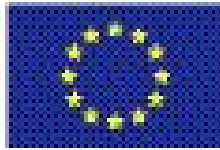


Magyarország célba ér

A projekt az Európai Unió társfinanszírozásával,
az Európa terv keretében valósul meg.



INFORMATIKA



HEFOP 3.3.1-P.-2004-06-0071/1.0

**Ez a kiadvány a
„Gyakorlatorientált képzési rendszerek kialakítása
és minőségi fejlesztése az agrár-felsőoktatásban”
című program keretében készült**

INFORMATIKA

© DE AMTC AVK 2007

Szerkesztő:

Dr. Harnos Zsolt
Budapesti Corvinus Egyetem

Dr. Herdon Miklós
Debreceni Egyetem

Szerző:

Dr. Herdon Miklós
Debreceni Egyetem

Dr. Fazekasné dr. Kis Mária
Debreceni Egyetem

Magó Zsolt
Debreceni Egyetem

Németh Zoltán
Budapesti Corvinus Egyetem

Lektor:

Dr. Nagy Elemérné
Szegedi Tudományegyetem

Dr. Szenteleki Károly
Budapesti Corvinus Egyetem

© DE AMTC AVK 2007

ISBN 978-963-9732-52-0

**E tankönyv teljes mértékben megegyezik a Debreceni Egyetem honlapján,
a <http://odin.agr.unideb.hu/hefop/> elérési úton megtalálható, azonos című tankönyvvel.**

Első kiadás

A kiadvány szerzői jogvédelem alatt áll. A kiadványt, illetve annak részeit másolni, reprodukálni, adatrögzítő rendszerben tárolni bármilyen formában és bármilyen eszközzel – elektronikus úton vagy más módon – a kiadó és a szerzők előzetes írásbeli engedélye nélkül tilos.

Kiadó:

Debreceni Egyetem Agrár- és Műszaki Tudományok Centruma
Agrárgazdasági és Vidékfejlesztési Kar
Debrecen, 2007.

Tartalomjegyzék

1. AZ INFORMATIKA ALAPJAI	9
1.1 Adat, információ.....	9
1.2 A Neumann-elvek	11
1.3 Mértékrendszerek.....	12
1.4 Kódrendszerek.....	13
2. SZÁMÍTÓGÉP-ARCHITEKTÚRÁK	15
2.1 Az architektúra fejlődése a XX. században.....	15
2.2 A központi egység.....	16
2.2.1 A processzor.....	16
2.2.2 Memória	19
2.2.3 A sínrendszer és az alaplapp	22
2.3 Perifériák	24
2.3.1 Háttértárolók.....	24
2.3.2 Adatbeviteli eszközök	28
2.3.3 Adatkiviteli (megjelenítő) eszközök	28
2.4 Platformok – modern CPU-architektúrák	30
2.5 Összefoglalás.....	31
3. OPERÁCIÓS RENDSZEREK.....	33
3.1 Alapfogalmak.....	33
3.2 Az operációs rendszerek csoportosítása	34
3.3 Az operációs rendszerek alapeladatai	37
3.3.1 Karakteres felület kezelése.....	37
3.3.2 Ablakozó (esemény-vezérelt) rendszerek sajátosságai	38
3.3.3 Állományszervezés.....	40
3.4 Operációs rendszerek a gyakorlatban.....	44
3.4.1. Microsoft Windows.....	44
3.4.2 Linux	57
3.4.3 Hálózati operációs rendszerek.....	63
4. IRODAAUTOMATIZÁLÁS, FELHASZNÁLÓI ALKALMAZÁSOK.....	69
4.1. Szövegfeldolgozás.....	69
4.2. Táblázatkezelő rendszerek	76
4.3. Adatbáziskezelés	83
4.4. Számítógépes grafika	84
4.4.1 Grafikus fájlformátumok.....	86
4.4.2 A színes kép	87
4.4.3 Különleges számítógépes grafikai eljárások	89
4.5. Prezentációkészítés.....	91
4.6. Irodautomatizálás	98
5. SZÁMÍTÓGÉPES PROGRAMOK, PROGRAMOZÁS	104
5.1 A programozási nyelvek.....	104
5.1.1 Gépi nyelv és assembly	105
5.1.2 Magas szintű programnyelvek és fordítók	106
5.1.3 Neumann-elvű és nem Neumann-elvű nyelvek	107
5.2 Alapvető programozási struktúrák	109
5.2.1 Szekvencia.....	109

5.2.2 Döntés.....	110
5.2.3 Ciklus	111
5.3 Az objektum-orientált programozás alapelvei	112
6. ADATBÁZIS KEZELÉS	114
6.1. Az adatbázis	114
6.2. Adatmodellek	114
6.2.1 Hierarchikus adatmodell	115
6.2.2 Hálós adatmodell.....	116
6.2.3 Kétszintű hálós adatmodell	117
6.2.4 CODASYL-hálós adatmodell	117
6.2.5 Relációs adatmodell	118
6.2.6 Objektumorientált adatmodell.....	118
6.3. Relációs adatmodell	120
6.3.1 A relációs adatbázis objektumai.....	121
6.3.2 Relációs adatbázis-kezelő rendszer	129
6.4. Adatbázis-kezelő rendszerek.....	130
6.5 Adatbázis-kezelő rendszerek felépítése	131
6.6. Normalizálás. Normálformák.....	134
6.7. Relációs műveletek	141
7. RELÁCIÓS ADATBÁZIS KEZELŐ RENDSZEREK A GYAKORLATBAN	145
7.1. Adatműveletek, adattábla kezelése	145
7.2. Adatbázis tervezése (relációs adatbázis tervezése). Adatbázis létrehozása	147
7.3. Lekérdezések. Szűrések a lekérdezésben	150
7.4. Űrlapok készítése	155
7.5. Jogosultság, biztonság, adatvédelem.....	161
7.6. Adatbázisok strukturált lekérdező nyelvének, az SQL-nek, ismertetése	164
7.6.1. Adatbáziskezelő SQL utasítások.....	166
7.6.2. Táblakezelő SQL utasítások.....	166
7.6.3. Összesítő függvények.....	170
7.6.4. Hozzáférési jogok, jogosultságok kezelése az SQL-ben.....	173
8. SZÁMÍTÓGÉP-HÁLÓZATOK.....	175
8.1 Céljai, elemei.....	175
8.1.1 Hálózati struktúrák	175
8.1.2 Hálózati hardver	176
8.1.3 Hálózati szoftver	178
8.2 Hivatkozási modellek.....	180
8.2.1 Az OSI hivatkozási modell.....	181
8.2.2 A TCP/IP hivatkozási modell.....	183
8.3 A hálózat fizikai megvalósítása.....	185
8.3.1 A fizikai réteg.....	185
8.3.2 Hálózat felépítés (topológiájuk)	189
8.3.3 A hálózatok részelemei	190
8.4 Hálózatok összekapcsolása	191
8.5 Kliens-Szerver modellek	193
9. A TCP/IP PROTOKOLL ÉS AZ INTERNET.....	195
9.1. A TCP/IP protokoll	195
9.2. Címzési rendszer	196
9.3. A Domén Név Rendszer (DNS – Domain Name System).....	200
9.4. Internet szolgáltatások.....	201
9.4.1. Elektronikus levelezés.....	202

9.4.2.	Állományok átvitele - FTP - File transfer protokoll	202
9.4.3.	TELNET	205
9.4.4.	A World Wide Web.....	206
9.4.5.	A WWW alkalmazások fejlesztésének eszközei.....	209
9.4.6.	Valós idejű kapcsolattartás (IP telefon)	212
9.5.	Kapcsolódás az Internetre	213
9.5.1.	Telefonvonalon való kapcsolódás	213
9.5.2.	ADSL	216
9.5.3.	Wi-Fi	216
9.5.4.	Wi-MAX	217
9.5.5.	Mobil Internet.....	217
10.	INFORMÁCIÓS RENDSZEREK	220
10.1.	Rendszer modell.....	220
10.1.1.	A rendszer fogalma	220
10.1.1.	Rendszerszemléletű megközelítés, rendszertípusok, rendszervizsgálat.....	224
10.2.	Információs rendszer fogalma	229
10.3.	Vezetői tevékenységek és adatszükségletek	234
10.4.	Vállalati információs rendszerek.....	235
10.5.	Ágazati információs rendszerek	241
10.5.1.	Az agrárstatisztika	242
10.5.2.	A Piaci Információs Rendszer.....	243
10.5.3.	A Tesztüzemi Rendszer.....	243
10.5.4.	Az Integrált Igazgatási és Ellenőrző Rendszer.....	244
10.5.5.	A Mezőgazdasági Számlák Rendszere.....	244
10.6.	Elektronikus kereskedelem, elektronikus üzletvitel.....	245
11.	TÉRINFORMATIKAI ALAPOK, ALKALMAZÁSOK	252
11.1.	A térinformatikai rendszerek, digitális térképek	252
11.1.1.	Kapcsolat a digitális helyzeti és alfanumerikus adatok között.....	253
11.1.2.	Kataszteri rendszerek	253
11.1.3.	Digitális magasságmodellek.....	253
11.1.4.	Földrajzi Információs Rendszerek	254
11.1.5.	Valódi térbeli (3D-s) információs rendszerek	255
11.1.6.	Térinformatikai rendszerek típusai.....	255
11.1.7.	A helymeghatározás, geometriai adatok	257
11.2.	Raszteres és vektoros adatok, adatkezelés, alapl műveletek	258
11.2.1.	Raszteres adatmodell átalakítása vektoros adatmodellé (raszter-vektor konverzió).....	258
11.2.2.	A vektor-raszter konverzió.....	259
11.2.3.	Síkbeli transzformációk.....	259
11.2.4.	Távolságfogalmak	260
11.2.5.	Hossz-kerület, terület és súlypont számítások.....	261
11.3.	Grafikus adatmodellek	261
11.3.1.	Vektoros modellek	263
11.3.2.	Raszteres-teszellációs modellek.....	265
11.4.	A térinformatika fontosabb eszközei, alkalmazása	266
11.4.1.	A digitalizálás.....	267
11.4.2.	Szkennelés	269
11.4.3.	Fej feletti digitalizálás	270
11.4.4.	Fotogrammetriai módszerek.....	271
11.4.5.	Térinformatikai rendszerek, az adatok kezelése	271

11.5.	Precíziós gazdálkodás	273
11.5.1.	Adatgyűjtés.....	273
11.5.2.	Gazdálkodási adatok integrációja.....	274
11.5.3.	Adatok interpolálása.....	275
11.5.4.	Adatelemezés	276
11.5.5.	Eredmények megjelenítése.....	276
11.6.	Országos mezőgazdasági térinformatikai rendszerek	276
11.6.1.	Mezőgazdasági Parcella Azonosító Rendszer (MePAR)	277
11.6.2.	A TAKARNET rendszer.....	279
11.6.3.	Országos Távérzékeléses Szántóföldi Növénymonitoring és Termésbecslés	280
12.	DÖNTÉSTÁMOGATÓ RENDSZEREK	282
12.1.	A döntéstámogató rendszerek alapjai.....	282
12.1.1	A döntéstámogató rendszerek használatának előnyei	284
12.1.2.	A döntéshozatal folyamata	285
12.1.3.	A döntési folyamat támogatása	287
12.1.4.	A döntéshozatal módjai	289
12.2.	Döntéstámogató rendszerek részei	292
12.2.1.	Adatkezelő alrendszer	292
12.2.2.	Modellkezelő alrendszer	294
12.2.3.	Kommunikációs alrendszer	296
12.3.	Döntéstámogató szoftverrendszerek és alkalmazások	297
12.3.1.	OPTRANS OBJECT	297
12.3.2.	Visual IFPS/Plus	297
12.4.	Üzleti intelligencia, mint vezetői támogatás	299
12.4.1.	On-line elemző feldolgozás.....	300
12.4.2.	Több dimenziós adatbázis	301
12.4.3.	Programcsomagok.....	304
13.	E-KORMÁNYZAT, E-SZAKIGAZGATÁS.....	307
13.1.	E-kormányzat fogalma	307
13.2.	Az önkormányzatokkal szembeni elvárások	310
13.3.	Elektronikus közszolgáltatások, ügyintézés	310
13.3.1.	E-ügyintézés: az önkormányzati ügyintézési szolgáltatások elektronizálása	311
13.3.2.	Az e-önkormányzat jövőkép funkcionalitása, alrendszerei, moduljai	312
13.3.3.	Az e-közigazgatás programja	312
13.4.	E-aláírás.....	315
13.4.1.	Az elektronikus aláírás funkciója	316
13.4.2.	Az elektronikus aláírás módszerei.....	317
13.4.3.	Elektronikus aláírással kapcsolatos szolgáltatások	318
13.4.4.	Aláírás-létrehozó adat elhelyezése	318
13.4.5.	Időbélyegzés.....	318
13.4.6.	Hitelesítési szolgáltatás – tanúsítvány	319
13.5.	Közigazgatási ügyfél-tájékoztató rendszer.....	320
13.6.	Agrár szakigazgatási intézmények	321
13.7.	e-Agrárium, e-szakigazgatás	323
14.	ÜZEMELTETÉSI KÉRDÉSEK	327
14.1	Segédprogramok, kommunikációs és egyéb alkalmazások	327
14.1.1	Tömörítés	331
14.1.2	Vírusok	332
14.2	Adatvédelem, adatbiztonság.....	334
14.2.1	Az adatvédelmi törvény	336

14.2.2 Informatikai vonatkozások.....	337
14.2.3 Adatvédelmi módszerek a gyakorlatban	338

1. AZ INFORMATIKA ALAPJAI

Mint minden komoly tudományterületnek, természetesen az informatikának (vagy ha úgy tetszik: számítástechnikának – bár ezt mindjárt belátjuk, hogy hibás megfogalmazás –, illetve ahogy manapság egyre elterjedtebben használják: információs- és kommunikációs technikáknak (ICT)) is megvan a maga fogalmi rendszere. Ahhoz, hogy a jegyzetben ismertetésre kerülő témakörökben foglaltakat pontosan megértsük, elengedhetetlen ezen fogalomrendszer tisztázása. És itt van az első bökkenő...

1.1 Adat, információ

Az, hogy az informatika az információval kapcsolatos, aligha igényel bővebb magyarázatot. Az azonban, hogy mi is az információ, már közel sem ennyire egyértelmű. Az informatika szakmai közösségében két fogalom: az adat és az információ szerepel, mint alapfogalmak. A gond csak az, hogy ezen két fogalom viszonya szerzőnként eltérő. Jelen esetben az állásfoglalást az olvasóra bizzuk, a teljesség kedvéért mindkét alapvető megközelítést bemutatjuk.

- Az első értelmezés szerint a környezetünkből érkező, a *befogadó személyre ható impulzusok az adatok*: mindaz, amit látunk, hallunk, érzünk (akár szaglás útján, akár tapintással, akár emocionális úton). Az adat az által válik **információvá**, hogy a befogadó az észlelésen túl *jelentéssel ruházza fel*. Ez azt jelenti, hogy az a tény, hogy süt a nap (ha egyébként nem érdekel bennünket, csak éppen tudomásul vesszük, hogy nem kell villanyt gyűjtanunk): adat, ami akkor válik információvá, ha ez a tény számunkra valamilyen jelentéssel bír (pl. azzal, hogy akkor most biztosan nincs éjféli). Egy másik (a két fogalom viszonyával foglalkozó szakirodalmi forrásokban gyakran említett) példával élve, ha meglátunk egy kínai nyelvű feliratot, érzékeljük az írásjeleit (adat), de – tisztelet a kivételnek – nem tudjuk, mit jelent a felirat, tehát nincs információ-tartalma – számunkra.

Ebben a megközelítésben tehát az adat részhalmaza az információ.

- A második megközelítés szerint a környezet ingerei közül a számunkra nem fontos ingerekről úgysem veszünk tudomást, következésképpen ezeket felesleges is kategorizálni. Maradnak a környezetünk azon impulzusai, amelyeket figyelembe veszünk – azaz számunkra valamilyen oknál fogva fontosak, jelentéssel bírnak: ezek az információk. Ebben a megközelítésben az információnak azt a jellemzőjét szokás hangsúlyozni, hogy megszerzésével az érzékelő személy „tudása” bővül – az **információ** tehát *érték(es észlelés)*. Az **adat** pedig nem más, mint az *információ tárolt* (rögzített vagy rögzítésre alkalmas) *formája*. Tehát az előző gondolatmenet kínai feliratánál maradva, ha nem tudunk kínaiul, akkor a fent említett táblán látható ákombákomok még csak nem is információk. Ha viszont el tudjuk olvasni a tábla feliratát, akkor ugyanazok az ákombákomok máris információvá válnak – sőt, adattá is (hiszen rögzített módon állnak rendelkezésre).

Vegyük észre, hogy e szerint a megközelítés szerint a rész-egész viszony pont a fordítottja az előzőnek: itt az információ a nagyobb halmaz, az adat pedig annak részhalmaza. Egyszerű, nem?

A két megközelítésben – látszólagos ellentmondásosságuk dacára is – van egy közös vonás: az információt mindkét megközelítés olyasvalaminek tekinti, ami a befogadó számára jelentéssel bír, fontos, értékes. (A továbbiakban az egyszerűség kedvéért nem teszünk különbséget az adat és az információ között – ahol mégis lényeges, ott hangsúlyozzuk, hogy melyik értelmezésben használjuk egyik vagy másik fogalmat.)

Természetesen az információ nem „csak úgy van” (persze, vannak ilyen típusú információk is, de a számítástechnika szempontjából ezek kevésbé jelentősek): azt valamilyen módon elő kell állítani (pl. méréssel, számítással, kutatással), értelmezni kell tudni (ld. az előző példát!), esetleg szükség lehet a tárolására (holnap ne kelljen már újra kiszámolni), stb. Azoknak az eszközöknek és módszereknek az összességét, amelyek az információ előállításával, tárolásával, feldolgozásával és továbbításával foglalkoznak, **információ-technológiának** nevezzük. Amennyiben ezt leszűkítjük egy konkrét feldolgozási folyamatra (azaz meghatározható a szóba jöhető információk köre, a feldolgozásukra szolgáló módszerek összessége, a tárolásuk módja, stb.), akkor beszélhetünk **információ (-technológiai) rendszerről** (IT rendszer).

Végül pedig azt a tudományágat, amely a fentieknek megfelelő IT rendszerek fejlesztési, üzemeltetési, elemzési kérdéseivel foglalkozik, **informatikának** hívjuk. (Fontos: a fentiekből természetesen következik, hogy az *informatika nem azonos a számítástechnikával*: ez utóbbi az informatika egyik részterülete, amely egy konkrét eszközrendszerre vonatkozóan végzi a fentebb megfogalmazott feladatokat!)

Előfordul, hogy az információ a keletkezési helyén kerül feldolgozásra, de (manapság már) nem ez az általános: az információt (általában) el kell juttatni egyik rendszerből a másikba, az ezzel kapcsolatos eljárásokat nevezzük összefoglaló néven **kommunikációnak**. A kommunikációs folyamatban legalább két fél vesz részt: az információt közlő (adó) és az információt fogadó (vevő). A kommunikáció csak akkor lehet sikeres, ha az adó által közölt információ ugyanazzal a jelentéstartalommal jelenik meg a vevőnél – ez pedig csak akkor biztosítható, ha a kommunikáció során mindkét fél betart bizonyos (előre meghatározott) szabályokat. **Protokollnak** nevezzük a kommunikációs folyamatra vonatkozó előírások gyűjteményét. Amennyiben egy kommunikációs folyamatban a részt vevő ICT eszközök összetartozását szeretnénk hangsúlyozni (vagyis azt a tényt, hogy az adott eszközök nem véletlenszerűen továbbítanak és fogadnak információt egymás között, hanem ezt valamilyen céllal teszik), akkor beszélünk **kommunikációs hálózatokról**.

Nos, vannak tehát információink, amelyek kommunikációs protokollok biztosításával jutnak el egyik informatikai eszköztől a másikig. Ezek az eszközök aztán valamit csinálnak az információval. Azt, hogy pontosan mit, azt nyilvánvalóan az információt értelmező személy határozza meg, jelen esetben a hangsúly az *automatizálhatóságon* van. **Algoritmusnak** nevezzük valamely feladat megoldását eredményező, véges számú, elemi (további műveleti részre már nem bontható) tevékenységek szabályokkal definiált sorozatát. Ha egy algoritmusra a következő feltételek teljesülnek, akkor az az algoritmus gépesíthető:

- egyértelműen meghatározott kezdőtevékenységgel rendelkezik (azaz pontosan egy „eleje” van)
- determinisztikus és rendezett (minden egyes tevékenység elvégzése után egyértelműen meghatározható a következő – elvégzendő – tevékenység)
- van legalább egy olyan tevékenysége, amelyet újabb tevékenység már nem követ (azaz van vége – *figyelem: amíg a kezdetre kikötés az egyértelműség, végállapotból több is lehet!*)

Azokat az információ-technológiai eszközöket, amelyek képesek az információ és a velük műveletet végrehajtó algoritmusok tárolására és automatikus végrehajtására, **számítógépeknek** nevezzük. A számítógép által értelmezhető algoritmusok neve **program**, a számítógépen tárolt és feldolgozott információ neve (egyértelműen!) **adat** – ez a két komponens **együttesen** alkotja a számítógép **szoftver** rétegét. Ezen kívül a számítógép részét képezi az adott eszközt ténylegesen alkotó (egyes megfogalmazások szerint: „kézzel

fogható”) „fizikai” elemek (elektronikai eszközök, vezetékek, csavarok, tokok, stb.) összessége, amelyet **hardvernek** nevezünk.

Összefoglalásként tehát azt mondhatjuk, hogy **számítógép** =

- **hardver**: alaplap, memória, processzor, merevlemez meghajtó, kábel, billentyűzet, stb.
- **szoftver**:
 - o adat: az adatokat kategorizálhatjuk
 - jellegük szerint: azaz milyen típusú információt tárolnak: numerikus (szám), szöveges, logikai, esetleg dátum értékű, stb.
 - belső szerkezetük szerint: az adatot alkotó információ összefüggése alapján (mező, rekord, állomány, stb.)
 - o program: a programokat általában feladatuk szerint csoportosítjuk:
 - operációs rendszerek, rendszerprogramok
 - rendszerközeli programok (segédprogramok)
 - felhasználói programok (alkalmazások)
 - fejlesztő eszközök (programozási nyelvek)
- (egyes források szerint a számítógép mint IT rendszer részét képezi a számítógépet használó személy (mint emberi erőforrás) is, ebben az értelmezésben az elnevezése „manware”)

1.2 A Neumann-elvek

A számítógépek fejlődése természetesen nem egyik pillanatról a másikra valósult meg (ha egy kicsit nyitottan szemléljük a fenti definíciókat, könnyű belátni, hogy már az ősember is használt informatikát...), de ennek a hosszú folyamatnak az egyik mérföldköve volt Neumann János¹ néhány tézise, amelyben megfogalmazta, hogy véleménye szerint mi kell ahhoz, hogy egy elektrotechnikai eszköz önálló működésre legyen képes. Ezeket az alapelveket nevezzük **Neumann-elveknek**, és azokat az eszközöket, amelyek ezeket megvalósítják, **számítógépeknek**.

A Neumann-elvek (az (elektronikus) számítógéppel szemben támasztott követelmény-rendszer):

- 1) Teljesen elektronikus működés (ennek az elvek a megértéséhez azt kell szem előtt tartanunk, hogy a korábbi számoló eszközökben valamilyen mechanikus szerkezet: fogaskerék, fogaslánc, csúszóléc, stb. vett részt a műveletekben), amelynek elemei logikai áramkörök (a logikai alaplóműveletek: ÉS, VAGY, NEM elvégzésére képes elektronikus eszközök)
- 2) Kettes számrendszer használata a műveletvégzésben. (Ez lényegében az elektronikus működés „szükségszerű” velejárója: egy elektronikus rendszerben a kétfázisú állapotok fizikai létrehozásából adódik az információ egyszerű megkülönböztethetősége, legyen szó az elektromos áram által okozott mágnesezettség létrehozásáról és megszüntetéséről, vagy a kettős fényvisszaverő tulajdonság könnyű és gyors cseréjéről, és más fizikai megvalósításokról.)
- 3) Tárolt program elve: az adat és program a rendszerben azonos módon (és helyen) tárolódik: ez a memória, amely azonos méretű és sorszámok alapján

¹ Neumann János kétségkívül a számítástechnika egyik legjelentősebb magyar képviselője volt. Életrajza számos helyen megtalálható, pl. a Neumann János Számítógép-tudományi Társaság honlapján is: <http://www.njszt.hu/neumann/neumann.head.page?nodeid=147>

megkülönböztethető részekből (rekesz) áll. Az adat és a program rendszerbe történő bevitele után a vezérlőegység (ld. 5. pont) külső (emberi) beavatkozás nélkül képes a memória tartalmát elérni, a benne tárolt adatokkal a kijelölt műveleteket elvégezni: működése automatikus.

- 4) Soros utasítás-végrehajtás: a számítógép egy időben csak egyetlen tevékenységet hajt végre – ez a Neumann-elvű gépek egyik legnagyobb hátránya, ugyanis párhuzamos működés mellett – értelemszerűen – lényegesen nagyobb teljesítmény lenne elérhető. *(Megjegyzés: ne tévesszen meg senkit az, hogy az általa ismert számítógépek egyszerre több dolog elvégzésére is képesek: ez csak látszólagos („virtuális”) párhuzamosság, amiről az operációs rendszerek működésekor részletesebben megismerkedünk...)*
- 5) A számítógép rendelkezzen a következő funkcionális egységekkel:
- a) **vezérlő egység** (CPU): rendszer-vezérlési feladatok, utasítások értelmezése és végrehajtása;
 - b) **műveletvégző egység** (ALU): elemi matematikai és logikai műveletek önálló elvégzése;
 - c) **tárolóegység** (memória): ld. 3. elv
 - d) **be- és kiviteli** (azaz kommunikációs) **egységek** (magyarul B/K, elterjedt angol rövidítéssel I/O) eszközök: amelyek révén a számítógépet használó ember kétirányú kommunikációt folytathat a számítógéppel

A Neumann-elvek azon túl, hogy immár több mint fél évszázada változatlanul határozzák meg a számítógépek felépítésének és működésének alapjait (ez még akkor is igaz, ha a mai korszerű számítógépekben a fenti szerkezeti elemek nem különülnek el szükségszerűen egymástól: a CPU és az ALU együttesen alkotja a processzort, stb.), alapvető hatást gyakorolnak az informatika mértékrendszerére is.

1.3 Mértékrendszerek

Az információ mértékegysége a bit.

A bit két diszkrét érték (egyértelmű) megkülönböztetését lehetővé tevő jel-pár. Megállapodás kérdése (és a 2. Neumann-elv hatása), hogy értékeit a kettes számrendszer jegyeivel (0, 1) jelöljük - lehetne akár a piros-kék szín pár is (mint az általános iskolában használt számolókorongok). A dolog szépséghibája, hogy a fentiekből az következik, hogy 1 bit összesen két állapot megkülönböztetését teszi lehetővé. Ha ennél több állapotunk van, akkor több bitre lesz szükségünk. n bittel pontosan 2^n különböző értéket tudunk jelölni.

Célszerűségi okokból a számítógépek felépítése és működése ezen bitcsoportok közül a 8 bitet tartalmazó csoportok köré szerveződött. A memória minden egyes tárolási egysége pontosan ennyi bit tárolására képes. 8 bit együttesét bájtnek nevezzük. (ilyen módon 1 bájton $2^8=256$ különböző érték ábrázolható.)

A bájt az információ-tárolás egysége.

A számítástechnikában a mértékek váltószámai (hasonlóan a tízes számrendszerhez) nagyságrendileg ezresenként követik egymást, de fontos tudni, hogy nem pontosan 1000 a váltószám, hanem $2^{10}=1024$! (Írásban – a bevett gyakorlat szerint – a bitet „b”, a bájtot „B” jelöli.) Ennek megfelelően

- 1 kilobájt (KB) = 1024 bájt (nagyjából ezer, 10^3)
- 1 megabájt (MB) = 1024 kilobájt = $1024^2 (=2^{20})$ bájt (nagyjából 1 millió, 10^6)
- 1 gigabájt (GB) = 1024 megabájt = $1024^3 (=2^{30})$ bájt (nagyjából 1 milliárd, 10^9)

- a további váltószámok növekvő sorrendben: tera (T, 10^{12}), peta (P, 10^{15}), exa (E, 10^{18}).

1.4 Kódrendszerek

A bit (illetve bájt) alapú adattárolásnak (pontosabban a kettes számrendszer kizárólagosságának) van egy másik hatása is: gondoljuk csak el, hogy hogyan magyarázzuk el a számítógépnek, hogy 29? Az ötlet egyszerű: váltsuk át kettes számrendszerbe, és kész. Rendben, működik. És hogy tároljuk le a 299-et? Az átváltással itt sincs gond, azonban a 299 kettes számrendszerbeli alakja hosszabb, mint 8 jegy (azaz 1 bájt, márpedig az előbb leszögeztük, hogy a memória bájtokat tárol)... És a gondok csak fokozódnak, ha az is felmerül bennünk, hogy hogyan tároljuk le azt a szót, hogy ALMA? Ezt még csak át se lehet váltani kettes számrendszerbe...

A számítógépek az adatot minden esetben kettes számrendszerbeli alakjukban (ún. bináris kódban) tárolják!

A számok esetében a bináris kód

- **fixpontos** ábrázolás esetén megegyezik a szám kettes számrendszerbeli alakjával (amennyiben a szám értéke olyan nagy, hogy egy bájton nem fér el a neki megfelelő kettes számrendszerbeli szám, akkor az egymást követő sorszámú rekeszekben tárolódik).

Példa: fixpontos ábrázolás esetén ha a memória egyik rekeszébe a 99-es értéket írjuk be, akkor a rekesz tartalma 01100011 lesz.

A fixpontos ábrázolás alapvetően két korlátozó tulajdonsággal rendelkezik: ábrázolási tartománya viszonylag kicsi és nem képes tárolni valós számokat.

- **lebegőpontos** ábrázolás esetén megegyezik a szám kettes számrendszerbeli alakjának normálalakjával.

Példa: lebegőpontos alakban a 99 érték tárolása helyett a $9,9 \cdot 10^1$ felírási mód egyes elemeit tároljuk, azaz a memóriarekesz(ek) tartalma a 99 és az 1 (amennyivel el kell tolnia tizedesvesszőt ahhoz, hogy az eredeti értéket megkapjuk) bináris kódja lesz. *(Megjegyezzük, hogy természetesen a normálalak is kettes számrendszerben képződik és az eltolást is 2 hatvány alakban kell érteni, a fenti példa csak az alapelv szemléltetését célozza!)*

- **BCD** (binárisan kódolt decimális) ábrázolás esetén megegyezik a szám jegyeinek kettes számrendszerbeli alakjainak sorozatával.

Példa: BCD ábrázolás esetén a 99 minden egyes jegyét külön-külön tároljuk (azaz nem a szám értéke, hanem az „alakja” kerül letárolásra): a memóriában 0000 1001 0000 1001 lesz található.

Szöveges értékek esetén a bináris kód

- valamilyen kódrendszer alapján az adott szöveges szimbólumot (karaktert, betűt) jelölő **sorszám**. Ismertebb kódolási rendszerek

- ASCII: eredetileg 7 bites kódolási rendszer, a karaktereket egy kódtáblázat alapján sorszámokkal látja el (pl. az „A” betűnek a 65 felel meg). A 7 bites kódolásnak köszönhetően ($2^7=128$) legfeljebb 128 szimbólum megkülönböztetését teszi lehetővé – ami az angol nyelv karakterkészletét tekintve elegendő, de ha figyelembe vesszük a nemzeti karaktereket is (pl. a magyar ékezetes betűket), akkor már nem. A **kiterjesztett ASCII** 8 bites kódolást használ olyan módon, hogy az első 128 sorszámhoz minden nyelvben ugyanazok (a 7 bites ASCII szabványnak megfelelő)

szimbólumok tartoznak, míg a 129-255 közötti sorszámokhoz az egyes nyelvek (szabványokban rögzített) eltérő karakterei kaptak helyet – ezeket a kiegészítéseket nevezzük **kódlapnak**.

- o MIME: ezt a kódolást elsősorban az elektronikus levelezési rendszerek alkalmazzák, hogy a levélben szereplő tetszőleges karaktereket a 7 bites ASCII kódtábla szimbólumaival le lehessen írni.
- o UNICODE: manapság legjelentősebb és legelterjedtebb kódolási rendszer az ASCII filozófiáját követi, de 2 bájtos kódolási rendszerének köszönhetően ($2 \text{ bájt} = 16 \text{ bit} \rightarrow 2^{16}$) hozzávetőlegesen 64 000 karakter különböztethető meg a segítségével.

Ebből az következik, hogy a számítógép számára egy memóriarekesz 8 bites tartalma bármi lehet: akár szám, akár egy karakter sorszáma, akár egy számsorozat egyik eleme – a számítógépnek nem tudja (és nem is tudja eldönteni!), ezért lényeges, hogy az adatokat feldolgozó programokat helyesen készítsük el!

Ellenőrző kérdések:

1. Határozza meg az adat és az információ fogalmát!
2. Mivel foglalkozik az információ-technológia (mint tudomány-terület)?
3. Melyek a számítástechnika szempontjából jelentős algoritmusok legfontosabb jellemzői?
4. Magyarázza meg a hardver és a szoftver fogalmát!
5. Ismertesse a Neumann-elveket!
6. Miben áll a „tárolt program elvének” jelentősége?
7. Melyek a számítógép (elvi) funkcionális egységei, mi a feladatuk?
8. Hogyan történhet a numerikus típusú adatok tárolása egy számítógépes rendszerben?
9. Milyen előnyös és hátrányos tulajdonságokkal rendelkezik a fixpontos, illetve a lebegő pontos ábrázolás?
10. Hogyan történik a karakterek tárolása a számítógépekben?

Irodalomjegyzék

[1] Kovács – Knapp – Ágoston – Budai: *Bevezetés a számítástechnikába (LSI, 1999)*

2. SZÁMÍTÓGÉP-ARCHITEKTÚRÁK

A XX. század közepétől a számítógépek történelme gyors és egyre gyorsuló ütemű fejlődést tükröz. Neumann János elvei mérföldkőnek bizonyultak – a mai napig meghatározzák a számítógépek többségének felépítését. Az első fejezetben ismertetett kategorizációt alapul véve a következő pontokban elsősorban a mikroszámítógépek és összetevőik jellemzőit tekintjük át, de először – különösen ezen összetevők funkcióinak megértéséhez – célszerű áttekintenünk az elmúlt évtizedek számítógép-generációit, a fejlődés kulcsmozzanatait. Az egymást követő generációk mindig forradalmian új felfedezések eredményei voltak, ennek megfelelően jelentősen átfarmálták hat évtizeddel ezelőtti gépek alkotóelemeinek fizikai és teljesítménybeli paramétereit.

2.1 Az architektúra fejlődése a XX. században

Az első elektronikus digitális számítógép, az ENIAC és kortársai fontos jellemzője az elektroncsöves technika. A hatalmas méretű és tömegű gépek programjai gépi kóddal készültek, a folyamatokat és minden adatforgalmat, így a perifériákra irányulókat is a központi vezérlő egység koordinálta. A Neumann által 1946-1951 között fejlesztett EDVAC² volt az első belső tárolású (Neumann elve szerint: „program és adat egy helyen”) gép, amely már a processzor helyett *memória*-centrikus következő generációk előfutárának tekinthető. A második generációt az 1954-től, a *tranzisztorok* – félvezető kapcsoló eszközök az elektroncsövek leváltására – első kísérleti alkalmazásától számíthatjuk, az 1960-as évek közepéig. E tíz évhez kapcsolódóan a tranzisztorok bevezetése mellett meg kell említenünk, hogy a bemeneti-kimeneti műveleteket a CPU-k helyett speciális perifériavezérlő áramkörök felügyelték. Ezek működésének szabályozására és szinkronizálására dolgozták ki a megszakítási eljárásokat, amelyek lényege: a periféria kezdeményezésére félbeszakított futó programot elmenti és a megszakítási eljárás befejezése után folytatja a gép.

A tranzisztorok gyakorlatilag minden modern elektronikai eszköz kulcselemei. Az 1960-as években folytatott kísérletek során kimutatták, hogy a méret csökkentésével a tranzisztorok sebessége arányosan nő, miközben a fogyasztása négyzetesen csökken. E felismerésből születtek a nagy mennyiségű tranzisztorból álló integrált áramkörök, amelyeket chip vagy mikrochip néven is ismerünk. A chipenként négyzetmilliméterenkénti tranzisztorszáma hatalmas ütemben nőtt, az ezredforduló után már milliós nagyságrendet ért el³.

E növekedés újabb korszakhatárokat jelent: harmadik generációs számítógépnek tekintjük az első, integrált áramkörökre épülő gépeket, amelyek tízes vagy százas nagyságrendben tartalmaznak tranzisztorokat, azaz esetükben alacsony (Small Scale Integration, SSI) vagy közepes (Medium Scale Integration, MSI, 1966 után) integráltsági fokról beszélhetünk. Az 1970-es évektől, a magas integráltsági fokú (Large Scale Integration, LSI) technológiák megjelenésétől számítjuk a számítógépek negyedik generációját. Ekkor vált általánossá az integrált áramkörökből felépülő mikroprocesszor mint központi feldolgozó egység, illetve az őt támogató memória- és ki-bemeneti áramkörök. Ahogy az integráltság növekedett (a szakirodalom a „Large” után „Very Large” és „Ultra Large” kifejezésekkel jellemzi a ’80-as, illetve a ’90-es évek technológiáját), a mikroprocesszorok újabb és újabb generációinak teljesítménye lettek a fejlődés újabb lépcsőfokai. 1981 óta, az első személyi számítógép megjelenése óta a legnagyobb újításokat e kategóriában a processzorok számának növekedése és a számítógép-hálózatok elterjedése hozták, azonban továbbra is – legalábbis e

² Elektron Discrete Variable Computer

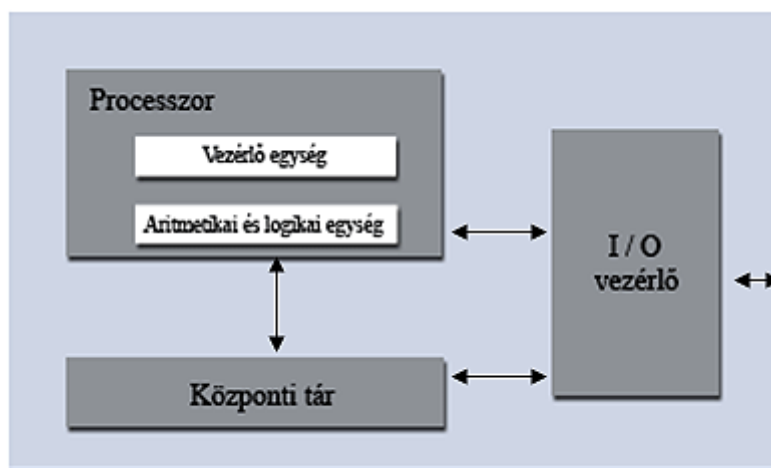
³ E növekedési ütemről szól Gordon Moore híres törvénye, amely szerint kb. kétfévente megduplázódik az egy áramkörbe zsúfolt tranzisztorok száma. Ezt az 1965-ös elméletet a mai napig helytállónak tekintik.

jegyzet írásáig és a jelenleg használt számítógépek döntő többségét tekintve – a negyedik generáció korát éljük. A következő nagy generációváltás a párhuzamos feldolgozásra képes, öntanuló és problémamegoldó számítógépekkel érkezhetsz el.

A számítógépek az 1. fejezetben látott működési elv szerint bemeneti, feldolgozó és kimeneti egységekből épülnek fel. Ezek a részegységek, azaz. modulok és a közöttük levő kapcsolatokat biztosító áramkörrendszerek a gépek nemzetközileg szabványosított alkotórészei, melyek közül elsőként a processzort és a memóriát vizsgáljuk meg.

2.2 A központi egység⁴

Az adatok feldolgozása utasítások sorozatával, azaz egy program végrehajtásával történik a számítógép központi egységében. A számítógépek első generációinak központi egységei a központi tárból (operatív memória), illetve a vezérlő egységből és számoló egységből (más néven: aritmetikai és logikai egység) álltak. E két utóbbi egység integrálásával született meg a mai értelemben vett processzor az integrált áramkörök bevezetésekor. Ekkorra a be- és kimeneti (I/O) műveletek irányítását a perifériák⁵ kezelését, kiszolgálását végző segéd-processzor vette át. Az I/O műveletek továbbító közegét, a központi egység, a központi tár és a perifériák között kapcsolatot teremtő áramkörök összességét sínrendszernek vagy más néven buszrendszernek nevezzük. Ezt a struktúrát mutatja a 2.1 ábra. A következő fejezetekben áttekintjük e hardverek fő jellemzőit.



2.1 ábra: A központi egység

2.2.1 A processzor⁶

A processzor a programok utasításait, adatait értelmezi és dolgozza fel, majd továbbítja az eredményt a számítógép adatútvonalain keresztül. A tulajdonképpeni

⁴ Megjegyezzük, hogy a „központi egység” és a „központi vezérlő egység” fogalmakat a szakirodalom olykor szinonimaként használja, jelen értelmezésünkben az előbbi magába foglalja az utóbbit, az I/O vezérlőket, a központi tárat és ezen elemek közötti kommunikációs csatornákat is.

⁵ A periféria gyűjtőnév: be- és kiviteli, valamint adattároló egységeket sorolhatunk ide. Lásd a 2.3 fejezetet.

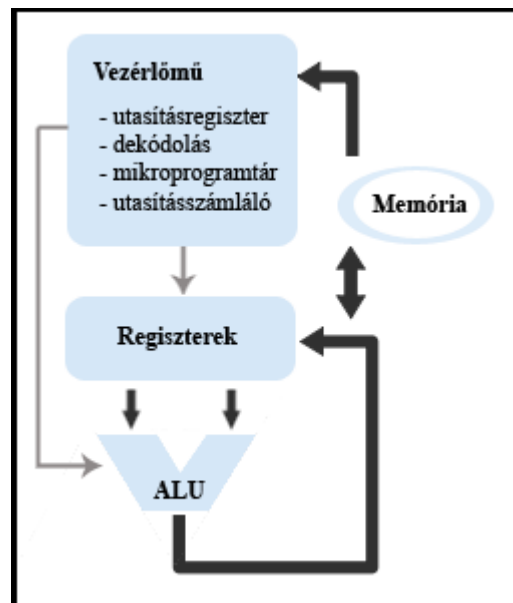
⁶ Processzor = Central Processing Unit. A magyar szakirodalomban „központi feldolgozó egység” és „központi vezérlő egység”-ként is hivatkoznak rá.

számítások végrehajtása mellett vezérli a belső adatforgalmat és a csatlakozó egyéb modulok tevékenységét. A processzor három alegységre tagolható:

- vezérlő egység (utasítás-előkészítő processzornak is nevezik),
- aritmetikai és logikai egység,
- regiszterek.

A processzorok egy művelet kezdetekor a feldolgozandó adatokat a memóriából a regiszterekbe másolják. A regiszterek tárolócellák (manapság általában 32 bit tárolókapacitással), amelyek magában a CPU-ban helyezkednek el; küldetésük a processzor működésének hatékonyabbá tétele, mivel a központi tárnál gyorsabb hozzáférést biztosítanak az adatokhoz. Nagy részük számadatok vagy memóriacímek tárolására szolgál, de van néhány speciális célú regiszter is, ezek közül néhányat ki is emelünk:

- Akkumulátor: a számítások első adatát és köztes (illetőleg az utolsó művelet után a végső) eredményeit tároló regiszter. A kortárs processzorokban akár 512 regiszterből álló tömb is betöltheti ezt a funkciót.
- Utasításregiszter: a soron következő művelet kódját tárolja.
- Utasításszámláló regiszter: a soron következő utasítás címét tárolja.



2.2 ábra: A processzor leegyszerűsített sémája

A 2.2 ábrán az eddig részletezett alkotóelemek és kapcsolatrendszerük egyszerű vázlata látható. Most röviden tekintsük át a fenti elemek működési mechanizmusát!

A vezérlőegység a program adott utasításának feldolgozásakor vezérlőjeleket küld az aritmetikai és logikai egységnek, az operatív tárnak, illetve adatok ki és bevitele esetén az I/O vezérlőegységnek, és utasítja őket a műveletek végrehajtására. Az első lépés az **utasítás beolvasása** az utasításregiszterbe az utasításszámláló által megadott címről. A következő lépés az utasítás **dekódolása**, részekre bontása. Az utasítások bináris jelsorozatok, *műveleti kódból* és *címrészből* állnak (előbbit *opkódnak*, utóbbit *operandus résznek* is hívják), a felépítésüket meghatározó szabályrendszert „gépi nyelv”-nek nevezik, magukat az utasításokat pedig gépi kódú programnak⁷. Minden CPU-architektúrának saját „nyelve”, *utasításkészlete* van: ez azon műveleti kódok összessége, amit a processzor értelmezni tud. Ez

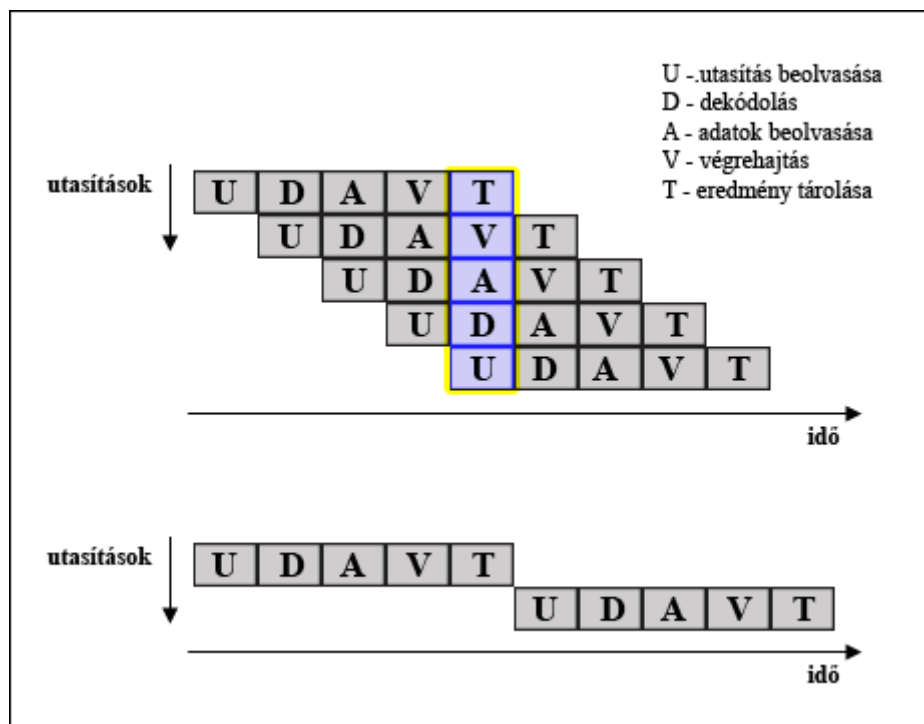
⁷ Minden magasabb szintű, azaz a programozók számára könnyebben értelmezhető programozási nyelvet gépi kóddá kell alakítani ahhoz, hogy a program végrehajtható legyen.

az értelmezés mikroprogramok segítségével történik: ún. mikroutasítások segítségével alakítják át a műveleti kódokat konfigurációs jelekké⁸. A dekódolt műveletek operandusai lehetnek konstansok, illetve egy adott regiszter- vagy memóriacellára mutató címzések. Miután ezek **beolvasása** megtörtént, az általuk megadott adatokkal az aritmetikai és logikai egység feladata a számítási műveletek **végrehajtása**. Az ALU kombinációs áramkörhálózata összeadást, bitléptetést és logikai műveleteket képes végrehajtani, az összetettebb műveleteket ilyen elemi részekre bontva oldja meg. A folyamat utolsó lépése az eredmény **viSSzaküldése** egy regiszterbe vagy a memóriába. Az utasítások egy másik csoportja nem az ALU-ban kezdeményez számítási műveletet, hanem például a programszámláló regiszter értékét módosítja: ezek *ugrási utasítások*. A folyamat ezután kezdődik előlről, a következő utasítás feldolgozásával.

