerepe, hogy eldönthessük, a vásárló regisztrált felhasználó-e, és jogosult-e a vásárlásra. A felhasználói adatállomány kapcsolatban kell, hogy álljon a megrendelők „megbízhatósági” adatállományával, amelyben a korábban visszaélésekkel próbálkozók adatai találhatóak. Ezenkívül a megrendelések adatállománnyal is közös kulccsal kell, hogy rendelkezzen, hiszen alapfeltétel a megrendelések teljesítéséhez, hogy kinek is kell kiszállítani a megrendelt árut.

Az árukészlet adatállomány a tárolóhelyek, és a járművek adatállományaival kell, hogy kapcsolatban legyen, így egyértelműen meghatározhatjuk a raktári, és szállítási anyagmozgatási feladatok járműigényét, ezenkívül a megrendelések adatállomány segítségével lehetővé válik a készletek aktualizálása.

Az Internetes áruház vásárlási folyamatát és az adatállományok legfontosabb kapcsolatait mutatja be a 3.1-es ábra. Az üzemeltetés és a hatékony működés szerves részét képezi az adatok, és az adatbázis-struktúra archiválása. Már a tervezéskor célszerű arra is gondolni, hogy egy esetleges későbbi adatbázis szerkezeti változás, és a korábbi lementett állapotok hogyan „fésülhetőek” egyszerűen össze.

Az adatmentés vonatkozhat mindig egy bázisidőponttól keletkezett adatmennyiségre, vagy a legutóbbi archiválás óta keletkezett adatmennyiségre. Előbbi archiválási eljárás alkalmazása indokolt például a felhasználók adatállományainál, az utóbbi pedig a megrendelések adatállománya.



3.1. ábra B2C elektronikus kereskedelem vásárlási folyamatábrája és fontosabb adatállományainak kapcsolatai

4. Adatbázisok Internetes alkalmazásainak előnyei

Az adatbázisokkal kapcsolatos alapfogalmak, mint a „normalizálás”, „tábla”, „rekord”, „index”, stb. éppúgy érvényesek az Internetes alkalmazások esetén is, mint az informatika bármely más területen.

Internetes áruházak tervezési folyamatainak egy jelentős részét az adatbázis modellezés, a tervezés teszi ki. A táblák tartalmának, a mezők típusainak meghatározása, valamint a táblák közötti kapcsolat átgondolt, optimalizált, mindenre kiterjedő megtervezésére épülnek a generáló, aktualizáló, és lekérdező programok. Az adatbázisok tehát az internetes áruházak alapkövei.

A szerver oldali programnyelvek, a dinamikus weboldalak elterjedésével, gyakorlatilag a böngészőnkön megjelenő teljes HTML oldal, a megjelenő képek, szövegek, animációk, de még maga a HTML kód is, adatbázisokban tárolható. A képek vagy a flash animációk esetében az adatbázisban csak a fájlok elérési útvonalát szokás tárolni. Ennek a technikának és egy jelszóval védett adminisztrációs oldal segítségével, így a HTML-hez és a programozáshoz kevésbé értő felhasználó is áttervezheti az internetes áruház arculatát, és aktualizálhatja annak tartalmát.

Az Internetes áruházak ideális esetben az ellátási lánc szerves részét képezik, közvetlen kapcsolatnak kell fennállnia a raktárak külső információs rendszere és az elektronikus piac tér között. Az elektronikus kereskedelemben a vevői oldalon felmerülő igények azonnal jelentkeznek a rendszerben, bemenő információként szolgálva a raktárak belső folyamatainak szempontjából, és a készletgazdálkodás eredményeképpen keletkező információk külső információként szolgálnak a beszállítóknak.

Ahhoz, hogy az ellátási lánc információáramlása, információfeldolgozása zavartalanul működhessen, és ne keletkezzenek szűk keresztmetszetek, sem az információ hiány, sem információ kibocsátás, sem információ fogadás közben, összehangolt információs rendszerre és ezen belül jól megtervezett adatállományokra van szükség.

Irodalomjegyzék:

Simon Z.: Adatbázis kezelés és szervezés. Műegyetemi kiadó 1995.
Prezenszki J.: Logisztika I. Budapest 1997.

Internetes források:

<http://www.aut.bme.hu>

<http://bme-geod.agt.bme.hu>