A fentiekben felvázolt folyamatból látható az utasítás útja a processzorban, és az is, hogy az egyes lépésekben más-más részegységek foglalkoznak vele. Ha a feldolgozást minden utasítás esetében részlépésekre bontjuk fel, amelyek egymás után haladnak végig az ismertetett dekódolási és végrehajtási fázisokon, ideális esetben a processzor valamennyi egysége folyamatosan dolgozhat. Ez a pipeline-elv (pipeline angolul csővezeték), amely ugyan egyetlen utasítás feldolgozását sem gyorsítja meg, de lerövidíti egy utasítássorozat teljes feldolgozási idejét. Ezt a folyamatok párhuzamosítására irányuló technikát szemlélteti a 2.3 ábra felső része, amelyen látható, hogy egy hosszú utasítássorozatot véve az ötödik utasítás beolvasásától kezdődően a processzor valamennyi részegysége munkában van. Figyeljük meg a módszer – elméleti – hatékonyságát: az ötödik utasítással is végzett a processzor annyi idő alatt, mint amennyi az alsó ábrán látható, lineáris szervezésű változatnak szükséges két utasítás feldolgozására⁹.

⁸ A mikroutasításokat a processzorok tervezőmérnökei alkotják meg, a lehető leggyorsabb végrehajthatóságra optimalizálva, és leggyakrabban egy csak olvasható *mikroprogramtárban* tárolják őket, olykor azonban ez a tár írható is. Mikroutasítás lehet például a következő: „Tárold az eredményt a 3-as regiszterben!”

⁹ A fentiekben bemutatott pipeline-elv továbbfejlesztett változatát alkalmazzák a modern *szuperskalár* processzorokban (az elnevezés magyarázata: egy órajel alatt több művelet), amelyekben egyes funkcionális egységek (például az ALU) számát megnövelték.



2.3 ábra: A pipeline-elv¹⁰

A feldolgozási folyamatban a processzor több részegysége önállóan dolgozik, e részegységek összehangolására szolgál az órajel-generátor, amely egy nagy pontosságú, kvarckristállyal vezérelt rezgéskeltő. Ez az „óra” meghatározott időpillanatonként szinkronizációs jeleket szolgáltat, két ilyen jel jelent egy *processzorciklust*, ami a feldolgozási idő mértékegysége (az utasítások bonyolultságától függ, hogy hány ciklus alatt hajtja végre őket a processzor). Az órajel nanoszekundum nagyságrendű, és reciprokát, a MHz-ben vagy GHz-ben megadott frekvenciát a CPU egyik fő teljesítménymutatójának tekinthetjük.

A modern processzorok minél hatékonyabb párhuzamosításra törekvő fejlesztési irányzatai – például a főleg tudományos számításokhoz használt vektorprocesszorok és az FPGA¹¹-k – már túlmutatnak e jegyzet keretein, egyes architektúrális jellemzőkre viszont még visszatérünk a 2.2.4. fejezetben.

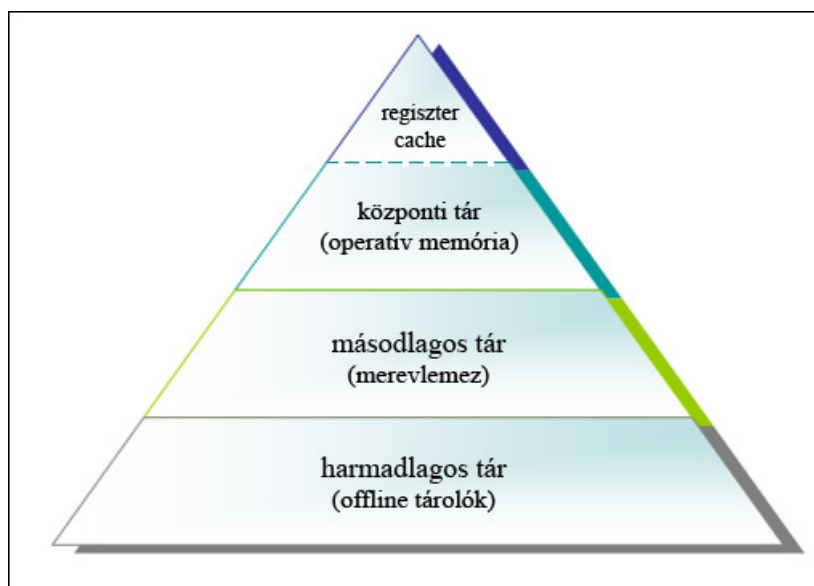
2.2.2 Memória

A memória a számítógép olyan egysége, amely képes adatokat és programokat tárolni valamekkora időtartamra – ezáltal a neumann „tárolt program elvének” alapját képezi. Szűkebb értelemben – és a gyakorlatban – a memória megjelölést a központi tár szinonimájaként használjuk, tágabb értelemben viszont ide tartoznak a regiszterektől a háttértárolókig mindazon elemek, amelyek képesek ideiglenesen vagy tartósan megőrizni a nekik átadott információt. A tárolóeszközök tehát sokfélék lehetnek, nem létezik minden igényt kielégítő univerzális típus. Az egyes változatok más-más funkciót valósítanak meg a

¹⁰ Megjegyezzük, hogy az ábra felső része egy olyan ideális működést mutat, ami a gyakorlatban ritkán valósulhat meg. Ez a technika akkor hatékony, ha a feldolgozandó utasítások egymástól függetlenek, tehát pl. a 2. utasítás végrehajtásához nincs szükség az első eredményére. Az ábrán egy „klasszikus” ötlépcsős pipeline látható, egyes modern CPU-k ennél jóval több lépcsővel működnek (példa erre a 31 lépcsős Intel Pentium 4).

¹¹ Field Programmable Gate Array, azaz programozható kapuáramkörök tömbje. A CPU átprogramozható a megoldandó feladat függvényében.

számítógépekben, és többféleképpen is kategorizálhatók, elsőként a számítógépek hierarchikus társzervezését mutatjuk be.



2.4. ábra: A számítógépek tárhierarchiája

Ahogy a 2.4 ábrán látható, a társtruktúrában fentről lefelé haladva megkülönböztetünk *elsődleges*, *másodlagos* és *harmadlagos* tárolókat:

- Az elsődleges tároló a processzorral közvetlen kapcsolatban áll. Háromféle tártípust sorolhatunk ide:
 - regiszterek: a leggyorsabb tárolók, a CPU-ba beépítve
 - cache memória¹²: más néven előmemória vagy gyorsítótár
 - operatív memória
- A másodlagos tárolót a központi egység az I/O csatornákon keresztül éri el; az elsődleges tárnál nagyobb kapacitású, de lassabb elérésű mágneses háttértároló. Tartós adattárolásra szolgál, tartalmát az elsődleges tárral ellentétben a számítógépek kikapcsolásakor is megőrzi.
- A harmadlagos tár fogalmába tartoznak az olyan off-line formák, mint az optikai lemezek, a floppy lemez, a mágnesszalagos tár vagy a flash memória (USB pendrive). Ide sorolhatók még olyan, nagygépes (hálózati) rendszerekben használatos tárolók, amelyeket a rendszer igényei szerint, emberi beavatkozás nélkül csatlakoztatnak az I/O csatornához (pl. jukebox).

Az első fejezetben már említettük a különbséget a csak olvasható, illetve a véletlen elérésű, írható/olvasható memóriatípusok, azaz a ROM és a RAM között. A számítógépek operatív memóriájaként funkcionáló, félvezető alapú, integrált áramkörökből álló egységére idővel „ráragadt” a RAM elnevezés – így a mai terminológiában RAM alatt már leggyakrabban magát a hardverelemet értjük. Ez az értelmezés félrevezető lehet, mivel a RAM mint típuskategória két fajtáját különböztetjük meg:

¹² A cache tartalmát a memóriából való olvasáskor tölti fel a CPU a leggyakrabban használt memóriacímekről, ezzel a CPU adatelérése gyorsabbá válik. Kezdetben a számítógépek alaplapján helyezték el, az ezredforduló környékén és azután fejlesztett mikroprocesszorok beépített, többszintű (2 vagy 3) cache-sel dolgoznak.

- A statikus RAM (SRAM) olyan áramkörökből (billenőkörök) áll, amelyek áramellátás mellett megőrzik a bennük tárolt információt frissítés nélkül; a cache memória ilyen típusú.
- a dinamikus RAM (DRAM) a SRAM-nál nagyobb sűrűségű és olcsóbb¹³, viszont rendszeres újratöltést (másodpercenként akár több ezer frissítést) igénylő típus. A központi tárat dinamikus RAM-ból építik fel.

A dinamikus RAM többféle típusát alkalmazták központi tárként az utóbbi három évtizedben, ezek közül a jelenleg általánosan használt *DDR* mellett csak érdekességként megemlítünk néhány korábbi fontosabb változatot is, amelyekkel még ma is találkozhatunk egy-egy korosabb gépben:

- FPM: Fast Page Mode (azaz „gyorslapozású”) DRAM, az első PC-k korából;
- EDO: Extended Data Out (azaz kiterjesztett adatkimenetű) DRAM, az FPM-nél 2-5%-kal gyorsabb és a kimenő adatokat tovább megőrző változat;
- SDRAM: Synchronous (a rendszersínnel és ezáltal a processzorral szinkronban működő) DRAM;
- Direct Rambus DRAM: az SDRAM-nál akár tízszer gyorsabb, de jóval drágább és ritkán (egyes Intel CPU-k mellett) használt változat;
- DDR SDRAM: Double Data Rate Synchronous (duplázott szinkron) DRAM; a nevéből adódóan az SDRAM-hoz képest kétszer annyi adatot képes mozgatni azonos frekvencián. A jelenleg legelterjedtebb változat (mind a központi tárat, mind a grafikus megjelenítőket tekintve)¹⁴.

A központi tárban egy adott időpillanatban az éppen aktív programok és a hozzájuk tartozó adatok, valamint a számítógép működését irányító operációs rendszer egy része (vagy egésze) található. (Az éppen nem használt programok és adatok a másodlagos / harmadlagos tárolókban helyezkednek el.) A tár legegyszerűbben rekeszek, memóriacellák összességéként képzelhető el, ahol az egyes cellákat azonosító számokat a cella címezésének nevezzük, ezek összessége a címtartomány. A processzor a feldolgozás során a címzéseken keresztül éri el a tárban tárolt adatokat. A központi tár működésének jellemző mutatói: a kapacitás (a cellák száma) és az elérési idő. Utóbbitt másképpen a memória válaszüjének is nevezik, ez a beérkezett kéréstől az adat rendelkezésre állásáig eltelt, nanoszekundum nagyságrendű idő.

Az operatív memória kapacitása a kortárs gépekben a felhasználási terület függvényében elég széles skálán mozog, pár száz megabájttól több gigabájtig, általános tendencia azonban, hogy az alkalmazások tárigénye folyamatosan növekszik. Előfordulhat, hogy a központi tárban nincs elég szabad kapacitás egy futtatandó program számára, ez a probléma azonban különféle tárkezelési technikákkal feloldható. Az ún. virtuális memóriakezelés felhasználja a központi tár és a másodlagos tároló kapacitását egyaránt: csak a végrehajtás alatt levő programrészek (és a hozzájuk kapcsolódó adatok) helyezkednek el a memóriában, a többi programrészt a merevlemezen tárolják¹⁵. Ez általában a lapozós technikán alapszik: ennek lényege, hogy a futtatandó program címtartományát és a memóriát rögzített méretű (általában 4 kilobájt) alegységekre, *lapokra*, illetve *adatblokkokra* bontják, és a lapok és blokkok megfeleltetésére egy ún. *laptáblázat*ot használnak. Így elegendő, ha az éppen aktív utasítás lapja bent van az operatív memóriában. Ha a soron következő utasítás lapja nincs bent, akkor a merevlemezről kell beolvasni, szükség esetén egy korábbi lap

¹³ A DRAM bitenként egy tranzisztort és egy kondenzátort igényel, a SRAM 6 tranzisztort.

¹⁴ A DDR generációiról még lesz szó a 2.2.4 fejezetben.

¹⁵ Microsoft operációs rendszereiben *lapozófájl*ban, míg Linux rendszerekben ún. *cserepartíció*kban.

eltávolítása árán. A folyamatot a processzorok memóriakezelő egysége (Memory Management Unit, MMU) irányítja, amely többek között a virtuális memória címezéseit képezi le a tár fizikai címekre.

2.2.3 A sínrendszer és az alaplapp

A számítógépek hardver moduljai között kapcsolatot teremtő vezetékek és jelerősítő áramkörök összessége a *sínrendszer* (bus system, buszrendszer). Belső sínek kapcsolják össze a processzor elemeit (ezekkel e jegyzetben nem foglalkozunk), ezért memóriával, valamint a be- és kiviteli eszközökkel való kommunikációt biztosító síneket szokás külső sínrendszernek is nevezni. Az adattovábbításhoz használt jeleket a vezetékek feszültség szintje adja¹⁶. A síneken továbbított tartalom szerint három kategóriát különböztetünk meg:

- vezérlősín: vezérlőjelek futnak rajta a CPU és a többi egység között, amelyek az adatforgalmat irányítják, de többek között ide tartoznak a megszakításokat vezérlő és a szinkronizáló jelek is;
- adatsín: a processzor, a központi tár és a perifériák közötti adatcsere közege;
- címsín: a memória egyes, írni vagy olvasni kívánt rekeszeit meghatározó jelek áramlanak rajta.

A mai számítógépekben alkalmazott sínrendszerek összetettek, számos funkcionális alegységgel bírnak, de közös általános jellemzőjük a sebesség (azaz a sín órajele, MHz-ben kifejezve) és az adatszélesség. Ez utóbbi a sín vonalainak, ereinek számából ered – minden vonalon egy időpillanatban egy bit információ továbbítható. Az adat- és címbuszok az első PC-k korában 8-16 bit szélesek voltak, a kortárs gépekben általában 32-64 bit szélesek. A címsín szélessége felső határt szab az operatív memória mennyiségének: például 32 bites szélesség esetén maximálisan 2^{32} bájt, azaz 4 gigabájt méretű memória használható.

A sínrendszer vezetékai egy többretegű, nyomtatott áramköri lapon, az alaplapon helyezkednek el. Az alaplapp a számítógép gerincét adja: különböző méretű és alakú csatlakozói és a ráintegrált áramkörök útján fizikai kapcsolatot teremt a processzor, a memóriamodulok, a ki- és bemeneti bővítőkártyák, illetve a háttértárak és egyéb perifériák között. Utóbbiak csatlakoztatására szabványos ún. interfészek (portok, kapuk)¹⁷ szolgálnak, amelyeknek két fő kategóriába tartoznak:

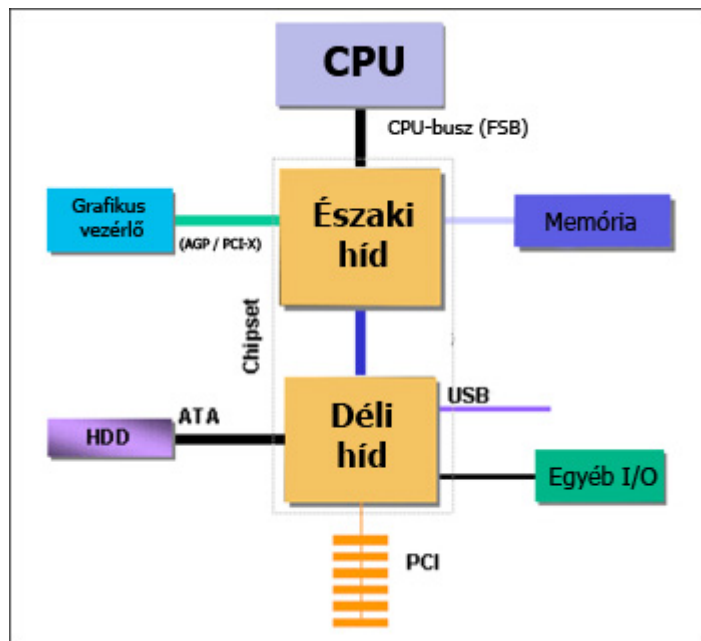
- soros elvű kommunikáció esetén egy időegység alatt egy bit adat átvitele történik (a bitek sorban követik egymást),
- párhuzamos elvű kapcsolat esetén többszálú az adatátvitel (például egy bájt minden bitje egyszerre lesz továbbítva).

Következzen néhány példa! A soros interfészek közé tartozik például a PC-kben népszerű RS-232-C szabvány, amit általában csak soros portként emlegetnek, és többek között egér vagy modem csatlakoztatására használták. A párhuzamos kapcsolatra példa az IEEE 1284 szabványú Centronics-interfész (a gyártó nyomán), amelyet párhuzamos portnak vagy nyomtatóportnak is neveznek. Az IBM által kifejlesztett PS/2 interfész billentyűzet és egerek csatlakoztatására használatos. A soros és párhuzamos port mára széles körben elterjedt utódjának tekinthető az USB (Universal Serial Bus, azaz általános soros busz) csatlakozófelület.

¹⁶ A bináris 0 és 1 értékeknek felel meg a vezetékek 0 V közeli, illetve szabványtól függően 12 V / 5 V / 3,3 V / 1,5 V stb. feszültség szintje.

¹⁷ Az interfész általános fogalmába beletartozik a fizikai csatlakozófelület mellett a hozzá kapcsolódó vezetékrendszer. Megjegyezzük továbbá, hogy a port kifejezés más jelentéssel is használatos: kommunikációs hálózatok csatlakozási pontja.

A sínrendszer működését az alaplapon egy vagy több vezérlő chip irányítja, ezek alkotják az alaplapon chipkészletét (chipset). PC-s környezetben az utóbbi évek trendje szerint általában két, különböző funkciókkal felruházott lapka, az északi (northbridge) és a déli híd (southbridge) látja el ezt a feladatot. A 2.5 ábrán ennek a struktúrájának egy leegyszerűsített vázlata látható:



2.5. ábra: Északi és déli híd¹⁸

Forrás: Intel Co.

Az északi híd feladata a processzor, az operatív memória és a grafikus vezérlő¹⁹, valamint a déli híd közötti kapcsolattartás; olyan változatok is léteznek, amelyek a chipbe integráltan tartalmazzák a grafikus vezérlőt. Az északi híd a PC-konfigurációk meghatározó eleme: ettől az egységtől függ, hogy hány darab, milyen típusú és órajelű processzort építhetünk be, mekkora mennyiségű, milyen típusú és órajelű RAM-modulokat alkalmazhatunk; egy északihíd-típus általában egyféle processzorarchitektúrát és RAM-ot támogat. A sínrendszer az északi hidat és a CPU-t összekötő része az előoldali busz (FSB: front side bus) vagy más néven CPU-busz, míg a híd és a memória kapcsolataért felelős részt memóriasínnek is nevezik²⁰.

A déli híd az északi hídon keresztül kapcsolódik a processzorhoz, és a központi egységeknél jellemzően alacsonyabb órajelen működő egységeket kapcsolja össze: ide

¹⁸ A „híd” elnevezés a két chip kapcsolatteremtő funkciójára utal, az északi és déli jelzők az ábrán látható elrendezésből adódnak. Megjegyezzük, hogy egyes újabb CPU-k beépített memóriavezérlővel dolgoznak, ezekben az architektúrákban az északi és déli hidak szerepét egyetlen chip látja el.

¹⁹ Jelenleg elterjedt szabványok: AGP (Accelerated Graphic Port) és az AGP-t felváltó PCI-Express

²⁰ A processzor órajele az FSB órajelének és a processzor belső szorzójának szorzataként adódik; a memóriasín órajele korábbi generációkban gyakran az FSB órajelével egyezik meg, a kortárs rendszerekben lehetőség van e sínek aszinkron működésére is. Az FSB tágabb értelemben a teljes CPU-híd-memória kapcsolatot jelenti.

kapcsolódik a PCI²¹-sín és az USB, továbbá a gyakran az alaplaphoz integrált hang- és hálózati (LAN) vezérlők, a merevlemez-vezérlők, illetve az ábrán „Egyéb I/O” felirattal jelzett egység (Super I/O chip), amely a soros, párhuzamos és PS/2 portokat, illetve a floppy meghajtó vezérlőjét kapcsolja a sínrendszerbe (amennyiben létezik, ugyanis az USB térhódításával ezek a kapcsolatok gyakorlatilag feleslegessé váltak).

A déli híd két fontos funkciója a megszakítás-vezérlés (Interrupt controller) és a közvetlen memória-elérés (DMA: Direct Memory Access) vezérlése. A megszakítások a CPU által éppen feldolgozott program ideiglenes szüneteltetésével lehetőséget biztosítanak azonnali beavatkozást igénylő feladatok „soron kívüli” megoldására. A megszakításokat általában valamilyen periféria kezdeményezi, a megszakítási eljárás befejezése után a CPU visszatér az eredeti feladatához. A DMA-vezérlő többek között a grafikus vezérlő és a háttértároló számára nyújt közvetlen kapcsolatot az operatív memóriával, a processzortól függetlenül – ilyen művelet például a memória egy adott méretű blokkjának a másolása a grafikus kártya saját tárába. A módszer nagy előnye a processzor tehermentesítése: nem tartják fel ezek a relatíve sokkal lassabb egységeket érintő műveletek.

2.3 Perifériák

A periféria a számítógép funkcióit bővítő hardverelemek általános elnevezése. Kétféle értelemben is használatos:

1. Szűkebb értelmezés szerint csak azokat az eszközöket soroljuk ide, amelyek külsőleg csatlakoznak a számítógéphez, használatuk opcionális. Ekkor például a monitort és a merevlemez nem soroljuk ide, mert a gépek többségének alapvető tartozékai.
2. Tágabb értelemben minden bemeneti / kimeneti eszköz és a háttértárak is ide tartoznak, ekkor a hardver kategóriáit a központi egység – perifériák szerinti tagolásban értelmezzük.

A továbbiakban a második értelmezést követve áttekintjük a legfontosabb eddig nem említett hardvertípusokat

2.3.1 Háttértárolók

A 2.4 ábrán bemutatott tárhierarchia másodlagos és harmadlagos szintjét jelentő eszközök a központi tárhoz viszonyítva relatíve nagy – és időben tartós – tárolási képességű háttértárak. Általánosan elmondható, hogy a tárhierarchia egyes szintjein lefelé haladva növekszik a tárolóeszközök kapacitása és elérési ideje, csökken egy tárolóegységre jutó költsége. A háttértárak közvetlen, perifériavezérlő rendszereken keresztül kapcsolódnak a központi egység sínrendszerére, legtöbbször kétirányú (adatok írása / olvasása) adatforgalommal. Az adatok tárolására többféle technika létezik, a legelterjedtebbek a mágneses és az optikai adattárolók. A fejezetben ezek néhány típusát ismertetjük, illetve említést teszünk az egyre népszerűbb memóriakártyákról.

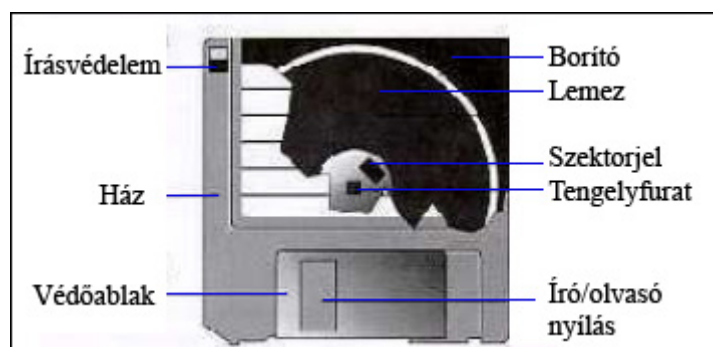
2.3.1.1 Mágneses tárolók

Az első széles körben használt, mágneses elven működő tárolók a mágnesszalagos meghajtók voltak. Ezekben a meghajtókban egy műanyag szalagra felhordott mágneses

²¹ Peripheral Component Interconnect; bővítőkártyák (pl. hang- vagy videovezérlők) általános csatolófelülete általában 33 MHz órajelen, az átviteli sebesség elméleti maximuma 133 MB másodpercenként. Utódja az egyelőre főként videovezérlők illesztésére használt PCI-Express, ami elméletileg harmincszor gyorsabb adatátvitelre képes.

rétegben több sávon, párhuzamosan rögzíthetők az adatok; a tárolás biztonsága érdekében hibajavító és ellenőrző kódokat is alkalmaznak. Az adatok visszaolvasásához a szalag megfelelő helyre való tekerése szükséges – ez soros elérési mód. A mágnesszalagos meghajtók ma már elavult technológiát képviselnek, főleg nagy méretű adattárak biztonsági mentésére használatosak.

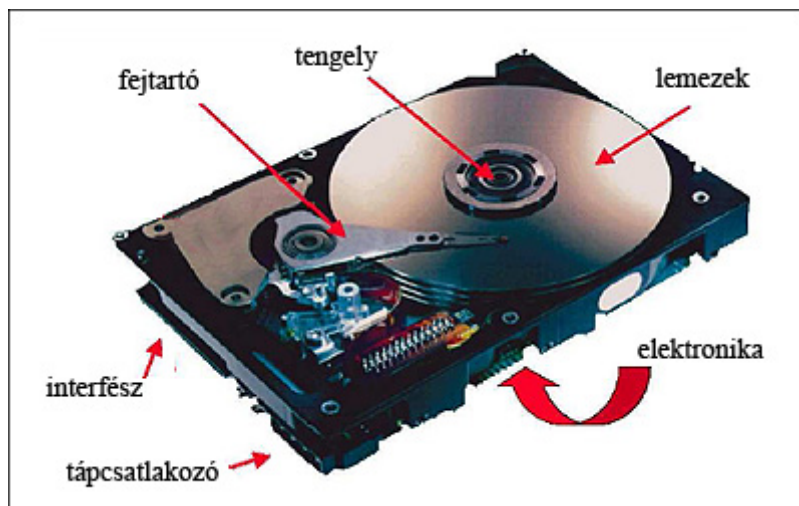
A mágneslemezes tárolókban a mágneses réteget forgó lemezekre hordják fel, az adatokat koncentrikus körök, ún. sávok mentén rögzítik. A lemez legkisebb írható/olvasható egysége a szektor, amely általában 512 bájt tárolására képes; több szektorból (2, 4, 8) állnak a klaszterek, amelyek a legkisebb logikai tárolási egységek (az operációs rendszer minimum egy klaszternyi terület allokál egy fájl tárolására). A lemezek száma és alapanyaga szerint beszélhetünk hajlékony- és merevlemezekről. A hajlékonylemez (FDD, Floppy Disk Drive) – más néven floppy – több, különböző (8”, 5,25”) méretű és adatsűrűségű generációja közül ma már többnyire csak a 2.6 ábrán látható, 3,5” méretű, 1,44 MB tárolására alkalmas típusokkal találkozunk. A floppyk cserélhető tárolók előnyük a hordozhatóság és a merevlemezhez viszonyítva alacsony költség, azonban az ugyanezekkel az előnyökkel és nagyságrendekkel nagyobb kapacitással bíró optikai tárolók és memóriakártya-típusok terjedésének köszönhetően az 1990-es évek vége óta jelentősége folyamatosan csökken²².



2.6. ábra: 3,5” hajlékonylemez

A merevlemez (HDD, Hard Disk Drive) – más néven winchester – a legelterjedtebb adattároló eszköz, felépítése a hajlékonylemezekéhez hasonló, de adatsűrűsége és írási/olvasási sebessége sokkal nagyobb. A merevlemez valójában több, egymás felett elhelyezett lemezből áll, amelyek folyamatosan, nagy sebességgel (percenként 5400, 7200, 10000 vagy még több fordulat) forognak – a lemezek felett mozgó író/olvasó fej így az egymás alatti lemezek azonos sávjait ugyanabból a pozícióból képes olvasni (a sávok vertikális összességét cilindernek nevezzük). A kortárs merevlemezek tárhatalma eléri a több száz GB-ot, az adatátviteli sebességük elméleti maximuma a jelenlegi leggyorsabb modelleknél 300 MB/sec.

²² Megjegyezzük, hogy a hajlékonylemezek továbbfejlesztésével születtek több száz MB adat tárolására képes változatok, például a Superdiskek és Zip-drive-ok, de ezek sem vehették fel a versenyt az egyre csökkenő költségű CD-vel, DVD-vel és a flash meghajtókkal.

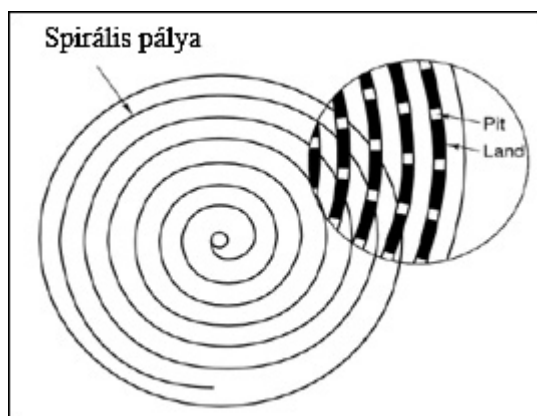


2.7. ábra: A merevlemez felépítése

A 2.7. ábrán látható merevlemez „elektronika” feliratú eleme a meghajtó működését vezérlő egység, amelyet az átvitel sebességét növelendő összeépítették a meghajtóval, innen származik a merevlemezek csatlakoztatására használt interfész egyik elnevezése, az IDE (Integrated Drive Electronics)²³.

2.3.1.2 Optikai tárolók

Az optikai lemezek két fő családja a kezdetben hanganyagok tárolására fejlesztett CD (Compact Disc) és a DVD (Digital Versatile Disc, Digital Video Disc), mindkét lemezfajtának több változata is elterjedt. Ezek közös tulajdonsága, hogy az adatokat spirális sávokban rögzítik valamilyen fényérzékeny anyag használatával. Az adattárolásra apró bemélyedések, gödrök (pit) szolgálnak; a lemez letapogatásakor a gödrökből visszaverődő fénymennyiség kisebb, mint ahol nincs bemélyedés (ezeket a szakaszokat „land”-nek hívják, lásd a 2.8 ábrát). Az optikai fej ezt a visszavert fénymennyiséget érzékeli, ebből állítja elő a bináris jelsorozatot.



2.8. ábra: Egy optikai lemez vázlata

²³ Ez valójában az ATA (Advanced Technology Attachment) szabvány első változatának másik elnevezése. Az ATA interfész eredetileg párhuzamos volt, utódja, a gyorsabb Serial Ata technológia soros elvű.

A CD-ROM elnevezést CD-k olvasására szolgáló meghajtókra alkalmazzuk leggyakrabban, de eredetileg ez a csak olvasható, gyári körülmények között készült lemezek megjelölése volt (Compact Disc Read-only Memory). A CD-R (CD Recordable) megjelölés egyszer írható lemezekre vonatkozik, amelyek az írófejjel ellátott meghajtókhoz, PC-s környezethez készültek. A CD-RW (ReWritable) változat a CD-R továbbfejlesztése, többször írható / törölhető lemez. A CD lemezek általában 650, 700 vagy 870 MB kapacitással bírnak, amit szokás a rögzíthető hanganyag hosszával is megadni, így beszélhetünk 74, 80 és 99 perces változatokról.

A DVD lemez tárolókapacitása a CD lemezeknél elérhető értékek sokszorososa; elsősorban nagyméretű hang- és videoanyagok tárolására fejlesztették, majd a megfelelő meghajtók (DVD-ROM) megjelenésével – többek között – a merevlemezek tartalmának archiválására is alkalmas lett. DVD lemezből létezik egyoldalas vagy kétoldalas, illetve egyrétegű vagy kétrétegű változat – az ezekből adódó négy kombinációból leggyakrabban az egyoldalas, egyrétegű, 4,7 GB tárolókapacitású DVD-5, illetve az egyoldalas, kétrétegű, 8,5GB tárolókapacitású DVD-9 formátumokkal találkozhatunk. A CD-khez hasonlóan a DVD lemezekből is létezik egyszer írható és újraírható változat, az összkép itt annyival bonyolultabb, hogy a nagy gyártók két csoportja kétféle technológiát fejlesztett, így beszélhetünk például DVD-R („mínuszos”) és DVD+R („plusszos”) kategóriákról. A technológiai kettősség problémáját az ún. multiformátumú, „mínuszos” és „plusszos” változatokat egyaránt támogató meghajtók oldották fel.

Megjegyezzük, hogy az optikai lemezek következő generációi, a Blu-Ray Disc (BD) és a HD-DVD (High Density DVD) e jegyzet írásakor már közvetlenül megjelenés előtt állnak, várhatóan 2006 második / harmadik negyedévében jelennek meg²⁴. A BD lemezek kapacitása nagyobb (rétegenként 25 GB szemben a HD-DVD lemezek 15 GB-os tárolási képességével), de a HD-DVD változatok előállításának költsége kezdetben alacsonyabbnak ígérkezik.

2.3.1.3 Chipkártyák, flash memória, pendrive

Említést kell még tennünk az egyre népszerűbb chipkártyákról, memóriakártyákról, amelyek hordozható adattárolókként való alkalmazása egyre népszerűbb. A kártyákba beépített chip lehet mikroprocesszor vagy memóriachip. Az első kategóriába tartoznak az olyan „intelligens kártyák”, mint az ATM-kártyák vagy a mobiltelefonokban használt SIM-kártyák, míg a memóriachipeket főleg digitális kamerákban, hordozható számítógépekben használjuk, illetve hordozható adattárolóként.

A memóriachipek többsége flash memória: olvasható, törölhető és újraprogramozható memóriacellákból áll. A flash memóriák speciális tranzistorai áramellátás nélkül is képesek információ tárolására; tartalmuk olvasásának/írásának sebessége az optikai lemezekéhez hasonló, de jóval ellenállóbbak. Egyes főleg kamerákban használt memóriakártyák tartalmát a számítógéphez (az USB porton) csatlakoztatható külső olvasóegység használatával másolhatjuk a számítógép háttértáaira, míg a *pendrive* néven is ismert USB flash meghajtók közvetlenül csatlakoztathatók az USB interfészre. A pendrive-ok kapacitása gyors ütemben nő, jelenleg eléri a 4-5 GB-ot is.

²⁴ Ismét két, egymással versenyző technológia jellemzi tehát a jövő optikai tárolóit, de vélhetően a kompatibilitási nehézségek ismét kikényszerítik a mindkét technológiát támogató meghajtókat.

2.3.2 Adatbeviteli eszközök

A beviteli (input) perifériák közé tartozik a billentyűzet, az egér és egyéb mutatóeszközök, valamint a jelolvasók. Az utóbbi kategóriába tartoznak többek között a scannerek (lapolvasók) – amelyek funkcióját gyakran ún. multifunkciós, bevitelre és nyomtatásra egyaránt alkalmas készülékek veszik át –, illetve különféle vonalkód-olvasók és mágneskártyáknál használt olvasók.

A billentyűzet a legalapvetőbb beviteli eszköz, amelynek kialakítása a régi írógépek billentyű-elrendezésén alapszik. Egy adott billentyű vagy billentyűkombináció lenyomásakor és felengedésekor egy-egy adott kódot továbbít a billentyű vezérlője a CPU-nak. A billentyűk többsége karakteres: egy-egy betű, számjegy vagy egyéb jel van hozzájuk rendelve. Speciális billentyűk:

- váltó (Shift, Ctrl, Alt),
- kapcsoló (Num Lock, Caps Lock, Scroll Lock)
- kurzormozgató (←, ↑, →, ↓, Home, End, Page Up, Page Down)
- törlő és beszúró (Backspace, Insert, Delete)
- funkcióbillentyűk (F1... F12)
- egyéb billentyűk (például az ún. WinKey )

A mutatóeszközök vagy pozicionáló eszközök a képernyő egy adott pontjára mutatnak. Legelterjedtebb ezek közül az egér, amely legalább két gombbal és valamilyen mozgáskövető mechanizmussal rendelkezik:

- A mechanikus egér használatkor a belsejében egy általában gumi anyagú golyó forgása közvetíti az elmozdítás eseményét: a mozgás irányát és sebességét görgők érzékelik, ezekből határozza meg az egér elektronikája az új (relatív) pozíció koordinátáit.
- Az optikai egérben fénykibocsátó LED-et alkalmaznak, a kibocsátott sugarak visszaverődése alapján állapítja meg az elmozdulás irányát és sebességét.
- A hordozható számítógépekben alkalmazott érintőpad (touchpad) nyomásra érzékeny szenzorokkal működik.

Az egéren kívül egyéb mutatóeszközök is léteznek: például főleg játékokhoz, szimulátorokhoz használt mutatóeszköz a botkormány (joystick).

2.3.3 Adatkiviteli (megjelenítő) eszközök

Ebben a pontban a fő megjelenítő hardvereket tárgyaljuk: a monitor és a nyomtató jellemzőit foglaljuk össze, továbbá röviden kitérünk olyan multimédiás eszközökre, amelyek adatkiviteli és –beviteli funkciót egyaránt elláthatnak. A perifériák negyedik csoportjára, a számítógép-hálózatokban alkalmazott adatátviteli eszközökre a 8. fejezetben térünk vissza.

2.3.3.1 Képernyő (monitor)

A legfontosabb vizuális megjelenítő eszköz, amely a tv-technológia számítógépekre alkalmazott változatából alakult ki. Főbb jellemzői a következők:

- működési elv szerint katódsugárcsöves (CRT) vagy folyadékkristályos (LCD) monitorokat különböztetünk meg²⁵
- a képátló (hüvelykben), például 14”, 15”, 17” stb.

²⁵ A két említett, meghatározó kategória mellett természetesen egyéb technológiával is találkozhatunk a monitorokban, példa erre a gázkisüléses kijelző, ismertebb nevén plazma technológia.

- felbontás, színmélység és frissítési frekvencia

Az utolsó három tulajdonság egymással összefüggő, működés közben változtatható paraméter, amelyek lehetséges értékei a képernyő és a grafikus vezérlő adottságaitól egyaránt függenek. A felbontást képpontban (pixelben) adjuk meg, a megjelenített sorok és oszlopok számával (például 1024*768). A színmélység a megjeleníthető színek számát határozza meg, (például 16 bites színmélység esetén $2^{16} = 65536$ szín jeleníthető meg). A frissítési frekvencia két, egymással összefüggő mutatót jelent: míg a vízszintes frissítés az időegység alatt kirajzolható sorok számát adja meg, a függőleges a teljes képre vonatkozólag jelzi ugyanezt. A két értéket a felbontás kapcsolja össze. Elvileg tehát a maximális vízszintes frekvencia a felbontásban megadott sorok számával elosztva a függőleges frekvencia maximális értékét adja.

A megjelenítésért az alaplapha illeszthető bővítőkártya, a VGA-kártya (vagy ritkábban az alaplaphra integrált vezérlő) felel. A VGA-kártyákat grafikus alrendszernek is nevezhetjük, saját, grafikus számításokra optimalizált célprocesszorral (GPU – Graphics Processing Unit) és speciális memóriával (video RAM) rendelkeznek. A kortárs VGA-kártyák a központi egységhez a speciálisan e célra fejlesztett, a PCI-szabványnál gyorsabb adatátvitelt biztosító AGP- vagy PCI-Express-sínen kapcsolódnak.

2.3.3.2 Nyomtató (printer)

A nyomtatók feladata információ papírra rögzítése, lényegében pontok mintázatát állítja elő. Legfontosabb tulajdonságai: a megjeleníthető színek száma és a nyomtatási sebesség (általában lap/perc-ben mérjük), továbbá a nyomtatás minőségét meghatározó nyomtatási felbontás (a képpontok száma és egymástól való távolsága), illetve a felhasználható papír mérete. Két fő csoportjuk van: egyik az érintéses nyomtató, amelynél a papírhoz érő mechanikus eszköz (nyomtatófej) festéknyomata jelenik meg a lapon (ilyen a főleg számláknál, több példányos nyomtatáskor a mai napig használt mátrixnyomtató). A nem érintéses nyomtatók főbb típusai:

- tintasugaras nyomtató: a nyomtatófej apró fűvókáin keresztül finom tintacseppeket (másodpercenként több ezer) juttat a papírra;
- lézernyomtató: egy fényérzékeny hengerre gyenge lézersugár rajzolja fel a nyomtatandó karaktereket vagy képeket, ami (elektrosztatikus töltések útján) először magára vonzza a festéket, majd a festék „átragad” a hengerrel érintkező papírra;
- hőnyomtató: az egyszerűbb típusnál speciális hőérzékeny papírt alkalmaznak, amelynek bevonata melegítéskor elszíneződik, míg a modernebb, jó minőségű fotónyomtatáshoz használt változatokban a kép megjelenítése a papírra szorított festékszalagról leolvadó festékkel történik.

2.3.3.3 Multimédia

A hangkártya akusztikus jelek előállítására képes (általában a PCI-sínre illeszthető) bővítőkártya, amely audio kimenettel és bemenetekkel is rendelkezik: így hanglejátszásra (analóg vagy digitális jel küldése hangfalaknak vagy erősítőnek) és hangfelvételre (bemenő vonalon vagy mikrofon segítségével), digitalizálásra is alkalmas. A hangkártya funkcióját betöltheti a számítógép alaplapjára integrált audiovezérlő is. A hangkártya mellett szintén multimédiás funkciót betöltő bővítőkártyák a különféle videodigitalizáló eszközök, például a tévéadások fogadására, rögzítésére használatos TV-tuner-kártyák.

2.4 Platformok – modern CPU-architektúrák

A platform fogalom a számítástechnikában többféle értelemben is használatos, de minden esetben valamilyen keretrendszert, fizikai vagy logikai strukturális modellt jelent. A hardverek világában ilyen átfogó kategória lehet egy-egy gyártó teljes termékskálája vagy csak annak egy része – így például beszélhetünk gyártók szerint Intel- vagy IBM-platformokról, vagy termékcsaládok elkülönítése céljából Itanium- vagy GeForce-platformokról (előbbi az Intel CPU-családja, utóbbi az NVidia egyes grafikus processzorainak gyűjtőneve). A platform logikai keretrendszer értelmezése főleg operációs rendszerekre jellemző, gyakran találkozunk olyan kifejezésekkel, hogy az adott programot például Windows- vagy Linux-platformra fejlesztették, esetleg többfélét is támogat – ekkor platformfüggetlennek nevezzük. Az operációs rendszerekre a 3. fejezetben térünk ki, ebben a pontban a kortárs processzorok néhány fontos jellemzőjét vizsgáljuk meg, konkrét termékeket hozva fel példaként.

Az egyik első mikroprocesszort az Intel Corporation tervezte: az 1971-ben megjelent, 4004-es sorozatszámú CPU tranzisztorból állt, és maximum 4 KB memóriával volt képes együttműködni. A 4004-es *szóhosszúsága*²⁶ – ami azoknak a biteknek a száma, amit a CPU egyidejűleg fel tud dolgozni – 4 bit volt. A mikroprocesszorok teljesítménye azóta jelentősen megnőtt, a 8 bites változatokat 1978-ban követte az immár 16 bites Intel 8086 és utódja, a 8088. Erre a processzorra épült az 1981-ben megjelent első IBM gyártmányú személyi számítógép; a 8086-os pedig az ún. x86-os processzorarchitektúra²⁷ első példánya volt. Ez az architektúra uralja a mai napig az asztali és hordozható személyi számítógépek, valamint a minigépek, munkaállomások piacát.

Az x86-platform első 32 bit szóhosszúságú processzora az Intel 386-os terméke volt, és az ekkor debütáló 32 bites architektúra egészen a 2003-ig, a 64 bites változatok megjelenéséig számos processzorgeneráció alapja volt. A piacvezető Intel mellett számos másik gyártó, az AMD (Advanced Micro Devices), a Cyrix és a VIA is belépett az x86-os piacra, de közülük egyedül az AMD-nek sikerült az Intel termékeivel versenyképes CPU-kat fejleszteni. Jelenleg is ez a két gyártó állítja elő a legtöbb PC-be szánt mikroprocesszort; a legmodernebbek az Intel a Pentium márkanévű sorozatának IV. generációja és az AMD Athlon64-család.



2.9 ábra: Modern mikroprocesszorok

A processzorok teljesítménye növelésének egyik módja az órajel, azaz a CPU belső ütemezésének növelése. A kortárs csúcsmoდეllek a 740 kHz-en üzemelő 4004-es CPU órajelének kb. ötezerszeresén üzemelnek, azonban – többek között – a további órajelnövelés fizikai korlátai (például a hőtermelésként jelentkező veszteség) a fejlesztés fő irányvonala az

²⁶ A szóhosszúságnak megfelelően alakul például a regiszterek többségének mérete, a memóriából egy időegység alatt beolvasható bitek száma és a memória címzése.

²⁷ Az elnevezés az egymást követő processzorgenerációk típusjelében – 8086, 80186, 80286, 386, 486 – alkalmazott „86” számjegyekből adódik

órajelnövelés helyett a párhuzamosítás lett, azaz a processzormagok meg többszörözése. Az új típusoknál tehát nem az órajel lesz a teljesítmény elsődleges mutatója²⁸, ezért a gyártók különféle új skálákat társítanak az egyes termékekhez (példa erre a 2006 nyarán megjelent Intel Core 2 Duo E6400 vagy az AMD Athlon 64 X2 3800+).

Az x86-os processzorokra jellemző az ún. CISC-architektúra (Complex Instruction Set Computer = bonyolult utasításkészletű számítógép). Az elnevezés magyarázata, hogy a processzorok fejlődése során a korábbi utasításkészleteket mindig újabbakkal egészítették ki. Az összetettebb utasítások viszont bonyolult mikroprogramokat igényelnek; a változó hosszúságú utasítások nehezebben optimalizálhatóak. Az 1970-es években az IBM, a Sun Microsystems és más fejlesztők olyan architektúrát dolgoztak ki, amely több, egyszerűbb utasítást alkalmaz egyetlen komplex utasítás helyett. A RISC (Reduced Instruction Set Computer, azaz csökkentett utasításkészletű számítógép) –technológia mikroprogramok helyett kötött hosszúságú utasításokat, egyszerűbb és gyorsabb dekódolást alkalmaz. A kötött utasításhossz a belső párhuzamosítás szempontjából is előnyös, a pipeline-elv is hatékonyabban alkalmazható ezeknél a processzoroknál. Elterjedt RISC-processzor például a Sun SPARC, a DEC Alpha és a Hewlett Packard PA-RISC – általában elmondható, hogy a mainframe gépek alapja a RISC-technika. Az x86-os platform alapvetően CISC-jellegű, de a Pentium-sorozattól kezdődően a két technológia együttes alkalmazása jellemző.

Az x86-platform a személyi számítógépek többségére jellemző, de említést érdemel még az Apple által gyártott Macintosh (Mac)-sorozat, amely a IBM PowerPC processzoraira, egy RISC-processzorcsaládra épül.

2.5 Összefoglalás

A fejezet során áttekintettük a számítógépek legfontosabb hardverelemeit. A számítógép működése során bemeneti adatokat dolgoz fel és alakít át kimeneti adatokká. A feldolgozást végző központi egység, a beviteli eszközök, a megjelenítő eszközök és a háttértárak a fő hardverkategóriák. A különböző funkcionális egységek közötti kapcsolatot a sínrendszer biztosítja, amelyre az egyes hardverelemek szabványos interfészekon kapcsolódnak.

A számítógép „gerince” az alaplapp, az alaplapi foglalatokban helyezzük el a processzort, az operatív memóriát és a bővítőkártyákat, és ide csatlakoztatjuk a háttértárakat. Az egyes hardverek tetszés szerint nem kombinálhatók; az egyes platformokon alkalmazott architektúra jelentősen eltérhet.

²⁸ Megjegyezzük, hogy a számítási teljesítmény mérésére használt fő mutató a MIPS (Million Instructions Per Second) = millió utasítás másodpercenként.

Ellenőrző kérdések:

1. Mit jelent a negyedik generációs számítógép fogalma?
2. Milyen egységekből épül fel a modern mikroprocesszor?
3. Vázolja fel a processzor működését!
4. Ismertesse a számítógépekben alkalmazott tárhierarchia szintjeit!
5. Jellemezze a processzor és a többi hardverelem kapcsolatrendszerét!
6. Mit jelent az alaplapi chipkészlet fogalma?
7. Milyen háttértároló-kategóriákat ismer?
8. Mit jelent a képernyő frissítési frekvenciája?
9. Sorolja fel a nyomtatók főbb típusait!
10. Foglalja össze a CISC- és RISC-technika különbségeit!

Irodalomjegyzék

Csala Péter – Csetényi Arthur – Tarlós Béla: Informatika alapjai. ComputerBooks Budapest, 2001

Csiki András – Molnár István – Orbán Anna: Informatika. Pannongraf, 2003

Elektronikus irodalom

<http://developer.intel.com/technology/memory/>

<http://www.amd.com/us-en/Processors/TechnicalResources/>

<http://www.kingston.com/tools/umg/>

3. OPERÁCIÓS RENDSZEREK

Amint azt az előző fejezetben is láthattuk, a számítógépet alkotó fontosabb hardver eszközök száma meglehetősen nagy. Ezek között vannak olyanok (ld. Neumann-elv!), amelyek nélkül nem képzelhető el működő számítógép, és vannak olyanok, amelyek csupán opcionálisak. Még az azonos eszközök tekintetében sincs azonban egység: a különböző gyártók (természetesen) más technikai és működésbeli paraméterekkel rendelkező egységeket készítenek – a felhasználók saját igényeik szerint alakíthatják ki a számítógépüket. (Egyes vélemények szerint (nyilván sok egyéb mellett) ez a fajta modularitás: az egyes alkatrészek független cserélhetősége, bővíthetősége a számítógép népszerűségének az oka.) Azonban mindezen sokszínűség dacára a számítógépek felépítésének és működésének az alapelvei (a konkrét hardver kiépítettségétől függetlenül) azonosak. Az egyik ilyen azonosság az, hogy – csakúgy, mint a hardver elemek között – a szoftverek között is van egy olyan szoftver-csoport, amelynek hiányában a számítógép nem képes működni. Ezeket a szoftvereket operációs rendszereknek nevezzük.

3.1 Alapfogalmak

Az **operációs rendszerekre** vonatkozóan többféle **definíció** is létezik:

- olyan program(rendszer), amely felügyeli és vezérli a számítógépen futó valamennyi folyamatot (az ISO szabvány szerinti megfogalmazás)
- a számítógépet alkotó hardver eszközök működését felügyelő és vezérlő program (technológiai megközelítés)
- a számítógép tevékenységét meghatározó programokat felügyelő és vezérlő szoftver (funkcionális megközelítés)
- (a számítógépes rendszerben rendelkezésre álló) erőforrásokat elosztó szuperfolyamat (folyamat-centrikus szemlélet)
- olyan program, amely kapcsolatot teremt (és tart fent) a számítógépet alkotó technikai-technológiai (hardver) elemek és a (számítógéppel tevékenységet végző) felhasználó között (felhasználói szemléletű definíció).

A fentiekkel kapcsolatban két dolgot kell kiemelnünk. Egyrészt az egyes megfogalmazások között (első olvasásra legalábbis) nincs sok különbség (sőt, levonhatjuk azt a tanulságot, hogy az operációs rendszer elsősorban „felügyel és vezérel”, hiszen ez szinte az összes definícióban szerepel), azonban minden megközelítés az operációs rendszer működésének más és más jellemzőjét emeli ki. Másrészt a fenti definíciók alapján az is jól látható, hogy az operációs rendszernek (a felügyeleti feladatokon túl) számos egyéb feladat ellátására is képesnek kell lennie, és hogy ezek a feladatok vonatkozhatnak akár a számítógép hardver elemeivel kapcsolatos műveletekre, akár a felhasználó igényinek kiszolgálására – következésképpen az operációs rendszernek egyfajta közvetítő szerepet is be kell töltenie (a felhasználó és a hardver között).

Az operációs rendszerek legfontosabb **alapfeladatai** a következők:

- **kapcsolattartás** a felhasználóval (felhasználói felület biztosítása, állományszervezés) – ld. később!
- a számítógépet alkotó **hardvereszközök** különbözőségének „elfedése”, **kezelésének egységesítése**. Ezt nevezik „*virtuális gép koncepció*”-nak, lényege az, hogy az operációs rendszer úgy viselkedik, hogy a felhasználó úgy érzékeli, mintha az összes számítógép egyforma lenne – azaz nem kell ismernie az általa használt hardver-eszközök működésének specifikumait.

- **erőforrás-gazdálkodás.** Természetes, hogy egy számítógépes rendszerben a rendelkezésre álló erőforrások (pl. a memória szabad rekeszei, a processzor „ideje”, a perifériák állapota, stb.) folyamatos ellenőrzést igényel, hiszen ha pl. egyszerre két forrásból akarják ugyanazt az eszközt használni (mondjuk nyomtatási kérés érkezik a nyomtatóhoz), az hibás működéshez vezethet. Az operációs rendszer feladata az elérhető erőforrások nyilvántartása, illetve a hibamentes működéshez szükséges elosztása.
- **állapot-felügyelet** (rendszernaplók készítése, a felhasználó tájékoztatása a hibákról, lehetőség szerint automatikus hibakezelés, stb.).
- **programkészítés támogatása.** (Ezen feladat alapvető jellege vitatható. Ha azonban elfogadjuk azt a működési modellt, amely szerint az operációs rendszer az egyetlen „ab ovo” létező program egy számítógépen, akkor nyilvánvaló, hogy valamilyen módon lehetőséget kell biztosítani további programok létrehozására is. Ez lehet (mint ahogy a gyakorlatban az is) egy programfejlesztő környezet, de (szükséges esetben) lehet az operációs rendszer egy vagy több (önálló vagy sem – pl. a forráskód elkészítése valószínűleg csak egy egyszerű szövegszerkesztőt igényel) szolgáltatása is.)
- **hálózati szolgáltatások.** Szintén vitatható, hogy önálló feladatnak tekinthető-e, de a mai hálózat-centrikus informatikai modellben egy olyan komplex feladat, mint egy hálózati kommunikációs folyamat kiszolgálása, nem igazából illeszthető be egyik előző csoportba sem.

Összefoglalva tehát azt mondhatjuk, hogy az **operációs rendszer egy olyan programrendszer** (nem egyetlen program, hiszen szerkezetileg több önálló, egy vagy több alapeladat kiszolgálásában közreműködő komponensből tevődik össze), amely **felügyeli** a számítógép **működését** és **biztosítja** a szükséges **kommunikációt** a **felhasználó és a hardver eszközök között**.

Az operációs rendszert alkotó szolgáltatások közül azokat, amelyek a hardvereszközök kezelését végzik, az operációs rendszer „**kernel**”-jének (magyarul leginkább „rendszermag”-nak fordítják), a felhasználói felülettel kapcsolatos szolgáltatásokat pedig az operációs rendszer „**shell**”-jének (magyarul „burok” vagy „héj”) nevezzük.

(A kernel és a shell különválasztásának létezik egy másik értelmezése is: ha nem a szolgáltatás igénylője (hardver vagy felhasználó), hanem a szolgáltatás és a rendszer működése közti összefüggés alapján csoportosítjuk a tevékenységeket. Ez esetben a rendszer működésének biztosítása szempontjából kritikus komponensek alkotják a kernelt, a „kevésbé fontos” (pl. meghibásodás esetén a rendszer működését nem (vagy csak kis mértékben) befolyásoló) komponensek pedig a shellt. Nyilvánvaló, hogy a kétfajta értelmezésben vannak átfedések, hiszen a számítógép alapvető hardver eszközeinek a kezelése (memória, processzor) mindkét esetben a kernel feladata, a felhasználói tevékenységek mindkét esetben a shellre tartoznak – a különbség leginkább a két réteg határán megjelenő szolgáltatások besorolásában van: míg a periféria-kezelő (pl. nyomtatás-vezérlő) komponensek az első esetben kernel, a másodikban shell szinten helyezkednek el.)

3.2 Az operációs rendszerek csoportosítása

Az operációs rendszerek feladatainál láthattuk, hogy számos feladata lehet ezeknek a programoknak, aligha meglepő hát, hogy operációs rendszerből nem csak egy létezik. (Igazság szerint a különböző operációs rendszerek számát valószínűleg csak becsülni lehet: az ismert (jelenleg is használt, vagy mára elavult, vagy csak fejlesztői célokkal készült, stb.) operációs rendszerek száma is legalább százaz nagyságrendű.) Az operációs rendszerek

csoportosítására számos szempont képzelhető el. A következőkben néhány fontosabb szempontot mutatunk be – az adott szempontnak megfelelő konkrét operációs rendszer feltüntetésével együtt. Természetesen ez a csoportosítás nem feltétlen teljes és (mint ahogy az néhány rendszer esetében látható is) az egyes rendszerek besorolása egy-egy csoportba (az esetek többségében) nem kizárólagos.

Operációs rendszer-kategóriák (példák)

- 1) Felhasználói felület szerint:
 - a) karakteres (UNIX, DOS)
 - b) grafikus (Apple Mac OS)
- 2) Felhasználók száma szerint
 - a) egy-felhasználós (BeOS)
 - b) több-felhasználós (Microsoft Windows XP)
 - i) hálózati (Novell Netware)
- 3) Folyamatkezelés szerint
 - a) köteget (Microsoft MS DOS)
 - b) multiprogramozott
 - i) valós idejű (BeOS, QNX)
 - ii) időosztásos (Multics, UNIX)
- 4) Hardver-architektúrák szerint
 - a) számítógép-kategóriák
 - i) mainframe (SUN Solaris)
 - ii) szerver (Microsoft Windows 2000, Linux)
 - iii) személyi számítógép, munkaállomás (IBM OS/2 Warp)
 - iv) mikroszámítógép (Commodore 64)
 - v) kézi számítógép (PalmOS)
 - b) processzor-architektúrák szerint
 - i) CISC alapú (Linux, Microsoft Windows)
 - ii) RISC alapú (Hewlett-Packard HP-UX)
 - c) sínrendszer alapján
 - i) 16 bites (IBM PC DOS)
 - ii) 32 bites (Microsoft Windows XP)
 - iii) 64 bites (Macintosh OS X)
- 5) Jogállás szerint
 - a) szerzői jogvédelem alá tartozó (SCO OpenServer)
 - b) nyílt forráskódú (Linux)
- 6) „Történelmi” kategóriák
 - a) korai operációs rendszerek
 - b) UNIX-alapú rendszerek
 - i) UNIX verziók, POSIX-kompatibilis rendszerek
 - ii) Linux disztribúciók
 - c) Windows rendszerek

Az operációs rendszerek csoportosításának legtriviálisabb, ugyanakkor legkevésbé egyértelmű szempontokat meghatározó módszere lehet a **felhasználói preferenciák** szerint történő csoportosítás. Ide sorolhatók mindazok a szempontok, amelyek szerint a felhasználók döntenek az általuk használt operációs rendszer mellett: felület, kezelhetőség, megbízhatóság, gazdaságosság, támogatás, gyártó, stb. Ezen kategóriák közül csak az operációs rendszerek **felhasználói felület** szerinti csoportosítását ismertetjük, eszerint megkülönböztetünk karakteres és grafikus felületű rendszereket.

Karakteres felületű (*CLI: Command Line Interface*) rendszerek esetében a parancskiadás eszköze a billentyűzet, a számítógép számára az elvégzendő utasításokat egy általában zárt nyelvi struktúra (utasítás-készlet) kifejezéseinek a begépelésével lehet megadni.

A **grafikus felületű** (*GUI: Graphics User Interface*) rendszerek esetében a felhasználói műveletek kiválasztása valamilyen előre definiált grafikus szimbólumkészlet és egy pozícionáló eszköz együttes használatával lehetséges. Az előbbi csoportba tartozó operációs rendszerek szokás még parancsvezérelt, az utóbbiakat pedig eseményvezérelt rendszereknek is nevezni – ez esetben nem a felhasználói felület külalakja, hanem a felhasználói munkavégzést támogató eszközrendszer képezi a csoportosítás alapját. (A parancsvezérelt elnevezés egyértelmű, az eseményvezérelt elnevezés magyarázata pedig az, hogy a grafikus felületű rendszerekben a grafikus környezet változásait (a kijelölő eszköz – pl. az egér – elmozdítását vagy aktiválását (kattintás), vagy egy grafikus szimbólum kiválasztását) eseményeknek nevezik.)

Lényegesen egzaktabb kategorizálásra ad lehetőséget az operációs rendszert futtató számítógép **hardver architektúrája** alapján történő csoportosítás. Ilyen értelemben beszélhetünk a különböző *számítógép-kategóriák* (mainframe kategóriájú számítógépek, szerver számítógépek, személyi számítógépek (munkaállomások), kézi számítógépek, stb.) jellemző operációs rendszereiről, vagy a számítógép *hardver felépítésének* támogatásán alapuló kategorizálásról (processzor-architektúra szerint (CISC vagy RISC alapú rendszerek) egy- vagy többprocesszoros rendszerek, 16-32-64 bites rendszerek, stb.).

Az operációs rendszerek csoportosításának következő eszköze lehet az egyes rendszerek **működésbeli sajátossága** is. Ilyen értelemben vizsgálhatjuk az operációs rendszerben futó programok számát és együttműködésének módját, a rendszer által kiszolgálható felhasználók számát és az erőforrások megosztásának lehetőségeit, stb.

Az **egy időben futtatható alkalmazások száma** szerint megkülönböztetünk *egyfolyamatos (mono-módú vagy kötegelt)* és *többfolyamatos (multitasking)* rendszereket – a hangsúly ebben az esetben az egyidejűségen van: a kötegelt rendszerekben egy program futásának ideje alatt újabb program indítására nincs lehetőség, míg a multiprogramozott rendszerekben egyszerre (egymással párhuzamosan) több program is futtatható. (Azonban nem szabad elfeledkezni arról, hogy a Neumann-architektúrájú számítógépeken a műveletvégzés minden esetben szigorúan soros, így a multiprogramozott operációs rendszerek esetében is csak virtuális párhuzamosságról beszélhetünk – azaz a felhasználó számára úgy érzékelhető, mintha programjai egyszerre működnének, valójában azonban csak az egyes programrészek végrehajtása közötti gyors váltásokról van szó.)

A **felhasználók száma szerint** egy- és több-felhasználós rendszereket különböztethetünk meg aszerint, hogy az operációs rendszer rendelkezik-e olyan azonosítási szolgáltatással, amely lehetővé teszi a számítógéppel dolgozó felhasználók (és munkájuk) megkülönböztetését. Az *egy-felhasználós („single user”)* rendszerekben semmilyen azonosítási rendszer nem működik, így nem eldönthető, hogy az adott számítógép egyes erőforrásaihoz ki és milyen jogokkal férhet hozzá. Egy *több-felhasználós („multi-user”)* operációs rendszer esetében minden felhasználói tevékenység kiegészül az elvégző felhasználó azonosítójával és ilyen módon (megfelelő biztonsági rendszer kialakításával) az ugyanazon a számítógépen dolgozó felhasználók tevékenysége teljesen elkülöníthető egymástól. (A több-felhasználós rendszerek esetében célszerű további különbséget tenni aszerint, hogy a különböző felhasználók egyidejűleg kezelhetik-e a számítógépet – amennyiben igen, akkor *hálózati operációs rendszerről (NOS: Networking Operating System)* beszélünk.)

Az operációs rendszerek működésbeli jellemzői alapján történő csoportosításának következő szempontja az **operációs rendszer válaszüteme** lehet. A több-folyamatos rendszereknél említettük, hogy csupán virtuális párhuzamosságról van szó, amely mögött alapvetően kétféle technikai megoldás képzelhető el. Az *időosztásos elven* működő („time sharing”) operációs rendszerek esetében a rendszerben futó folyamatok között szétosztásra kerül a rendszer teljes működési ideje: minden folyamat kap egy „időszelvet”, ennyi ideig használhatja a processzort („futhat”), majd át kell adnia a következő folyamat számára, és ez a tevékenység-sorozat mindaddig ismétlődik, míg minden program be nem fejeződik. A *valósütemű (RTOS: Real Time OS)* rendszerek esetében az egyes programok azonnal végrehajthatók, így a futásuk eredménye („a válasz”) is azonnal (vagy legalábbis emberi léptékkal mérve reális időn – legfeljebb néhány másodperc – belül) rendelkezésre áll. (Természetesen egy valós ütemű rendszerben sincs lehetőség két tevékenység egyszerre történő végrehajtására, így amennyiben két program akar ugyanazon időpillanatban elindulni, az egyiknek meg kell várni a másik befejeződését – ebből viszont az is következik, hogy az ilyen rendszerek hatékony működésének feltétele a kis méretű (és ebből következően gyorsan végrehajtható) programok alkalmazása. A valós ütemű rendszereket jellemzően olyan helyzetekben alkalmazzuk (többnyire felügyeleti szerepben) ahol az idő szerepe kritikus, mint pl. egészségügy, atomenergia, stb.)

3.3 Az operációs rendszerek alapfeladatai

Dacára azonban a fentebb bemutatott sokszínűségnek, az operációs rendszerek eltérő működési módjuk és különböző megjelenésük ellenére is lényegében ugyanazokat a feladatokat látják el. Részleteiben nem mutatjuk be az operációs rendszerek valamennyi szolgáltatásának megvalósítási módszereit, a következőkben csak a felhasználói szempontból fontos alapfeladatok: a felhasználói felület kezelésének alapelvei és az állományszervezés bemutatására szorítkozunk.

Ahogy azt az operációs rendszerek csoportosításakor már láttuk, az operációs rendszer felülete (megjelenési módja, azaz amit a felhasználó „lát” az operációs rendszerből) alapvetően kétféle lehet: karakteres vagy grafikus.

3.3.1 Karakteres felület kezelése

A karakteres felületű operációs rendszer esetében a shell legfontosabb feladata a felhasználó által begépett (karakteres felületről lévén szó sokkal több lehetőség nincs az utasítás-kiadásra...) utasítás értelmezése. Az operációs rendszer által ismert (értelmezhető) parancsok együttese az **utasításkészlet**. Az utasításkészlet a parancsok felsorolásán túl azok használatával, megadásának módjával (esetleg feltételével) kapcsolatban is tartalmaz(hat) előírásokat, ezek a szabályok határozzák meg az adott operációs rendszer **utasítás-szerkezetét**. Az utasítás-szerkezet határozza meg, hogy az utasítás-készlet egyes elemeihez (a konkrét parancsokhoz) hány és milyen módosító tartozhat, milyen módon kell az egyes részeket elhatárolni, mekkora lehet egy utasítás-sor maximális hossza, stb. Az utasítás-szerkezet általában egy utasításon belül a következő három funkcionális egység megkülönböztetését teszi lehetővé: kulcsszó – paraméterek – kapcsolók.

A **kulcsszó** az utasításkészlet eleme: olyan karaktersorozat, amely azonosítja az elvégzendő tevékenységet. Az egyes feladatokat azonban (természetesen) nem minden esetben kell ugyanazokkal az adatokkal és ugyanolyan módon elvégezni. A **paraméterek** (a parancs aktuális végrehajtása során) a műveletben érintett adatok körét jelölik ki, a **kapcsolók** pedig a tevékenység végrehajtásának a körülményeit befolyásolják.

Ahhoz, hogy ezt megértsük, nézzük meg a következő (képzeltbeli!) parancsszerkezetet:

PRINT levél –NOHEADER

(A példa szándékosan használ angol kifejezéseket: a parancsvezérelt rendszerek általában nem rendelkeznek nemzeti (nyelvi) támogatással...)

A fenti utasításban a PRINT a nyomtatási művelet parancsa, a levél a nyomtatandó adatállomány azonosítója (paraméter: egy másik állomány nyomtatásakor ezen a helyen egy másik azonosító szerepel), a –NOHEADER pedig azt jelzi, hogy a nyomtatási művelet során a fejlécben szereplő információk megjelenítését nem kérjük (kapcsoló: ha a következő nyomtatási műveletben kérünk fejléct, akkor nem szerepeltetjük).

A fenti parancsszerkezet látszólagos egyszerűsége ellenére is számos kérdést vet fel:

- melyek a kötelező és melyek az opcionális elemek?
- milyen sorrendben kell az egyes elemeket megadni?
- hogyan lehet eldönteni, hogy az utasítás módosítója paraméter vagy kapcsoló?

Általános érvényű válasz ugyan nincs ezekre a kérdésekre, de vannak olyan irányelvek, amelyek az operációs rendszerek többségében (kvázi-szabványként) megtalálhatók:

- az utasításszerkezetnek minden esetben kulcsszóval kell kezdődnie.
- az utasítás végét az operációs rendszer számára egy kitüntetett szimbólum (általában billentyű(kombináció): pl. Enter) jelzi.
- a paraméterek/kapcsolók száma utasításonként eltérő lehet (megengedve a paraméterrel/kapcsolóval nem rendelkező utasítást is).
- amennyiben a paraméterek és kapcsolók sorrendje nem kötött, akkor az adott elem előtt álló szimbóluma dönti el a szerepét (pl. a kapcsolók egy – jellel kezdődnek).
- a több paraméterrel/kapcsolóval rendelkező utasítások esetében az egyes elemeket legalább egy elhatároló szimbólum (általában szóköz) választja el egymástól.

A parancsok kiadása tehát az utasításkészlet és az utasítás-szerkezet ismeretében (viszonylag) egyszerű feladat – ez azonban nem jelenti azt, hogy az eredmény garantáltan hibátlan is. Az operációs rendszer az utasítás lezárása („utasítás-kiadás”) után szintaktikailag ellenőrzi a kapott parancsot (ld. fenti szempontrendszer), majd megpróbálja a megadott paraméterek és kapcsolók értékei alapján végrehajtani. A végrehajtás eredményével kapcsolatos információk kezelése szempontjából alapvetően két megközelítés fordul elő a különböző rendszerekben, a pozitív és a negatív nyugtázás. **Pozitív nyugtázás** alatt azt értjük, hogy az operációs rendszer a sikeresen elvégzett parancs végrehajtásáról ad valamiféle visszajelzést a felhasználónak (általában az elvégzés sikerességének igazolása mellett – amennyiben értelmezhető – az eredménnyel kapcsolatos információt is megjeleníti). A **negatív nyugtázásos** rendszerben ezzel szemben csak a hibás parancsok váltanak ki reakciót az operációs rendszer részéről – a sikeresen végrehajtott parancsok sikerességét a hibaüzenet hiánya(!), illetve a készenléti jel megjelenése jelzi.

3.3.2 Ablakozó (esemény-vezérelt) rendszerek sajátosságai

A karakteres felület hatékony, de körülményes használatát előbb kiegészítendő, majd kiváltandó jelentek meg az első grafikus felületű rendszerek (**GUI: Graphic User Interface**), amelyek kényelmes (egyszerű) használhatóságuk miatt hamarosan az elterjedt informatikai eszközök elsődleges felhasználói felületévé léptek elő. A grafikus felületű rendszerek működésének hátterében ugyanazok a tevékenységek zajlanak, mint a karakteres felület esetén, csak éppen az eszközrendszer különbözik.

Habár a grafikus felületű rendszerek felületükben nem feltétlen, de működésükben általában mutatnak rokonságot. A szabványos komponensek angol nevéből képzett mozaikszóval ezeket a rendszereket szokás **WIMP** elvű rendszereknek is nevezni: **ablak** (*window*), **ikon** (*icon*), **menü** (*menu*) és **pozícionáló eszköz** (*pointing device*) **együtteséről** van szó.

Ablak: minden folyamathoz hozzárendelhető a képernyő egy (alapértelmezés szerint téglalap alakú) területe: ez a terület a program futási környezete (a grafikus felület szempontjából).

Multiprogramozott rendszer esetében egy időben tetszőleges számú ablak lehet a képernyőn, de a felhasználó egy időben csak eggyel kommunikálhat (**aktív ablak**, általában az utoljára kiválasztott). Az ablakok alapértelmezés szerint a képernyő tetszőleges tartományán elhelyezkedhetnek (megengedve a két végletet: a képernyő teljes terjedelmét, illetve a képernyő-területet nem foglaló állapotot és a kettő közti összes lehetséges átmenetet), akár egymást átfedő módon is. Az ablakok alapvetően három csoportba sorolhatók:

- *csoportablak*: azonos típusú objektumok (általában ikonok, ld. később) rendezett megjelenítésére szolgál;
- *alkalmazás-ablak*: az adott alkalmazás (program) munkaterülete,
- *párbeszéd-ablak*: a felhasználói interakció eszköze.

Ikon: kis méretű grafikus ábra, amely egy objektum vagy tevékenység reprezentálására szolgál. Az ikonok az általuk megjelenített objektum tartalma és viselkedése szerint lehetnek:

- program(indító) vagy *parancsikonok* (tevékenységet reprezentálnak),
- *állományszervezési ikonok* (mappaikonok, állomány-ikonok: a tárolási rendszer különböző szintjeit ábrázolják),
- *hivatkozás-ikonok* (tényleges (fizikai) elhelyezkedéstől független (logikai) csoportosítást tesznek lehetővé) és
- *alkalmazás-ikonok* (a képernyőn meg nem jelenő ablakokat jelképezik).

Menü: (az operációs rendszer vagy egy adott program) tevékenységek szöveges megjelenésű – de grafikus viselkedési jellemzőkkel is rendelkező –, strukturált rendszere.

A menüszerkezetben menüpontok (csoportok) és menüparancsok (utasítások) szerepelnek. A „grafikus viselkedés” azt jelenti, hogy ezek a szöveges objektumok (az eseményvezérlő működésének megfelelően) képesek megváltoztatni a megjelenési módjukat (a leggyakrabban alkalmazott ilyen változtatás a parancs kiválaszthatóságának színnel történő jelzése).

A menüpontok között vannak szabványos (minden alkalmazásra jellemző, többé-kevésbé azonos menüparancsokat tartalmazó) elemek, ilyenek a Fájl menü („File”, jellemzően az állománykezeléssel, nyomtatással kapcsolatos parancsokkal), a Szerkesztés menü („Edit”, kijelölés, vágólap, keresés parancsokkal) és a Súlyó.

A **pozícionáló eszközök** tekintetében általánosságokat meglehetősen nehéz megfogalmazni, hiszen ezen eszközök családja meglehetősen népes: ide tartoznak a különböző egerek, gördítő-csúszó kijelölők (touch-ball, touch-pad), érintő (marker) -eszközök (fényceruza, érintőképernyő), stb.

Az ablakozó rendszereket gyakran emlegetik **esemény-vezéreltként** is. A két fogalom nem szükségszerűen fedi egymást, de tény, hogy grafikus felületű rendszerek működése általában esemény-vezérelt. Alapelve, hogy az operációs rendszer a felhasználó tevékenységét (és a rendszer működésében megjelenő állapotváltozásokat is) folyamatosan figyeli – ez

természetesen nem csak a GUI jellemzője, de ez esetben a szokásos műveletek kiegészítéséről van szó. Amennyiben a rendszer környezetében valamilyen változás áll be (a fenti tevékenységek valamelyikének hatására) – ezt nevezzük eseménynek –, akkor erről a grafikus felület értesítést küld az operációs rendszernek – ez az üzenet –, az pedig az üzenet információi alapján meghatározza az elvégzendő műveletet. (Tulajdonképpen a karakteres felület parancsértelmező komponensének a megfelelőjéről van szó.) Az elv – látszólag – egyszerűsége ellenére is tökéletes, ez azonban nem feltétlen igaz: a grafikus felületű rendszerek (nem kötelező jelleggel, de leggyakrabban) multiprogramozottak is – ekkor viszont felmerül a kérdés, hogy egy esemény megjelenésekor hogyan azonosítható a forrása?

Amennyiben az esemény egy pozicionáló eszköztől származik (pl. az egér egyik gombjának lenyomása), akkor az eszköz pozíciójának ismeretében meghatározható a kérdéses program. Ha az esemény billentyűzetről származik, akkor az operációs rendszer alapértelmez: a futó programok között mindig van egy aktuális (általában a legutoljára kiválasztott), és ha az eseményből a forrás nem meghatározható, akkor az az esemény az aktuális programra vonatkozik.

Végül két megjegyzés:

- o egyrészt legyen bármilyen jól szervezett egy grafikus felhasználói felület, a felhasználói interakció nem feltétlen merül ki az előre definiált menük és ikonok használatában, ezért a GUI-k ablakkezelői az egyes ablakokat további (a felhasználó döntési lehetőségeit kiterjesztő vagy leegyszerűsítő) ún. vezérlőelemekkel ruházzák fel (mint pl. beviteli mezők, parancsgombok, listák, kiválasztó vezérlők: rádiógombok, jelölőnégyzet), stb.) – ezek az eszközök a szabvány kvázi kiegészítésének tekinthetők;
- o másrészt, hogy egy grafikus felhasználói felület sem nélkülözheti teljes egészében a karakteres vezérlés lehetőségét: az egyes (általában gyakran használt) tevékenységekhez billentyűkódok épp úgy hozzárendelhetők (és általában ilyen hozzárendelések léteznek is, sőt bizonyos esetekben a felhasználó igényeinek megfelelően módosíthatók, kiegészíthetők), mint ikonok, illetve a WIMP komponensek közül legalább a menü kezelhető és a pozicionáló eszköz helyettesíthető (szimulálható) a billentyűzetről.

3.3.3 Állományszervezés

A shell-szolgáltatások közül a másik (a felület-kezeléshez hasonlóan alapvető) fontos tevékenységi kör a felhasználói információk tárolásával kapcsolatos tevékenységek támogatása. Az előző fejezetben megismerkedtünk a különböző elven működő háttértárak (mágnestape, mágnesszalag, CD/DVD, pen-drive) adattárolási rendszerének fizikai jellemzőivel, a tárolási elvekkel és módszerekkel. Könnyű belátni, hogy a winchester számára „levél.doc” fájl nem létezik – a tárolás (csakúgy, mint a memória esetében) bájtok alkalmas sorozata, valamilyen sorszámmal azonosított rendszerben. Akkor viszont hogy találja meg a számítógép, ha kiadom az ezen állomány megnyitására vonatkozó parancsot? Természetesen az operációs rendszer segítségével.

Az operációs rendszer **állományszervezéssel kapcsolatos komponensének alapfeladata**, hogy a számítógép (illetve az aktuálisan használt háttértár) tárolási rendszere – nevezzük ezt a továbbiakban az adattárolás **fizikai szintjének** – és a felhasználói igényeknek megfelelő azonosítási rendszer – ez lesz az adattárolás **logikai szintje** – közötti **megfeleltethetőséget** biztosítsa.

Ennek eszköze az állományszervezés logikai egységekbe történő szervezése. Az állománykezelő rendszer logikai alapegysége az állomány. **Állomány** (fájl): **valamely**

háttértárolón elhelyezkedő adatok egyedi azonosítóval rendelkező együttese, az adattárolás egysége a felhasználó szempontjából. A két megfogalmazás az állomány fogalmának két (egyenértékű) megközelítését mutatja: az első, ahogyan a számítógép, a második ahogyan a felhasználó számára megjelennek az adatok tárolásával kapcsolatos tevékenységek.

Állományszervezés: az operációs rendszer azon tevékenységeinek összessége, amely a felhasználói információk háttértárolón történő elhelyezésével, azonosításával és visszakeresésével kapcsolatos. Az operációs rendszernek azt a komponensét, amely ezeket a tevékenységeket végzi, szokás **állománykezelőnek** is nevezni (*NEM összetévesztendő a hasonló nevű segédprogramokkal!*).

A felhasználó által megadott azonosító alapján az állománykezelő meghatározza a keresett információ fizikai helyét, és (szükség esetén az adott háttértár kezelésére szolgáló kiegészítő program – az **eszközvezérlő** - segítségével) utasítja a háttértár író/olvasó mechanizmusát a megfelelő művelet elvégzésére. Tekintve, hogy a számítógép és a háttértárak műveletvégzési sebessége jelentősen eltér, a gyakorlatban az állománykezelő és a hardver eszköz közé egy gyors átmeneti tárolót (**puffer**, „buffer”, „cache”) is beépítenek.

Azt, hogy az egyes operációs rendszerek mit tekintenek „egydi azonosítónak”, nem rögzítik szabványok, de általánosan elmondható, hogy a felhasználó valamilyen alfanumerikus karaktersorozattal hivatkozhat a tárolt adatokra – ez az állomány **neve**.

A névhasználat szabályai (amelyek szintén operációs rendszerenként változnak) határozzák meg, hogy az állományok neve milyen hosszú lehet (hány karakterből állhat), milyen szimbólumokat tartalmazhat (általában azok a karakterek, amelyekhez az operációs rendszer valamilyen egyéb tevékenységet rendel hozzá, tiltottak), adott esetben a név felépítésére is vonatkozhatnak megkötések.

Az operációs rendszertől függ az is, hogy milyen egyéb adatokat tart nyilván az egyes állományokról. A legáltalánosabban használt **tulajdonságok** a következők:

1. **típus:** mivel a tárolt információ elemzése nem az operációs rendszer feladata, de az egyes állományok tartalma alapvetően meghatározza a vele (rajta) elvégezhető műveletek körét, az operációs rendszerek általában az állomány nevét kiegészítik egy azonosítóval, ami segít annak meghatározásában, hogy milyen műveletek értelmezettek az adott adatcsoporthal. A DOS alapú operációs rendszerekben örökségeként szokás ezt a típusazonosítót **kiterjesztésnek** is nevezni.
2. **időbélyegek:** az állományok használatakor más és más időpillanatokban különféle tevékenységeket végezhetünk az adatokon, amelyek feljegyzése részben adminisztratív, részben biztonsági célokat szolgál. Általánosan használt időbélyeg az állomány **keletkezésének időpontja** (amikor az adatcsoporthoz rögzítésre került a háttértárolón), de egyes operációs rendszerek rögzítik az állománnyal végzett műveletek (**utolsó megnyitás**, **utolsó módosítás**) időpontját is.
3. **méret:** az állományban tárolt információ mennyisége, általában bájtban kifejezve. Fontos tudni, hogy az állomány tartalma és a tárolásához szükséges hely nagysága általában nem azonos! (*Kérdés: miért?*)
4. **jellemzők** (attribútumok): operációs rendszerenként változó kiegészítő információk (általában 1 biten tárolt értékek), amelyek az operációs rendszer számára hordoznak járulékos információt az egyes állományokról.
5. **jogosultságok:** amennyiben az operációs rendszer támogatja, megadható (korlátozható) az egyes állományokon végezhető műveletek (pl. módosítás, törlés,

megnyitás, tulajdonságok megváltoztatása, stb.) köre. Elsősorban a többfelhasználós rendszerekben alkalmazott tulajdonság, amely a felhasználók azonosításának képességével együttesen szabályozott állománykezelést tesz lehetővé.

Példák:

DOS: az állomány neve 1-8 karakter hosszú lehet, amihez „.” (pont) után kapcsolódik az 0-3 karakter hosszú kiterjesztés. (A kiterjesztést az operációs rendszer csak a számára értelmezhető utasításokat tartalmazó, ún. parancsállományok megkülönböztetésére használja: SYS, COM, EXE, BAT). Mind a név, mind a kiterjesztés egyaránt tartalmazhat betűket, számokat és az ASCII tábla első 127 karaktere közül még néhány szimbólumot, a szóköz és \ * ? + | , : ; / < = > karakterek kivételével (ide tartozik a . is: csak a név és a kiterjesztés elválasztására szolgál, nem ismétlődhet). Az időbélyegek közül csak a létrehozás dátumát (és időpontját) használja, az egyes állományok R (csak olvasható), H (rejtett – az operációs rendszer parancsai nem vonatkoznak rá), S (rendszerállomány), A (tartalma – az utolsó archiválás óta – megváltozott) jellemzőkkel rendelkezhetnek, jogosultságokat nem kezel.

Windows: a Windows 95/98 előtti verziókban a névhasználati szabályok a megegyeztek a DOS-nál megszokottakkal. A Windows 95 vezette be az ún. **hosszú fájlnev** fogalmát, amelynek értelmében az állománynévben nem (kötelezően) válik külön a név és a típusazonosító, a kettő együttes hossza 1-254 karakter lehet, kis- és nagybetű egyaránt használhatók, tiltott karakterek a / \ : * ? ” ’ < > |. (Megjegyzés: a DOS-szal való kompatibilitás fenntartása végett minden állományhoz létezik egy ún. **rövid név** is, amely megfelel a DOS névhasználati szabályainak, valamint a kis-és nagybetűk megkülönböztetése csak látszólagos – azaz a Windowsban ugyanazon a helyen nem hozható létre egy alma.txt és egy ALMA.txt nevű állomány.) Az időbélyegek között a létrehozás mellett szerepel az utolsó módosítás és az utolsó hozzáférés adatai is. A hálózati Windows változatok (NT, 2000, XP) kezelik az állományszintű jogosultságokat is.

UNIX: bármilyen szimbólum szerepelhet az állomány nevében, a kis- és nagybetűs elnevezések között van különbség, a név maximális hossza 255 karakter, a típus nem a név része (igazából a „kiterjesztés”-nek megfelelő fogalom nem is létezik a UNIX alatt, a „.” szimbólum a névben csak akkor kap speciális szerepet, ha ez a név első karaktere), nincsenek (legalábbis az előzőkben bemutatott operációs rendszerekben értelmezett) jellemzők (viszont ide sorolható az előző operációs rendszerek által nem ismert „tulajdonos” érték), több szinten kezeli a jogosultságokat.

A felhasználói munkavégzés során új állományokat hozunk létre, létező állományokkal végzünk műveleteket vagy megszüntetjük a feleslegessé vált állományokat. Az állományok életciklusában (létrehozás – felhasználás: olvasás, módosítás, futtatás, stb. – megszűnés) a keletkezés lényegesen gyakoribb tevékenység, mint a megszűnés, így az egyes háttértárolókon az állományok száma általában nő – ez viszont felveti a tárolás és a visszakeresés néhány problémáját. Például: az egyértelmű azonosítás kényszere végett minden állománynak egyedi nevet kell adni, márpedig nagyszámú állomány esetén nem könnyű megjegyezni, mely neveket használtuk már fel. Másrészt: egy sok állományt tartalmazó listából a keresett állomány kiválasztása lényegesen bonyolultabb feladat. Célszerű tehát az állományok tárolását valamilyen rendszerbe foglalni.

A háttértáron az adatok rendszerezetten helyezkednek el – ez azonban az operációs rendszer és a háttértár tárolási szerkezetének megfelelő rendszer, ami nem követi és nem tükrözi a az egyes állományok között a felhasználó számára jelentkező összefüggéseket. Az igény tehát az, hogy az adatok a tárolási szerkezettől független logika szerint legyenek visszakereshetőek: a felhasználó számára összetartozó információk a felhasználó számára

(látszólag) azonos helyen legyenek elérhetők. A helyzet tehát ugyanaz, mint az állományoknál: ugyanaz a fogalom mást jelent és másként jelenik meg a számítógép és a felhasználó számára.

Könyvtár (directory):

1. az operációs rendszer által használt adminisztratív célokat szolgáló állomány, amely a felhasználói információt tartalmazó állományok logikai csoportosítását a fizikai elhelyezkedésük tárolásával teszi lehetővé.
2. a (felhasználói) állomány elhelyezkedésének logikai struktúráját tükröző bejegyzés.

A könyvtárak tehát tulajdonképpen állományok (ami természetes is, hiszen az operációs rendszer számára minden, a háttértáron elhelyezkedő információ állomány), amelyek azzal a speciális tulajdonsággal rendelkeznek, hogy tartalmuk nem a felhasználó, hanem az operációs rendszer számára hordoz értelmezhető információt: azt, hogy azok az állományok, amelyek a felhasználó számára azonos helyen levőnek **látszanak**, hol helyezkednek el **ténylegesen** a háttértárolón.

A lehetséges tárolási szerkezetek közül az operációs rendszerek a **hierarchikus fa struktúrát** használják legelterjedtebben. Ez a struktúra egy körmentes, irányított gráffal írható le, amelynek pontosan egy kiinduló csúcspontja (**gyökér**) van és minden más pontjába pontosan egy út vezet a gyökértől. Az egyes csúcsokból tetszés szerinti számú (irányított) él vezet más csúcsokhoz, azt a csúcsot, ahonnan egy él kiindul, **szülőnek**, ahová vezet, **gyereknek** (**alkönyvtár**) nevezzük. A gyökérnek nincs szülője és minden gyereknek pontosan egy szülője van..

A könyvtárak névhasználatára általában az állományokkal azonos szabályok vonatkoznak (*kivéve a gyökérkönyvtár: a gyökérkönyvtár nevét nem a felhasználó, hanem az operációs rendszer határozza meg*), típusuk és méretük általában nincs, az állományoktól egy attribútum különbözteti meg őket, a jogosultsági tulajdonságok (bizonyos értelemszerű megkorlátozások mellett) az állományokkal azonos módon értelmezhetőek a könyvtárakra is.

Elérési útnak nevezzük a könyvtár helyének megadását a hierarchikus könyvtárstruktúra szerkezetének megfelelően. Az elérési út a könyvtár azonosítására szolgál. Mivel a gyökértől minden könyvtárhoz pontosan egy út vezet, ennek az útvonalnak a megadása egyértelműen azonosítja a könyvtárat. A megadás során az elérési út könyvtárait az operációs rendszer speciális szimbóluma (általában a / vagy a \) választja el egymástól. **Abszolút elérési út** használata esetén a gyökérkönyvtártól a hivatkozott könyvtárig az összes tartalmazó könyvtárat a hierarchiában elfoglalt helyének megfelelő sorrendben fel kell sorolni. Az abszolút elérési út használatával minden könyvtár (és minden állomány) egyértelműen azonosítható, azonban ez a fajta megadási mód egy összetettebb (már akár 4-6 szintű) hierarchia esetén meglehetősen kényelmetlen lehet. Egyszerűbbé tehető a hivatkozás, ha a könyvtár helyét nem a gyökérhez, hanem a pillanatnyilag használt könyvtárhoz képest adjuk meg. Ez viszont egyrészt azt jelenti, hogy nyilván kell tartanunk az aktuálisan használt könyvtárat (szokás **munkakönyvtárnak** is nevezni), másrészt szükségünk lesz olyan hivatkozások bevezetésére, amelyek a „visszafelé” hivatkoznak: a gyerekkönyvtárakhoz határozzák meg a szülőt. Az elterjedt hivatkozási könyvtár-szimbólumok a „.” a könyvtár saját adataira, a „...” pedig a szülőkönyvtárra mutató speciális hivatkozás. Ezek bevezetésével megadható **relatív elérési út** is, ahol a hivatkozott könyvtár elérését az aktuálisan használt (munka-) könyvtárhoz képest írjuk fel.

A könyvtárak szervezése a tárolási rendszereknek (azaz konkrétan a háttértárak) megfelelően további szinteken folytatódhat. A logikai tárolási rendszer legmagasabb szintjén

a háttértár egészének vagy részének megkülönböztetését lehetővé tevő azonosítók állnak. Ilyen a **partíció**: *a háttértár fizikailag összefüggő (folytonosan sorszámozott szektorokat tartalmazó) önálló része*, amely az operációs rendszer számára önálló adattároló egységként kezelhető és ilyen a **kötet**: *a háttértároló egyedi azonosítóval rendelkező területe*. (A kötetet szokás **meghajtónak** is nevezni, azonban fontos tudni, hogy a kötet és a fizikai meghajtó közötti megfeleltetés egyáltalán nem egyértelmű: elképzelhető a háttértároló teljes kapacitásának egyetlen azonosítóhoz (egy eszköz – egy kötet, pl. hajlékony lemez) rendelése, de egy partícionált winchester esetében (minden rész külön nevet kap, önálló kötetként használható) az összefüggés már egy eszköz – több kötet; fordított esetben (több meghajtó tárolási területének összevonására egyetlen logikai név alá) pedig több eszköz – egy kötet.)

Az egyes partíciók az operációs rendszer számára tehát úgy jelennek meg, mintha fizikailag különálló háttértárak (pl. újabb és újabb winchesterek) lennének. Mivel a hierarchikus fa struktúrájú könyvtárszervezési módszer alkalmazása esetén egy fizikai háttértárolón csak és kizárólag egyetlen gyökerkönyvtár helyezkedhet el, kézenfekvő, hogy több gyökerkönyvtárú struktúra kialakításához a merevlemez partíciókra kell osztani.

A kötet azonosítására egyes operációs rendszerek az angol ábécé betűit használják, mások megengedik a kötetek nevesítését.

3.4 Operációs rendszerek a gyakorlatban

3.4.1. Microsoft Windows

A Microsoft és Bill Gates az elmúlt 25 évben ugyanolyan meghatározó tényezővé vált a számítástechnikában, mint korábban Hollerith vagy Neumann. A Windows (minden hibájával és számos kritikájával együtt is) töretlen sikere és elterjedtsége révén az egyik legismertebb operációs rendszer-család.

Az egyes verziók jellemzői

1. Windows 1.x – 2.x: **nem** tekinthető **önálló operációs rendszernek** (grafikus felület a DOS-hoz).
2. Windows 3.x: **kooperatív multitasking** (az időosztásos elv szerint párhuzamosan futó folyamatok az operációs rendszer felügyelete nélkül (saját hatáskörükben) döntenek az időszelvény lejártakor a CPU átadásáról – aminek következtében, ha az aktuális (a CPU-t birtokló) alkalmazás lefagy, akkor nem tudja a CPU-t továbbadni, így az összes többi (egyébként még futásra képes) alkalmazás is várakozó állapotba kerül, amit a felhasználó úgy érzékel, hogy valamennyi lefagyott...), **16 bites alkalmazások**.
3. Win 3.11 (Windows for Workgroup): **egyenrangú hálózatok** (munkacsoport).
4. Windows NT: **önálló** operációs rendszer (saját kernel!), **preemptív multitasking** (a párhuzamosan futó folyamatok között a CPU kiosztása az operációs rendszer feladata), 32 bites alkalmazások, **NTFS** fájlrendszer, **kliens-szerver hálózati architektúra** (tartomány).
5. Windows 95/98/ME: **FAT32** (nagyobb tárolókapacitás, hosszú fájlnevek használata), újratervezett **felhasználói felület** (Asztal, Tálca), fejlesztett multimédia-támogatás (DirectX), **integrált Internet** (amiért később az ún. „Antitröszt-per”-ben el is marasztalták a Microsoftot).

(Megjegyzés: a Win95 esetében megoszlanak a vélemények, hogy valódi operációs rendszerről beszélhetünk-e, vagy még mindig egy DOS kernel grafikus felületéről van szó...)

6. Windows 2000/XP: központosított hálózati nyilvántartási rendszer (LDAP alapú címtár: **Active Directory**), **NTFS v5** (titkosító fájlrendszer (EFS), lemezkvóták, dinamikus kötetek).
7. Windows 2003: **.NET integráció, csak szerver** verziója van.

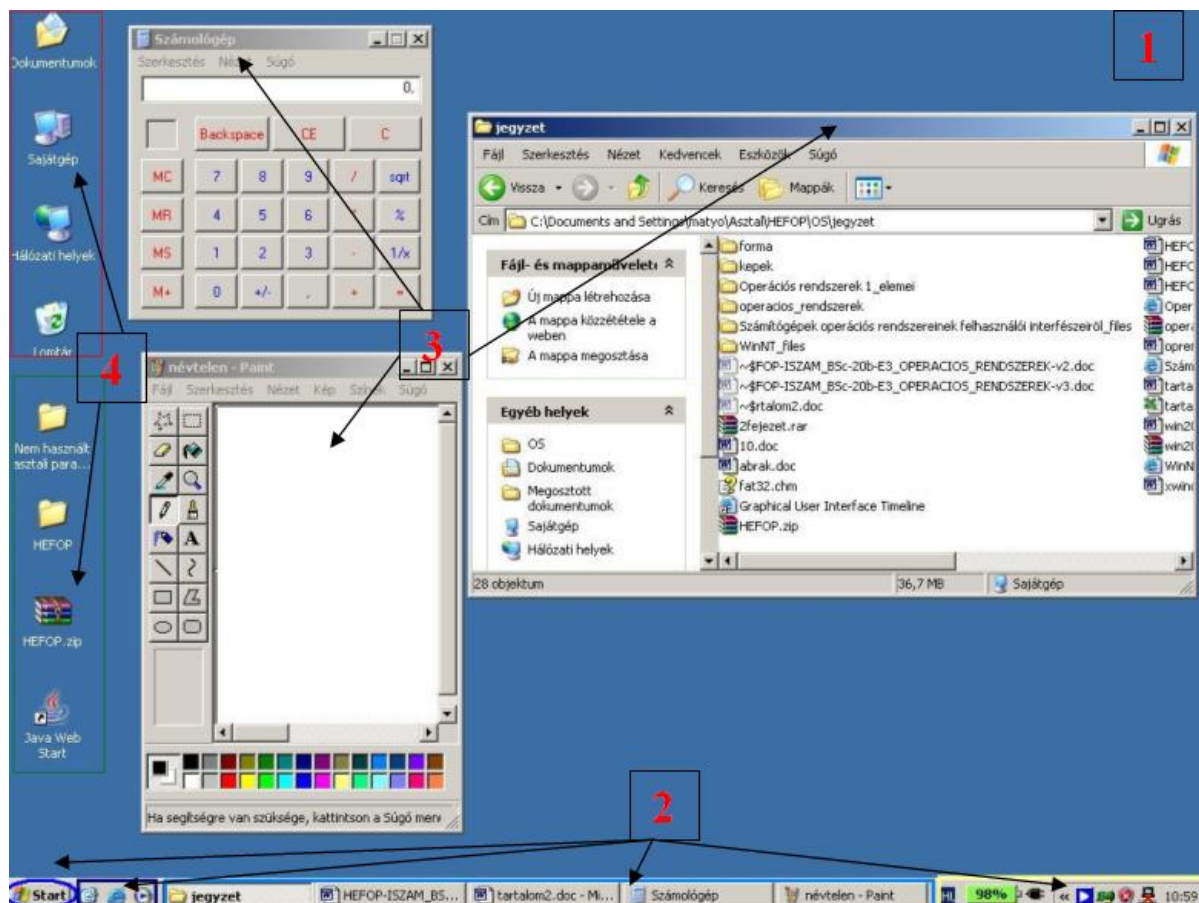
A jegyzet írásának időpontjában már sokat lehet hallania és olvasni a következő Windows verzióról (a legvalószínűbb, hogy „Windows Vista” név alatt fog megjelenni), de egyelőre csak a teszt-verziók érhetők el. A leírások tanúsága alapján a Vistában ismét számos újdonság jelenik meg, a legérdekesebbnek az NTFS kiterjesztéseként beharangozott új fájlrendszer tűnik (lehetővé válik többszörös hivatkozások és összetett keresések alkalmazása), de a felületen megjelenő újdonságok (3D Asztal: Aero, interaktív másodlagos tálca: „sidebar”, stb.) is bizonyára sokakat kápráztatnak majd el.

A Windows XP kezelésének legfontosabb jellemzői (GUI fölött)

A Microsoft fogalmai szerint a felhasználói munkaterület az Asztal – felfogható legmagasabb („root”) szintű ablaknak is. Az Asztal megjelenési formája (ahogy az Asztal a felhasználó számára megjelenik) a Tapéta (vagy Háttér). Az Asztal valamelyik szegélyénél helyezkedik el a Tálca, ami lényegében parancsikonok gyűjteménye. Az Asztalon alkalmazások (a hozzájuk tartozó Ablakban) illetve ikonok helyezhetők el.

Az Asztal szabványos elemei:

A Windows XP számos újdonsága közül a felhasználó elsőként valószínűleg az áttervezett felhasználói felülettel találkozik. A (korábbi verziókban megszokotthoz képest) megújult felület részint vizuális megjelenésében, részint felépítésében és bizonyos mértékben műveleteiben is eltér a korábbiaktól. A *3.1. ábra* az Asztal szokásos elrendezését és szabványos komponenseit szemlélteti – *de nem az alapértelmezés szerinti új, hanem a „klasszikus”-nak nevezett hagyományos megjelenítési módban!*



3.1. ábra: A Windows XP grafikus felülete

Az Asztal a felhasználói munkavégzés eszköze (3.1. ábra, 1). A Tapéta az Asztalon megjelenő grafikus elem, lehet egyszínű kitöltés (az 3.1. ábrán a Windows XP alapértelmezett kék színe látható), vagy valamilyen grafikus formátumú állomány tartalma (ez utóbbi esetben a kép méretének és a grafikus felbontásnak (az Asztal „méretének”) függvényében különböző megjelenítési módok alkalmazhatók: átméretezés (a két méret kiegyenlítése a kép méretének arányos nyújtásával vagy zsugorításával), megjelenítés (a kép saját méretében az Asztal közepén helyezkedik el), ismétlés (a kép az Asztal bal felső sarkától kiindulva, amennyiben kisebb az Asztalnál, akkor jobbra és lefelé ismételve jelenik meg). Az Asztal kinézetével kapcsolatban a Windows XP számos előre definiált összeállítást tartalmaz, amibe nem csak a Tapéta „képe” tartozik bele, hanem az Asztalon megjelenő valamennyi (ablak)komponens formai leírása is – ezek a Sémák –, de ezek a megjelenítési beállítások (akár elemenként is) egyénileg is kialakíthatók.

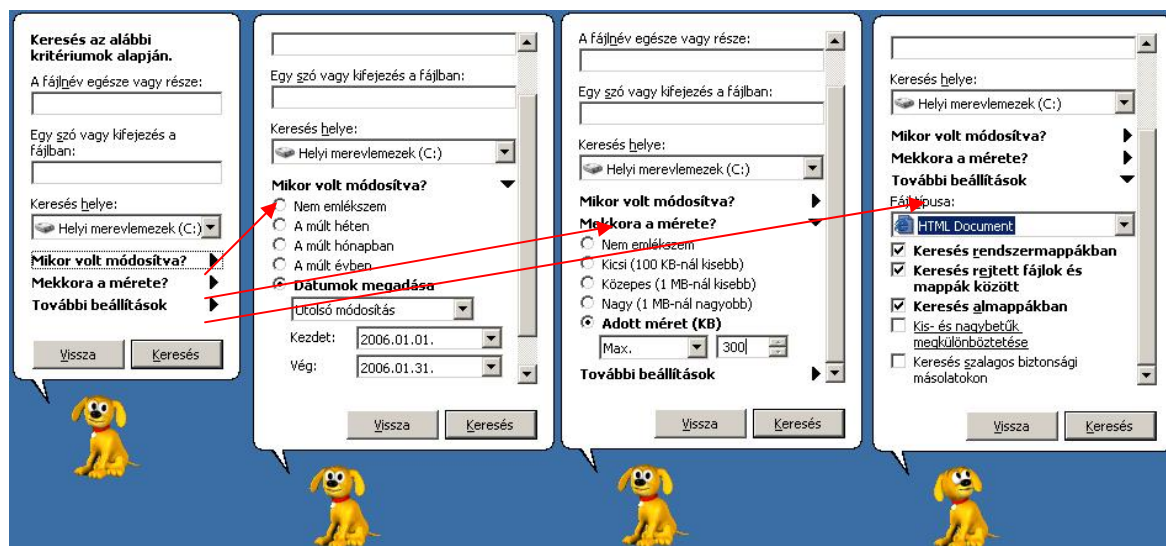
A Tálca a Windows kezelésének alapvető vezérlési eszköze (3.1. ábra, 2). Alapértelmezés szerint az Asztal alsó szegélyénél helyezkedik el, de bármelyik szegélyre (alsó, felső, bal, jobb) áthelyezhető, továbbá mérete is változtatható (legfeljebb az Asztal területének feléig növelhető, illetve akár teljesen el is rejtethető). A Tálca részei (balról jobbra): START menü, gyorsindító eszköztár (lényegében a START menü Programok elemének kivonatolt változata: a felhasználó által leggyakrabban használt programok elindítására szolgáló ikonok), alkalmazás-ablakok ikonizált állapotának megjelenítési területe (a futó programokhoz tartozó ablakok ikon méretű megfelelői, lehetővé teszi az alkalmazások közti váltást, a Windows XP újdonsága, hogy az azonos alkalmazáshoz tartozó ablakokat képes egyetlen ikon-csoportban megjeleníteni), státusz (állapot-jelző) terület (a rendszerszolgáltatások és a háttérben futó felhasználói folyamatok (azaz olyan programok,

amelyekhez nem tartozik az Asztalon ablakterület) ikonjainak gyűjteménye, amelyen keresztül ezek az alkalmazások felügyelhetők).

A START menü a Windows alapértelmezett kezelési eszköze. Parancsikokon csoportosított (és szervezett) gyűjteménye, amelyben egyaránt megtalálhatók az operációs rendszer felhasználói szolgáltatásaira vonatkozó elemek (felhasználói munka befejezésével kapcsolatos feladatok, beállítási műveletek, támogatási szolgáltatások, rendszerkomponens alkalmazások) és a felhasználó munkavégzéséhez kapcsolódóan létrejövő saját elemek (utoljára használt állományok – illetve alkalmas felületi beállítás esetén programok is – jegyzéke, felhasználó által telepített alkalmazások (programok) elindítására szolgáló parancsikonok). Az egyes elemekkel kapcsolatban a következőket érdemes tudni:

1. leállítás: a felhasználói tevékenység befejezésének szabályos eszköze (a számítógép - fizikai szintű – kikapcsolása az operációs rendszer megfelelő leállítása nélkül adatvesztést, illetve a rendszerállományok sérülését okozhatja!). A számítógép leállításával kapcsolatban több lehetőség is elképzelhető:
 - a. a leállítás a számítógép fizikai kikapcsolását eredményezi (régebbi típusú alaplapok esetén csak lehetővé teszi);
 - b. az újraindítás az operációs rendszer teljes újratöltését eredményezi (mintha csak most kapcsoltuk volna be a számítógépet);
 - c. a kijelentkezés az adott felhasználó és a hozzá kapcsolódó rendszerszolgáltatások befejezését valósítja meg (alapvető különbség az újraindításhoz képest, hogy nem a teljes rendszerbetöltési folyamat ismétlődik meg, csak a bejelentkezéstől a felhasználói környezet kialakításáig terjedő rész, azaz nem minden operációs rendszer szolgáltatás indul újra);
 - d. a felhasználó-váltás anélkül teszi lehetővé más felhasználói környezetre (és munkamenetre) történő áttérést, hogy az adott felhasználó tevékenysége(i) befejeződne(nek) – a jelenlegi felhasználói környezet működése mellett egy új felhasználói környezet indítására ad lehetőséget;
 - e. a készenléti állapot és a hibernálás pedig a munkamenet felfüggesztésének (energiatakarékos) eszközei, mindkét esetben a kijelölt állapotból visszatérve a felhasználói tevékenység a felfüggesztést megelőző állapotból folytatható (azaz a felület, a futó programok, stb. ugyanaz lesz, mint a felfüggesztés előtt). Alapvető különbség a két leállítási lehetőség között, hogy készenléti állapotban csak alacsonyabb energiaszintre kapcsolódnak a számítógép hardver egységei (a nem használt eszközöket az operációs rendszer lekapcsolja), hibernálás esetén pedig a teljes környezet (a felület, a programállapotok, a memóriatartalom, stb.) lementésre kerül a háttértárra. (Ebből következően készenléti állapotban egy esetleges áramszünet adatvesztést okoz, hibernálás esetén nem, másrészt a hibernálás használatához szükséges a fizikai memóriaméretnek megfelelő szabad tárhely valamelyik háttértáron.)
2. futtatás: a felhasználói parancskiadás eszköze (lényegében felfogható egyetlen parancs kiadására alkalmas karakteres felületnek is). Használható olyan alkalmazás elindítására, amely nem rendelkezik parancsikonnal (ilyenek pl. a karakteres felület parancsai, de ilyen a (felügyeleti eszközök között tárgyalásra kerülő) rendszerleíró adatbázis kezelő programja (regedit), vagy a Felügyeleti Konzol (mmc) is), vagy egy parancsikonnal rendelkező alkalmazásnak (az ikonban rögzítettektől) különböző paraméterekkel történő indítására.

3. sűgó: az operációs rendszer felépítésével, működésével, kezelésével, hibaelhárításával kapcsolatos tudnivalók több szinten kategorizált, tematikusan és kulcsszó alapján egyaránt kereshető rendszere.
4. keresés: a számítógépen (illetve hálózati környezetben az adott hálózati szegmensben) elérhető információk többszintű keresését lehetővé tevő eszköz. Legáltalánosabban állományok keresésére használható, a háttértáron elhelyezkedő állományok helyének meghatározásának eszközrendszerét bővíti: (a logikai tárolási rendszerben – könyvtár-struktúrában – elfoglalt hely megadása mellett) további keresési ismérvek kezelésére is képes. A legfontosabbak a tartalom szerinti keresés, illetve az állományazonosító részleges ismeretében használható (helyettesítéses) keresés – ez utóbbi esetben az állomány ismert részeit helyettesítő- (maszk, joker) karakterekkel: „*” (tetszőleges számú, tetszőleges karakter) vagy „?” (pontosan egy darab tetszőleges karakter) egészíthetik ki –, de szinte az összes állománytulajdonságra (típus, méret, időbélyegek) alkalmazható (3-2. ábra).



3.2. ábra: A kereső különböző szolgáltatásai

5. beállítások: a rendszer működési paramétereinek módosítását lehetővé tevő (egyik, ld. a „felügyeleti eszközök” c. részt) felhasználói eszközök gyűjteménye (Vezérlőpult).
6. dokumentumok: ilyen néven két objektum szerepel a Windows alapértelmezett felhasználói felületén, némiképp hasonló szerepben, de alapvetően eltérő szerkezettel: a START menü Dokumentumok eleme a felhasználó által utoljára szerkesztett állományokra vonatkozó hivatkozások jegyzéke, a Dokumentumok mappa (az Asztalon) pedig tényleges állományokat tartalmaz.
7. programok: az operációs rendszer és a felhasználó által telepített alkalmazásokhoz tartozó parancsikonok gyűjteménye.

Az ablakok (3.1. ábra, 3) a felhasználói tevékenységeket megvalósító alkalmazásokhoz rendelt képernyő-részek. Az ablakok általános szerkezete (címsor: az ablak által reprezentált program és (esetlegesen) a program által feldolgozott adatállomány ikonja és/vagy neve; menü; eszköztárak (az alkalmazáshoz kapcsolódó tevékenységekhez rendelhető ikonok); munkaterület, gördítősávok; állapotsor; keret) minden alkalmazás esetében azonos. Az ablakok kezelésével kapcsolatban az alapvető műveletek (mozgatás, alapértelmezett

méretezési műveletek:  (kicsinyítés ikonméretre),  (az ablak az Asztal teljes felületét lefedi),  (váltás a teljes és az előző méret között)) a címsoron keresztül valósíthatók meg (az előzőektől különböző méret a szegély segítségével állítható be). Az inaktív ablakok címsora alapértelmezés szerint szürke, az aktív ablaké élénk (kék) – mindkét beállítás módosítható az Asztal megjelenítési sémáinak segítségével.

Az Asztalon alapvetően 3 (hálózati kapcsolat esetén 4, illetve 5) rendszerszintű ikon jeleníthető meg (3.1. ábra, 4): a Sajátgép (a Windows alapértelmezett állománykezelője), a Dokumentumok (rendszerkönyvtár, a felhasználó saját adatállományainak alapértelmezett tárolási helye), Lomtár (a (logikailag) törölt állományokra mutató hivatkozásokat tartalmazó speciális rendszerkönyvtár), továbbá (amennyiben a számítógép rendelkezik hálózati csatlakozóeszközzel (hálózati kártya, modem, stb.) a hálózatok kezeléséhez kapcsolódóan a „Helyi hálózatok” (az adott helyi hálózatban elérhető megosztott erőforrások áttekintésére, kezelésére szolgáló vezérlő) és alapértelmezett böngészőprogramként az „Internet Explorer”). A fentiek mellett a felhasználó szabadon helyezhet el az Asztalon újabb ikonokat (3.1. ábra, 4), amelyek alapvetően két típusúak lehetnek: közvetlen (egy, a háttértáron fizikailag létező objektumot reprezentáló) ikon, illetve közvetett (a Windows „parancsikonnak” nevezi ezeket az ikonokat még akkor is, ha az ikon által reprezentált – a parancsikonnak megjelenési helyétől független elem – nem futtatható parancs- vagy programállomány (alkalmazás), hanem adatfájl) – ez utóbbit az ikon bal alsó sarkában megjelenő nyíl alakú szimbólum jelzi.

A grafikus felület megjelenésével kapcsolatban fontos megjegyezni, hogy az operációs rendszer megjelenése a Windows filozófiája szerint szorosan kötődik a felhasználóhoz. Ez a gyakorlatban azt jelenti, hogy minden felhasználó saját felhasználói felülettel rendelkezik – ez alapértelmezés szerint természetesen azonos, hiszen ugyanazon rendszerszintű felhasználói felület másolataként jön létre, de a felhasználó saját beállításai ezt módosíthatják. A felhasználóra jellemző felület beállításai, további bizonyos szolgáltatások, alkalmazások működéséhez kötődő beállítások együttesen alkotják a felhasználói profilt. A profil alapértelmezett tárolási helye a rendszerkönyvtár „Documents and settings” nevű (rendszer)könyvtárának a felhasználó nevével megegyező nevű mappája.

A felhasználói interakció eszközei:

Grafikus felületű operációs rendszerről lévén szó, a felhasználói tevékenységek kiválasztásának, megadásának alapvető eszköze valamilyen pozicionáló (kijelölő) eszköz (leggyakrabban az egér, illetve – elsősorban hordozható számítógépek esetében – az egér működésével azonos működési elvű eszköz), de mint azt a GUI alapelveinél is említettük, ez nem kizárólagos. Az egér használatakor a felhasználói tevékenység hatásával kapcsolatban a Windows az egér szimbólumának megváltoztatásával ad tájékoztatást, a legfontosabb állapotok a következők:

- o nyíl  : alapértelmezett egérkursor-alak, általánosan használható kijelölésre, kiválasztásra.
- o kettős nyíl  : objektum méretezése.
- o négyes nyíl  : objektum mozgatása.
- o homokóra  : (az előzőleg kiválasztott) tevékenység végrehajtásának időtartama alatt megjelenő kursor, amennyiben a végrehajtás alatt más tevékenység nem végezhető.
- o tiltás  : a kijelölt művelet (az egér jelenlegi helyén) nem végezhető el, nem értelmezett(hető).

- o a különböző alkalmazásokban a fentiek mellett különböző egéralakok is elképzelhetők, pl. szövegbevitelkor a szövegkurzor: , stb.

Az egér kezelésével kapcsolatban a Windows rendszerszinten két gombot különböztet meg, alapértelmezés szerint a bal oldali az elsődleges (az egyes tevékenységek kiválasztásához használandó), a jobb gomb az ún. gyors (vagy helyi) menük (olyan dinamikus változó parancsokból felépülő menük, amelyek aktuális tartalma az egér pillanatnyi helyzetétől függ) megjelenítését teszi lehetővé. (A két gomb szerepe felcserélhető, továbbá az adott eszköz hardverkiépítésétől függően további szolgáltatások is definiálhatóak: pl. harmadik gombhoz, görgő használatához, stb.) A gombok kezelése mellett a rendszerszinten definiált műveletek (események) az egyszeres (alapértelmezés szerinti hatása: kiválasztás) és a kétszeres (dupla, alapértelmezés szerinti hatása: végrehajtás) kattintás, valamint a vonszolás (az egér elsődleges gombjának nyomva tartása mellett az egér mozgatása, alapértelmezés szerinti végrehajtása: mozgatás).

A mutatóeszköz használata mellett az alapvető tevékenységek billentyűzetről is megadhatók, rendszerint valamilyen billentyű-kombináció (egy vagy több vezérlőbillentyű lenyomva tartása közben kell a megfelelő billentyűt lenyomni) formájában. A teljesség igénye nélkül néhány általánosan használható billentyűkombináció:

- o CTRL+Esc – START menü aktiválása
- o F1 – súgó (az aktív ablakra vonatkozóan)
- o F3 – keresés (az állománykezelő alkalmazásokban)
- o F2 – objektum átnevezése (az állománykezelő alkalmazásokban)
- o SHIFT+Del – objektum fizikai törlése (az állománykezelő alkalmazásokban)
- o Alt+Esc – következő ablak aktiválása
- o Alt+F4 – aktív ablak bezárása
- o F10 – menüsor elérése
- o Alt+<betű> - a <betű> kiemeléssel (menü szövegében aláhúzott karakter) jelölt menüelem kiválasztása
- o SHIFT+F10 – gyorsmenü
- o Alt+Enter – kiválasztott objektum tulajdonságai
- o CTRL+C, CTRL+X, CTRL+V – vágólap kezelése (másolás, kivágás, beillesztés)
- o +M – minden ablak minimalizálása (ikonállapotban a Tálcára)
- o +D – Asztal megjelenítése
- o PrtScr („Print Screen”) – az Asztal aktuális kinézeti képének másolása a Vágólapra (Az Alt vezérlőbillentyűvel együtt használva csak az aktuális ablak tartalmát másolja.)

(A fenti felsorolás természetesen nem teljes, a  szimbólummal jelölt vezérlő billentyű pedig nem található meg minden billentyűzeten.)

A felhasználói munkavégzés során a tevékenységek természetesen nem csak az Asztalra (vagy annak részeire), illetve az ablakok szabványos elemére vonatkozhatnak: a különböző felhasználói információk megadására számos további (grafikus) eszköz is rendelkezésre áll, ezeket nevezzük vezérlőknek. A vezérlők jelentősége túlmutat az operációs rendszerek kezelésének témakörén, hiszen bármilyen alkalmazásban a felhasználói beavatkozás lehetőségét határozzák meg – ebben az értelemben akár programozási komponenseknek is felfoghatóak –, azonban mivel működésük erősen kötődik az operációs rendszer használatához, ezért a legfontosabbakat az alábbiakban (3.3. ábra) összefoglaljuk:

- o szöveg- (vagy beviteli) mező: billentyűzetről származó adat fogadását és megjelenítését végző vezérlő.

- o rádiógomb: egyszeres kiválasztást lehetővé tevő vezérlő: általában több, egymást kizáró alternatívát jelenít meg, az egyik kiválasztásának hatására az (esetleges) korábbi kiválasztás törlődik.
- o jelölőnégyzet: többszörös kiválasztást lehetővé tevő vezérlő: a rádiógombtól annyiban különbözik, hogy az alternatívák között megengedi több kiválasztását is.
- o nyomógomb: kiválasztó vezérlő, általában tevékenységhez kötődik.
- o toló- („pot-”) méter: kiválasztás típusú vezérlő, értéktartományban történő dinamikus (analóg) kiválasztást tesz lehetővé.
- o lap (fül): felsorolás típusú vezérlő, tartalma további vezérlők tematikus gyűjteménye, használatával egyetlen párbeszéd-ablakon belül több vezérlőcsoport is megadható
- o lista: felsorolás típusú vezérlő, előre definiált értékek közötti választást tesz lehetővé.
- o kombinált lista: felsorolás típusú vezérlő, az előre definiált értékek kiválasztása mellett (a listában nem szereplő) új érték megadását is lehetővé teszi (tulajdonképpen egy szövegmező és egy lista együtt)



3.3. ábra: Párbeszédablakok vezérlői

Az eddig megismert eszközöket számos kernel-szolgáltatás egészíti ki, amelyek közül külön említést érdemel a Vágólap. A Vágólap az operatív memóriának egy olyan része, amely valamennyi alkalmazás számára elérhető (emlékezzünk: szegmentált memória-kezelés esetén ilyen memória-terület létezése egyáltalán nem triviális!). A Vágólaphoz vonatkozó műveleteket az adatmozgatás iránya és jellege szerint csoportosíthatjuk: másolásnál a kijelölt objektum marad a helyén, tartalmáról képződik egy másolat a Vágólapon, kivágásnál a kijelölt objektum eredeti helyéről törlődik, és csak a Vágólapon érhető el a továbbiakban, beillesztés során pedig a Vágólap aktuális tartalma a kurzor aktuális helyére másolódik. A Vágólap kezelésére vonatkozó parancsok (nevük az egyes műveletek nevei) elérhetők minden alkalmazás Szerkesztés menüjében, a legtöbb alkalmazás eszköztárában, a helyi menükben és akár billentyű-kombinációk formájában is. A Vágólap kezelésével kapcsolatban két dolgot kell megjegyeznünk: az egyik, hogy alapértelmezés szerint vágólaphoz csak egy van (egyes alkalmazások lehetővé teszik több vágólap kezelését is – pontosabban ebben az esetben is csak egyetlen közös területről van szó, de az alapértelmezéstől eltérően ezen a területen több, különböző forrásból származó információ is tárolható), így minden, a vágólaphoz adatot mozgató művelet (másolás, kivágás) törli az aktuális tartalmat (a beillesztés viszont nem! – tehát a vágólap aktuális tartalma többször is felhasználható); a másik, hogy a vágólap tartalma csak olyan alkalmazásban használható fel, ami képes az aktuális típusú információ fogadására

(pl. egy képet elhelyezve a vágólapon, ha megpróbáljuk egy formázatlan szöveget tartalmazó dokumentumba beilleszteni – ld. Jegyzetomb –, nem járunk sikerrel).

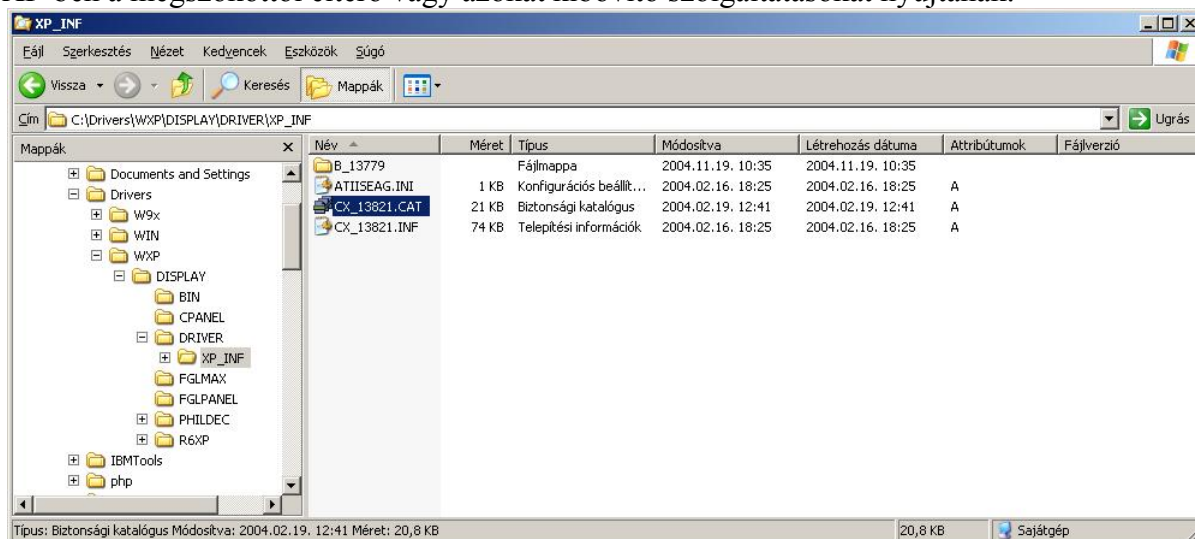
Állománykezelés

A grafikus felületű operációs rendszerekben az ablakkezelő mellett van még egy szabványosnak tekinthető komponens, az operációs rendszer állománykezelő szolgáltatásainak grafikus megfelelője. (Ez többé-kevésbé természetes is, ha belegondolunk a GUI működési elvébe: az ablakkezelő eseményvezérlést végez, de csak a GUI komponensek vonatkozásában – márpedig ahogy azt a Windows példájában láthattuk, ezen komponensek között szerepel(het)nek olyanok is: mappa, állomány, parancsikon, amelyek nem a grafikus felületnek, hanem az állomány-rendszernek képezik a részét.) A Windows ezt az elemet Explorer-nek (Intézőnek) hívja. A Windows Intéző tehát egyrészt valamennyi, az állományszervezéssel kapcsolatos művelet megvalósításáért felelős eleme a grafikus környezetnek, másrészt a Windows ugyanígy nevezi (nem véletlenül!) a grafikus felület (rendszerszintű) állománykezelő komponensét is. (A Sajátgép és az Intéző ugyanaz a folyamat, csak eltérő alapértelmezett felületi beállításokkal.)

Ahogy azt az állománykezelő rendszerek elméleti áttekintése során láttuk, az operációs rendszer ezen komponensének feladatai az állományok logikai és fizikai elhelyezéséhez kötődnek (a felhasználó szempontjából természetesen a logikai szint az elsődleges). Az Intéző tehát az állományszervezés logikai szintjén megjelenő feladatok elvégzésének eszköze.

A Windows Intéző (3.4. ábra) munkaterülete alapvetően két részre oszlik: a bal oldali a „böngészősáv”, a jobb oldali a „megjelenítési terület”: az előbbi a tevékenységek, az utóbbi az objektumok (mappa, állomány) kiválasztására szolgál (mindkét rész megjelenési módja változtatható a Nézet menü parancsainak segítségével). Az Intézőben elvégezhető tevékenységek mind a logikai állományszervezés struktúrájához (meghajtóra (kötetre), könyvtárra (mappára) és állományokra vonatkozó műveletek), mind hagyományos tevékenységeihez (létrehozás, módosítás (jellemzők, elhelyezkedési információk megváltoztatása) törlés) illeszkednek. Az egyes műveletek többnyire mindhárom megismert vezérlési mód (a menürendszer, az eszköztár ikonjai vagy a gyorsmenük használata) segítségével elvégezhetők.

A következőkben ezen műveletek közül (az alapvető állománykezelési tevékenységek elvégzésének lépésenkénti ismertetésétől eltekintve) azokat tekintjük át, amelyek a Windows XP-ben a megszokottól eltérő vagy azokat kibővítő szolgáltatásokat nyújtanak.



3.4. ábra: A Windows Intéző felülete

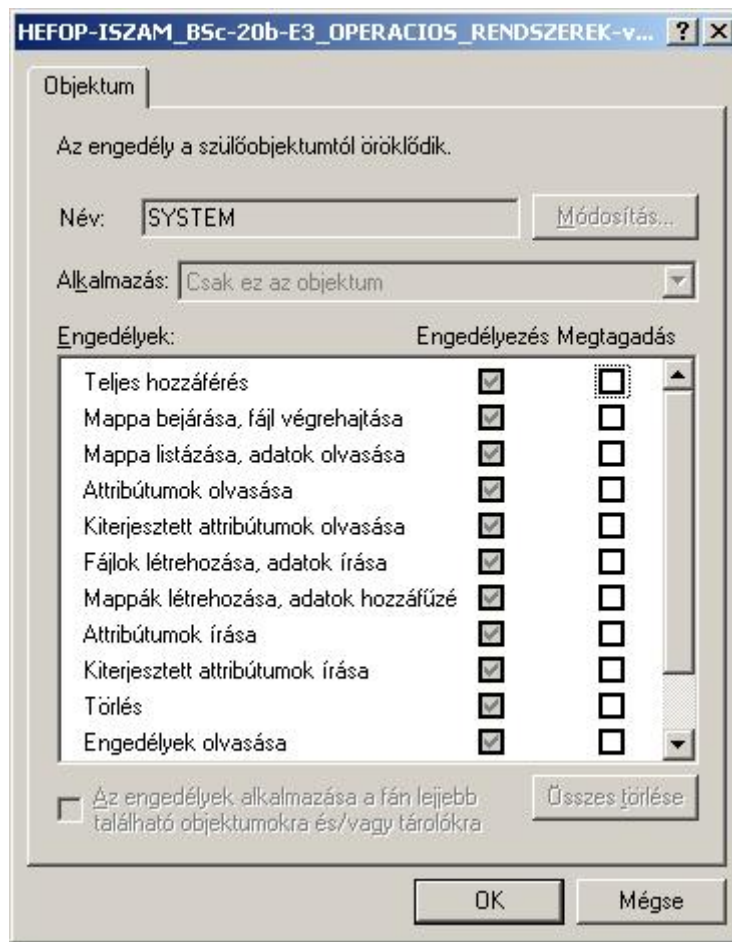
Fájlrendszerek: a Windows XP alapvetően mindhárom (a Microsoft esetében hagyományosnak mondható) fájlrendszert képes kezelni: FAT(16), FAT32, NTFS – de hatékony, és az operációs rendszer szolgáltatásait maximálisan kihasználó módon csak NTFS partíció. (FAT alapú partíciók használata csak abban az esetben indokolt, ha az adott partíció tartalmának a Windows korábbi verzióit futtató számítógépeken is elérhetőnek kell lennie, a FAT alapú partíciók adatvesztés nélkül konvertálhatók NTFS partícióvá – ez visszafelé nem érvényes!) Az egyes fájlrendszerek között komoly különbségek húzódnak meg mind az elérhető szolgáltatások körében, mind a technikai korlátok mértékében (3.1. táblázat):

3.1. táblázat: A Windows XP által támogatott fájlrendszerek

	FAT	FAT32	NTFS
Kötetméret	1 MB – 4 GB	512 MB – 2 TB (elvi) 512 MB – 32 GB (WXP gyak)	10 MB -
Fájlméret	max. 2 GB	max. 4 GB	nem korlátozott
Hosszú fájlnévek	alkalmazható	alkalmazható	alkalmazható
Jogosultság- kezelés	nincs	nincs	alkalmazható
Tömörítés	nincs	nincs	alkalmazható
Titkosítás	nincs	nincs	alkalmazható
Kvóta	nincs	nincs	alkalmazható

Az NTFS működési újításai közül az egyik legfontosabb a dinamikus kötetek kezelése. Dinamikus kötetet dinamikus lemezen lehet létrehozni – ez utóbbit pedig a lemezkezelés beépülő modul segítségével a hagyományos háttértár konvertálásával készíthetjük el. A dinamikus lemezek lehetővé teszik több lemezes kötetek kialakítását, a tárterület dinamikus növelését, illetve RAID rendszerekre jellemző szolgáltatások (csíkozás, tükrözés) megvalósítását.

Az NTFS az állománytulajdonságok kibővült körét tartja nyilván az egyes háttértárakon elhelyezkedő információkról, ezek közül felhasználói szempontból a jogosultsági rendszerhez kapcsolódó állománytulajdonságok a legfontosabbak. Tekintettel arra, hogy a Windows XP alapvetően több-felhasználós operációs rendszer, természetes, hogy az egyes állományokra vonatkozóan a felhasználók (vagy felhasználói csoportok) szintjén meghatározható az elvégezhető műveletek köre. Az NTFS partíció elhelyezkedő állományok esetében a jogosultsági rendszer kétszintű: alapvetően összetett jogok beállítását támogatja (ezek: írás, olvasás (és végrehajtás), módosítás), de elemi jogok segítségével ennél lényegesen finomabb műveleti szabályozás is elképzelhető. (Az elemi jogok beállításához az adott objektum tulajdonságai között a Biztonság fülön található Speciális parancsgomb segítségével juthatunk, 3.5. ábra.)



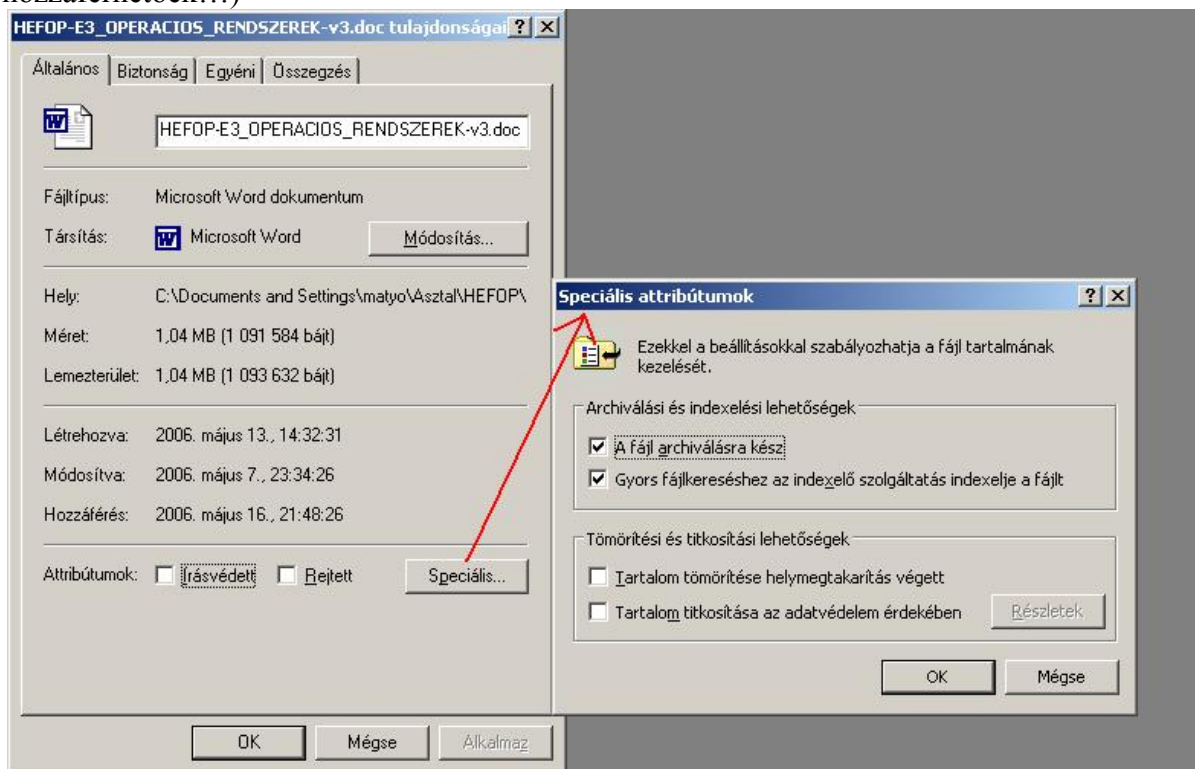
3.5. ábra: Elemi állományszintű jogosultsági beállítás

A jogosultsági rendszerhez szorosan kötődik a szintén csak NTFS partíción elhelyezkedő állományok esetében alkalmazható titkosítási művelet (3.6. ábra). A titkosítás felhasználói szempontból az állomány egyik (kiterjesztett) attribútuma, az operációs rendszer EFS (Encryption File System) szolgáltatása valósítja meg. (Kiterjesztett attribútumok alatt a Windows hagyományos RHSA („read only” – írásvédett, „hidden” – rejtett, „system” – rendszerfájl, „archiv” – archív(ált) állomány) állomány-jellemzőit kiegészítő attribútumokat értjük. A Windows XP kezeli (már csak kompatibilitási okokból is) a négy hagyományos attribútumot – annak ellenére, hogy grafikus felületen (az állomány tulajdonságai között) csak kettőt, az írásvédelmet és a rejtettséget jeleníti meg.) A titkosítást az állomány (vagy mappa) tulajdonságai között az „Általános” fül „Speciális” parancsgombja segítségével állíthatjuk.

Amennyiben egy állomány rendelkezik „titkosított” attribútummal, úgy a háttértáron nem az eredeti tartalma, hanem annak egy kódolt változata kerül letárolásra. Ennek ellenére az állomány a hagyományos állományokkal azonos módon használható (a titkosítás-visszafejtés „transzparens” folyamat) – legalábbis a titkosítást végző felhasználó számára, bárki más „hozzáférés megtagadva” hibajelzést kap. A titkosítás attribútuma vonatkozhat mappára is, ebben az esetben (értelemszerűen) a mappa teljes tartalma titkosításra kerül.

(Technikailag a titkosítás során a felhasználó egyedi tanúsítványával és egy biztonsági okokból létező, „helyreállítási ügynök”-nek nevezett speciális azonosító tanúsítványával mint kulccsal történik a titkosítás. Mivel a tanúsítványban a felhasználóra egyedileg jellemző információk tárolódnak, (létrehozáskor olyan módon generálódik, ami biztosítja az egyediségét), ezért ha letöröljük a titkosító felhasználót, hiába hozunk létre ugyanolyan

jellemzőkkel egy újat, az előző felhasználó által titkosított állományok nem lesznek hozzáférhetőek...)



3.6. ábra: Kiterjesztett attribútumok használata

Szintén a kiterjesztett attribútum-készletbe tartozik a röp-tömörítés tulajdonság is (3.6. ábra). A Windows XP alapvetően két tömörítési eljárással rendelkezik: a „tömörített mappák” szolgáltatás felhasználói tevékenység, a röp-tömörítés (a titkosításhoz hasonlóan a felhasználó számára észrevehetetlen – transzparens – módon megvalósuló) rendszerszolgáltatás. A röp-tömörítés a titkosítással azonos lapon kapcsolható attribútum, amennyiben egy állomány (vagy mappa) rendelkezik ezzel az attribútummal, úgy a tartalma a háttértárra történő írási művelet során automatikusan tömörítésre kerül (megnyitáskor pedig – értelemszerűen – kitömörítésre).

A „tömörített mappák” ezzel szemben nem automatikus tömörítési eljárás (lényegében egy ZIP kódolású tömörítő algoritmus megvalósításáról van szó, amelyet integráltak az operációs rendszer szolgáltatásai közé), hanem felhasználói tevékenységről: amennyiben egy állományt (vagy mappát) be szeretnénk tömöríteni, úgy a (Fájl menü vagy a gyorsmenü) „Küldés” parancs „Tömörített mappák” elemét kiválasztva a tömörítés megtörténik. A tömörített mappa tulajdonképpen nem más (a hagyományos tömörítő programok: ZIP, RAR, ICE, stb. terminológiájával), mint egy archívum – amelyet azonban a Windows XP a mappákkal analóg módon kezel. Ennek megfelelően ebben a tömörítési módszerben is tetten érhető bizonyos mértékű transzparens működés: amennyiben már létezik egy ilyen mappánk, úgy állományokat helyezve ebbe a mappába megtörténik egy betömörítési művelet, állományokat kimoizgatva ebből a mappából pedig lezajlik egy kibontási folyamat.

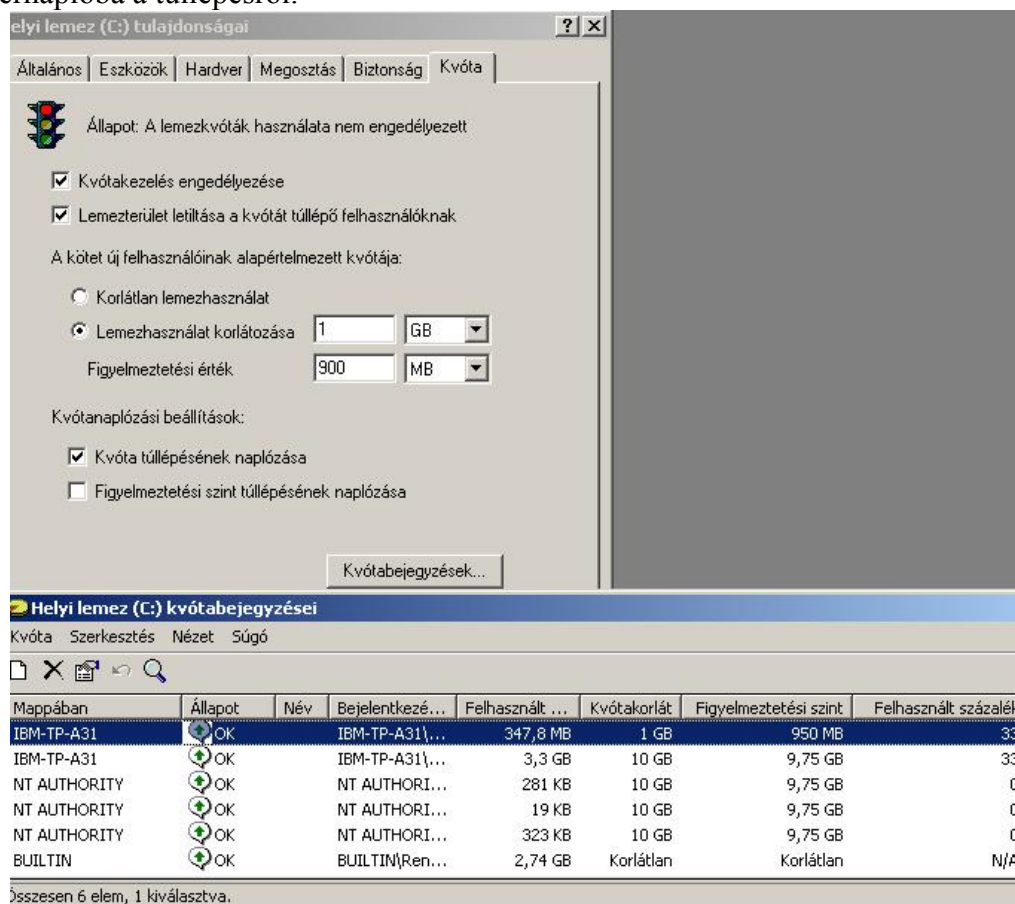
A két módszer azonban (a felhasználási terület jellemzőinek megfelelően) nem azonos módon működik: a röp-tömörítés gyorsabb, ugyanakkor (a tömörített állapot méretét tekintve) kevésbé hatékony, mint a tömörített mappák. (Ez természetes, ha belegondolunk abba, hogy röp-tömörítés használatakor a felhasználó nem elsődlegesen a tömörítés műveletével akar foglalkozni, hanem az állomány tartalmával, ezzel szemben a tömörített mappák használatakor maga a tömörítés a cél.)

(Végezetül egy utolsó megjegyzés a kiterjesztett attribútumok működésével kapcsolatban. Az ilyen attribútummal rendelkező állományok helyválttatása időnként (első ránézésre) meglepő eredményeket produkál:

1. ha lemásolunk egy röptömörített állományt egy ilyen attribútummal nem rendelkező könyvtárba, úgy törlődik ez az attribútum (azaz a célhelyen már nem lesz tömörített),
2. ha ugyanezt az állományt mozgatással helyezzük át egy másik (de az eredetivel azonos kötetben található!) könyvtárba, úgy a tömörítettsége megmarad.

A magyarázat az állománykezelő rendszer működésében rejlik: a mozgatás az állománykezelő rendszer számára csak a katalógus-bejegyzés módosítását jelenti, tehát (az operációs rendszer szerint legalábbis) ebben az esetben az állomány helye nem változik meg...)

Ahogy az a fentiekből is kiderült, a Windows XP az állománykezelés hagyományos feladatait számos többlet-szolgáltatással egészíti ki, amelyek ha úgy tetszik, kényelmi eszközök, de ha szem előtt tartjuk a több-felhasználós működési modellt, akkor ezek szükségszerű komponensek. A helyfoglalás korlátozása is egy ilyen eszköz, amely a hálózati operációs rendszerek jellemző szolgáltatásai közül került a Windows XP eszköztárába. A kvóták használatát kötet szinten lehet engedélyezni (a kötet tulajdonságai között található „Kvóta” fül segítségével, 3.7. ábra). Amennyiben egy kötet rendelkezik kvótával, úgy az egyes felhasználók legfeljebb a kvótában megadott területet foglalhatnak le az adott kötetben – beállítás kérdése, hogy a kvótában beállított adatmennyiség elérése után további adatmentés az adott felhasználó számára nem lesz lehetséges, vagy csak egy bejegyzés készül a rendszernaplóba a túllépésről.



3.7. ábra: Lemezkvóták beállítása és lekérdezése

Az állománykezelés gyakorlatában a megszokott műveletekhez képesti eltérések ismertetésekor a típusok szerepéről kell még szót ejtenünk. A Windows XP a hagyományos állománytulajdonságok (azonosító, méret, időbélyegek, jellemzők) közül az azonosító tulajdonság végének külön jelentőséget tulajdonít. Szokás ezt kiterjesztésnek nevezni, más források típusként hivatkoznak rá, technikailag az állomány azonosítójában az utolsó pont után elhelyezkedő karakterekről van szó. A típus jelentősége abban áll, hogy az operációs rendszer ez alapján meghatározza az adott állománnyal elvégezhető (nem állomány-, hanem alkalmazás szintű) műveletek körét, illetve hozzárendeli (amennyiben elérhető az adott számítógépen) az állományt feldolgozni képes alkalmazás(oka)t – ezt nevezzük társításnak. A Windows XP-ben lehetőség van egy típushoz többszörös társítási leírást is létrehozni, ilyen módon ugyanazon állománnyal (a felhasználás célja függvényében) más és más alkalmazások is képesek dolgozni. Ez az oka annak, hogy amennyiben megváltoztatjuk egy állomány azonosítóját, akkor az operációs rendszer egy (a kezdő felhasználót gyakran megtévesztő és elbizonytalanító) üzenetben figyelmeztet a típusváltásból adódó (lehetséges) társítási problémákra.

A társítás, illetve az állomány-típusok kezelése azonban közel sem triviális: alapértelmezett telepítés mellett azok a típusok, amelyeket az operációs rendszer „ismer” – azaz rendelkezik az adott típusra vonatkozó társítási listával –, az Intézőben nem jelennek meg. Ezt az „Eszközök” menüpont „Mappa beállításai” parancsának „Nézet” fülén lehet felülbírálni („Ismert fájltypus kiterjesztésének elrejtése”), és ugyanezen párbeszédablak „Fájltípusok” füle szolgál a társítások áttekintésére, módosítására. (Magát a társítást egyébként közvetlen módon is elvégezhetjük a „Fájl” menüpont vagy a gyorsmenü „Társítás” parancsának használatával – ugyanez a parancs szolgál a többszörös társítással rendelkező típusok esetében az aktuálisan használni kívánt alkalmazás kiválasztására is.)

Természetesen a társítás művelete csak akkor alkalmazható eredményesen, ha az adott típushoz hozzárendelt alkalmazás képes feldolgozni az állomány tartalmát: egyrészt amennyiben olyan típussal találkozunk, amelynek a kezeléséhez nem rendelkezünk megfelelő programmal, azt társítva sem fogjuk tudni (érdemben) megnyitni, másrészt attól, hogy egy állományhoz másik alkalmazást társítok, attól annak a tartalma még nem fog megváltozni (ha pl. a JPG típushoz hozzárendeljük a Jegyzettömböt, attól a grafikus állományból nem lesz szöveg...)

3.4.2 Linux

A Linux egy 32/64 bites, POSIX szabványt követő UNIX változat, egy valódi 32/64 bites többfelhasználós (multiuser) és többfeladatos (multitasking) operációs rendszer. Eredetileg csak karakteres felülettel rendelkezett, a mai verziókban már szinte kivétel nélkül van lehetőség grafikus felület kialakítására is. A rendszer kidolgozottsága olyan fokú, hogy egyre több helyen alkalmazzák munkaállomásként vagy hálózati szerverként. Előnye az elterjedtebb operációs rendszerekkel szemben a nagyfokú megbízhatóság és az alacsony ár (léteznek olyan változatai – ezeket nevezzük disztribúciónak – is, amelyek ingyenesek). A Linux nyílt forráskódú és szabad terjesztésű szoftver – ami egyszerre lehet előny (bárki megnézheti, hogy mit és hogyan csinál az operációs rendszer, sőt akár át is írhatja azt a saját igényeinek megfelelően) és hátrány (ez a hozzáférhetőség nyilván rosszindulatú célokra is felhasználható, pl. „kiskapuk” beépítésére...). Manapság egyre több cég foglalkozik Linux disztribúciók fejlesztésével, az ilyen rendszerek elterjedése az utóbbi időben jelentősen megnőtt.

A Linux eredetileg egy egyetemista szakdolgozatának készült, fejlesztésének kezdetén Linus Torvalds a 80386 processzor védett módú (protected mode), feladat-váltó (task-switching) lehetőségeivel szeretett volna megismerni. Az alaprendszer egyes verzióit

verziószámok különböztetik meg egymástól, egy speciális sorszámozási eljárás szerint. A verziószámot három, ponttal elválasztott nem-negatív egész jelzi. Az első a fő verziószám, ami csak a rendszermag lényegét érintő változásoknál vált eggyel nagyobbra. A második szám elég speciális jelentésű: ha páros, akkor stabil, tesztelt kerneltől van szó, amit bárkinek ajánlanak használatra, míg a páratlan szám tesztváltozatot jelöl, amit inkább azoknak javasolnak, akik tesztelni, fejleszteni szeretnék a kernelt, és akiknek nem számít, ha a rendszer néha "elszáll". A harmadik szám pedig kisebb módosításokkor ugrik egyet.

A fent említett nyílt forráskódú jellegből következően a Linuxnak számtalan változata alakult ki az idők folyamán. Az egyes verziók ugyanazon a rendszermagon de eltérő szolgáltatásokkal, nem ritkán eltérő felülettel rendelkeznek. Ezeket a változatokat nevezzük disztribúcióknak. Manapság elsősorban a RedHat és a SuSE disztribúciók tekinthetők elterjedtnek – azzal a kiegészítéssel, hogy mindkét disztribúcióból számos újabb (ezek alapjait átvevő, azokat továbbfejlesztő vagy kiegészítő: Debian, Caldera, stb.) verzió jelent meg és létezik. Ezek a disztribúciók a könnyű telepíthetőséggel, a fejlett (és igény szerint testre szabható) csomagkezeléssel nyújtanak többet a hagyományos LINUX verziókhoz képest.

Külön említést érdemelnek a „load-and-run” típusú disztribúciók: az operációs rendszer méretcsökkenése lehetővé tette az egyetlen optikai adathordozóra elhelyezhető – **és betöltés után működőképes!** – operációs rendszerek megjelenését, ilyen irányú fejlesztésekre a KNOPPIX disztribúciók mutatnak jó példát.

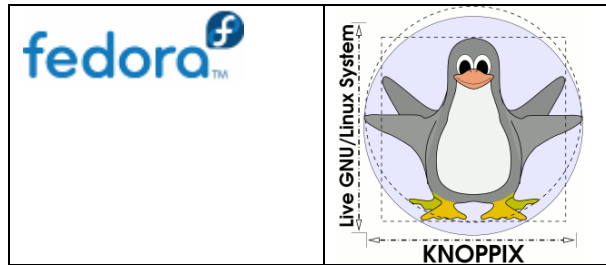
A LINUX tehát *nem egy operációs rendszer*, sokkal inkább egy „operációs rendszer-család” vagy inkább számítástechnikai (jelen esetben a szó alatt a számítógép kezelésének képességét értve) *filozófia*.



Disztribúciók:

<http://www.linux.org/dist/list.html>





Alapfogalmak

A Linux alapú rendszerek működési sajátosságainak áttekintésekor (a Windowsnál bemutatott gyakorlati megközelítés helyett) azt az elvet követjük, hogy azokat a fogalmakat illetve módszereket mutatjuk be, amelyek a Linux alatt a hagyományosan megszokotthoz képest más (pl. kibővített) értelemben szerepelnek. Tesszük mindezt részint azért, mert felhasználói szinten a Linux alapú rendszerek használata (dacára a hagyományosan karakteres felületnek) manapság egyre inkább ugyan azokat a módszereket követi, amelyet a Windows esetében bemutattunk – még akkor is, ha a munkaterület megjelenése eltér.

account: a Linuxban alapfogalom. Tekintve, hogy eredendően többfelhasználós rendszerről van szó, Linux alapú rendszert csak felhasználói azonosítás után tudunk használni. Az egyes felhasználókat egyértelműen megkülönböztető logikai leírás (azonosító) az account. (Magyarul felhasználói számlának is szokták fordítani, a „számla” kifejezés abból ered, hogy a felhasznált processzoridőért, tárolóterületért pénzt számoltak/számolnak fel.)

UID: a felhasználó azonosítója, az „account” fizikai megjelenési formája: a rendszerben minden egyes felhasználónak egy ilyen egyedi azonosítója van.

GID: csoportazonosító. A Linux alapú rendszerekben (mint minden többfelhasználós/hálózati operációs rendszerben) léteznek csoportok, amelyek a felhasználók kezelésnek egyszerűsítését támogató/lehetővé tevő logikai kategóriák. Minden felhasználó be van osztva egy (vagy több) csoportba, a GID annak a csoportnak az azonosítója, amelybe a felhasználó tartozik.

EUID (*effektív user id*): általában egyenlő az UID-del, de bizonyos esetekben (ld. jogosultsági rendszer „setuid-bit”-jei – ld. később!) más is lehet. Ilyen módon egy adott folyamatnak több jogot lehet adni, mint ami a folyamat elindítójának van.

EGID (*effektív group id*): mint EUID, csak a csoport-azonosítóra.

root: kitüntetett szerepű felhasználó, alapértelmezés szerint minden joggal rendelkezik (*rendszergazda, szuperfelhasználó, supervisor*).

process (job): a Linux rendszerben a process a legkisebb párhuzamos feldolgozásra alkalmas egység. Az aktuálisan futó program egy process-t határoz meg, ebből következően minden futó programhoz a rendszer egy process-t rendel, amely rendelkezik egy azonosítószámmal.

PID: folyamat-azonosító. A felhasználókhoz hasonlóan – a Linux filozófiája szerint – a rendszer valamennyi erőforrás-használója (*így a folyamatok is!*) egyértelmű azonosítóval (sorszám) rendelkeznek.

multitask: több feladat egyidejű végrehajtását jelenti. Egy processzorral rendelkező számítógépeken az egyidejű végrehajtás csak látszólagos, hiszen a processzor csak egy feladattal tud foglalkozni egyszerre (*ld. folyamatkezelés!*). A legkisebb egység amely párhuzamos feldolgozásra kerülhet - a *processz*. A feladatok váltogatását az ütemező végzi, amely különböző stratégiák szerint dolgozhat (*ld. folyamatkezelés!*). A Linux prioritási

szinteket használ lehetővé téve, hogy a felhasználó megváltoztassa a saját processzeinek a prioritását.

A Linux *preemptív multitaszkos* operációs rendszer, ami azt jelenti, hogy amikor egy adott folyamat számára kijelölt időszak letelt, akkor a kernel megszakítja a folyamat futását, és másik folyamatnak adja át a vezérlést. Az operációs rendszer nem teszi lehetővé, hogy egy folyamat a végtelenségig magánál tartsa a vezérlést, és így megakadályozza a többi folyamat futását. Azonban a prioritási soron belül lehetőség van a szokásos (Round robin) ütemezés helyett FIFO ütemezés kérésére is - ezáltal szoft-real-time ütemezést is meg lehet valósítani. Ilyenkor a rendszer nem veszi el a futás jogát a processztől, csak ha az lemond róla. Azonban a Linux nem real-time operációs rendszer (de van ilyen irányú fejlesztés is), és ez azt jelenti, hogy több futó folyamat esetén bizonyos időközönként mindegyikre rákerül a vezérlés, azonban a két aktív állapot között eltelt időre nincs szigorú időkorlát.

multiuser: több felhasználó egyidejű kiszolgálását jelenti. Ez nem kifejezetten fájlok megosztását jelenti, hanem inkább programok futtatását. Tehát egy gépre több ember jelentkezhet be egyszerre, és egyszerre tudnak dolgozni anélkül, hogy zavarnák egymás munkáját. Ez maga után vonja azt, hogy a rendszernek meg kell tudnia különböztetni egymástól a felhasználókat. Minden felhasználóhoz előre definiált jogok és engedélyek tartoznak, amelyeket a rendszer mindig ellenőriz, amikor a felhasználó szeretne hozzáférni valamihez. Ez szükséges az erőforrások megfelelő szétosztásának, és a biztonságnak az érdekében.

memóriakezelés sajátosságai: a modern operációs rendszerek képesek arra, hogy látszólag több memóriát biztosítsanak a programoknak, mint amennyi fizikailag a rendelkezésükre áll (*ld. memóriakezelés*). Annak érdekében, hogy a merevlemez virtuális memóriakezelésre használni lehessen, *swapfájl* vagy *swappartíciót* kell létrehozni. Linux alatt ez dinamikusan, menet közben is változtatható, tehát az operációs rendszer leállítása nélkül lehetőségünk van a virtuális memória méretének megváltoztatására. Egy swap partíció mérete maximum 128 MB lehet, de használhatunk belőle többet is egyszerre, maximum 16 darabot.

gyorsítótár (buffer cache): a memóriakezeléshez szorosan kapcsolódó fogalom, a Unix rendszerek lemezeléréshez használt gyorsítótárja, amelyet a kernel kezel. Linux alatt a buffer cache mérete dinamikusan változik a rendszer terhelésétől függően - az éppen szabad fizikai memória egészét erre a célra használja. A lemezre írandó anyag is először a buffer cache-be kerül, és vagy egy megadott idő elteltével (30 másodperc) íródik ki a lemezre, vagy pedig akkor, ha a rendszer számára elegendő mennyiségű anyag összegyűlik a cache-ben. *Ezért fontos, hogy ne kapcsoljuk simán ki a számítógépet, hanem mindig szabályosan állítsuk le a rendszert a megfelelő parancsokkal.*

demand paging: szintén a memória-kezeléshez kötődő teljesítményt növelő módszer. Azt jelenti, hogy egy futtatható fájl végrehajtásakor nem az egész fájl töltődik be a memóriába, hanem mindig csak azok a lapjai, amikre a végrehajtás során éppen szükség van. Mivel minden programnak vannak olyan részei melyek csak egyszer (vagy akár egyszer sem) futnak le, ezeket a részeket vagy be sem tölti a rendszer, vagy miután lefutottak felszabadítja az általuk elfoglalt memóriaterületet.

osztott kódkönyvtárak: használatának alapelve az, hogy a programok C nyelven íródnak, és valószínűleg sokban van olyan függvény, amely más programokban is előfordul. Ezeket felesleges lenne minden programmal a memóriába tölteni, elég egyszer, és meg kell mondani a programoknak, hogy hol keressék ezeket a függvényeket a memóriában. Ezt csinálja a *dinamikus linker*, amely a programokba beépített programrészletnek segítve

gondoskodik a függvények megtalálásáról, illetve a memóriába töltésükről, amennyiben még nem lennének betöltve.

shell: a rendszer parancsértelmező komponense, de *nem alapértelmezett része az operációs rendszernek!* Minden felhasználó bejelentkezésekor egy parancsértelmező indul el. A parancsértelmező szabványos bemenete és kimenete a terminál. Egy *promptot* jelenít meg (ami egyénileg beállítható), jelezve, hogy készen áll a feladatok végrehajtására. Ha a felhasználó elindít egy parancsot, akkor a parancsértelmező elindít egy gyermekprocesszt, ami lefuttatja a kért parancsot. A gyermekprocessz futása közben a parancsértelmező annak a megszűnésére vár. A gyermekprocessz megszűnésekor a parancsértelmező újra megjeleníti a promptot.

fájlrendszer: a fájlrendszer a lemezen tárolt adatok kezelhetőségét biztosítja. Annak érdekében, hogy a Linux más fájlrendszerek támogatását is tudja biztosítani, a kernel és a fájlrendszerek között létezik egy szint, amelyet *virtuális fájlrendszernek* neveznek. Ez rendelkezik azokkal a rutinokkal, amelyek szükségesek egy fájlrendszeren történő műveletvégzéshez. Ez biztosítja a különböző fájlrendszerek közötti átjárhatóságot, mert a felhasználónak nem is kell tudnia, hogy milyen fájlrendszeren történik a műveletvégzés, csak kiadja a parancsot, és az érintett fájlrendszer-kezelő lefordítja a megfelelő fájlrendszer-hívásokra.

(hozzáférési) jogosultságok: a fájlokhoz tartozó olyan jogosultságok, amelyek meghatározzák, hogy melyik felhasználó melyik fájlra milyen műveletet hajthat végre.

kernel: az operációs rendszer magja, szerepe a programok és a hardver közötti kapcsolattartás. Memóriaterületet biztosít az összes futó processz (program) számára, ezenkívül egy kényelmes felületet nyújt a programoknak, amelyen keresztül kommunikálhatnak a hardverrel.

kernel frissítés: lényegében az operációs rendszer újabb szolgáltatásokkal történő kiegészítése, illetve a meglevő szolgáltatások javítása. Az újabb kernelnek általában több eszközmeghajtójuk van, általában jobb a processz-kezelésük, gyorsabb vagy stabilabb, és javítja a korábbi verzió ismert hibáit.

eszközök: A Linux rendszerek *mindent fájlként kezelnek:* a merevlemezeket, a terminálokat, az audio-eszközöket, meghajtókat, stb – az elv lényege, hogy ilyen módon az eszközöket (bármilyen eszközről – perifériáról – legyen is szó!) a fájlokhoz hasonló módon tudják elérni. A /dev könyvtárban találunk meg minden fájlt, ami az eszközökhöz tartozik. Az eszközöknek két fajtájuk van, karakter- és blokkeszközök. *Karakterorientált* eszköz például a terminál, a soros port. *Blokkorientált* eszközök például az adattároló eszközök. Az eszközöket két szám jellemzi: a fő- és az aleszközsám. A főeszközsám adja meg az eszköz típusát. Ha ugyanabból a típusból több eszköz is van, akkor ezeket az aleszközsám különbözteti meg egymástól.

démonok: speciális processzek, amelyek a háttérben futnak, párhuzamosan más programokkal. Az operációs rendszer nagy egységei önálló programként így futnak. Konfigurációjuk módosítása esetén anélkül újraindíthatóak, hogy magát az operációs rendszert is újra kellene indítani. Jellemző példák: nyomtató démon: lpd; rendszernaplózás: syslogd; időzítési feladatok: cron, at; Internet démon: inetd; fájlrendszer-kezelés: rpc.nfsd. (Fontos megjegyezni, hogy az egyes szolgáltatásokhoz egy jól meghatározott *port-szám* tartozik, amelyeken keresztül el lehet érni az adott szolgáltatást: a rendszer külső védelme szempontjából ennek igen nagy jelentősége van!)

konzol: (a Linux alapértelmezése szerint) a számítógépes rendszernek nem része a monitor és billentyűzet, csak hozzá lehet csatlakoztatni, éppen ezért szükség van egy

kommunikációs felületre – ez a konzol: a számítógép és a felhasználó közötti kommunikáció felülete. Alapértelmezés szerint ide érkeznek azok az üzenetek, amiket a rendszer küld, biztosítja a parancskiadást (*ld. parancsértelmező*) és létezik grafikus felülete is (xconsole). A Linux lehetővé teszi, hogy a fizikailag egyetlen számítógépet virtuálisan úgy használjuk, mintha több különböző számítógépen (terminálon) dolgoznánk. Természetesen egyszerre csak egy terminálon tudunk dolgozni, de az elindított programok párhuzamosan futnak egymással. Az egyes *virtuális konzolok* között váltogathatunk az Alt-Fx (x=1...12) billentyűkombinációval.

Alaptevékenységek

Felhasználók bejelentkezése, kijelentkezése, leállítás

A több-felhasználós rendszerekben kulcskérdés az egyes felhasználók egymástól való elkülönítése, valamint a felhasználói jogosultságok kezelése. A felhasználói azonosítás folyamata a Linux-ban alapvetően kétféleképpen valósulhat meg aszerint, hogy a bejelentkezés helyi vagy távoli számítógépről történik.

(Az egyszerűség kedvéért a karakteres terminálon történő bejelentkezés menetét ismertetjük, a grafikus felület használata esetén a műveletek azonosak, csak a megjelenésben van különbség).

Lokális bejelentkezés során a "getty" program várja a felhasználók bejelentkezését (többek között kiírja a "login" feliratot). Amikor a "getty" megkapja, hogy a felhasználó bejelentkezik a rendszerbe, akkor meghívja a "login" programot. Ez a program megváltoztatja a felhasználó környezetét, és meghívja a beállított "shell"-t. A "login" program a "/etc/passwd" fájlt használja. A "/etc/passwd" fájlban meghatározhatunk más programot is, nem csak "shell"-t, ami a felhasználó bejelentkezésekor elindul. A belépés a rendszerbe csak a megfelelő felhasználónév ("username") és jelszó ("password") megadásával lehetséges. Ennek kiosztása a "root" jogosultsággal rendelkező rendszergazda feladata. A felhasználónak csak a saját jelszó megváltoztatására van jogosultsága.

A hálózaton keresztüli belépés annyiban különbözik az előbb ismertetett módszernél, hogy a bejelentkezés során a két számítógép közötti kapcsolat egy kommunikációs folyamatban épül fel. Ennek megfelelően az "inetd" démon fogadja az felmerült igényeket, a biztosított szolgáltatások körét a "/etc/inetd.conf" fájlban határozhatjuk meg. Általánosságban azt mondhatjuk a rendszer alaphelyzetben mindig több szolgáltatást engedélyez, mint amit szeretnénk publikussá tenni. Ezek korrekt beállítása tehát kulcskérdés, mielőtt szerverünket kiajánljuk. A szolgáltatások elérését a "/etc/hosts.allow" és a "/etc/hosts.deny" fájlokkal tudjuk szabályozni.

A kijelentkezés legegyszerűbben a shell process lezárásával történhet.

A rendszer leállítása szigorú szabályok szerint történik. A leállítási folyamatot a "shutdown" program végzi. Minden process-nek tudni kell, hogy most rendszerleállítás jön, tehát le kell zárnia minden nyitott fájlt kilépés előtt. Ha ezt nem tenné meg, a nyitott fájlok sérülhetnek vagy elveszhetnek a leállítás során. Használhatjuk a "halt" parancsot a leállításhoz és a "reboot" parancsot az újraindításhoz.

Karakteres felület legfontosabb parancsai

Habár a Linux rendszerek is rendelkeznek grafikus felülettel, hagyományosan karakteres operációs rendszerről lévén szó tekintsük át röviden a legfontosabb parancsokat.

Könyvtárakkal, állományokkal kapcsolatos alapvető parancsok:

- **mkdir** [kapcsolók] könyvtárnév: könyvtár létrehozása.
- **pwd**: az aktuális könyvtár teljes elérési útvonalát adja meg.

- **cd** [könyvtárnév]: az aktuális könyvtár kiválasztására (beállítására) használható. (Az alapértelmezett aktuális könyvtár az aktuális felhasználó "home" könyvtára.)
- **ls** [kapcsolók] fájl vagy könyvtárnevek: a fájl- és könyvtár-információk megjelenítése.
- **cp** [kapcsolók] fájlnev fájlnev: állományok másolása.
- **mv** [kapcsolók] forrás cél: fájlokat vagy könyvtárakat átnevez vagy átmozgat.
- **rm** [kapcsolók] fájlok: egy vagy több fájl vagy könyvtár törlése.
- **ln** régi-fájlnev új-fájlnev: új nevet hoz létre a fájl számára (tulajdonképpen egy hivatkozási név, ún. link („csatolás”) jön létre, fizikai másolás nem történik).
- **tar** [kapcsolók] [archív] [fájlok]: beépített tömörítőprogram, kapcsolói határozzák meg, hogy be- vagy kitömörítést hajtson végre.

Jogosultság kezelésére szolgáló parancsok

- **chmod** [mód] név: állomány és könyvtárak hozzáférési engedélyek megadása, elvétele.
- **chown** [kapcsolók] tulajdonos [csoport] fájl: megadott fájlok tulajdonosát és csoportját módosíthatjuk ezzel a parancsal.

Process-kezelés parancsai

- **ps** [kapcsolók]: aktuálisan futó "process"-ek listája
- **kill**: futó "process" befejeztetése, leállítása.
- **shutdown** [kapcsolók] időpont [üzenet]: a rendszer futási szintjét módosítja, vagy lezárja a rendszert. Az időpont és a figyelmeztető üzenet argumentumként átadható. Azonnali lezáráshoz az időpont értéke: now.
- **su** [-] [felhasználó] [argumentum]: a "shell" programot indítja másik felhasználóval. Olyan terminálon való bejelentkezésre használható, amelyet másik felhasználó használ. Ha nem adjuk meg a felhasználót akkor egy root "shell" programot indít. Az új "shell" programot az exit vagy [Ctrl-d] beírásával zárhatjuk le.

Egyéb, gyakran használt parancsok

- **mount** [kapcsolók] [eszköz] [csatolási_ hely]: új fájlrendszer csatolása egy könyvtárstruktúrába. A meg nem adott paramétereket a parancs az "/etc/fstab" fájl megfelelő bejegyzéseiből veszi. (Fájlrendszer leválasztása: umount.)
- **find** útvonal [opciók]: az útvonalban megadott katalógusból kiindulva, rekurzívan végigjárva a katalógusokat az opcióiban megadott tulajdonságú állományt megtalálja.
- **grep**: egy meghatározott szövegmintát keres egy vagy több fájlban.
- **more**: képernyő oldalanként jeleníti meg a megadott fájlt.
- **man**: kézikönyv az egyes parancsok szerkezetéről és működéséről („súgó"). A legfontosabb és leggyakrabban használt parancs. Az oldalak a "/usr/man" könyvtárban találhatók vagy a MANPATH környezeti változóban megadott könyvtárban. Kilépés a "man"-ból "q" billentyű lenyomásával történik. Lapozás előre [Space], vagy "f" billentyűvel, vissza "b" billentyűvel.

3.4.3 Hálózati operációs rendszerek

Az operációs rendszerek között külön kategóriát képviselnek a hálózati operációs rendszerek. A hálózatokkal a későbbiekben még részletesen foglalkozunk, jelen fejezet

szempontjából a helyi hálózatok (viszonylag kis kiterjedésű, jellemzően egy szervezet számítógépeit összekapcsoló hálózati rendszer) kezelésével kapcsolatos rendszerek érdekesek. Hálózatot alapvetően **erőforrás-megosztási céllal, kommunikációs közegként vagy az adatok biztonságának növelése érdekében** alakíthatunk ki – helyi hálózatok esetén mindhárom szerep egyformán fontos, éppen ezért nem véletlen, hogy a hálózati operációs rendszerek már olyan széleskörű hálózati szolgáltatásokkal rendelkeznek, hogy minden jellemző hálózati tevékenység megvalósítható segítségükkel.

Hálózati modellek

A hálózatba bekapcsolt számítógépeket **hálózati csomópontnak** nevezzük, és elsődlegesen a hálózatban betöltött szerepük szerint szokás *megkülönböztetni* őket (bár ezek csupán fogalmi különbségek, nem biztos, hogy működésbeli vagy teljesítménybeli eltéréseket is jelentenek). Ilyen értelemben beszélhetünk **hálózati kiszolgálókról** (*szerver*: olyan számítógépek, amelyek valamilyen erőforrást vagy szolgáltatást nyújt (megoszt) a csomópontoknak) és **munkaállomásokról** (*ügyfél- vagy kliens-számítógép*: a szerver által nyújtott szolgáltatásokat igénylő vagy használó számítógép).

A helyi hálózatok sajátossága, hogy ugyanazon számítógép akár többféle szerepben is képes működni a hálózatban. A hálózati csomópontoknak a hálózatban elfoglalt lehetséges szerepei alapján két jellemző helyi hálózati modellt különböztethetünk meg:

- az **egyenrangú** („*peer-to-peer*”) modellben a hálózat **valamennyi számítógépe képes szolgáltatásokat nyújtani** (erőforrásokat felajánlani) a többi számítógép számára és **ugyanakkor használni** (igényelni) **más csomópontok által kínált szolgáltatásokat**. Erre a modellre tehát az jellemző, hogy nincs kitüntetett számítógép: mindenki lehet szerver és munkaállomás – *akár azonos időben is!*
- a **hierarchikus** (*ügyfél-kiszolgáló, „server-client”*) modellben ezzel szemben **minden esetben van** egy (vagy több) **kitüntetett szerepű számítógép** (a szerver), „aki” a hálózati szolgáltatásokat **kizárólagosan birtokolja** – a hálózat többi számítógépe (a munkaállomások) csak a szerver által felajánlott szolgáltatásokat használhatja. (Ez persze azt is jelenti, hogy amennyiben két munkaállomás kommunikálni (pl. adatot cserélni,) szeretne egymással, ezt is csak a szerveren keresztül tehetik meg!) Ebben a modellben további kategorizálásra ad lehetőséget, ha
 - azt is megvizsgáljuk, hogy **a szerver** (a hálózat kiszolgálásán és felügyeletén **túl**) **milyen feladatok elvégzésére** képes: **dedikált szerver** esetében *semmilyen egyéb tevékenységet* nem folytat (azaz felhasználói értelemben vett „munkát” nem végezhetünk rajta), a **nem dedikált szerver** a hálózat vezérlésén túl teljes értékű munkaállomásként is működik („kettő az egyben”),
 - azt is megvizsgáljuk, hogy **a munkaállomás** mennyire alkalmas **önálló** (hálózati kapcsolat nélküli) **munkavégzésre**: azokat a munkaállomásokat, amelyek csak a szerver erőforrásai birtokában tudnak valamilyen feladatot ellátni (megkülönböztetendő a hálózatba kapcsolt önálló számítógépektől) **terminálnak** nevezzük (ebben a megközelítésben a kiszolgálót is szokás „host”-ként emlegetni)

A két modell között (természetesen) számos eltérés van, azonban mégsem állíthatjuk azt, hogy az egyik vagy a másik jobb lenne. A használhatóságukat minden esetben az igények és a rendelkezésre álló források határozzák meg: **az egyenrangú hálózat** kiépítése olcsóbb, kihasználhatósága hatékonyabb (nem kell egy plusz – általában a munkaállomásokhoz képest) nagyobb hardverigénnyel rendelkező (és adott esetben effektív munkavégzésre nem is használható) számítógépet beszerezni), **a hierarchikus hálózat** felügyelete egyszerűbb (egyetlen számítógépen (a szerveren) kell csak a beállításokat elvégezni), nagyobb a rendszer biztonsága (centralizált mentés, központi vírus- és betörésvédelem), rugalmasabb

munkavégzést tesz lehetővé (felhasználók nagyobb szabadsággal használhatják a hálózat bármely munkaállomását).

A megismert operációs rendszerek közül a Windows hagyományos változatai (95, 98, XP) egyenrangú hálózati rendszerbe szervezhetőek, míg a szerverváltozatok (2000, 2003) segítségével akár hierarchikus hálózat is kialakítható. A Linux alapú rendszerek alapértelmezés szerint mindkét modellt támogatják. Vannak továbbá olyan operációs rendszerek, amelyek kimondottan hálózati célok kiszolgálására készültek, a legismertebb ilyen a Novell cég Netware nevű rendszere. (Az, hogy az egyes rendszerek között mennyire nem egyértelműek a határvonalak, jól példázza, hogy a Novell a legújabb operációs rendszerét már a Linux egyik disztribúciójára (SuSE) alapozva készítette el...)

Hálózati szolgáltatások

A hálózati operációs rendszerek többlétszolgáltatásai (a hagyományos operációs rendszerekhez képest) a több felhasználói igény kiszolgálása köré szerveződnek. A legfontosabbak ezek közül:

- **erőforrások megosztása: hozzáférés biztosítása a számítógép** valamely *lokális* (saját: fizikailag az adott számítógéphez csatlakozó) **erőforrásához** (és a helyi hálózatok esetében ez szinte bármilyen típusú erőforrás lehet: *hardveres* (memória, háttértár, nyomtató) vagy *softveres* (program vagy adat)) **a hálózat más számítógépei számára**. A megosztott erőforrások használata a felhasználó felé transzparens módon valósul meg – azaz a felhasználó nem érzékel különbséget aközött, hogy saját számítógépével vagy a hálózaton található erőforrással dolgozik. A helyi hálózatokban leggyakrabban megosztott **hardveres erőforrások** tehát a **háttértárak** (merevlemez, illetve az itt tárolható állományok és könyvtárak) és a **nyomtatók**.
- **jogosultságok kezelése:** a megosztások kezelésével egyidejűleg felmerül egy olyan probléma, amely egy-felhasználós környezetben nem jelentkezik: nevezetesen a **hozzáférés** kérdése. Egyrészt, a közös tárolásnak csak akkor van értelme, ha a kiszolgáló által biztosított háttértárat minden, a hálózatba bekapcsolt számítógépről elérhetjük – másrészt viszont jogos elvárás az is, hogy bizonyos anyagokat ugyanolyan bizalmasan (másoktól védetten) kezelhessünk, mintha a saját számítógépünkön tárolnánk. Egyszerre legyen tehát nyilvános és privát... **Jogosultságoknak** nevezzük a hálózat azonosított felhasználói és az egyes erőforrásokkal végezhető tevékenységek körét.
- **hozzáférés:** a hálózati megosztások közös jellemzője, hogy azonosításukra olyan módszert kell alkalmazni, amelyet a hálózat valamennyi számítógépe képes értelmezni: legegyszerűbben a megosztott erőforrások egyedi névvel rendelkeznek. A névadás szabályai esetenként változhatnak, azonban pont az egységesség miatt létezik egy szabvány, az UNC, amivel szinte minden rendszerben azonos módon hivatkozhatunk megosztott erőforrásra. Az **UNC szintaxisa** a következő: **\\szerver\erőforrás**, ahol a „szerver” annak a számítógépnek a **hálózati azonosítója**, ahol közvetlenül (fizikailag) létezik a megosztott erőforrás („aki megosztja”), az „erőforrás” a megosztott objektum **hálózati megnevezése** (aminek nem kell megegyeznie a szerveren használt (tényleges) elnevezéssel!).
- **felhasználói információk kezelése:** a hálózati operációs rendszerben alapvetés, hogy egy olyan rendszerről van szó, amelyet több felhasználó használ. Ez pedig számos kérdést vet fel:
 - meg kell tudni különböztetni az *egyes felhasználókat egymástól*,

- o de célszerű a hálózat *jogosult felhasználóit* megkülönböztetni azoktól, akik nem használhatják az adott hálózat szolgáltatásait;
- biztosítani kell az *időben azonos hozzáférés* lehetőségét,
 - o de egyszerre *legfeljebb csak egy bizonyos számú* felhasználó kérését kiszolgálva (sem a számítógép műveletvégző képessége, sem az adatátviteli csatorna szélessége nem végtelen...),
- célszerű *külön választani* az egyes felhasználók által elérhető *erőforrásokat*,
 - o de úgy, hogy közben lehetőséget kell adni a felhasználóknak arra, hogy igény szerint akár *kicserélhessék adataikat egymással*, stb.

A gyakorlatban a felhasználók megkülönböztetésére valamilyen szöveges azonosítót szokás használni, ez az adott felhasználó adott hálózatra érvényes **felhasználói neve** („*user name*”). A felhasználói fiókok biztonságának fokozása érdekében a hálózati rendszerek (szinte minden esetben) az azonosításhoz egy *ellenőrző kódot* is megkövetelnek (**jelszó**, „*password*”). Ez a jelszó (elvileg) csak a felhasználó által ismert, és ily módon garantálja, hogy az adott felhasználói név alatt ténylegesen csak bizonyos személy használhassa a hálózat erőforrásait. *(Megjegyzés: az olyan rendszerekben, ahol az egyértelmű azonosítás fontos követelmény, a felhasználói név és jelszó (ismeret alapú azonosítás) helyett gyakran alkalmaznak ún. birtoklás alapú azonosítási rendszereket (ilyenek pl. a bankkártyák), illetve ma már léteznek tulajdonság alapú azonosítási rendszerek (biometria azonosítók, pl. ujjlenyomat) is.)*

Azt a *folyamatot*, amely során a felhasználó azonosító adatait megadja illetve azt a hálózati kiszolgáló ellenőrzi, **azonosításnak** („*authentication*”), a sikeres ellenőrzést követően a hálózati kapcsolat kiépítését és a megfelelő hálózati szolgáltatások helyreállítását pedig **bejelentkezésnek** („*log in*”) nevezzük. Természetesen a hálózati munkavégzés befejezésekor a *hálózati kapcsolatot* illik (ne fogjunk feleslegesen erőforrásokat mások elől!) és célszerű (biztonsági okokból) *lebontani*, ez a **kijelentkezés** („*log out*”) folyamat.

Ahhoz, hogy mindezeket az adatokat a hálózati operációs rendszer ellenőrizni tudja, nyilván kell őket tartania. A különböző operációs rendszerek számos (bizonyos vonásaiban hasonló, másokban jelentősen eltérő) nyilvántartási módszert alkalmaznak, manapság a **címtár alapú nyilvántartó rendszerek** tekinthetők a legkorszerűbbnek (és lassan a legelterjedtebbnek is, de ezzel kapcsolatban meg kell jegyezni, hogy egy – a hálózati operációs rendszerek korához képest – új technológiáról van szó, így egy régebb óta működő (és nem frissített) rendszerben könnyedén találkozhatunk nem ilyen elven működő nyilvántartással is).

A **címtárak lényege**, hogy egy kvázi-szabvány rendszerben (a címtárak jellemzően az LDAP vagy az X.500-as szabványok definícióit követik, de nem maradéktalanul valósítják meg a szabványokban foglaltakat) együtt tárolják a hálózat működéséhez szükséges valamennyi adatot (úgy a felhasználókról, mint a hálózat összes erőforrásáról) és az adott szervezet számára jelentőséggel bíró egyéb információkat, ily módon a hálózati nyilvántartás bármilyen más nyilvántartás alapját képezheti illetve képes más rendszerek számára adatot szolgáltatni.

A **hálózati szolgáltatások használatának** (szabványos) lépései tehát a következők:

1. a munkaállomás (amely rendelkezik a megfelelő hálózati erőforrás eléréséhez szükséges valamennyi – úgy hardver, mint szoftver – eszközzel) által nyújtott felületen (kliens-program) a **felhasználó jelzi igényét** a szolgáltatást nyújtó számítógép (szerver) felé;
2. a **szerver** (amennyiben az erőforrás használata nem korlátlan) **elkéri** a munkaállomástól (aki pedig a felhasználótól) a **felhasználó azonosító adatait** (a

nevet és a jelszót), ezek az adatok (általában) titkosítva továbbítódnak a munkaállomástól a szerverhez;

3. **a szerver ellenőrzi** (összehasonlítja a kapott adatokat a saját nyilvántartásában tárolt információval), hogy a megadott név érvényes-e, a hozzá tartozó jelszó megfelelő-e és végül, hogy az így módon azonosított felhasználónak van-e joga a kért erőforrás használatához – háromszoros **pozitív válasz** esetén **elküldi** a kért adatokat a kliensnek (azaz *engedélyezi a hozzáférést*).

Novell Netware

A különböző hálózati operációs rendszerek esetében az első (és legfontosabb) kérdés, hogy a szerver milyen szolgáltatás(oka)t nyújt a hálózat felhasználói számára. Gyakorlatilag az IntraNet fogalmának megjelenésétől azonban ebben a tekintetben nincs lényeges különbség az egyes megvalósítások között: az alapvető helyi hálózati feladatokon (fájl- és nyomtatógosztás) túl minden LAN operációs rendszer számos többlétszolgáltatással rendelkezik, részint beépített szolgáltatások, részint (külön megvásárolható) bővítmények formájában.

A Netware egy **kliens-szerver alapú, dedikált szerveres hálózati operációs rendszer**. A szerveren (természetesen) Netware operációs rendszer fut (a Netware a szervert futtató számítógép elindítására (boot-olására) és a hálózati operációs rendszer magjának (*server.exe*) betöltésére valamilyen DOS – az újabb verziókban Linux – verziót használ). A munkaállomásokon bármilyen operációs rendszer fut, a hálózati kérések kezelését a Netware hálózati programja, a **Netware kliens** (*Novell Netware Client*) végzi, szinte minden jelentős operációs rendszerhez készült kliensprogram.

A Netware operációs rendszerek a Novell hagyományos állomány- és nyomtatószerver szolgáltatásain túl számos, a hálózatos környezet hatékonyabb kihasználását lehetővé tevő szolgáltatást tartalmaz, melyek közül a legfontosabbak a következők:

- **Novell eDirectory**: a hálózati erőforrások központosított nyilvántartási rendszere,
- **ConsoleOne**: a szerverkonzol grafikus felületű kiterjesztése, lehetővé teszi a szerver és a hálózat távoli felügyeletét és menedzselését
- **iFolder**: adatszinkronizációs és backup megoldás egyben, lényege, hogy a felhasználók a szerveren tárolt állományait (akár böngészőprogramon keresztül, azaz szinte) bárhol elérhetik, és az esetleges változások automatikusan átvezetődnek a szerverre, így az adatok mindenhol ugyanazt az állapotot tükrözik
- **iPrint**: a hagyományos hálózati nyomtatási szolgáltatások kibővített funkcionalitású eszköze, amely segítségével a Netware szerver képes hálózati nyomtatóként kezelni a szerverhez csatlakozó (lokális), a hálózat valamely munkaállomásához csatlakozó (távoli) vagy akár Interneten keresztül elérhető nyomtatót is
- **Native File Access Pack**: az ismertebb operációs rendszereket (Windows, Unix, Macintosh) futtató munkaállomások számára a szerver kötetének közvetlen elérését biztosító komponens – az ilyen operációs rendszerrel rendelkező számítógépekre (amennyiben csak a Netware-en tárolt állományokat akarjuk elérni) nem kell külön kliensprogramot telepíteni
- **Novell Storage Service (NSS)**: a Netware 5-ben bemutatott, de csak a Netware 6-tól alapértelmezett állományszervezési mód, amely elméletileg korlátlan méretű (háttér)tárolórendszer kialakítását és kezelését teszi lehetővé
- **Novell Cluster Services**: 2 Netware szerverből álló hálózati fürt („cluster”) kialakítását támogató komponens
- **többprocesszoros (SMP) működés**: egy Netware szerver (megfelelő alaplappal) akár 32 processzort képes együttesen kezelni

- **NetWare Web Access:** teljes körű Internet szolgáltatást biztosító eszközök gyűjteménye. Részei: DNS- és DHCP-szerver, Netware AMP (Apache, MySQL, PHP: az Internet legjelentősebb webes (nyílt forráskódú) alkalmazásainak integrált és Netware-re optimalizált változata), Internetes fejlesztő eszközök (Java, XML, CGI, WebDAV) támogatása. Érdekessége, hogy lehetővé teszi az https protokollon alapuló távoli szerverfelügyeletet is (azaz a rendszergazda bárholonnan (egy böngészőprogramon keresztül) elvégezheti mindazokat a tevékenységeket, amelyek hagyományosan csak a konzolon kiadható parancsokkal oldhatóak meg).

Ellenőrző kérdések:

1. Definiálja az operációs rendszer fogalmát!
2. Sorolja fel az operációs rendszerek alapvető feladatait!
3. Hogyan csoportosíthatók az operációs rendszerek? Mondjon példákat!
4. Mit jelent az operációs rendszerek „több-feladatos” és „több-felhasználós” jellege?
5. Hogyan történik az utasítások kiadása karakteres illetve grafikus felületen?
6. Mit nevezünk állománynak, illetve könyvtárnak?
7. Melyek a legfontosabb állomány-tulajdonságok?
8. Melyek a Windows operációs rendszerek grafikus felületének általános jellemzői?
9. Mutassa be a Windows operációs rendszerek állomány-kezeléssel kapcsolatos szolgáltatásait!
10. Mit értünk LINUX disztribúció alatt? Mondjon példákat!
11. Magyarázza meg a következő fogalmakat a LINUX operációs rendszerben: process, konzol, démon!
12. Melyek a LINUX operációs rendszer legfontosabb állomány-kezelő parancsai? Mutassa be egy-két parancs működését is!
13. Melyek a helyi hálózatok alapvető modelljei? Hasonlítsa össze az egyes modelleket (előnyök, hátrányok)!
14. Mit jelent a „dedikált szerver” kifejezés?
15. Melyek a legfontosabb feladatai egy hálózati operációs rendszernek?

Irodalomjegyzék

- [1] Olajos – Magó: *Operációs rendszerek* (Számalk, 2003)
 [2] Knapp: *Operációs rendszerek* (LSI, 1998)
 [3] Magó– Nagy: *Hálózati felhasználói ismeretek* (Számalk, 2003)

4. IRODAAUTOMATIZÁLÁS, FELHASZNÁLÓI ALKALMAZÁSOK

4.1. Szövegfeldolgozás

A számítógépes szövegfeldolgozást, szövegszerkesztést ma már nagyon sok helyen alkalmazzák. Legismertebb fajtái az Microsoft Office programcsomagok részét képezik. Egy másik népszerű hasonló funkciókkal rendelkező programcsomag az OpenOffice. A jegyzet terjedelmi határait figyelembe véve a szövegszerkesztő program részletes ismertetésére nincs lehetőség. Ebben a fejezetben csupán néhány funkció kerül bemutatásra, amelyeket a szerző fontosnak tart. A szövegszerkesztő program egyszerűbb funkcióit, a hallgatók középiskolai tanulmányainkból, ismertnek tekintjük.

A bekezdések szerepe

A Word szövegszerkesztőnek a karakterek után nagyságrendben következő szövegegysége a bekezdés. Nagyon sok beállítás bekezdések egészére vonatkozik, ezért célszerű a bekezdéseket egyben kezelni. Ismert, hogy egy bekezdés befejezésére és új bekezdés megnyitására az ENTER billentyű szolgál.

Amikor a felhasználó a sor végére ér a gépeléssel, a szövegszerkesztő automatikusan a következő sor elején kezdi írni a szöveget. Ha a felhasználó mégis azt szeretné a bekezdés egy pontján, hogy egy új sorban folytatódjon a szöveg, bár még nem ért a sor végére, és nem is akar új bekezdést kezdeni, ezt a SHIFT+ENTER billentyűkombinációval érheti el.

A bekezdések határa és még sok más hasznos információ is megtudható, ha az eszköztár *mindent mutat* ikonja be van kapcsolva.

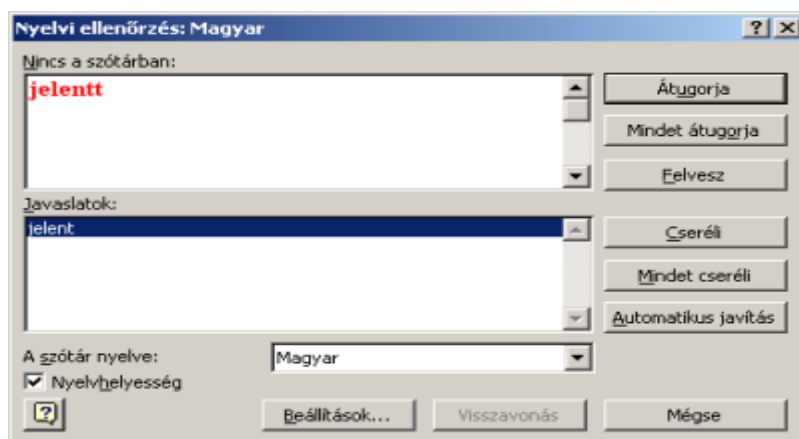


Az automatikus helyesírás-ellenőrző használata

A Word biztosít egy olyan eszközt, amely automatikusan ellenőrzi a szöveg helyesírását, és lehetőséget ad a korrekt javításra is.

Ez az **Eszközök** menü **Nyelvi ellenőrzés** menüponttal vagy az eszköztár **Hibaellenőrzés** ikonjával indítható el. Ekkor megjelenik egy párbeszédablak, amelyik a felmerülő hibákat megmutatja, és lehetőséget ad a számítógép által esetleg felajánlott javítási lehetőségek valamelyikének elfogadására, vagy a felhasználó saját maga javíthatja ki a hibás szót.

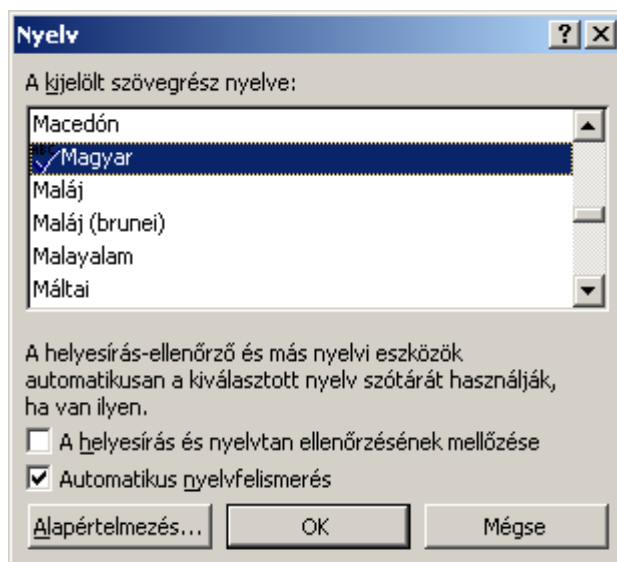
A helyesírás ellenőrzés szövegösszefüggést nem vizsgál, hibák maradhatnak a szövegben, ha az elgépelte szó önmagában is értelmes (4.1. ábra).



4.1. ábra: Nyelvi ellenőrzés ablak

Nyelv megadása

Az Eszközök menü Nyelv funkció Nyelv megadása menüpontban van lehetőség megadni a dokumentum, vagy a kijelölt szövegrész nyelvét. Ez lehetővé teszi, hogy idegen nyelvű szövegrészeket is nyelvtanilag ellenőrizze a szövegszerkesztő. Megadható az is, hogy melyik nyelv legyen az alapértelmezett (4.2. ábra).



4.2. ábra: Nyelv ablak

Tabulátorok

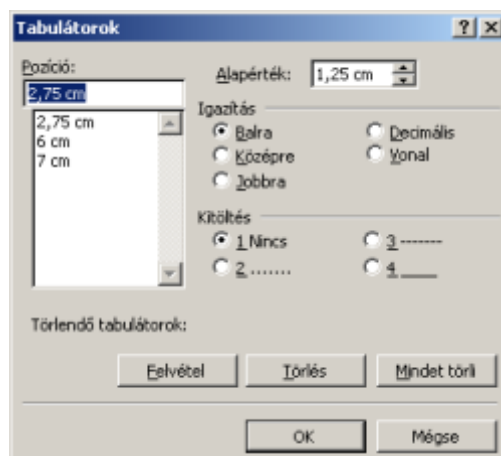
A tabulátor olyan eszköz, amellyel egy soron vagy egy bekezdésen belül a szöveget megadott pozícióba lehet igazítani. A szöveget a TAB billentyűvel a következő tabulátorpozíciókba lehet vinni. A tabulátorok típusai: balra igazító, középre igazító, jobbra igazító, tizedesjelhez igazító.



A tabulátorok legegyszerűbben a vonalzóra kattintva helyezhetők el. A létrehozott tabulátor típusa a vonalzó bal szélén látható ábrával egyezik meg, amely azonban kattintással módosítható. A már létrehozott tabulátorok a Formátum menü Tabulátorok paranccsal módosíthatók, illetve itt is létrehozhatók újak.

A Tabulátorok ablakban a Pozíció mezőben kell megadni a tabulátor pozíciókat. Meglévő tabulátorhely módosításához ki kell jelölni azt, majd a Pozíció mezőben módosítható az értéke.

Az Igazítás mezőben a kijelölt bekezdéseknek a behúzásokhoz viszonyított igazítása állítható be. Ha a szöveg és a tabulátorhely közötti üres részt ki kell tölteni, akkor kiválasztható a Kitöltés mezőben a megfelelő (pontosított, szaggatott vagy folytonos) vonaltípus. Az Alapérték mezőbe beírt értékkel az alapértelmezett tabulátorok távolsága módosítható. Egy tabulátor eltávolításához a Törlés, az összes nem alapértelmezett tabulátor egyidejű törléséhez a Mindet törli gombra kell kattintani (4.3. ábra).



4.3. ábra: Tabulátorok ablak

Az Élőfej és élőláb megadásakor szükség van a szakasz fogalmának ismeretére.

Szakasz

A szakaszok nélkülözhetetlen elemei a dokumentumoknak, mivel számos szerkesztési feladat csak szakaszok létrehozása után kivitelezhető. A szakaszok használata a következő esetekben szükséges:

- Változó hasábszámot kell megadni,
- Különböző papírméretre és/vagy tájolásra van szükség,
- Eltérő margóbeállításnál,
- Változó tartalmú és/vagy pozíciójú fej- és lábléc megadásánál,
- Fejezetenként (szakaszonként) újra kezdődő számozású láb- és/vagy végjegyzetre van szükség,
- Változó kezdőértékű, formátumú vagy helyzetű oldal- és/vagy sorszámozás megadásánál.

•

A szakasz a dokumentum része, így egy dokumentumon belül egy vagy több szakasz is létezhet. A dokumentum létrehozása után mindaddig egyetlen szakaszból áll, amíg külön paranccsal új szakasz létrehozása nem történik. A dokumentum tetszőleges számú szakaszra osztható, és az egyes szakaszok egymástól teljesen eltérő módon formázhatók.

A dokumentum egyes szakaszait a szakaszhatárok választják el egymástól. Ezek a határjelek rendeltetésüket és funkciójukat tekintve nagyon hasonlítanak a bekezdés-jelekhez. Új szakasz megkezdése előtt egy szakaszhatárjel segítségével zárhatjuk le az előzőt és nyithatjuk meg az újat.

Szakaszhatár beillesztése a dokumentumba:

- A dokumentumban a beviteli pontot abba a pozícióba kell vinni, ahol új szakaszt kell kezdeni,
- A **Beszúrás** menüpont **Töréspont** parancsát kell választani,
- A négy szakasz határtípus egyikét kell kijelölni
- OK gomb választása.

Szakasz határtípusok:

- Új oldal,
- Folyamatos,
- Páros oldal,
- Páratlan oldal.

Élőfej és élőláb

Az élőfej és az élőláb olyan szöveg vagy grafika, amelyek a dokumentum egyes oldalainak a tetején, illetve alján láthatók. Az élőfej a felső, míg az élőláb az alsó margón kerül nyomtatásra. A dokumentumban az egyes szakaszok első oldalán a többitől eltérő élőfej vagy élőláb használható, illetve ott el is hagyhatók. A dokumentum egyes szakaszaiban eltérő élőfej és élőláb adható meg a páros, illetve a páratlan oldalakon.

Eltérő élőfej létrehozása az első oldalon

A Nézet menüpont Élőfej és élőláb parancs választása. Az Élőfej és élőláb eszköztáron kattintani kell az Oldalbeállítás gombra, majd az Elrendezés lap választása. Az Első oldal eltérő jelölőnégyzet bejelölése, majd az OK gomb választása. Ekkor a Word a dokumentum vagy a szakasz első oldalának élőfejét vagy élőlábát jeleníti meg. Meg kell adni az első oldalon megjeleníteni kívánt élőfejet vagy élőlábat. Ha az első oldalon nem kell élőfejet vagy élőlábat használni, üresen marad a megfelelő terület.

A dokumentum vagy a szakasz többi oldalán megjelenő élőfej vagy élőláb elkészítéséhez az egérrel ki kell jelölni az Élőfej és élőláb eszköztáron a Következő , illetve a Váltás az élőfej és az élőláb között  gombot.

Eltérő élőfej létrehozása páros és páratlan oldalakon

A Nézet menüpont Élőfej és élőláb parancs választása. Az Élőfej és élőláb eszköztáron az Oldalbeállítás gombra kell kattintani. Majd az Oldalbeállítás lap választása. A Páros és páratlan eltérő jelölőnégyzet bejelölése következik, majd az OK gomb választása.

A páros oldalakon megjelenő élőfej vagy élőláb létrehozásához választani kell az  Előzőt, illetve a  Következőt mutat gombot, és el kell készíteni az élőlábat vagy az élőfejet.

A Word a Páros és páratlan eltérő beállítást a teljes dokumentumra érvényesnek tekinti, hacsak ez az egyes szakaszokban másképpen nem lett megadva.

Fejezetenként vagy szakaszonként eltérő élőfej és élőláb

Amikor egy szakaszban élőfej vagy élőláb szerepel, akkor a Word a dokumentum valamennyi szakaszába beilleszti ugyanezt az élőfejet vagy élőlábat. A dokumentum adott szakaszaihoz azonban ettől eltérő élőfej vagy élőláb is készíthető, ha az élőfejek illetve élőlábak folytonossága megszakad. A Nézet menüpont Élőfej és élőláb parancs választása. Az egérrel kattintani kell arra a szakaszra, amelyben eltérő élőfejet vagy élőlábat kell készíteni, majd az  Azonos az előzővel gomb kijelölését újra kattintással meg kell szüntetni és (Élőfej és élőláb eszköztáron) módosítható a szakasz meglévő élőfeje vagy élőlába, illetve új hozható létre.

Élőfej és élőláb törlése

Az egérrel arra a szakaszra kell kattintani, amely a törölni kívánt élőfejet vagy élőlábat tartalmazza. Ezután a Nézet/Élőfej és élőláb parancs választása. Ki kell

jelölni a törölni kívánt előfejet vagy élőlábat, és meg kell nyomni a DEL billentyűt.

Körlevélkészítés

A körlevélkészítés akkor használható, ha nagyszámú, egymástól csak néhány szóban különböző dokumentumot kell gyorsan és hatékonyan készíteni. Körlevelek készítésekor két dokumentumot kell szerkeszteni: a **törzsdokumentumot**, amely tartalmazza az állandó szöveget, és minden levélben azonos, valamint a változók neveit (a mezőneveket), az **adatforrás dokumentumot**, amely táblázatban tartalmazza a változó adatokat. A Word ebből a két dokumentumfájlból automatikusan létrehozza a körleveleket.

A szövegszerkesztő programban varázsló segíti a körlevélkészítést. A menüben az **Eszközök** menüpont **Levelek és küldemények** parancs **Körlevél varázsló** funkció választásával indul a körlevélkészítés. A varázsló hat lépésben készíti el a leveleket.

1. lépés: **Dokumentum típus** kiválasztása, Körlevél esetén a **levél** kijelölése.
2. lépés: **Alapul szolgáló dokumentum** megadása. Három lehetőség közül lehet választani:
 - aktuális dokumentum alapján,
 - sablon alapján,
 - egy már meglévő dokumentum alapján.

Az első lehetőség választásakor, a levél megírása után, a törzsdokumentumot célszerű elmenteni.

3. lépés: **Címzettek** kiválasztása. Három módon lehetséges:
 - létező listából,
 - outlook névjegyalbumból,
 - új lista létrehozásával.

Az első lehetőség esetén a varázsló ablakban megjelenő **Létrehozás** ikont kell választani. A megjelenő **Új címlista** ablakban lehet kiválogatni a levél címzettjeihez tartozó mezőneveket a **Testreszabás** gomb megnyomása után. A **Törlés**, az **Átnevezés** és a **Hozzáadás** gombok értelemszerűen alkalmazhatók az aktuális címzettek mezőneveinek előállítására. Ezután következik a címzettek adatainak begépelése. Az **Új bejegyzés** gomb választása után adható meg új cím. A címzettek adatainak megadása a **Bezárás** gombbal történik. A megjelenő **Címlista mentése** ablakban a **Fájlnev** és a **Hely** megadásával lehet elmenteni az adatokat. Ezután megjelenik a **Körlevél címzettjei** ablak, amelyben áttekinthető a címlista.

4. lépés: **A levél megírása**. Ekkor kell elhelyezni a törzsdokumentumba a címzethez tartozó mezőneveket. Ez egy ismétlődő tevékenységsorral történik addig, amíg minden mező a helyére kerül.
 - A kurzort a dokumentumban a **címlista** mező helyére kell vinni.
 - További elemek ikon választása.
 - Az **Adatmező beszúrása** ablakban az elhelyezendő mezőnév kiválasztása, **Beszúrás** gomb, majd a **Bezárás** gomb megnyomása.

Ha formázni kell az egyesített adatokat, például, ha egy mezőnevet félkövérré kell tenni, az egérrel a mezőnévre kell kattintani, majd alkalmazni kell a kívánt formátumokat.

5. lépés: **A levelek megtekintése**. A képernyőn sorban előre vagy hátra lehet lapozni az elkészült levelek között a << illetve a >> jelekkel.
6. lépés: **Az egyesítés befejezése**. Kétféle módon történhet:
 - nyomtatás (az elkészült levelek kinyomtatása),

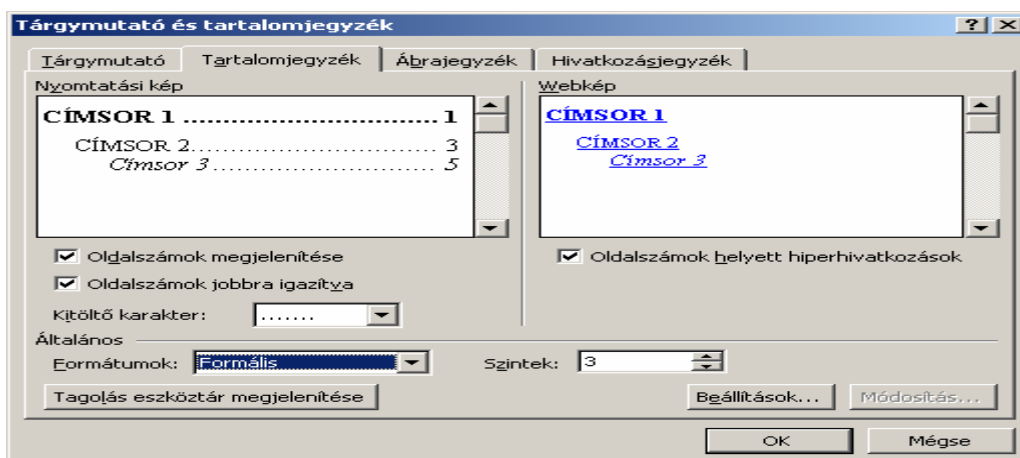
- az **Egyes levelek szerkesztése** (van lehetőség még az egyes levelek utólagos módosítására, majd elmentésére, és későbbi időpontban történik a levelek kinyomtatása).

Tartalomjegyzék

A tartalomjegyzék a dokumentum adott szintű címeit sorolja fel, és megadja az oldalszámukat. A tartalomjegyzék összeállításakor a Word megkeresi az adott stílusú bekezdéseket, címszint szerint rendezi ezeket, beírja az oldalszámot, és megjeleníti a tartalomjegyzéket a dokumentumban. A tartalomjegyzék tetszőleges stílusú bekezdésekből felépíthető, de készítésének legegyszerűbb módja, hogy a dokumentum címeit a beépített címszintekkel formázzák meg (Címsor1...Címsor9). A Word számos módszert kínál a tartalomjegyzék formátumának kialakítására, és a választott formázást és stílusokat a **Nézet** mezőben megjeleníti.

Tartalomjegyzék létrehozása

Az egérrel a dokumentumban arra a helyre kell kattintani, ahol a tartalomjegyzék elhelyezésre kerül, majd választani kell a **Beszúrás** menüpont **Hivatkozás** funkció **Tárgymutató és tartalomjegyzék** parancsot, ezután pedig a **Tartalomjegyzék** lapot. A rendelkezésre álló formátumok valamelyike kiválasztható a **Formátumok** listából (4.4. ábra).



4.4. ábra: **Tartalomjegyzéklap**

Tartalomjegyzék frissítése

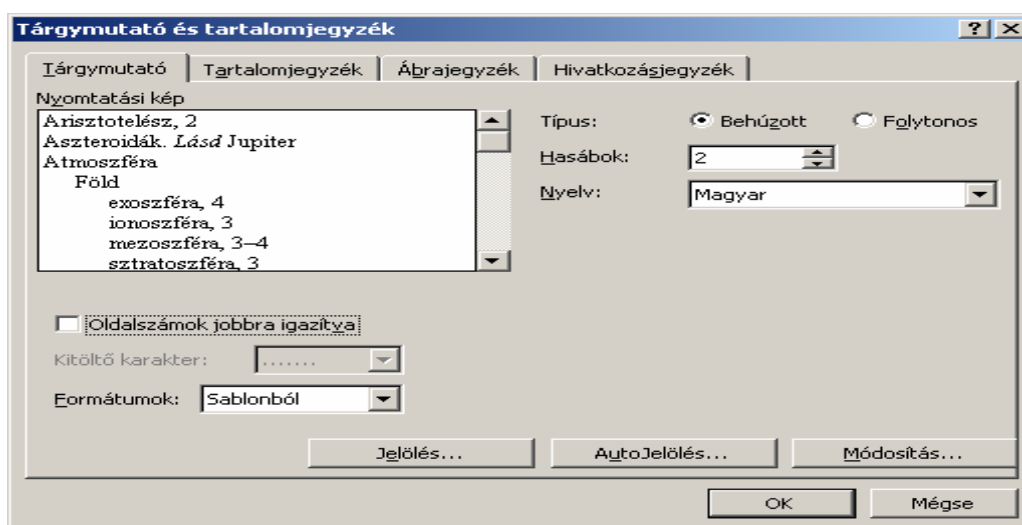
A kurzort a tartalomjegyzékbe kell vinni és meg kell nyomni az **F9** gombot. Alternatív megoldás: az egér jobb gombjával kattintani kell, és választani a **Mezőfrissítés** parancsot.

Tárgymutató

A tárgymutató a kinyomtatott dokumentumban a keresett szavak, fogalmak, vagy mondatok oldalszámát adja meg. Tárgymutató készítésekor ki kell jelölni a szerepeltetni kívánt szöveget vagy bejegyzést, formátumot választani, majd összeállítható a tárgymutató. A Word számos formázási lehetőséget biztosít és meg is mutatja a választott formázásnál a tárgymutató elrendezését, az illető formázás előtt. A tárgymutató összeállításakor a Word összegyűjti a tárgymutató-bejegyzéseket, sorba rendezi azokat, beírja a megfelelő oldalszámokat, megkeresi és törli az azonos oldalra mutató ismétlődő bejegyzéseket, végül megjeleníti a tárgymutatót a dokumentumban.

Tárgymutató készítése

1. A dokumentumban létre kell hozni a Tárgymutató-bejegyzéseket. (A tárgymutató-bejegyzés olyan mezőkód, amely azt jelzi, hogy az adott szöveg be fog kerülni a tárgymutatóba. A Word egy rejtett szöveggént formázott XE (Tárgymutató) mező beszúrásával készít tárgymutató-bejegyzést. A dokumentumban ki kell jelölni azt a szöveget (szót, mondatot), amely tárgymutató-bejegyzés lesz, majd meg kell nyomni az ALT+SHIFT+X billentyűket. Főbejegyzés létrehozásához a Főbejegyzés mezőben szerkeszteni lehet vagy be kell írni a bejegyzés szövegét. Ha Albejegyzést kell készíteni, akkor az Albejegyzés mezőbe kell beírni a szöveget. (Az albejegyzés olyan tárgymutató-bejegyzés, amely egy általánosabb címszó alá tartozik. Például a "bolygók" tárgymutató-bejegyzés albejegyzése lehet a "Mars" vagy a "Vénusz" címszó. A Főbejegyzés és/vagy Albejegyzés megadása után az egérrel a Jelölés gombra kell kattintani. Ha a szöveg valamennyi előfordulását meg kell jelölni, ez a Mindet jelöli gombbal történhet. A további bejegyzések szövegének kijelölése alatt a Tárgymutató-bejegyzés párbeszédpanel nyitva marad.
2. A dokumentumban arra a helyre visszük a kurzort, ahol a kész tárgymutató lesz.
3. A Beszúrás menüpont Hivatkozás parancs Tárgymutató és tartalomjegyzék menüpontok választása, majd a Tárgymutató lapé (4.5. ábra).
4. A rendelkezésre álló formátumok valamelyikét lehet használni a Formátumok mezőben.



4.5.ábra: Tárgymutatólap

Ellenőrző kérdések:

1. Ismertesse az automatikus helyesírás ellenőrzés használatát a szövegszerkesztő programban!
2. Ismertesse a nyelv megadását a szövegszerkesztő programban!
3. Ismertesse a tabulátorok használatát a szövegszerkesztő programban!
4. Ismertesse az élőfej, élőláb használatát a szövegszerkesztő programban!

5. Ismertesse a körlevélkészítést a szövegszerkesztő programban!
6. Ismertesse a tartalomjegyzék készítését a szövegszerkesztő programban!
7. Ismertesse a tárgymutató használatát a szövegszerkesztő programban!

Irodalomjegyzék

Bodnár Ibolya – Nagy Zoltán (1998): Szövegszerkesztés, Talentum Kft, Budapest.

Kunos Zsolt – Sörös Tamás (1995): A sokoldalú szövegszerkesztő Word 6.0, LSI Oktató Központ, Ligatúra Kft – Váci Ofset Kft.

Mogyorósi Istvánné (2003): Szövegszerkesztés, Kossuth Kiadó, Szekszárd.

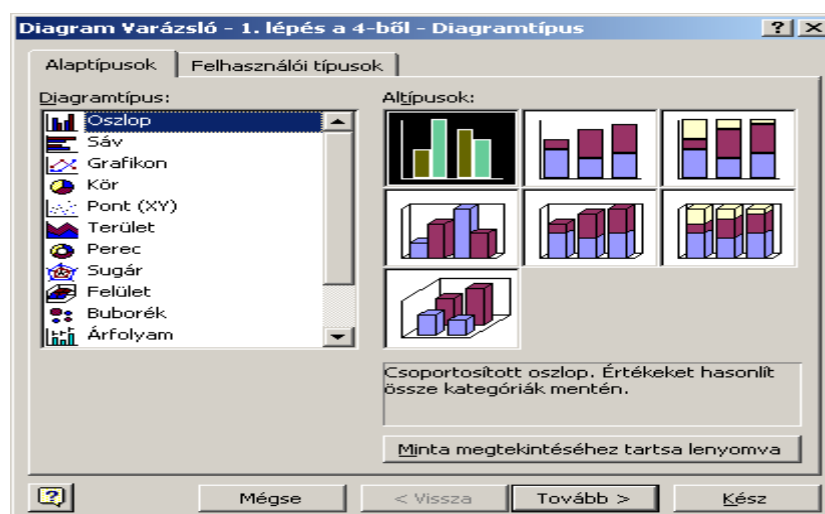
4.2. Táblázatkezelő rendszerek

A táblázatkezelő szoftverek azt az igényt elégítik ki, hogy gyakran kell ábrázolni számokat táblázatokban, amelyeknek azután oszloponként vagy soronként fel kell tüntetni azok összegét, és például százalékos arányát. Ebből a rutinfeladatból nőtték ki magukat a táblázatkezelők, amelyek ma már sokkal komplexebb feladatokra is alkalmasak: tudományos és statisztikai számításokra, nyilvántartások vezetésére, árlisták és grafikonok készítésére, valamint egyszerűbb adatbázis-kezelő műveletek végzésére. Az egyik legelterjedtebb táblázatkezelő a Microsoft Office programcsalád tagja, az Excel. A program néhány fontos, kiragadott funkciójának a bemutatására van csak lehetőség, a jegyzet korlátozott terjedelme miatt.

Diagram készítése

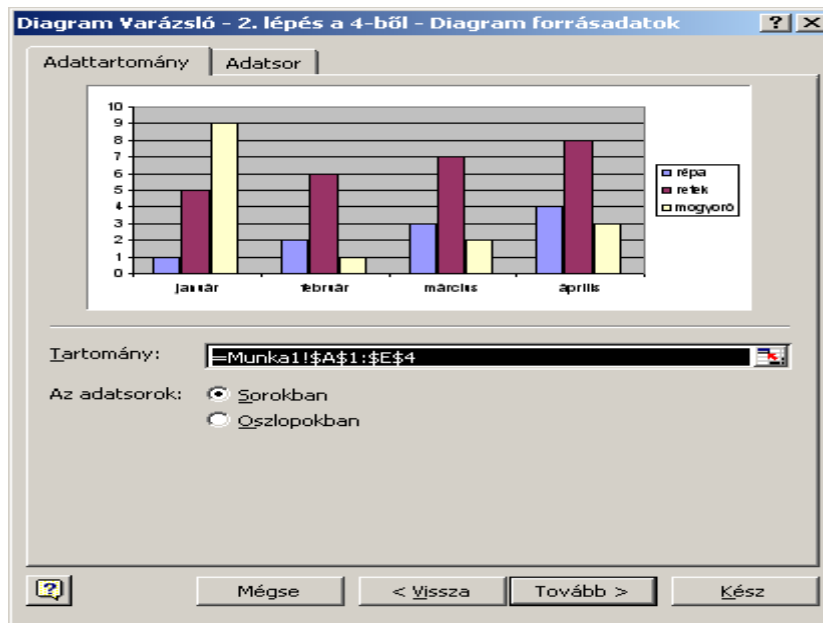
Elkészült táblázatból pillanatok alatt lehet készíteni egy tetszőleges formájú diagramot. A diagram készítése kétféle módon kezdhető el: a **Beszúrás** menüpont **Diagram** parancs vagy az eszköztár a diagramvarázsló ikon  kiválasztásával. Ezek hatására megjelenik a diagramvarázsló ablak, amely segítséget nyújt a diagram elkészítésében.

1. lépés: diagramtípus kiválasztása (4.6. ábra).



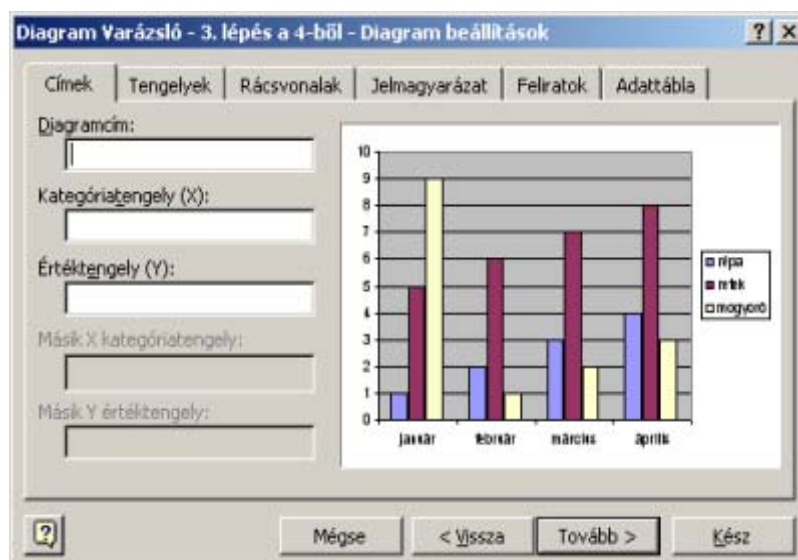
4.6. ábra: Diagramtípus választása

2. lépés: megadható, hogy az Excel táblázat mely részéből készül a diagram (4.7. ábra). Az Adatsor lapon kijelölhető A kategória (X) tengely feliratait, valamint az Adatsoroknak nevet is lehet adni.



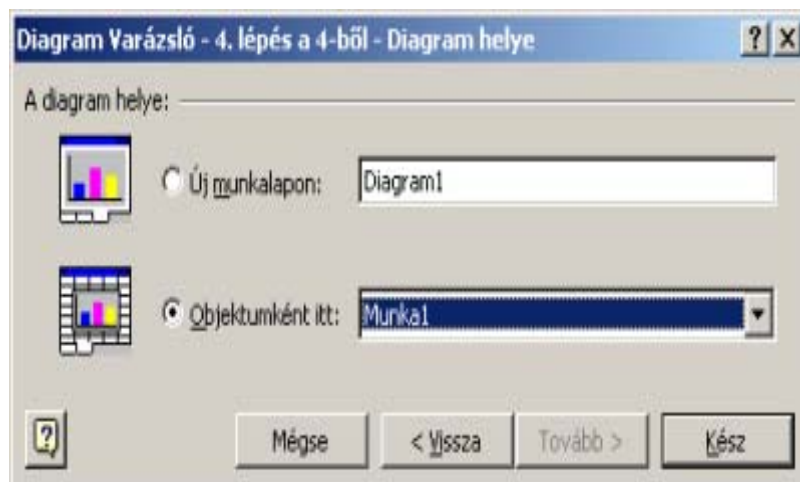
4.7. ábra: Diagram forrásadatok

3. lépés: megadhatók a diagram beállítások (4.8. ábra). Itt beállítható a diagram megjelenítése: Diagramcím, a Tengelyek, a Rácsvonalak, a Jelmagyarázat és a Feliratok, valamint hogy készüljön-e Adattábla. (Az Adattábla a diagramok alján megjeleníthető táblázat, amely a diagramok létrehozásához használt számértékeket tartalmazza, a diagram kategória tengelyéhez kapcsolódik.)



4.8. ábra: Diagram beállítások

4. lépés: megadható, hogy az új diagram hol jelenjen meg (4.9. ábra). Létrehozható egy Új munkalapon, amit a diagram teljesen ki fog tölteni, vagy Objektumként elhelyezhető az aktuális munkalapon.



4.9. ábra: Diagram helye

Ha a diagram készítése közben valami kimaradt, vagy rosszul lett megadva, vagy csak egyszerűen utólag kell a diagramban valamit megváltoztatni, természetesen megtehető. Általánosságban elmondható: ha valamit át kell formázni, akkor arra az objektumra vagy duplán kell kattintani az egérrel, vagy a jobb egér gombbal rákattintani, majd a megjelenő gyorsmenüből kiválasztani a megfelelő menüpontot.

Ha a diagram készítése közben több tartományt kell kijelölni, ez úgy történik, hogy az első tartomány kijelölése után le kell nyomni a billentyűzet CTRL gombját, és mindaddig lenyomva kell tartani, amíg az összes tartományt ki nem lett jelölve.

Példa néhány formázásra

Ha nincs szükség a jelmagyarázatra, akkor egyszerűen ki kell jelölni és a Delete gombbal le lehet törölni. Ha esetleg valamelyik feliratot meg kell változtatni, akkor rá kell kattintani arra a felíratra, és vagy átírható, vagy a Delete gombbal letörölhető. Ha valamelyik oszlopnak a színét kell módosítani, csak rá kell duplán kattintani az oszlopra, majd a megjelenő ablakból ki lehet választani egy másik színt. Ugyanezt meg lehet oldani az egér jobb gombjával is, majd a megjelenő gyorsmenüből ki lehet választani az Adatsorok formázása menüpontot. A diagram szinte bármelyik elemét át lehet formázni: háttérszín adható meg, újabb rácsvonalakat lehet beszúrni, az egyes oszlopok értékeit rá lehet írni az oszlopokra, át lehet alakítani egy másik diagramtípusra a diagramot, stb.

Ha a táblázat valamelyik cellája megváltozott, miután már elkészült a diagram, akkor természetesen a diagramon is hamarosan megjelenik a változás.

Excel adattábla használata

Az Excel egy táblázatkezelő program, de egyszerűbb adattáblák kezelésére is alkalmas. A tábla első sorában vannak a mezőnevek, amelyek azt határozzák meg, hogy az egyes oszlopokban milyen adatok lesznek. Az összes többi sorban vannak a tényleges adatok. Az egyes sorokat rekordoknak nevezzük. A rekordokban egy cella egy mező (4.10. ábra).

	A	B	C	D	E
1	név	város	utca	házsám	irányítószám
2	Nagy Béla	Budapest	Rózsa	12	1234
3	Fehér Aladár	Győr	Szív	56	5678
4	Tóth Ödön	Szeged	Holló	45	9012
5	Kiss Jakab	Pécs	Pipa	65	3456
6	Kovács Elemér	Miskolc	Akácfa	104	7890
7	Sós Ilona	Budapest	Dob	99	
8	Halász Elek	Budapest	Vadász	25	1234

4.10. ábra: Excel adattábla

A rekordok sorbarendezése úgy végezhető el, hogy első lépésként ki kell jelölni a táblázatot, amelyiket rendezni kell. A kijelölés után ki kell választani az **Adatok** menüpont **Sorbarendezés** menüpontját. A képernyőn megjelenő **Rendezés** ablak alján megadható, hogy van-e **rovatfej** az adatbázisban vagy nincs. Ez annyit jelent, hogy a táblázat első sora mezőazonosítókat tartalmaz-e vagy sem. Ugyanis ha az első sorban mezőazonosítók vannak, akkor azt a sort nem kell rendezni.

Az adattábla egyszerre több szempont szerint rendezhető. A **Rendezés** ablakban a legfelső mezőben kiválasztható, hogy legyen rendezés, például a **név** mező alapján emelkedő sorrendben. Ha véletlenül két azonos nevű ember van, akkor ezt a rendezést tovább lehet bővíteni, vagyis megadható egy másodlagos illetve egy harmadlagos rendezés is valamelyik mező alapján. Vagyis megadható, hogy a két azonos nevű ember közül az kerüljön előbbre, akinél például az **irányítószám** kisebb. Ilyenkor a második rendezési feltételnél ki kell választani, hogy legyen rendezés **irányítószám** alapján emelkedő sorrendben. A program az OK gomb hatására sorba rendezi a rekordokat. Az eszköztáron is található két sorbarendezés ikon:  , azonban ezekkel csak egy oszlop alapján lehet rendezni.

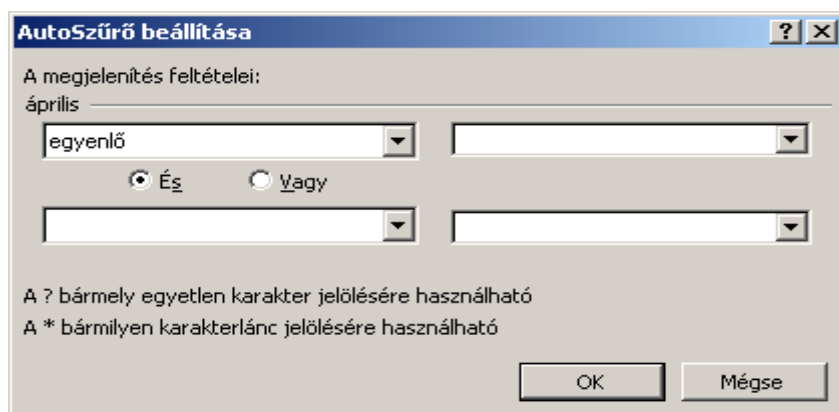
Rekordok szűrése

A következő fontos művelet, amit egy adattáblában gyakran el kell végezni, a rekordok szűrése. Előfordulhat, hogy csak azokat az embereket kell megjeleníteni, akik Budapesten laknak. Ehhez ki kell jelölni az adattáblát, vagy elég csak az adattáblába kattintani az egérrel. Ezek után ki kell választani az **Adatok** menüpont **Szűrő** funkció **AutoSzűrő** menüpontját. Ennek hatására minden mezőnév mellett megjelenik egy lefelé mutató kis nyíl. A város mezőnév melletti nyílra kattintva, megjelenik egy kis legördülő lista, amelyben megjeleníti a program az abban az oszlopban található különböző városneveket. A listából kiválasztható Budapest, akkor csak azok a rekordok fognak látszani, melyekben a város mező értéke Budapest. Ha az összes rekord megjelenítésére van ismét szükség, akkor vagy kiválasztható a listából a **Mind**, vagy az **Adatok** menüpont **Szűrő** funkció **Minden látszik** parancs.

Ha az feladat, hogy csak azok a rekordok látszódjának, amelyekben a város mező Budapest, valamint a név mező Nagy Béla, akkor már csak egy sor látszik. Az is előfordulhat, hogy olyan kombináció lett beállítva, aminek hatására egy sor sem jelenik meg.

A mezőnevek melletti legördülő listákban van egy **Egyéni** elem is (4.11. ábra). Ebben az ablakban megadhatók egyéb szűrési feltételek is. Ilyen

feltételeket nem csak számokat tartalmazó mezőkre lehet megadni, hanem szövegekre is. Ha például a név mezőnél az a beállítás szerepel, hogy csak azok jelenjenek meg, amelyeknél a név mező '>h', akkor csak azoknak az embereknek a nevei jelennek meg, amelyek az ABC-ben a 'h' betű után vannak. A mezőnevek melletti listában van olyan választási lehetőség is, hogy azok a rekordok jelenjenek meg, amelyek üresek, vagy éppen nem üresek.



4.11. ábra: Autoszűrő beállítása

Függvények

Az Excelben több száz beépített függvény van. A táblázat tartalmazza a leggyakrabban használt függvényeket. A függvény neve az adott szoftver nyelvétől is függhet, így előfordulhat, hogy az ÁTLAG függvényt AVERAGE néven érjük el.

SZUM	összeadja a paraméterlistájában megadott cellákat
ÁTLAG	átlagot számol
MAX	megkeresi a paraméterlistájában megadott legnagyobb számot
MIN	megkeresi a paraméterlistájában megadott legkisebb számot
HA	megvizsgál egy logikai kifejezést, amely ha igaz, akkor végrehajt valamit, ha nem igaz, akkor egy másik műveletet hajt végre
SZUMHA	a SZUM és a HA függvény keveréke; amennyiben a megadott feltétel igaz, akkor egy megadott tartományban összegez
FKERES	egy táblázat első oszlopában megkeres egy adott értéket, majd eredményképpen a táblázat egy meghatározott oszlopának az értékét adja vissza
RÉSZLET	a törlesztési időszakra vonatkozó törlesztési összeget számítja ki állandó nagyságú törlesztő részletek és kamatláb esetén
DARAB	megszámolja, hogy a paraméterlistájában mennyi szám van
MA	visszaadja a mai dátumot
ÉS	két logikai művelet ÉS kapcsolata
VAGY	két logikai művelet VAGY kapcsolata

Átlag()

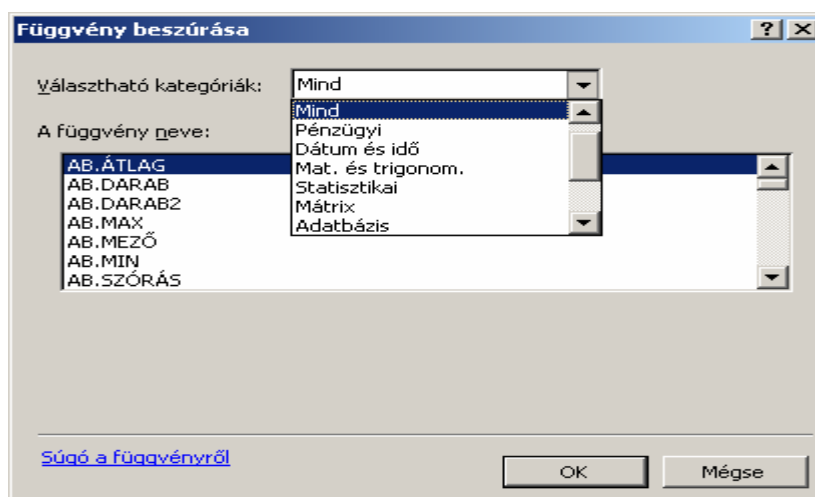
Az 4.12. ábra táblázatában az átlag meghatározása a feladat.

Első lépésként az egérrel abba a cellába kell kattintani, ahová a képlet kerül (B7), majd a **Beszúrás** menüpont **Függvény** parancsát kell választani. Ennek hatására előbukkan a **FüggvényBeszúrása** ablak (4.13. ábra).

	A	B	C	D	E	F	G	H
1	Osztályzatok	Matematika	Fizika	Biológia	Történelem	Átlag	Legjobb jegy	Legrosszabb egy
2	Nagy Aladár	3	4	5	4			
3	Fehér Béla	3	2	3	4			
4	Tóth Aladár	4	5	1	2			
5	Kiss Enikő	5	5	5	5			
6	Kovács Ödön	3	5	5	4			
7	Átlag							

4.12. ábra: Adattábla átlag függvény számításához

A Választható kategóriák listájából lehet függvény kategóriát választani. Ha nem ismert, hogy a keresett függvény melyik kategóriában van, akkor érdemes a **Mind** kategóriában keresni. Miután az **ÁTLAG** függvény kijelölése megtörtént, meg kell nyomni a **Tovább** gombot.



4.13. ábra: Függvény beszúrása ablak

A következő ablakban meg kell adni a paraméterlistát. Erre több lehetőség van: vagy a billentyűzetten az átlagolandó cellák cellahivatkozásait be kell gépelni, vagy az egérrel ki kell jelölni az átlagolni kívánt cellákat. Érdemes az egérrel történő kijelölést használni. Kijelölés után az Excel beírja a tartományhivatkozást a képletbe. A megfelelő tartomány kijelölése után, meg lehet nyomni a **Kész** gombot. A teljes képlet a következőképpen néz ki: **=ÁTLAG(B2:B6)**.

Az összes többi függvényt hasonlóan kell használni: az egérrel bele kell lépni abba a cellába, amelyikbe a függvényértékét kell elhelyezni, majd a függvényvarázslóban meg kell keresni a függvényt, és a **Tovább** gomb megnyomása után meg kell adni a paramétereket.

Részlet()

A **RÉSZLET** függvény a törlesztési időszakra vonatkozó törlesztési összeget számítja ki állandó nagyságú törlesztő részletek és kamatláb esetén.

A példában ki kell számolni, a felvett 2.500.000Ft hitelt évi 22%-os fix kamattal, és 5 éves törlesztési idő esetén, havonta mennyi összeggel kell törleszteni (4.14. ábra). A függvény paraméterezése a következő (4.15. ábra).

	A	B
1	Felvett hitel	2500000
2	Éves kamat	22%
3	Törlesztési idő	5 év
4	Havi törlesztő részlet	69047

4.14. ábra: Adattábla részlet függvény alkalmazásához

RÉSZLET

Ráta: B2/12 = 0,018333333

Időszakok_száma: B3*12 = 60

Mai_érték: B1 = 2500000

Jövőbeli_érték: = szám

Típus: = szám

= -69047,28009

A kölcsönre vonatkozó törlesztési összeget számítja ki állandó nagyságú törlesztőrészek és állandó kamatláb esetén.

Ráta az időszakonkénti kamatláb a kölcsön esetére.

Érték: 69 047,28 Ft

Kész Mégse

4.15. ábra: Részlet függvény paraméterezése

- Ráta** Az időszakonkénti kamatláb.
- Időszakok_száma** A fizetési időszakok száma a törlesztési időszakban.
- Mai_érték** A jövőbeli kifizetések jelenértéke, vagyis az a jelenbeli egyösszegű kifizetés, amely egyenértékű a jövőbeli kifizetések összegével.
- Jövőbeli_érték** A jövőbeli érték vagy az utolsó részlet kifizetése után elérni kívánt összeg. Ha a jövőbeli_értéket elhagyjuk, a program 0-nak tekinti (például egy kölcsön jövőbeli értéke 0).
- Típus** Értéke 0 vagy 1; azt mutatja, hogy mikor esedékesek a résztörlesztések. 0 vagy hiányzik: az időszak végén van 1: az időszak kezdetén van

A paraméterek közül az utolsó kettőt nem kötelező megadni. A példában a ráta B2/12, ugyanis a havi részletet kell kiszámítani, ezért az éves kamatot át kell számítani havi kamatra, tehát osztani kell 12-vel. Az időszakok_száma B3*12, vagyis a törlesztési időt is át kell váltani évről hónapra. A mai_érték az az összeg, amely ha most egy összegben lenne kifizetve, akkor nem kellene havonta részleteket törleszteni, vagyis ez maga a felvett hitel összege. A képlet =RÉSZLET(B2/12;B3*12;B1). A függvény neve előtt azért áll egy negatív előjel, mert különben negatív számot adna vissza eredményül a függvény.

Ellenőrző kérdések

1. Ismertesse az Excel diagramkészítés funkcióit!
2. Hogyan történik az Excel adattáblák rendezése?
3. Hogyan történik az Excel adattáblákban a szűrés?
4. Adjon meg néhány Excel függvényt! Ezek mire használhatók?

Irodalomjegyzék

Baczoni Pál (1997): Könnyen is lehet – Excel for windows 95, Panem-McGraw-Hill, Budapest.

Békefi Zoltán (1999): Táblázatkezelés, Talentum Kft, Budapest.

Kovalecsik Géza (1995): Excel for Windows 5.0, ComputerBooks, Budapest.

Kovácsné Cohner Judit (1995): Az Excel 5.0 függvénye, ComputerBooks, Budapest.

4.3. Adatbáziskezelés

A jegyzet 6. fejezete részletesen taglalja az adatbázis-kezelés elméleti alapjait. A 7. fejezetben, az adatbázisok gyakorlati alkalmazásai keretében, az Access program áttekintése, a program egyes részei, valamint az adatbázisok szabványos lekérdező nyelve, az SQL kerül ismertetésre.

Az Access adatbázis-kezelő rendszer

Az Access a Microsoft által kifejlesztett relációs adatbázis-kezelő program. Az Office programcsomag professzionális változatának része, de külön is megvásárolható. Felhasználóbarát rendszer, egyszerű alkalmazások interaktívan, programírás nélkül elkészíthetők. Az Access önálló rendszer, de sokoldalúan együttműködik a Microsoft SQL Server adatbázis-kezelővel.

Egy Access adatbázis az alábbi típusú objektumokat tartalmazhatja.

- *Tábla*: relációs adattábla, amelyben az adatokat tárolják. A tábla sorokból és oszlopokból áll. A sorokat rekordoknak, az oszlopokat mezőknek nevezik. Az oszlopokban az egyedek tulajdonságait tárolják, ezt a mezőnév mutatja. A sorokban az egyedek összes tulajdonsága szerepel.
- *Lekérdezés*: segítségével a táblákban tárolt adatok közül lehet kiválogatni azokat, amelyekre szükség van. A lekérdezés speciálisan megfogalmazott kérdések, feltételek összessége. A lekérdezés eredménye származhat egy vagy több adattáblából, amelyek adataiból számított értéket is tartalmazhat.
- *Úrlap*: az adatok rekordjainak bevitelét, módosítását vagy törlését felhasználóbaráttá teszi.
- *Jelentés*: az adatbázis adatainak papíron való megjelenítésére elsősorban a jelentések szolgálnak. A jelentés adatforrása lehet az adattábla, vagy egy lekérdezés eredményhalmaza. Az adatok megjelenítésén kívül részösszegek, végösszegek, számított képletek is megadhatók. A nyomtatott lapon megadható élőfej, élőláb vagy egyéb grafikus elem is.

- *Makró:* az adatbázis kezelésekor a felhasználó gyakran ismétlődő feladatait automatizálja a makró. A makró több műveletet egyetlen parancsba fog össze.
- *Modul:* Visual Basic nyelven eljárásokat, függvényeket lehet készíteni. A megírt program kódok modulok formájában tárolódnak.

Az egy adatbázishoz tartozó valamennyi tábla, űrlap, lekérdezés és jelentés egy közös *.mdb* kiterjesztésű adatbázisfájlban tárolódik.

Az említett objektumokról és további Access lehetőségekről bővebben szó lesz a jegyzet 7. fejezetében.

4.4. Számítógépes grafika

A számítógép hardver és a grafikus felhasználói felületű operációs rendszerek fejlődése nagyban elősegítették, hogy a számítástechnikán belül a grafika önálló területté vált. Megteremtődött a lehetőség a nagy mennyiségű képi információk számítógépes feldolgozására. A számítógépes grafika (computer graphics) képek, különféle rajzok számítógépen történő előállításával, valamint ezek számítógépes adathordozón való tárolásával foglalkozik. Tágabb értelemben idesorolhatók még a képek beolvasása (szkennelés, digitális fényképezés) és a képek megjelenítése.

A számítógépes grafika a *dimenzió alapján* lehet *2D* (kétdimenziós) vagy *3D* (háromdimenziós). A *képfajta*k lehetnek *vonalasak*, *szürkék*, *színesek*. A számítógépes grafika *interaktivitás* szerint készülhet *off-line rajzolással*, *interaktív rajzolással* vagy *az objektum előre meghatározásával és körüljárásával*, *interaktív tervezéssel*. A *kép szerepe* szerint a számítógépes grafika lehet *végtermék* vagy *közbülső termék*.

A számítógépes grafika fajtái

A rajzolóprogramok két csoportba oszthatók: a vektorgrafikus és a pixelgrafikus (bittérképes) rajzolóprogramok csoportjába. A két csoport közti különbség a rajzok elkészítésében és azok tárolási módjában van. A vektorgrafikus rajzolóprogramokkal vektoros képeket lehet jól feldolgozni. A pixelgrafikus rajzolóprogramok pixeles képek feldolgozására jól alkalmazhatóak. A prospektusokat pl. jól lehet készíteni pixelgrafikus programokkal.

Vektorgrafikus rajzó programok

A vektorgrafikus rajzóprogramok alkalmasabbak műszaki ábrák (CAD), egyszerű vonalas grafikák, kiadványok, prospektusok, vagy hosszabb szövegek kezelésére. A pixelgrafikus rajzóprogramokkal könnyen készíthetők egyszerűbb grafikák, illusztrációk, alkalmazhatók fotótechnikai trükkök, különleges effektusok.

A vektorgrafikus rajzóprogramok a képek felépítésére egyszerű alakzatokat (téglalap, ellipszis, sokszög, stb.) és ún. Bézier-görbéket (csomópontokkal, a csomópontok közt húzott görbékkel és érintőszakaszokkal felépített görbéket) használnak. A képek nem képpontokból állnak, tetszőlegesen nagyíthatók és kicsinyíthetők, a végeredmény minősége csak a képmegjelenítő eszköztől függ. Egy bittérképnél, egy pixelgrafikus rajzóprogrammal készített grafikánál, a kép méretével a képfájl mérete is növekszik, hiszen több képpont adatait kell tárolni. Mivel a vektorgrafikus rajzóprogramok a képeket csomópontok segítségével építik fel, a képfájlok méretét a csomópontok és a görbék száma határozza meg. Minél több csomópont szükséges a kép leírásához, azaz minél több görbéből áll a kép, annál nagyobb a vektoros állomány mérete.

Mivel a kép nagyításával, illetve kicsinyítésével nem változik a csomópontok száma, így nem változik az állomány mérete sem. Bonyolultabb grafikák esetében több MB méretű vektoros állomány is előállhat (4.16. ábra).



4.16. ábra: Vektorgrafikus kép

A vektorokból álló objektumokkal felépített képen minden objektum kitölthető valamilyen színnel. Fotótechnikai eljárások pixelgrafikus rajzolóprogramokkal végezhetők. A mai rajzolóprogramok képesek a vektorgrafikus rajzokat pixeles formátumra konvertálni, amelyet azután tovább lehet módosítani.

Pixelgrafikus rajzoló programok

A pixelgrafikus rajzolóprogramok a képeket mátrixszerűen elrendezett képpontokból, pixelekből, építik fel. A sorokat és oszlopokat alkotó képpontok különböző színűek lehetnek, ezekből a pontokból áll össze a rajz. A bitmap grafika (rasztergrafika) egy kép tartalmát egy négyzetrácson elhelyezkedő színes képpontok összességéként, ún. pixelekként írja le. Ahogy a képen látható, a falevél képe a képpontok helyének és a képpontok színértékeinek tárolásával hozható létre. A pixelekből álló képeket a kép felépítésére utalva bittérképeknek is nevezik. A bittérképek egyik legfontosabb tulajdonsága a felbontás. A kép minőségét több felbontás-típus egyszerre határozza meg (4.17. ábra).



4.17. ábra: Pixelgrafikus kép

Felbontás

Eszközök, képállományok jellemzője. Egy adott felületen tartalmazott, vagy megjeleníthető képpontok számát adja meg. Minél nagyobb a felbontás, annál több információt - részletet - tartalmaz a kép.

Képméret

A képméret a kép fizikai méreteit (hosszúság, szélesség) jelenti.

Képfelbontás

A képfelbontás a képen belüli képpont-távolságot mutatja. Mértékegysége a képpont/hüvelyk (pixels per inch: ppi). Ha egy kép felbontása 72 ppi, ez azt jelenti, hogy egy négyzethüvelyknyi területen $72 \times 72 = 5184$ képpont található.

Nagyobb felbontás esetén jobb a kép minősége, azaz több részlet jelenik meg. A képfelbontás elméleti érték, mivel a kép minősége függ a kép fizikai méretétől és a kimeneti eszköz felbontásától is.

Színmélység

A bitfelbontás vagy színmélység megmutatja, hogy egy képpont színe hány biten tárolódik, vagyis maximálisan hány színt használtak a képben. A nagyobb színmélység több színt, vagyis az eredeti kép pontosabb színvisszaadását teszi lehetővé, de egyben a képfájl méretét is növeli. A 8 bites színmélység esetén 256, a 16 bites (High Color) színmélység esetén 65536, a 24 bites (True Color) színmélység esetén 16777216 szín használható.

Monitorfelbontás

A monitorfelbontás a megjelenítő eszköz képfelbontását jellemzi. Mértékegysége a dpi (pont/hüvelyk: dot per inch). A monitor felbontása a megjeleníthető kép méretét mutatja, például egy 192 ppi képfelbontású kép egy 96 dpi felbontású monitoron eredeti méretének kétszeresében jelenik meg, mivel a 192 képpontból csak 96 jeleníthető meg a képernyő egy hüvelykjén.

Rácsfelbontás

A rácsfelbontás vagy rácsfrekvencia az egy hüvelykre eső, tónusképzéshez használt elemi egységek számát mutatja. Mértékegysége a vonal/hüvelyk (lines per inch: lpi). Ha egy színes képet fekete-fehérben kell kinyomtatni, vagy a nyomdai feldolgozáshoz alapszíneire kell bontani, fekete-fehér rácsmintát kell használni a színek szimulálásához. A képminőség függ a rácsfelbontástól és a kimeneti eszköz felbontásától. A kimeneti eszköz felbontása jellemzi a kész képet megjelenítő eszköz felbontását.

Egy pixelgrafikus ábrázolással tárolt kép fájlmérete arányos a kép felbontásával. Egy nagyobb felbontású kép részletgazdagabb, mint egy azonos méretű, kisebb felbontású kép. A fájl méretet befolyásolja a használt színmélység, illetve a fájlformátum megválasztása is.

4.4.1 Grafikus fájlformátumok

Vektorgrafikus fájlformátumok

Ahány vektorgrafikus program annyiféle vektorgrafikus fájlformátum. Az ilyen fájlformátumok mindegyike vektorosan, matematikai alapon írja le a programmal készített rajzot.

.EPS: A vektorgrafikus formátumok között talán legelterjedtebb az Encapsulated PostScript formátum, amelyet az Adobe fejlesztett ki a 80-as években. Az .EPS formátumot a grafikus programok legtöbbször ismeri, így a fájlformátumok közti átjárás egyik eszköze.

.CDR: CorelDRAW fájlformátuma.

.WMF: Windows operációs rendszer vektoros képformátuma.
.EMF: Elterjedt, általános vektorgrafikus formátum.
.PDF: Portable Data Format. Az Adobe Acrobat család fájlformátuma.
.DWG: AutoCAD fájlformátuma.
.DXF: Az Autodesk fájlformátuma.

Pixelgrafikus fájlformátumok

Sok pixelgrafikus fájlformátum ismert. A pixelgrafikus programokban vannak olyan, szinte szabványként használt fájlformátumok, amelyek más pixelgrafikus rajzoló programokban lehetővé teszik a pixelgrafikus rajzokkal való munkát. Gyakori, hogy egy különleges hatást csak egy bizonyos rajzolóprogramban lehet alkalmazni, ezért merült fel az igény a pixelgrafikus programok közti átjárhatóságra.

.BMP: Windows, ill. OS/2 bitmap. Maximum 24 bites színmélységet tárol, kizárólag az operációs rendszer által meghatározott színeket használja. Továbbfejlesztett verziója: **.DIB** (device-independent bitmap).

.PCX: Paint, illetve más egyszerű rajzolóprogramokkal készített, tömörítetlen vagy RLE tömörítést használó fájlformátum. Maximum 24 bit/pixel színmélységet használ.

.TIF, .TIFF: Tagged Image File Format, az Aldus cég által kifejlesztett, széles körben elterjedt képformátum.

.JPG, .JPEG: kifejlesztője a Joint Photographic Expert Group. Adatvesztő tömörítési eljárást használó formátum. A tömörítési arány 5:1 és 20:1 közötti. A .JPEG formátum 24 bites színmélységet képes kezelni (több mint 16 millió színt). A tömörítési eljárás az emberi szem becsaphatóságán alapul és adatvesztéssel jár. Az Internet elterjedt kép-fájlformátuma.

.GIF: Graphics Interchange Format. Fejlesztője a Compuserve, maximum 256 színű, vagy 256 fokozatú grayscale. A .GIF formátumnak két verziója van, mindkettő legfeljebb 256 szín tárolására képes. A .GIF87-tel szemben a .GIF89 képes több képkocka egyetlen fájlban való tárolására, valamint az Internetes böngészőkben a lassabb hálózatokon hasznos azonnali megjelenítésre, majd a kép finomítására alkalmas (interlaced .GIF). Így a .GIF-kép teljes méretben látható már a letöltés elején is. Az Interneten gyakori formátum, az utóbbi időben visszaszorulóban van.

.PNG: Portable Network Graphics. A harmadik fő internetes kép-fájlformátum. Szabad formátumú internetes fájlformátum, a .GIF formátumot válthatja fel.

.TGA: Targa. Fejlesztője: Truevision. 16/24/32 bit/pixel színmélységet tárolhat. Ipari szabvány, a fontosabb grafikus programok kezelik.

.MAC: A Macintosh számítógépek beépített MacPaint rajzoló programjának fájlformátuma.

.PSD: Adobe Photoshop fájlformátuma.

4.4.2 A színes kép

Az emberi szem kb. a 400-700nm hullámhosszúságú fénytartományra érzékeny, ez a látható fénytartomány. Ez alatt az ultraibolya, e fölött az infravörös tartomány található. Szemünk a látható spektrumból a kb. 550nm-es zöld színre, majd a vörös és a kék fénysugarakra a legfogékonyabb. A szín és a fényérzékelés attól függ, hogy a rájuk eső fény mely hullámhosszúságú részét nyelik el, verik vissza.

Színek, színérzet jellemzői

- *Színezet, árnyalat* alatt értjük, hogy milyen színt érzékelünk (sárga, zöld, kék, stb).
- *Telítettség* jelentése, hogy mennyire tiszta a szín, mennyi fehéret tartalmaz. Minél több benne a fehér, annál világosabb, annál fakóbb.

- *Világosság* a szín helyét adja meg, a fekete (0%) és a fehér (100%) között, ami a megvilágítás erősségétől vagy a fekete szín mennyiségétől függ.

Színek keverése

A színkeverésnek két módja különböztethető meg: az additív és a szubsztraktív módszer.

Additív színkeverés, RGB

Az additív színkeverés alapszínei a vörös (Red), a zöld (Green) és a kék (Blue), vagyis azok a színek, amelyekre szemünk a legjobban reagál. Az angol színnevekből jött létre az **RGB** rövidítés, ami ennek a színkeverésnek a másik neve.

A három alapszín együtt egyenlő arányban fehér fényt ad. A kék+zöld kombináció kékeszöldet, azaz ciánt, a kék+vörös színek együtt bíort eredményeznek, míg a vörös+zöld színekből sárga szín jön létre. A kapott kevert színek az alapszínek komplementerei, kiegészítő színei. Két alapszín és a komplementerük együtt mindig fehér színt ad. A vörös komplementere a cián, a zöldé a bíbor, a kéké a sárga.

Szubsztraktív színkeverés, CMYK

A szubsztraktív színkeverés a cián (Cyan), a bíbor (Magenta) és sárga (Yellow) színekből indul ki. E három színt megfelelő színerővel egymásra nyomtatva feketét kapunk, míg a sárga+bíbor vörös, a cián+sárga zöld, a cián+bíbor pedig kék színt eredményez. Úgy is mondható, hogy ez a fajta színkeverés a RGB ellentéte. Az alapszínekből előállított fekete szín telítettségének javítására a három alapszín mellé feketét kell keverni. A **CMYK** rövidítés az angol szakirodalomból származik, ahol a **CMY** a három alapszín nevének kezdőbetűiből ered. A **K** betű az angol **black** fekete szóból. Más megfontolás szerint a **K** betű a **Key color**, kulcsszín, kifejezésből származik. A nyomdaipar a **CMYK** eljárást alkalmazza.

A számítástechnikában egyszerűbb grafikai programok az **RGB** háromszínnyomásos módszert használják. Az ilyen programoknak outputja általában a képernyő, ami fényt bocsát ki, és nem a papír, ami fényt nyel el.

Az **RGB**-ben és **CMYK**-ban előállított színek tartománya nem fedi tökéletesen egymást. A szoftverek különféle színmodelleket használhatnak, ezért gyakorlatilag nem ugyanazt látjuk az egyik gépen, illetve programban, mint a másikon. Papíron is ugyanaz másként nézhet ki.

Képtömörítés

A képek mérete a képet alkotó képpontok számától, valamint az alkalmazott színmélységtől függ. Például egy teljes képernyős kép helyigénye, 800*600 felbontásnál 3 byte-os, azaz valós, true color, színmélység esetén 800*600*3, azaz 1440000 byte.

A nagy képméretok miatt szükség van a képek tömörítésére. A Web böngészők a képek átviteléhez 2-256 szín esetén veszteségmentes **.GIF**, 16 milliós valós szín esetén veszteséges **.JPEG** formátumot használnak.

Veszteségmentes tömörítéskor a visszaállított kép minősége azonos az eredetivel. Tömörítéskor elérhető méretcsökkenés a kép tartalmának függvénye. Az átlagos tömörítési arány 1:4-re tehető (pl.: **.GIF**).

Veszteséges tömörítés alkalmazásakor a tömörített, majd visszaállított kép minősége rosszabb az eredetinel. A minőségromlás foka a használt tömörítési eljárástól függ. A tömörítési arány kb.: 1:10 körül van (**.JPEG**).

Ismert a *fractal tömörítés*, amely szintén veszteséges tömörítés. Ez 4-5-ször jobb tömörítési arányt eredményez, azonos képminőség mellett, mint az előzőekben ismertetett technológiák. 1:45 körüli méretcsökkenést is el lehet vele érni. Az előző példa esetén 800*600 képpontos true color kép esetén 32 Kbyte-ot jelent. A fractal tömörítéssel tömörített fájlokat .FIF fájloknak nevezik (Fractal Image Format). A visszaállítás felbontás független. Az eredetnél nagyobb méretben is visszaállítható a kép, akkor sem lesz "szemcsés". A módszer hátránya, hogy maga a tömörítés idő- és számolásiigényes folyamat, néhány percig is eltarthat. A tömörített képfájlok kicsomagolása gyors, de alul marad a .JPEG kitömörítéssel szemben.

A legelterjedtebb grafikus programok

Képnézegetők: csak megjelenítésre és képformátumok közötti átalakításra alkalmasak (ACDSee, Picture Viewer, Image).

Videó- és hanglejátszók: Videó fájlok (pl. AVI, MPEG, DIVX) és hangfájlok (pl. MP3, WAV) lejátszására alkalmas programok (Windows Media Player, Mplayer, Totem).

Animátorok, média-átalakítók: animációk, multimédiás állományok készítésére, vágására, szinkronizációra alkalmas programok (PhotoPaint Animation Shop, Easy CDgrabber).

Rajzoló programok: dokumentumokba, bemutatókba egyszerű ábrák beillesztésére szolgáló programok (MS Paintbrush, MS Paint).

Szövegszerkesztők és kiadványszerkesztők: komolyabb dokumentumok, kiadványok készítésére alkalmas programok (MS Word, StarOffice, MS Publisher, Quark Xpress).

Művészi grafikák, képek készítésére: speciális, komplex raszteres és vektoros szolgáltatásokat nyújtó célprogramok (Adobe Photoshop, Gimp, Corel Draw).

Műszaki rajzok készítése, 3D-s megjelenítés: AutoCAD, ArchiCAD.

A grafikus programok közül több szabadon letölthető az internetről (pl. Infanview, Mplayer, Virtualdub, Dia, stb).

Képgyűjtemény (ClipArt)

A legtöbb program tartalmaz clipart gyűjteményt, amelyek általában .WMF fájlok. Ezek a kis állományok egy-egy dolog sematikus rajzát, vagy beszkennelt képet tartalmaznak.

4.4.3 Különleges számítógépes grafikai eljárások

Fraktál

A fraktálok a matematikai Káosz-elmélet alapján számítógéppel generált képek. Nincs szükség fraktál archívumokra, mivel a kiinduló paraméterek ismeretében egy adott fraktál mindig újragenerálható.

Morphing

Az egyik objektum alakját elveszítve "átfolyik" valami másba. Ilyen hatást előállító programok legtöbbje csak kétdimenziós objektumokat képes egymásba átalakítani. Egy hálózati torzításával kell kijelölni, hogy a forráskép adott területei a célkép mely részeibe transzformálódnak.

3D modellezés

A 3 dimenziós modellező programok térbeli tárgyak tervezésére és illusztrációk készítésére alkalmas programok, jól használhatók valósághű megjelenítésre. Fejlettebb változataikkal a 3D-s képek sorozataként animáció is készíthető, ami a számítógépes reklámgrafika egyik fontos területe.

Rendering

A rendering módszer úgy készíti a képet, hogy az ábrázolni kívánt test minden pontjának megvizsgálja a fényforrások és a kamerához viszonyított helyzetét, matematikai egyenletekkel modellezett fizikai jelenségek alapján számolja ki az adott képpont színét. Az egyszerűbb modellek akár valós időben is számíthatóak (videó játékok), a bonyolultabbakon hónapokig dolgoznak több ezer gépen (A függetlenség napja, Robotok, Terminátor 2).

Animáció

A számítógépes animáció attól számítógépes, hogy a benne szereplő képeket valamilyen számítógépes grafikai módszerrel állítják elő. Legtöbbször a módszer rendering vagy morphing. A külön animációs csomagok ritkábbak, hiszen egyszerűbb azokat beépíteni egy adott grafikai módszert megvalósító programba. Az animációk többféleképpen rögzíthetők: .FLI, .FLC, .MPEG és .AVI formátumba. Mindegyik formátum valamilyen módon tömörít, csak azokat a képrészeket tárolja, amelyek különböznek az őket megelőző képkocka ugyanazon helyén lévő adatától. Az .MPEG, állóképeknél megismert .JPEG formátumhoz hasonlóan, némi minőségvesztéssel járó tömörítést valósít meg.

Virtual Reality

Egy régen leírt, de csak nemrégien megfilmesített gondolat, amely filmekben a futurisztikus jellege miatt a legjobban megfogja a számítástechnikában kevésbé jártas nézőket. Manapság játékokban és tervezői programok kiegészítéskor alkalmazzák ezt a technológiát.

Interpolációs technika

Interpolációs technikának nevezik a kép helyreállítási folyamatát. Háttér helyreállításra alkalmas interpolációs technikák: a bilineáris interpoláció, 2 dimenziós polinominális interpoláció, a bicubic interpoláció. Előtér helyreállításra lehet alkalmazni a következőket: lineárisan összeillesztett intenzitási interpolációt, a lineárisan összeillesztett behelyettesítési interpolációt.

Ellenőrző kérdések:

1. Jellemezze a számítógépes grafikai programok csoportjait!
2. Mit jelent a képfelbontás, színmélység, rácsfelbontás, monitor felbontás? Milyen mértékegységekkel jellemezhetők az előzőek?
3. Nevezzen meg néhány vektorgrafikus fájlformátumot!
4. Nevezzen meg néhány pixelgrafikus fájlformátumot!
5. Adja meg a színérzet jellemzőit!
6. Mit jelent a számítógépes grafikában az RGB, jellemezze ezt?
7. Mit jelent a számítógépes grafikában az CMYK, jellemezze ezt?
8. Mit tud a képtömörítésről?
9. Jellemezze a legelterjedtebb grafikus program típusokat!
10. Milyen különleges számítógépes grafikai eljárások vannak? Jellemezze ezeket!

Irodalomjegyzék

Bodnár Ibolya - Nagy Zoltán (1998): Számítógépes prezentáció és grafika, Pc-Start studio, Budapest.
Budai Attila (1999): A számítógépes grafika.

Nagy Zoltán (2001). Számítógépes grafika és prezentáció, Pedellus Novitas Kft, Debrecen.
Székely Vladimír - Poppe András (1994): A számítógépes grafika alapjai IBM PC-n, ComputerBooks, Budapest.
Szirmay-Kalos László (2001): Számítógépes grafika, CommuterBooks, Budapest.

4.5. Prezentációkészítés

A prezentációkészítő PowerPoint programot a Microsoft Office programcsomag tartalmazza. A program részletes ismertetésére nincs lehetőség a jegyzet korlátozott oldalszáma miatt. Így a teljesség igénye nélkül csak a program néhány lényeges funkciója kerül bemutatásra.

A PowerPoint program lehetőséget nyújt bemutatók, diák, jegyzetek, vagy egyszerű grafikák készítésére. Az elektronikus bemutatókhoz készített diák tartalmazhatnak szöveget, diagramokat, rajzobjektumokat és alakzatokat, valamint ClipArt képeket, mozgóképeket, hangokat és más programokban készített ábrákat. Az utolsó pillanatban is elvégezhetőek módosítások a bemutatón, és diaáttűnések, időzítés és animálás használatával határozható meg, hogyan kövessék egymást a diák. A bemutató tipikusan egy gépen fut (előadások, kirakati reklámok), de előfordul, hogy több kijelzőn is nézik párhuzamosan (néhány TV adó) vagy időben eltolva (NIIF videó archívum), esetleg párhuzamosan és időben eltolva (Cisco Virtual University).

A bemutató könnyebb követhetősége érdekében a hallgatóság számára emlékeztető készülhet a diák kisméretű változatának felhasználásával, amelyekről egy oldalra egy, kettő, három, négy, hat vagy kilenc nyomtatható. Ezen kívül előadói jegyzetek is kinyomtathatóak. Miközben készül a bemutató, kinyomtatható a vázlat, beleértve a diacímeket és a főbb pontokat. Ezen felül használható a **Fájl\Küldés** parancs, amellyel a diaképek és jegyzetek elektronikus levélben elküldhetők a munkatársaknak egyenként, vagy körözvényként a felhasználók csoportjának.

Diabemutató tervezése

A diabemutatón a tartalom legyen az események középpontjában. A használt eszközök, például az animációk és az áttűnések arra szolgálnak, hogy segítsék mondanivaló kifejtését, nem pedig arra, hogy elvonják a hallgatóság figyelmét, és a különleges hatásokra irányítsák.

Ha a hallgatóság azt szokta meg, hogy balról jobbra olvas, akkor a diák felépítését tervezzük úgy, hogy az egyes újabb pontok baloldaltól érkezzenek. Ha ezután egy lényeges pontot hangsúlyozni kell, akkor az jobb oldalról lépjen be.

Alkalmanként zene bejátszása vagy valamilyen hangeffektus a dia áttűnése vagy felépítése alatt a hallgatóság figyelmét a diabemutatóra irányíthatja. A túl gyakran használt hangeffektusok viszont elterelhetik a hallgatóság figyelmét a lényeges kérdésekről.

A bemutató gyorsasága szintén befolyásolja a hallgatóság reagálását. Ha túl gyorsan követik egymást a diák, akkor kimeríthetik a hallgatóság tagjait; ha túl lassan, akkor érdeklődésüket veszítik. Ha úgy tűnik, hogy túl sok szöveg szerepel egy-egy dián, akkor a túlsúlyos dia szövegét meg kell próbálni két vagy három diára szétosztani, majd növelni kell a betűméretet.

Dianézetek

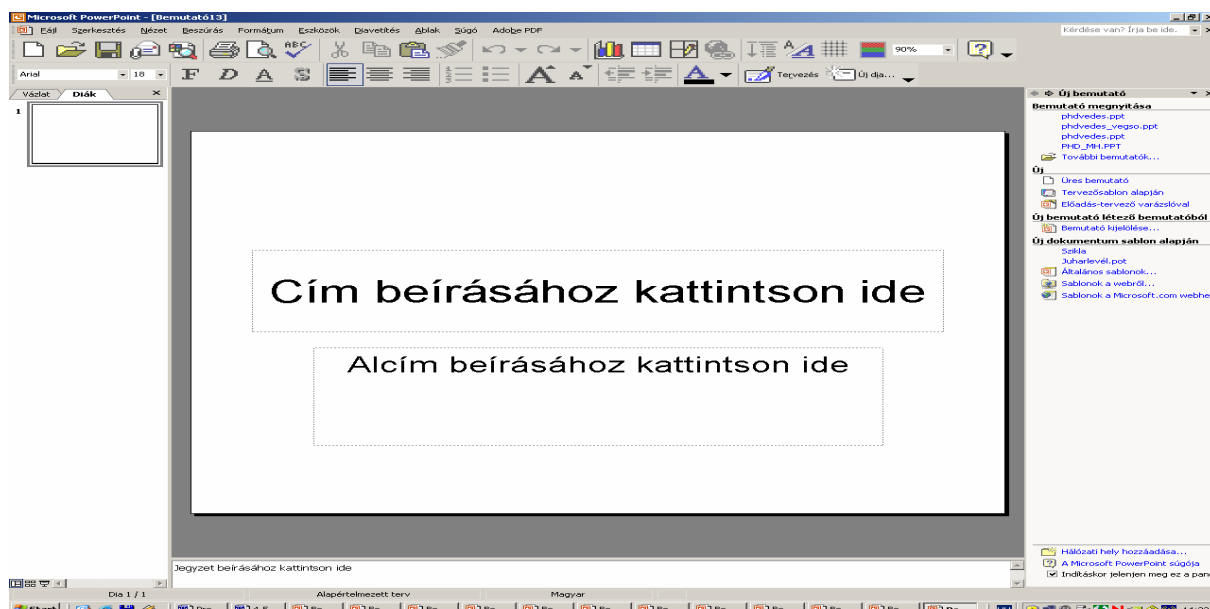
Normál nézet: Az alapértelmezett nézet három ablaktáblát tartalmaz: egyszerre lehet dolgozni a vázlat, a dia és a jegyzet táblákban.

Diarendező nézet: A diarendező nézetben egyszerre tekinthető meg a bemutató diái, egérekattintással a kívánt diára lehet ugrani, illetve kijelöléssel és az egér húzásával egyszerűen megváltoztatható a diák sorrendje.

Jegyzetoldal-nézet: Minden egyes diához kísérő jegyzetoldal tartozik, amely a dia kicsinyített mását tartalmazza, valamint elegendő helyet ahhoz, hogy jegyzeteket lehessen írni a lapra. A jegyzetek kinyomtathatók, majd segítségükkel emlékezetbe idézhetők a mondanivaló főbb pontjai a bemutató során.

Új bemutató létrehozása

Amikor új bemutató készül, többféleképpen lehet elindulni: lehet dolgozni varázslóval, használható vázlat, amely meghatározza a bemutató tartalmát és szerkezetét, vagy választható egy PowerPoint tervezősablon, amely megadja a bemutató színekészletét, betűtípusait és más alaptulajdonságait. El lehet indulni üres bemutatóból is: ekkor a színekészlet, a betűtípusok és más alaptulajdonságok az alapértelmezett értékek lesznek. A választási lehetőségeket a PowerPoint az indításkor felajánlja (4.18. ábra).



4.18. ábra: Új bemutató készítési lehetőségek

Üres bemutató létrehozása

Ha olyan új bemutatót kell készíteni, amely a PowerPoint alapértelmezés szerinti bemutatójának színekészletét, cím- és szövegstílusát használja, akkor be kell jelölni az Üres bemutató választókapcsolót (4.18. ábra).

Sablonok használata

Ha az új bemutatót egy PowerPoint Tervező sablon alapján kell elkészíteni, amely a színekészletet, valamint a cím- és a szövegstílust is tartalmazza, az új bemutató létrehozásakor az egérrel kattintani kell a megfelelő Tervező sablonra (4.18. ábra).

Vázlat használata

Az Előadástervező Varázslóra kattintva gyorsan megadható egy új bemutató tartalma; csak a feladatra vonatkozó kérdésekre kell válaszolni. Az Előadástervező Varázsló egy módosítható szövegű sémát jelenít meg (4.18. ábra).

Létező bemutató megnyitása

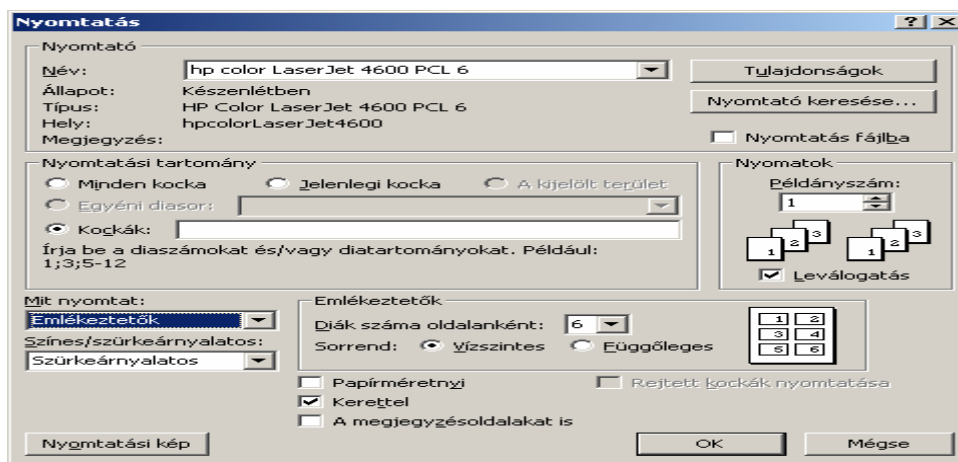
Már meglévő bemutató megnyitásakor, az egérrel kattintani kell a **Fájl** menü **Megnyitás** parancsra, vagy használható a **CTRL+O** billentyűkombináció, illetve választható a **Megnyitás** ikon  az eszköztáron. Egyszerre több bemutató is megnyitható, az **Ablak** menü segítségével át lehet lépni egyikből a másikba (4.18. ábra).

Bemutató mentése

A bemutató mentésére a Word-ből ismert módon több lehetőség van: az eszköztár **Mentés...**  ikonja vagy a **Fájl** menü **Mentés** parancs.

Nyomtatás

A bemutató kinyomtatható színesben vagy fekete-fehérben: a diák, a vázlat, az előadói jegyzet és a hallgatóságnak szánt emlékeztetők. A nyomtatás a **Fájl** menü **Nyomtatás** parancssal, a **CTRL+P** billentyűkombinációval, vagy az eszköztár  ikonjával kezdeményezhető (4.19. ábra).



4.19. ábra: Nyomtatás ablak

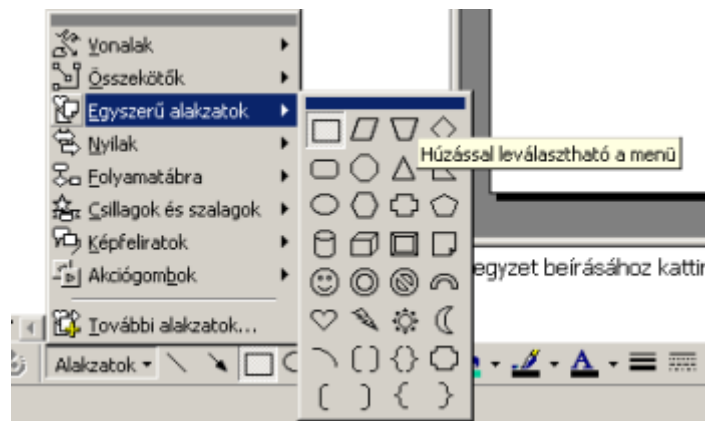
Egyszerű rajzelemek készítése

Egyszerű alakzat rajzolása a **Rajzolás** eszköztár egyes ikonjaival elvégezhető. Így rajzolható: vonal, téglalap, négyzet, ellipszis, kör.

Beépített rajzobjektumok

A PowerPoint előre elkészített beépített alakzatokkal is rendelkezik. Alakzat rajzolásához az egérrel kattintani kell a **Rajzolás** eszköztár **Alakzatok** gombjára, ki kell választani a megfelelő típust, és az egér húzásával megrajzolható az alakzat a dián (4.20. ábra).

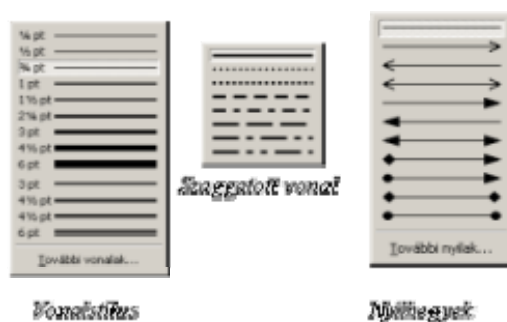
Rajzolás eszköztárból kezdeményezhető ClipArt kép beillesztése a diába, és kép beszúrása.



4.20. ábra: Beépített rajzobjektumok választási lehetőségei

Szín, kitöltés, árnyék, szegély

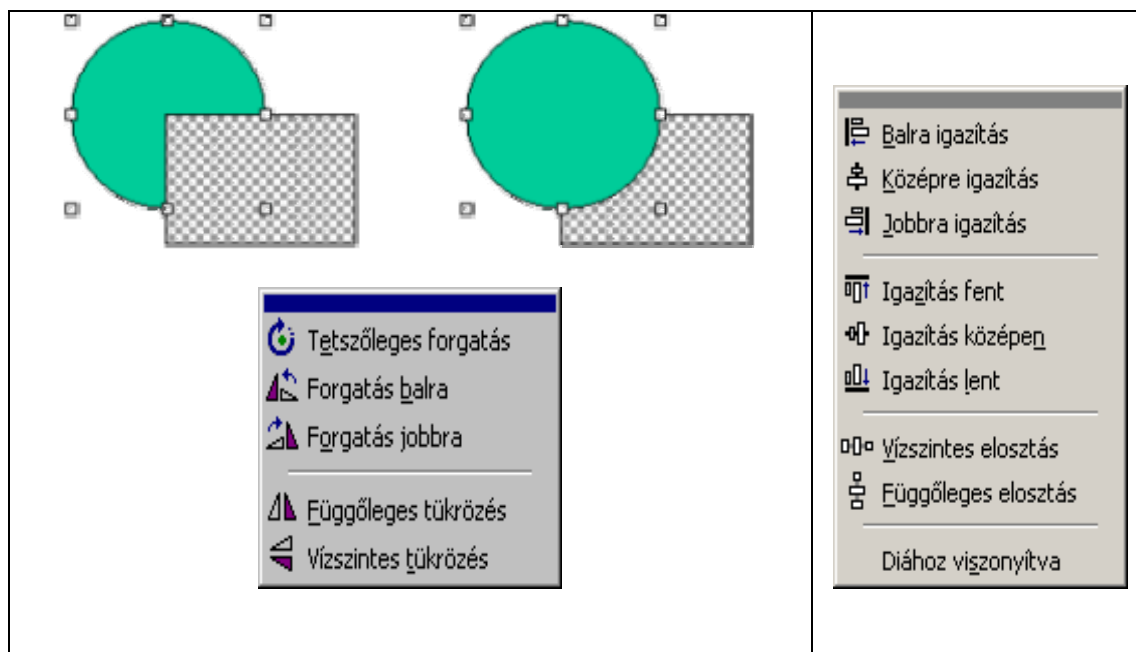
A rajzobjektum színének megadásához ki kell választani az objektumot, majd az egérrel kattintani kell a **Rajzolás** eszköztár **Vonalszín** ikonjára . Zárt alakzat rajzolásakor megadható a kitöltés színe az eszköztár **Kitöltő szín** gombjára kattintva. Az objektumárnyékolás bekapcsolásához az egérrel kattintani kell az eszköztár **Árnyék** stílus gombjára. A rajzobjektum körvonalának módosítására használható az eszköztár **Vonaltípus** és **Szaggatási típus** gombjai, illetve vonal rajzolásakor a **Nyílstílus** gomb is. Mindhárom esetben a legördülő menüből választható ki a vonal új tulajdonsága (4.21. ábra)



4.21. ábra: Vonalak rajzolási lehetőségei

Rajzobjektumok kezelése

A rajzobjektumokat lehet mozgatni, méretezni, csoportosítani, csoportokra bontani, egymásra halmozni, objektumokat egymáshoz viszonyítva előrehozni, hátraküldeni, forgatni, tükrözni, igazítani (4.22. ábra).



4.22. ábra: Rajzobjektumok kezelése

Szöveg bevitelle és formázása

Szöveget dia készítésekor az ún. szövegmezőben lehet elhelyezni. Megfelelő típusú dián ennek helyét szaggatott körvonalú terület jelzi, de szabadon is létrehozható **szövegdoboz** az **Rajzolás** eszköztár **Szöveg** gombjával . A szövegnek rövidnek, lényegre törőnek, áttekinthetőnek és jól olvashatónak kell lennie. A diákra ezért csak címek, vázlatpontok kerülnek, a többit az előadó mondja el szóban. A vázlatpontokra jellemző, hogy több szintűek, azaz fő- és alpontok egymásba vannak ágyazva.

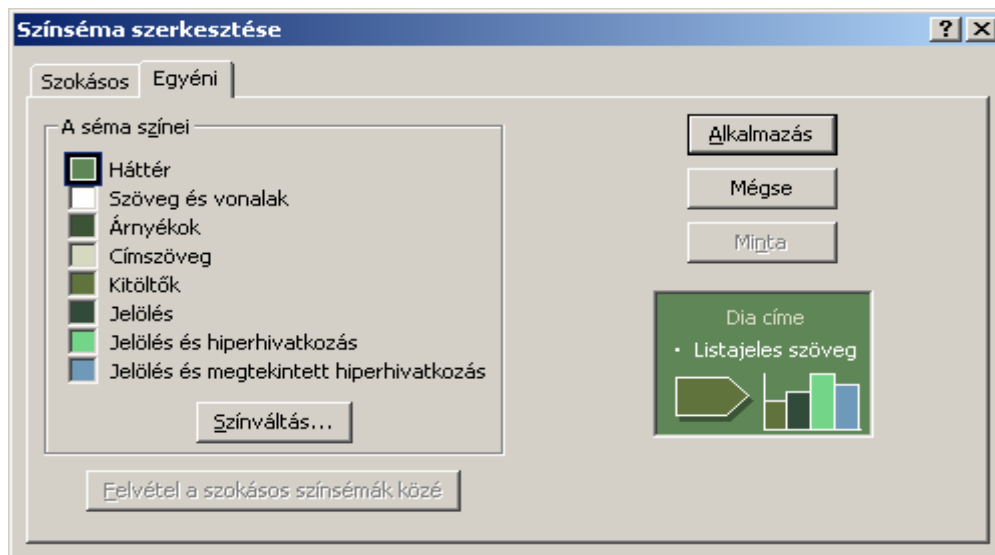
Animálási effektusok

Az objektumok és a szöveg felépítésére különböző elnevezésű animációs sémák közül lehet választani a **Diatervezés** munka ablakban az **Animációs sémák** funkció választása után.

Dia színválasztéka

A PowerPoint bemutatókban használt színek a program által létrehozott színsémákon alapulnak. Ez elérhető a **Formátum** menü **Diatervezés** parancs **Színsémák** funkcióval. A kiválasztott színsémákon még lehet módosítani az **Egyéni lapon** (4.23. ábra).

Egyéni háttér megadásához az egérrel kattintani kell a diára az egér jobb gombjával, és kiválasztani az **Egyéni háttér** parancsot vagy **Formátum** menü **Egyéni háttér** menüpontot.



4.23. ábra: Színséma szerkesztése

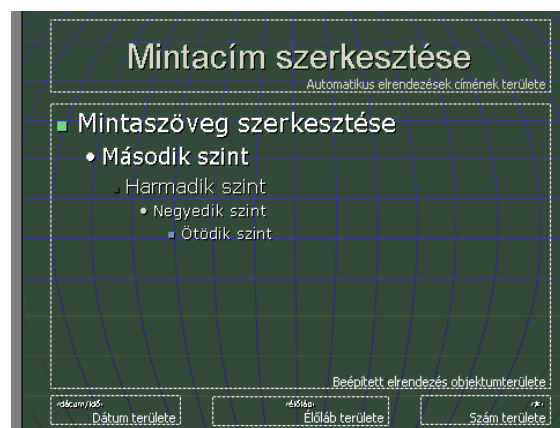
Mintadia

Az aktív dia mintájának megjelenítéséhez ki kell választani a **Nézet** menü **Minta** funkció **Dia-minta** parancsot. A dia-minta különleges dia, amelyen a diabemutató összes diájának formázása (például a diacím és diaszöveg formázása) megadható. A diabemutató minden egyes alkotórészének (dia, vázlat, jegyzet, emlékeztető) különálló mintája van. Ha egy képet, szöveget vagy egy különleges formázást minden dián, vázlatban, jegyzetben, illetve emlékeztetőben meg kell jeleníteni, be kell tenni a megfelelő mintába.

Az élőláb szövegdozobban a dia általános adatai helyezhetők el. Ilyen adatok lehetnek például a bemutató dátuma és időpontja, a dia száma, a bemutató címének szöveges magyarázata, vagy hogy milyen alkalomból kerül sor a bemutatóra. Amikor az élőláb tartalma már elkészült, akkor a megfelelő mintára átváltva módosítható az élőláb külalakja vagy helyzete a diákon, a jegyzetlapokon, az emlékeztetőkön és a vázlatokon (4.24. ábra).

Word táblázat beszúrása

Dia nézetre kell átváltani táblázat beszúrásához. A **Beszúrás** menü **Táblázat** menüpontban megadható a beillesztendő táblázat sorainak és oszlopainak a száma.



4.24. ábra: Mintadia

Egyéb objektumok beillesztése

Előfordulhat, hogy a diára olyan objektumot is be kell illeszteni, amelynek ikonja nem látható az eszköztáron. Ekkor választani kell a **Beszúrás** menü **Objektum** parancsot. A leggyakoribb egyéb objektumok: Microsoft Egyenlet, Microsoft Excel diagram, Microsoft Szervezeti diagram, Microsoft Word rajz, Microsoft WordPad dokumentum.

Diarendezés

A diabemutató előtti utolsó simítások, a diák közti áttűnések a diarendező nézetben végezhetők el. A diák sorrendje a **Diarendező** nézetben az egérrel való húzással egyszerűen módosítható, áttűnések iktathatók be, valamint beállítható a számítógépes diavetítés időzítése. Lehetőség van az egyes diák törlésére is.

A **Diarendező** ablakban egy dia egérkattintással jelölhető ki. Ha egyszerre több dia tulajdonságait kell módosítani, lenyomva kell tartani a **SHIFT** billentyűt, és az egérrel kattintani kell a megfelelő diákra.

Az áttűnések, felépítési effektusok és az időzítés beállításakor a munkát a **Diarendező** eszköztár megkönnyíti. Áttűnés létrehozásához **Diarendező** nézetben ki kell jelölni azt a képkockát vagy képkockákat, amelyekhez áttűnést kell rendelni. Ezután az egérrel kattintani kell az eszköztár **Áttűnés** gombjára. Az **Áttűnés** munkalapon választható az áttűnés típusa, a sebessége, a továbbítás módja. A **Továbbítás** mezőben megadható az a módszer, amely a kijelölt képkocka megjelenítését szabályozza diabemutató közben. Kézi léptetés kérésekor a **Csak kattintáskor** kapcsolót kell bejelölni. Automatikus vetítéskor be kell jelölni az **Automatikusan...** **X. másodperc múlva** kapcsolót, és meg kell adni egy számot, hogy milyen időközönként kövessék egymást a képkockák. A **Hang** mezőben pedig az állítható be, hogy milyen hangeffektus kísérje az átmenetet.

Diavetítés

Nézet menü **Diavetítés** menüpontból vagy a **Diavetítés** menü **Diavetítés** parancsával kezdhető el a diabemutató. **Dia nézetbe** az **ESC** billentyűvel lehet visszatérni.

A diavetítés beállításai **Diavetítés** menüpont **Vetítési beállítások** panelen adhatók meg. Az **Ismétlés az ESC lenyomásáig** jelölő négyzettel megadható, hogy a diavetítés végén automatikusan előlről kezdhető a bemutató.

A diavetítés az **F5** funkciós billentyű lenyomásával is elindítható. **Diavetítés** közben az egér balgombjának lenyomásával a következő dia jeleníthető meg, a szóköz billentyű megnyomása is ugyanezt a hatást váltja ki. **Diavetítés** közben az egér jobbgombjának lenyomásakor megjelenő helyi menüben ki lehet választani a következő vagy az előző diát, illetve a bemutatóhoz felhasznált tetszőleges diát. **Diavetítés**kor a **PageDown** funkciós billentyűvel előre, míg a **PageUp** funkciós billentyűvel visszafelé haladhatunk a bemutató diái között.

Ellenőrző kérdések:

1. A diabemutató tervezésének fontosabb szempontjait adja meg!
2. Milyen dianézetek vannak? Jellemezze ezeket!
3. Milyen módokon hozható létre diabemutató?
4. Milyen egyszerű rajzelemeket és beépített rajzobjektumokat lehet használni a PowerPoint-ban?
5. Hogyan lehet a dián a színeket megváltoztatni?

6. Mire szolgál a minta-dia?
7. Mire használható a diarendezés?

Irodalomjegyzék

Bodnár Ibolya - Nagy Zoltán (1998): Számítógépes prezentáció és grafika, Pc-Start studio, Budapest.

Gerő Judit (1995): PowerPoint4, ComputerBooks, Budapest.

Nagy Zoltán (2001). Számítógépes grafika és prezentáció, Pedellus Novitas Kft, Debrecen.

4.6. Irodaautomatizálás

Az irodaautomatizálás fogalma először az 1950-es években bukkant fel. Eleinte csak a számítógépek alkalmazását jelentette az irodai munka területén. A mai értelmezéshez közelebb álló irodaautomatizálásról, azóta beszélhetünk, amióta az IBM cég 1964-ben elkészítette az első mágneses háttértárral felszerelt irodai szövegfeldolgozó berendezését. A számítógépes alkalmazások megjelenése és a számítógépnek, mint eszköznek az elterjedése tette lehetővé az automatizálást. Az automatizálás egy olyan rendszer kialakítását jelenti az irodában, amely az iroda feladatait vagy annak egy részét önállóan képes megvalósítani. Az irodai automatizálás nagyon sokáig csak az egyes tevékenységek kiváltását jelentette. Az irodai folyamatok elemzésével és modellezésével kiderült, hogy algoritmizálhatók az irodai munkák. Ezért ma már az informatikán belül külön szakággá fejlődött ki az irodaautomatizálás.

A hagyományos irodai funkciók problémái

Az irodai funkciók sokrétűek, szövevényesek, nehéz kiigazodni bennük. A hagyományos iroda nem képes a feladatokat szerves egységben kezelni. Erős szabályozókkal próbálják az ügykezelési rendet fenntartani. Ez azt eredményezi, hogy a működés szabályozott, de nem elég gördülékeny. Az erős szabályozás erős felügyeletet igényel. Ezzel bonyolultabbá válnak az ügyek, lelassul az ügyintézés. Az apparátus létszáma növekszik. A különböző funkcionális megoldások inkompatibilitása miatt akadozottan, lassan, rugalmatlanul és nem kellő hatékonysággal működnek a folyamatok.

Az irodaautomatizálás fő feladata, hogy kiküszöbölje ezeket a hiányosságokat, egy szabályozott ügymenetet lebonyolítani képes elektronikus iroda kialakításával.

Elektronikus iroda

Az elektronikus irodában az irodai funkciók gépi eszközök segítségével valósulnak meg. Az esetek döntő többségében ezek az eszközök számítógépek és azok kiegészítő elemei. Elektronikus irodában leegyszerűsödnek, könnyebbé és gyorsabbá válnak a tevékenységek.

Automatizált iroda

Az automatizált iroda fogalma több mint az elektronikus irodáé. Automatizált irodáról akkor lehet beszélni, amikor egy elektronikus irodában kialakítják az irodai folyamatok, az ügymenetek önműködő algoritmusát. Az ügyek meghatározott rendben, a bekövetkezett eseményeket figyelembe véve zajlanak.

Általában nem teljes az automatizálás, mivel bizonyos tevékenységeket csak emberi közreműködéssel lehet végrehajtani. Az ügyintézők nem változtathatják meg az ügy előírt menetét, az csak az események függvényében, a rendszerben megadott feltételeknek megfelelően módosulhat.

A hagyományos iroda funkciói

- az iratok érkeztetése,
- iktatás, előzményezés,
- határidőzés, kiszignálás,
- kézbesítő-átadójegyzékek készítése, elosztás,
- vélemények, javaslatok bekérése,
- határidő figyelés,
- ügykövetés,
- határozathozatal, lezárás, elintézés,
- kiadványozás,
- irattárolás, selejtezés.

Az automatizált iroda főbb funkciói

- ügyiratbevitel,
- számítógépes iktatás,
- irattárolás, dokumentumkezelő rendszerek,
- archiválás,
- iratkészítés,
- elektronikus aláírás,
- elektronikus levelezés,
- elektronikus adatcsere,
- ügymenetkezelés (workflow),
- védelmi rendszer.

Összehasonlítva a hagyományos és az automatizált iroda funkcióit, látható, hogy a funkciók csak részben fedik egymást. Emiatt nem könnyű a hagyományos irodát automatizált irodává alakítani. Nincs a piacon olyan rendszer, amely standard módon képes kielégíteni bármely irodai ügymenetet, iratfeldolgozást. Az új rendszer bevezetése csak átfogó rendszerfejlesztéssel valósítható meg, ami jelenti a hagyományos irodai tevékenységek felülvizsgálatát is.

Workflow

A workflow vagy munkafolyamat-automatizálási rendszer rögzíti az ügymeneteket, automatikusan kezeli az ügyfél halmazt, ellenőrizhetővé, átláthatóvá teszi az irodai folyamatokat, lényegesen olcsóbb és gyorsabb a bevezetésük egy új információrendszer elkészítéséhez képest. A workflow a munkafolyamat-vezérlő program révén alkalmas modellezett eljárások értelmezésére, a munkafolyamatok résztvevőinek összekapcsolására, megfelelő információtechnológiai eszközök és alkalmazások integrálására.

Workflow management

A workflow management (ügyviteli folyamatirányítás, ügymenet-irányítás, ügyviteli folyamat-menedzser) az irodai rendszerek elemzésére, tervezésére,

dokumentálására kidolgozott átfogó, strukturált módszertan, amely a folyamatorientált tervezést részesíti előnyben. Két feladatot valósít meg:

- egy irodára specifikált rendszerszervezési módszertan, amellyel a teljes iroda vagy az irodán belüli folyamatok logikai modellje megalkotható;
- egy olyan case eszköz, amely közvetlen kapcsolatot teremt a fejlesztő és az iroda keretrendszere között, segítségével a logikai modell fizikai modellé alakítható, egyben realizálható is.

Irodai rendszerek fejlődésének szakaszai

1. generáció: A számítógép nélküli hagyományos irodákat az jellemzi, hogy az irodai tevékenységeket manuális eszközökkel próbálták megvalósítani, de bizonyos tevékenységeket kiváltottak cél-automatákkal.
2. generáció: A számítógépek megjelenésével megvalósult a tevékenységek számítógépes kiszolgálása. Lehetővé vált az iratok elektronikus tárolása, párhuzamosan a papírmásolatokkal. A kommunikáció továbbra is alapvetően szóban történik. Az ügymenetkezelés időnként nehézkessé vált a vegyes rendszer és a gyakori programhibák miatt.
3. generáció: A számítógépes hálózatok kialakítása nagy előrelépést jelentett az iroda automatizálásának irányában. Szinte mindent meg lehetett oldani, „barkácsolással”. A korszerű megoldások sem voltak a külvilág számára szabványosak.
4. generáció: Ezek a rendszerek már teljes megoldást kínáltak az egyedi megvalósítás helyett. Az egyes gyártók által kidolgozott megoldások zárt rendszert jelentenek. Több rendszer összekapcsolása nehézkes.
5. generáció: Alapvető változást jelent az új technikai lehetőségek integrálása és a nemzetközi szabványok figyelembe vétele. Ez lehetővé teszi, hogy bármilyen rendszer bárhonnan elérhető és gond nélkül összekapcsolható egy másik rendszerrel.

Integrált irodai rendszerek

A 4. generációtól kezdve az irodaautomatizálási rendszerek megnevezésére az integrált irodai rendszerek (IIR) kifejezést is használják. Ezek a rendszerek általában több alrendszerből épülnek fel, az egyes modulok az automatizált iroda funkcióit valósítják meg. Az IIR moduljait négy csoportba lehet sorolni.

1. Irodai programok: Az egyes irodai tevékenységeket végrehajtó programok tartoznak ide. Office programok (szövegszerkesztő, táblázatkezelő, stb), egyedi fejlesztésű programok.
2. Groupware eszközök: A csoportmunkát és a kommunikációt megvalósító programok tartoznak ide. Hálózat, internet hozzáférés, elektronikus levelezés, telefax, csoport-naptárak, EDI (elektronikus adatsere).
3. Dokumentum- és adatkezelő rendszer: Ebbe a csoportba az irattárolást és az adattárolást megvalósító és segítő programok tartoznak. Az irodai rendszerek legfontosabb részét képezik. Irattárolás, archiválás, iktatás, adatbázis-kezelés.
4. Workflow rendszer: Az ügymenetkezelést megvalósító és működtető programok tartoznak ide. Ügymenetvezérlés, eseménykövetés, naplózás.

Elvárások egy automatizált irodai rendszerrel szemben

Az IIR moduljai egy egységes ún. keretrendszerre építhetők fel. Egy automatizált irodai alkalmazás megvalósításához három területet meghatározása szükséges.

1. Az irodai rendszer architektúrális felépítése.

2. Az alkalmazni kívánt szoftver eszközök.
3. Az irodai rendszer kialakításának hardvereszköz igénye.

Az irodaautomatizálási feladatok megoldására a kliens/szerver architektúra kiépítése javasolt. Ez lehetővé teszi a szervezetben már régebb óta használt irodai eszközök és az új alkalmazások integrálását. Az így kiépített rendszer biztosítja a háttérkapacitás rugalmas bővítését a növekvő igények kiszolgálását. A közös adatok a szerveren találhatók, ehhez szinte korlátok nélkül bárholnan hozzá lehet férni. Az architekturális felépítést jellemzi a rendszerkapacitás, a modularitás, a szabványos interfészek, a bővíthetőség, a rugalmasság és a közös munkavégzés.

Az irodai rendszerektől elvárt szoftvereszközök: operációs rendszerek, fejlesztői eszközök, relációs adatbázis kezelés, szöveges adatbázis-kezelés, alkalmazói programok, hálózati felügyelet, adatvédelem, adatbiztonság.

Jellegzetes irodai rendszerek

Első próbálkozások, kliensprogramok

FRAMEWORK: első próbálkozás az integrált irodai munka megvalósítására. Tartalmazott szövegszerkesztőt, táblázatkezelőt, adatbázis-kezelőt, grafikák, ábrák készítését is támogatta. Nem támogatta a csoportos munkát, nem volt kommunikációs kapcsolata a külvilággal.

PRISMA OFFICE: MS WORKS programhoz hasonlító szövegszerkesztő. Tartalmaz adatbázis-kezelőt és beépített elektronikus levelezést is.

MICROSOFT WORKS: tartalmaz szövegszerkesztőt, táblázatkezelőt, adatbázis-kezelőt, modemkezelést, és hívható belőle elektronikus levelezés.

OFFICE PROGRAMOK: A leggyakoribb irodai tevékenységek megvalósíthatók, a legismertebb programsomag az irodákban.

Csoportmunka támogatása

WINDOWS 3.1: modem segítségével lehetőséget biztosít távoli számítógépre történő bejelentkezésre.

WINDOWS FOR WORKGROUPS: biztosít megosztható dokumentumkezelést, erőforrások megosztását, kommunikációs lehetőséget, elektronikus levelezést, fax-, modemillesztést, csoportnaplár használatot.

WINDOWS 95: A kommunikációs lehetőségek bővültek, alkalmazhatók iratgyűjtők, tárcsázó használható, internet hozzáférés biztosított.

Integrált irodai rendszerek

BULL OFFICE TEAM: modulszerűen felépíthető, UNIX platformon működő, kliens/szerver architektúrájú integrált rendszer.

DIGITAL LINKWORKS: irodai munkák számára kifejlesztett objektumorientált alkalmazás, felhasználói oldalon grafikus felületet használ, a szerver oldal biztosítja az adatkezelést, adatbiztonságot. Ezzel a rendszerrel kialakítható az irat- és eljáráskezelés, meghatározhatók iratkezelési- és ügymeneti módok. A felhasználók a megszokott szövegszerkesztőt, táblázatkezelőt, vagy egyéb alkalmazást integrálhatják a rendszerbe.

ICL TEAMWARE: kliens/szerver alapú irodaautomatizálási rendszer.

IBM FLOWMARK rendszere: objektumorientált fejlesztőeszköz, grafikus felhasználói felülettel, interaktív kezelő rendszerrel. A rendszerszervezésben egy konkrét integrált alkalmazásfejlesztő eszköznek tekinthető. Kliens/szerver koncepcióra épül.

LOTUS NOTES: a legismertebb és legelterjedtebb integrált irodai rendszer. Kliens/szerver felépítésű, irodai munkák elvégzésére tervezett keretrendszer, tartalmaz fejlesztő és futtató környezetet is. Széleskörűen használható, működik a legismertebb platformokon.

LOTUS NOTES

A Notes legfontosabb tulajdonságait a következőkben kerülnek ismertetésre.

- Dokumentum-nyilvántartás és –keresés: Az adatbázis az egyéni igényeknek megfelelő struktúrában tárolja a dokumentumokat. A dokumentumok tetszőleges szoftverrel készülhetnek, tartalmazhatnak szöveget, képet, táblázatot, stb. A dokumentumok az azonosító mezők alapján gyorsan visszakereshetők.
- Teljes fejlesztő eszközkészlet: Ezzel definiálhatók űrlapok, adatmezők, adatcserék más alkalmazásokkal, létrehozható automatikus ügymenet.
- Beépített elektronikus levelezés: Teljes beépített levelezési rendszert tartalmaz, amely bármely alkalmazásból használható. Összekapcsolható más levelezőrendszerekkel. Faxot lehet küldeni és fogadni.
- Replikációs technológia: A különböző helyeken működő Notes alkalmazások ugyanazon adatokat tartalmazzák, az adatbázis tartalma az egyes szerverek között automatikusan frissítődik. A replikáció során nemcsak az adatok, hanem a hozzáférési jogok és a fejlesztés során létrejövő változások is szinkronizálódnak a teljes rendszeren belül.
- Biztonsági rendszer: Az adatbázisokhoz 7 szintű hozzáférési jogokat lehet definiálni.
- Rendszerfelügyelet: A rendszergazda helyben és távolról is képes felügyelni a szerver működését. Grafikus rendszer-felügyeleti eszközt tud használni.
- Internet kapcsolat: A Notes kapcsolódhat az internethez, a szerver használható internet információ szolgáltatásra is.
- Ügymenetkezelés: Megadható a dokumentum útja, figyelhető az egyes folyamatok állása, de a rendszer automatikusan is figyelmeztetheti a felhasználót. Grafikus ügymenettervező is beépíthető a rendszerbe.
- Papíralapú dokumentumok kezelése: A Notes segítségével a papírdokumentumok elektronikus képpé alakíthatók, megtekinthetők, módosíthatók.

Workflow rendszerek

A workflow programcsomaggal az egyének vagy a csoportok egy adott cég üzleti céljai érdekében strukturált vagy strukturálatlan környezetben kezelhetnek ismétlődő, vagy nem ismétlődő eseményt.

Néhány workflow rendszer

KEYFLOW: A munkafolyamatok tervezése grafikus felületen történik.

STAFFWARE WORKFLOW: Kliens/szerver alapú rendszer. A rendszer fontosabb összetevői: grafikus workflow tervező, munkalaptervező, személyi workflow, konsolidált workflow kezelő, vezetői rendszer.

ULTIMUS WORKFLOW ON THE WEB: Grafikus modellező felülettel rendelkező szoftver. Interneten keresztül üzemeltethető munkafolyamat-kezelő rendszerrel rendelkezik.

Az irodaautomatizálás emberi tényezői

Az iroda átalakítása során figyelembe kell venni, hogy az irodában emberek dolgoznak, nekik kell üzemeltetni az új rendszert. Az emberek többsége a változást sokszor szorongással éli meg. Félnek az ismeretlentől, a kihívásoktól, egzisztenciális bizonytalanságuk ébred. Felléphet hatalomféltés vagy a kiharcolt

posztok elvesztésének lehetősége is felvetődik. Az ilyen jellegű problémák megoldása nagyon nehéz. Be kell mutatni a dolgozóknak a jelenlegi rendszer buktatóit, meg kell magyarázni a változás szükségszerűségét. Be kell mutatni, mit várhatnak a dolgozók az új rendszertől, mire képes, mennyivel gyorsabb, hatékonyabb, mennyivel egyszerűsíti a munkát. Meg kell mondani a dolgozóknak, hogy az átalakítás pontos munkát igényel, és sok nehézséggel jár. Ki kell dolgozni, kit, milyen jellegű képzésben kell részesíteni ahhoz, hogy megfelelően képes legyen használni az új rendszert.

Ellenőrző kérdések:

1. Sorolja fel a hagyományos irodai funkciók problémáit!
2. Mit jelent az elektronikus iroda?
3. Mit jelent az automatizált iroda?
4. Sorolja fel a hagyományos iroda funkcióit!
5. Sorolja fel az automatizált iroda funkcióit!
6. Mit jelent workflow és mit valósít meg a workflow management?
7. Jellemezze röviden az irodai rendszerek fejlődésének szakaszait!
8. Jellemezze röviden az integrált irodai rendszerek (IIR) csoportjait!
9. Sorolja fel az automatizált irodai rendszerekkel szembeni elvárásokat!
10. Ismertesse röviden a jellegzetes irodai rendszereket és a csoport-munkát támogató szoftvereket!
11. Jellemezzen röviden néhány integrált irodai rendszert!
12. Jellemezze a Notes IIR-t!
13. Mit jelent a workflow rendszer és adjon meg néhány ilyen rendszert!
14. Milyen emberi tényezőket kell figyelembe venni irodaautomatizáláskor?

Irodalomjegyzék

Adamcsik János(1998): Irodaautomatizálás, Budapest.
Adamcsik János (1995): Irodai információs rendszerek, irodaautomatizálás, Budapest.
Dobay Péter (1996): Az iroda, Panem-Mcraw-Hill.
Poór József (1987): Irodaszervezés, irodaautomatizálás, Budapest, Közgazdasági és Jogi Könyvkiadó.

5. SZÁMÍTÓGÉPES PROGRAMOK, PROGRAMOZÁS

Az informatikai rendszereket tárgyalva a szoftver és program fogalmak többféle jelentéssel is bírhatnak. Egy összetett rendszert különálló komponensek, önálló szoftverek elegyének is tekinthetünk, de szokás a szoftver fogalmat gyűjtő kategóriaként, a számítógépen futtatható programok összességéként is definiálni. A program meghatározása már egyértelműbb: a számítógép által értelmezhető utasítások sorozata. A program tehát a bemeneti (input) adatok feldolgozásának lépéseit jelentő algoritmusok pontosan megfogalmazott leírása valamely programozási nyelven, lényegében feladat-meghatározás a számítógép számára.

Az első számítógépek működtetéséhez használt programok célja a hardver működésének hatékony, sikeres irányítása volt, és a megfelelő szoftver előállításának költsége jóval alacsonyabb volt a – mai mértékkel – csillagászati hardverkölségeknél. A század folyamán a technológiai fejlődés megfordította ezt az arányt, a fizikai alkotóelemek tömegtermelése jelentősen csökkentette azok árát, de ezzel párhuzamosan az egyre fejlettebb architektúra és a felhasználók táborának kiszélesedése egyre nagyobb teljesítményt követelt a programok fejlesztőitől. A XX. század végére megfigyelhetővé vált, hogy a szoftverek karbantartása – azaz: hibák javítása, módosítások a meglévő struktúrában, illetve új elemek beépítése – nagyobb jelentőségű, több emberi erőforrást foglal le, mint a teljesen új programok fejlesztése²⁹.

Meglévő programok módosításához, továbbfejlesztéséhez elengedhetetlen, hogy a program világosan értelmezhető, *olvasható* legyen. Ezen követelmény – és nem mellesleg a programozással foglalkozók körének jelentős kiszélesedése – hatására a programok nyelvezete és a nyelvek mögötti filozófia is sokat változott az első számítógépek óta. A következő fejezetben a programozási nyelvek történetével, osztályozásával foglalkozunk, és az elméleti keretek ismertetése után a Visual Basic és a C nyelv környezetét felhasználva példaprogramokat, egyszerű algoritmusokat tekintünk át.

5.1 A programozási nyelvek

Az első, elemi típusú és csak alapvető utasításokat ismerő programnyelvek óta körük jelentősen bővült, több száz programozási nyelvet tartanak számon, amelyekből általánosan legfeljebb egy-két tucat ismert és népszerű. Különböző célokra különféle nyelvek születtek, így például

- főleg tudományos (matematikai) számítások eszköze a FORTRAN és az ALGOL nyelv,
- ügyviteli, gazdasági alkalmazásokhoz, adatfeldolgozáshoz készült a COBOL,
- számítógépes szimuláció, modellezési technikák megvalósítása a GPSS és a SIMULA-nyelvek célja,
- rendszerprogramozásra született a C nyelv,
- oktatási célból fejlesztették a BASIC³⁰ és Pascal nyelveket.

²⁹ Gondoljunk például a Windows-sorozatra, amelyek kernelje az 1995 és 2002 között kiadott verziókban alig változott.

³⁰ **B**eginners **A**ll-purpose **S**ymbolic **I**nstruction **C**ode

A felsorolás természetesen nem teljes, és a felhasználási szempontok mellett számos egyéb szempont alapján is kialakíthatunk programnyelv-kategóriákat. A számítógép processzorának saját „nyelve” és a programnyelvek közötti távolságot, tehát azt, hogy a konkrét hardverelemek mennyire befolyásolják a program elkészítésének módját, a következő osztályozással szemléltethetjük:

- gépi kódú programozás (közvetlenül futtatható kódok)
- alacsony szintű, ún. assembly nyelvek
- magas szintű, feladatorientált nyelvek

Szokás ezt a megkülönböztetést a programnyelvek egymást követő generációiként megadni, így a gépi kód alkotja az első, az assembly a második, és a magas szintű nyelvek a harmadik generációt. Negyedik és ötödik generációs nyelveket is ismer az irodalom, a harmadik generációs nyelvek továbbfejlesztett változatai mellett speciális célú nyelvek, illetve a mesterséges intelligencia nyelvei tartoznak ide – az utóbbi legismertebb példája a PROLOG logikai nyelve, amely műszaki alkalmazásokban (robotok) és szakértői rendszerekben népszerű.

5.1.1 Gépi nyelv és assembly

Neumann János 1948-ban megfogalmazott követelményei a mai napig meghatározzák a számítógépek többségének architektúráját és működését. A belső programvezérlés elve szerint mind az adatok, mind a programok a gép belső tárolójában (a memóriában) helyezkednek el – a programok egyes lépései automatizáltan követik egymást. A számítógépek processzorai különböző nyelvekkel rendelkeznek, bizonyos utasításokat – beépített áramkörökkel – közvetlenül hajtanak végre: ezek alkotják tehát az egyes processzor-típusokra jellemző *utasításkészleteket*.

A **gépi kódok** elemei műveleti kódból és címrészből állnak (szokásos elnevezés még: műveleti és operandus rész). A műveleti kódban megadott utasítást hajtja végre a processzor a címrészben megadott 1, 2 vagy 3 operandussal – ezek határozzák meg, hogy a memória mely területén („rekeszében”) találhatóak a számításokhoz szükséges adatok. A gépi kódú program ezen utasítások sorozata, az adott utasítás végrehajtása után automatikusan a következő utasítást dolgozza fel a processzor.

Bár a gépi kód képes kihasználni a számítógép minden adottságát, megkívánja a számítógép teljes ismeretét, a fizikai felépítését is beleértve – nem meglepő, hogy gyakorlatilag az ötvenes évek óta nem vagy nagyon ritkán használják. Az egyes utasítások könnyebb megfogalmazására, jobb olvashatóságára és dokumentálhatóságára született meg az **assembly**, a gépi nyelvhez legközelebb álló, alacsony szintű nyelv. Egy assembly-program adatdefiníciókat és programutasításokat (adatmozgatás, aritmetikai és logikai utasítások, feltételes ugrások) tartalmaz, és általában egyértelmű megfeleltetéseket alkalmaz a gépi nyelvű utasításokra: műveleti kódok helyett 3-4 karakteres rövidítéseket, a memóriacím helyett pedig változókat. Egyes, gyakran használt utasítássorok ún. makrókba szerevezhetők, így használatuk jelentősen egyszerűsödik (az utasítássor alkalmazásához csak a makró nevét kell megadni). Az assembly nyelvű program végrehajtásához a szöveges forrást le kell fordítani gépi nyelvre, ezt az *assembler* végzi.

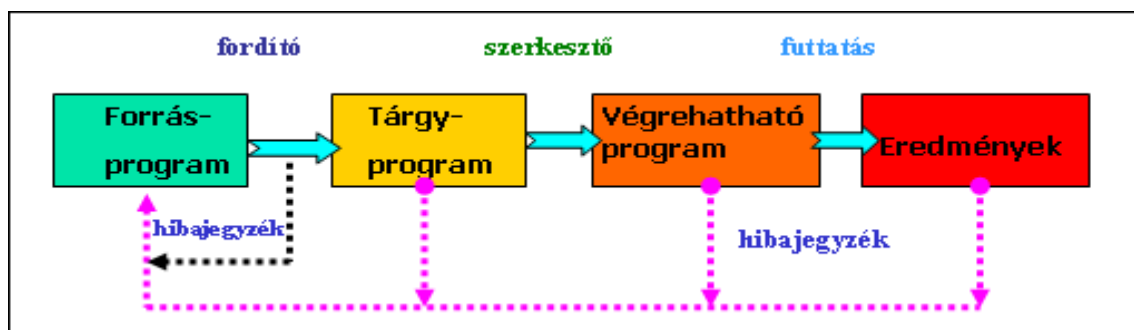
Elsősorban a hardverek működését szabályozó rendszerszoftverek, illesztőprogramok és operációs rendszerek készültek/készülnek assembly nyelven, de használata visszaszorulóban van platformfüggősége (a különböző processzorcsaládok, mint pl. Intel 8086 és utódai, IBM Power PC vagy a Sun SPARC assembly-nyelvjárásai legfeljebb szerkezetükben hasonlóak) és a magas szintű nyelvekhez képest nehéz olvashatósága,

módosíthatósága miatt. Azonban a mai napig léteznek olyan feladatok, amelyeknél a magas szintű nyelvek egy adott probléma megoldására nem nyújtanak megoldást, vagy – különösen méret- vagy időkritikus feladatoknál – nem gazdaságos az alkalmazásuk; ekkor célszerű lehet a magas szintű programkód kiegészítése egy assembly nyelvű egységgel.

5.1.2 Magas szintű programnyelvek és fordítók

Az első magas szintű programnyelv megalkotásakor a készítőket programok hatékonyabb fejlesztése vezérelte. A FORTRAN („Formula Translator” összevonásából) és utódai *általános célú nyelvek*, azaz saját szabályokkal rendelkeznek, amelyek mentén a programok elemei felépíthetők. Az így megalkotott programok gépfüggetlenek, szavadon hordozhatók. A felhasználó az adott nyelven ún. **forrásprogramot** készít, ehhez legtöbbször egy egyszerű szövegszerkesztő is elegendő, de a legnépszerűbb nyelvekhez önálló fejlesztői környezetet készítettek, amelyek használata jelentős segítség lehet a programírásban. Ha nem is nélkülözhetetlenek, de nagy jelentőségűek tehát a fejlesztői környezetek beépített funkciói, mint például a programkód hibáinak ellenőrzése, a „nyers” kód szövegének formázása és főleg az adott nyelv szabályairól tájékoztató segédletek és minták. A kész forrásprogramot át kell alakítani a gép processzora számára értelmezhető bitsorozattá, ez kétféleképpen valósulhat meg.

A **fordítóprogram** (compiler) ellenőrzi a forrásprogramot: megfelel-e a nyelv **szintaxisának**, azaz szabályainak. Hibátlan kódsor esetén a fordító **tárgyprogramot** készít, ami még szintén nem futtatható (legalábbis közvetlenül a processzor számára nem az), szerkesztésre szorul: különböző, az adott nyelvre vagy rendszerszoftverre jellemző rutinokkal egészül ki. A fordítás végeredménye a **végrehajtható program**. Ilyen mechanizmussal dolgozunk C, FORTRAN, Pascal nyelvek esetén.



2.1 ábra: A programfordítás szakaszai

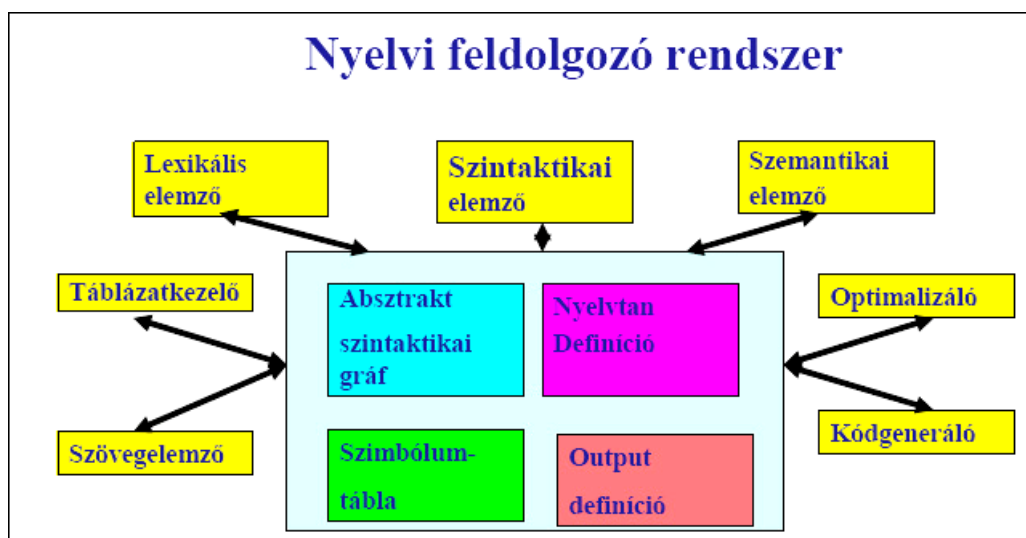
Az **értelmezőprogram** (interpreter) soronként halad a forráskódban, és az egyes sorok feldolgozása után azonnal végrehajtja az utasítást, tehát a programunk gépi kódú megfelelője soronként áll elő. Emiatt a szintaktikai hibákról itt nem jegyzék készül a fordítás után, mint a compilernél, hanem azok a fordítás során azonnal jelentkeznek. Az interpreter előnye tehát a hibakezelésben érvényesül, de komoly hátránya, hogy a végrehajtható program a fordítási művelet után már nem hozzáférhető, nem lesz tárolt változata. Így minden egyes futtatáshoz külön futtató környezet szükséges, ami legtöbbször maga a fejlesztői környezet. Interpretert

használnak többek között a BASIC nyelv régi változatai, de a kortárs webprogramozási nyelvek (PHP, Perl) is³¹.

A fordító- és értelmezőprogramoknak, bár más elvet követnek, vannak közös jellemzőik. Mindkettő alkalmaz forráskódkezelő alkotóelemeket, ezek közül néhány lényegesebb:

- Lexikális elemző: a karaktersorozatot szimbólumsorozattá alakítja (szimbólumtábla használatával). Meghatározza a szimbolikus egységeket (változók, kulcsszavak, operátorok).
- Szintaktikai elemző: feladata a nyelvi elemek felismerése és elemzése, szintaktikai hibák megállapítása.
- Szemantikai elemző: az általa keresett hibák elsősorban kompatibilitási, végrehajtási problémákhoz vezetnek (végtelen ciklus, logikailag értelmezhetetlen utasítás).
- Hibakezelő: hibák jelzése, amelyek súlyossága lehet „normál” vagy „fatális”.

A felsoroltakon kívül a compiler tartalmaz kódgeneráló és kódoptimalizáló elemeket is, előbbi a tárgykód előállítását végzi, utóbbi a futtatás gyorsaságát és erőforrásigényét igyekszik javítani. Csak az értelmezőkre jellemző a speciális végrehajtó egység.



5.2 ábra: Egy komplex fordítórendszer

5.1.3 Neumann-elvű és nem Neumann-elvű nyelvek

A magas szintű nyelvek mindegyik változatában valamilyen fokú absztrakció érvényesül – a gépi kód „nyers” lépéseire képest valamilyen gondolkodási módot, stílust sugallnak. A *paradigma* – amely egy tudományterület sarkalatos megállapítása – a programozási nyelvekben ezeket a módszertanokat jelenti, azaz meghatározza, hogyan használták fel a programozási alapfogalmakat a nyelvek létrehozásakor, milyen modellek szerint kell végrehajtani az algoritmusokat. Általában két fő paradigma-csoportot

³¹ Megjegyezzük, hogy sok nyelvhez mindkét fordítótípus létezik, és arra is van példa, hogy egyidejűleg használják: a JAVA nyelv esetében például egy ún. hibrid kódra történik a fordítás, majd ezt értelmezi a gép.

különíthetünk el, ezek az **imperatív** és a **deklaratív** elvű nyelvek, előbbihez a Neumann-féle architektúrához szorosan kötődő, algoritmikus nyelvek, utóbbihoz pedig főleg matematikai logikára épülő, eredményorientált, nem algoritmikus nyelvek tartoznak.

Az imperatív nyelvek fő programozási egységei az utasítások, amelyek egymásutánisága vezérli a processzort. Egy ilyen elvű programban a központi problémát az eredmény elérésének módja, az oda vezető út megadása jelenti. Változókat használ az adatok kezelésére, amelyeknek az értéke módosítható. Ide tartoznak az *eljárás-orientált* nyelvek (eljárás: szövegegység a forráskódban), mint a FORTRAN, COBOL vagy a BASIC nyelvek első változatai³², ezek mellett két fő modern irányzatát különíthetjük el: a *strukturált programozás* elvét (mint az eljárás-orientált paradigma egy formáját, egyik fő képviselője a Pascal nyelv), illetve az *objektum-orientált programozás* elvét (C++); mindkettőről lesz még szó a későbbiekben.

A deklaratív nyelveknél a programozó a megoldandó feladatot írja le, a megoldáshoz vezető út meghatározása a rendszer dolga. Így ezekben a nyelvekben az utasítás fogalma is ismeretlen, nem létezik vezérlési szekvencia, nincs szerepe a sorrendiségnek. Többek között ide sorolhatók a *funkcionális*, *logikai* és *automata-elvű* nyelvek, amelyekben tehát közös, hogy nem kötődnek a Neumann-féle architektúrákhoz.

- A funkcionális nyelvek magas szintű függvények használatára és operátor definíciókra épülnek, minden részproblémát függvényként írnak le.
- A logikai nyelvekben tényekből kiindulva következtetéseket vonhatunk le meghatározott szabályok mellett. A program itt egy logikai formula, amelyet ki kell értékelni; ebből adódóan erős matematikai kidolgozottságot igényelnek. Az ilyen programokban az adatok is formulák, nem különülnek el, így nincsenek változók, nem létezik értékadás sem. Ezen nyelvek egyik népszerű alkalmazását a szakértői rendszerek adják.
- Az automata-elvű nyelvek másik elnevezése „ipari robot nyelvek”. Egy meghatározott „állapottér”-ben működnek, az adatok állapotok, a végrehajtás egyszerű állapotok sorozata. A program maga egy állapotátmenet-függvény, elkülönül az állapotoktól. Változók, értékadások nincsenek, csak névvel ellátott állapotkomponensek – a nyelv utasításai általában egy állapotkomponenst változtatnak.

Megjegyezzük, hogy a programozási paradigmák fentebb tárgyalt képviselőin kívül számos más kategóriát ismer az irodalom. Léteznek például olyan nagyon magas szintű nyelvek (negyedik generációs paradigmaként is hivatkoznak ezekre), amelyek esetében a megoldandó feladat természetes nyelven, vagy diagramok használatával fogalmazható meg, többnyire az adatbázisok lekérdező nyelveit is ide sorolják.

Napjaink programozási eszközei főképp interaktív fejlesztői környezetek, programkód-generátorok, amelyek már túlmutatnak a magasszintű nyelvek osztályán. Azonban a programozó eszköztárát továbbra is a fenti paradigmák alapján határozhatjuk meg, és a bonyolult fejlesztőkörnyezetek használatához az alapvető programozási fogalmak és módszerek elsajátításán keresztül juthatunk el. A következő fejezetek a strukturális és objektum-orientált elvekbe nyújtanak bevezetőt, számos példával illusztrálva.

³² Az „első változat” megjelölés nem véletlen. A felsorolt nyelvekből a '70-es, '80-as években strukturált változatok is készültek, a BASIC legújabb változatai (Visual Basic) pedig részben objektum-orientáltak.

5.2 Alapvető programozási struktúrák

A magas szintű programnyelvek terjedésével az eljárásorientált technikák minél hatékonyabb alkalmazása és a programok olvashatósága, érthetősége a sikeres fejlesztői munka feltételévé váltak. A programok megalkotása során a kiindulási problémát egymástól jól elhatárolt részproblémákra bontva lehetőségessé vált a fejlesztési feladatok megosztása. Ezeket a külön kezelhető, cserélhető, tesztelhető részegységeket moduloknak, magát a technikát modularizációnak nevezzük.

Az algoritmusokban olykor ugrásokat definiálunk: ha például valamilyen feltételtől függően kell egy számítás „A” vagy „B” változatát elvégezni, akkor a két változatból csak az egyiket fogjuk futtatáskor alkalmazni, míg a másikat „átugorjuk”. Nem mindegy azonban, hogy a forráskód szövegében hogyan kezdeményezzük ezt az ugrást. Az első nyelvekben széleskörűen alkalmazott címkézős technika a GoTo (go to = menj oda) utasítással irányította át a folyamatot a kódszöveg valamely megjelölt területére, azonban az ilyen ugrópontok halmozása átláthatatlanná teheti az algoritmust, és korlátozza a modularizációt. Dijkstra³³ 1968-as cikkében a GoTo-féle ugrások ellen érvelt, és olyan programozási elvet ajánlott, amely szigorúan moduláris, világos felépítésű, jól elhatárolt részelemekkel. Ezek az alstruktúrák egyetlen bemenő és kimenő ponttal rendelkeznek, és hierarchikusan szervezett alá- és mellérendelések alapján kapcsolódnak össze teljes programmá.

A strukturált programozás mindhárom alapeleme zárt egységet alkot, amelyben egyszerű input / output / értékadó utasítások vagy eljáráshívások, illetve beágyazott struktúrák szerepelhetnek. A három alapstruktúra:

- **Szekvencia:** egymást követő utasítások sorozata, végrehajtásuk sorrendje adott.
- **Döntés:** valamilyen logikai feltétel kiértékelésétől függően hajtjuk végre a programkód egyik vagy másik szakaszát. Ezeket a szakaszokat ágaknak, magát a struktúrát elágazásnak is nevezzük.
- **Ciklus:** egy adott utasítássorozat ismétlése valamilyen feltétel függvényében.

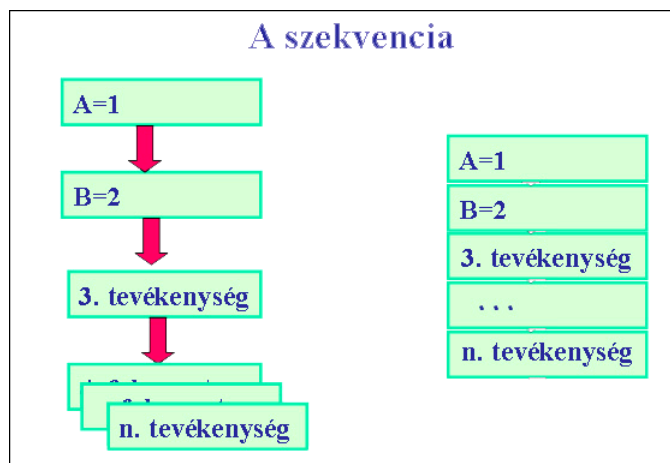
A következő pontokban röviden áttekintjük az alapstruktúrákat és alkalmazásuk körülményeit. A struktúrafajták általános jellemzése következik, programnyelvtől függetlenül. Egyúttal bemutatjuk az algoritmusok vizuális megadási formáinak (folyamatábra, struktogram) az adott alapstruktúrákhoz rendelt szimbólumait³⁴.

5.2.1 Szekvencia

A szekvencia a legegyszerűbb alapstruktúra, egymást követő utasítások együttese. A végrehajtás sorrendje a forráskódot olvasva triviális, a rajzos leírásokban általában „fentről-le” irányú, vagy nyilakkal megadott.

³³ Edsger Wybe Dijkstra (1930 -2002) holland tudós

³⁴ Itt jegyezzük meg, hogy a folyamatábrák elemeire nincs konkrét szabály, többé-kevésbé általánosak azonban az alábbi megfeleltetések: bekérés/kiírás/értékadás: tégalalp vagy paralelogramma, feltételes elágazás: elforgatott négyzet – utóbbit a ciklusok feltételének jelzésére is használják. Az egyes elemeket folytonos, egyenes vonallal vagy nyilakkal kötjük össze.



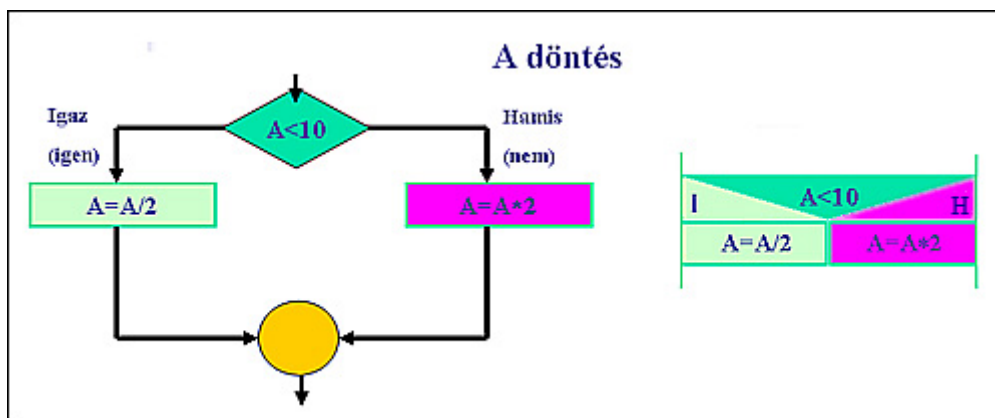
5.3. ábra: A szekvencia

Az algoritmusok grafikus illusztrálására alkalmazott két fő módszer látható a 2.3 ábrán, a baloldali folyamatábrának, a jobboldali struktogramnak hívjuk. A folyamatábra az egyes alapstruktúrákhoz különféle alakzatokat rendel, az egyszerű utasításokat általában téglalap vagy paralelogramma szimbolizálja. A struktogram mindig egy felosztott téglalap, a belső elemek formái itt is a struktúraelemek fajtáját mutatják.

A szekvencia alapvetően háromféle utasítástípus kombinációja, ezek: *beolvasás*, *kiírás* és *értékkadás*. Az értékkadás valamilyen változó (információ ideiglenes tárolására szolgál, lényegében egy vagy több adott memóriacímet szimbolizáló szöveges azonosító) értékét egy kifejezés értékére állítja. Általában az „=” vagy „:=” operátorok használatosak értékkadáskor, ilyenkor az egyenlőségjel jobb oldalán álló kifejezés kiértékelése után az eredmény a változóhoz rendelt memóriarekeszbe kerül. A beolvasás legegyszerűbb formája egy a billentyűzeten leütött karaktert ad értékül egy változónak, de fájlból is olvashatunk adatokat. A kiírás változók, kifejezések értékét vagy a forráskódban megadott szöveget jeleníti meg a képernyőn, vagy írja ki fájlba, küldi el nyomtatásra.

5.2.2 Döntés

A döntés az algoritmus csomópontja, amelyből több ágon haladhatunk tovább. Gyakran *feltételes elágazásként* (vagy a programnyelvekben szinte kivétel nélkül használt If kulcsszó alapján: *If-elágazásként*) hivatkozunk rá, mert valamilyen feltétel áll mindig a középpontban, a legegyszerűbb elágazásokban e feltétel teljesülése vagy nem teljesülése határozza meg, melyik ágon folytatódik a program.



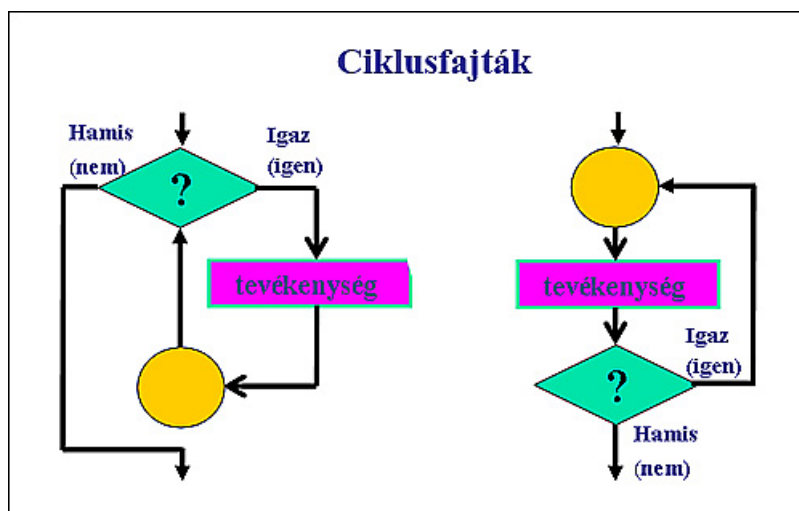
5.4. ábra: Feltételes elágazás

Az ábrán látható példában egy „A” azonosítójú változó értéke szerepel a feltételben: ha kisebb, mint 10, a változó értékét 2-vel osztjuk, egyébként 2-vel szorozzuk. Bár most mindkét ágban csak egy-egy utasítás szerepel, természetesen állhatna a helyükön szekvencia vagy akár egy másik elágazás is. Emellett gyakran előfordul olyan döntési pont is az algoritmusokban, ahol a két ág közül csak az egyikbe írunk utasításokat. A fenti példánál maradva, ha a hamis ág utasítását töröljük, az elágazás értelmezése: osszuk el az „A” változót kettővel, de csak akkor, ha 10-nél kisebb.

Az egymásba ágyazott elágazásoknak sok programnyelvben létezik egy speciális formája, több, a fentihez hasonló alapforma összevonásából. Ezt *kiválasztásnak* nevezzük (*Switch* vagy *Select* parancsok leggyakrabban), és a 5.4 fejezetben látunk rá példát.

5.2.3 Ciklus

Ciklusokat akkor alkalmazunk, ha egy adott utasítássorozatot – amely természetesen lehet egyelemű is – egymás után többször kívánunk végrehajtani. Az ismétlést feltételhez kötjük, és e feltétel helye, illetve megadásának módja szerint beszélhetünk különféle ciklusfajtákról.



5.5. ábra: Ciklusok

Az ábrán látható két típusban eltérő helyen vizsgáljuk meg a folytatáshoz szabott feltételt. Ennek megfelelően a baloldali változatot *előtesztelő*, a jobboldalit *háttesztelő* ciklusnak nevezzük. E típusok között egy fontos különbséget tehetünk, és ez az ábrán is jól

megfigyelhető: a hátultesztelő ciklus utasításai – ha nem használunk ugrásokat – legalább egyszer biztosan végre lesznek hajtva, míg az előlesztelőnél lehetséges, hogy egyszer sem fut le a ciklus.

A ciklusok kezdő utasítását *ciklusfej*nek, míg az ismétlődően végrehajtott utasításokat *ciklusmagnak* vagy *ciklustörzsnek* nevezzük. A ciklus feltételének megadására legtöbbször valamilyen kiértékelendő logikai kifejezést használunk (sok nyelvben a *While* kulcsszót követi ez a kifejezés), de az előlesztelő változatnak számos programnyelvben létezik egy speciális, *számlálós ciklus* néven ismert típusa (amelyet a *For* kulcsszóval adunk meg, ezért gyakran *For-ciklusnak* is hívjuk). Ez utóbbi típus esetén a ciklusfejben egy index kezdő- és végértékét, valamint lépésközét megadva konkrétan meghatározhatjuk, hányszor ismétljük meg a ciklusmag utasításait.

5.3 Az objektum-orientált programozás alapelvei

Az objektum-orientált programozási paradigma (gyakori rövidítéssel: OOP) a strukturált programozáson alapul, de szemléletmódjában meghaladja azt. A mára már „klasszikusnak” tekintett strukturált elvek szerint a program lényegében függvények, eljárások gyűjteménye, vagy csak a gép számára kiadott direkt utasítások sorozata – ezzel szemben az objektum-orientált megközelítési mód a programot önálló, egymással kölcsönhatásba kerülő, de zárt elemek, ún. objektumok együttműködésével adja meg. Ezek az objektumok jól elkülönített feladatokat és hatásköröket kapnak, és képesek üzenetek fogadására, adatok feldolgozására, és üzenet küldésére más objektumoknak. Erre a módszertanra a strukturált programozásnál jobban jellemző a felhasználóbarát jelleg, a programozó számára nagyobb rugalmasságot ad, a programok egyszerűbb átláthatósága, karbantarthatósága könnyebbé teszi a fejlesztői munkát. Míg a korábbi paradigmák elveit követve programíráskor az utasítások számítógépes végrehajtásának módját kellett szem előtt tartani, az objektumorientált elv az emberi gondolkodásmódot állítja a középpontba. Ezzel megnöveli az absztrakciós szintet a programozásban: az objektum-orientált nyelvek távolabb esnek a gépi kódtól, mint a korábbi eljárás-orientált vagy funkcionális nyelvek.

Következzen az OOP-elvek néhány fontos jellemzője:

Az objektum-orientált nyelvek struktúrájának alapjai az *osztályok*, amelyek adatdefiníciók és működési definíciók együttese. Az objektumok egy-egy osztály konkrét megjelenési formái, amelyek saját adatokkal, az osztálydefinícióban megadott szabályok mentén léteznek.

Az objektumok kívülről fekete doboznak tekinthetők: belső folyamataikhoz saját adat- és eljárásstruktúrával rendelkeznek, az egymással való kommunikációt pedig e célra kialakított interfészeken (csatlakozó felületeken) bonyolítják. Így az objektum adatok és függvények együttese „összecsomagolva”, és az interfésztől függ, hogy a csomag mely elemeihez férhetnek hozzá (csak olvasás vagy módosítás) más objektumok. Ez a *kapszulába záras elve* (egységbe záras elve, enkapszuláció).

Az egyes objektumok közötti hasonlóságok gyakran arra vezethetők vissza, hogy az osztályaik között átfedések vannak. Például egy ún. alaposztályból származtathatunk új osztályokat, amelyek az alaposztály valamennyi definícióját hordozzák, és ezek mellett saját jellemzőkkel bírnak – ez az *öröklődés*³⁵. Amikor a hasonlóság azt jelenti, hogy különböző

³⁵ Több alaposztály tulajdonságait is egyesítheti egy származtatott osztály, ekkor *többszörös öröklésről* beszélünk. Megjegyezzük, hogy a származtatott osztályokból lehetséges újabbakat

osztályokba tartozó objektumok egy közös interfészen keresztül reagálnak valamilyen üzenetre, *polimorfizmus*ról beszélünk. Ekkor ugyanazon műveletre más-más eredmény kapható az egyes objektumoktól, amelyek lehetnek a közös interfésztől eltekintve függetlenek, de gyakran valamilyen közös alaposztály származtatott osztályaiba tartoznak.

Az 5.1.3 pontban példaként említettük objektum-orientált programnyelvre a C++ nyelvet, amely a mai napig az egyik legismertebb és legnépszerűbb fejlesztői nyelv. További, népszerű objektum-elvű nyelvek a *Delphi* és a *Java*, meg kell még említenünk a C++ utódjának szánt C#³⁶ nyelvet, és részben ide sorolható a *Visual Basic* nyelv is. E jegyzetben nem célunk az objektum-orientált programozás mélyebb ismertetése, azonban néhány vonására ki fogunk térni a fejlesztői rendszerek kapcsán.

származtatni, ekkor az osztályok hierarchikus *örökléslánca* jön létre. *Absztrakciónak* nevezzük az OOP-elvű programnyelv azon képességét, amely lehetővé teszi, hogy az örökléslánc szintjeit félretéve egy származtatott osztálybeli objektumot egy általánosabb szinten kezeljünk. Például a származtassuk a „jegyzet” osztályt a „tankönyv” osztályból, amely viszont a „könyvek” osztályból származik. Ekkor egy jegyzet kezelhető könyvként, ha általános, minden könyvre jellemző tulajdonságait tekintjük.

³⁶ C# kimondva: C-sharp

6. ADATBÁZIS KEZELÉS

6.1. Az adatbázis

Ebben a fejezetben először röviden ismertetésre kerülnek a különböző adatmodellek. Továbbiakban a relációs adatmodell részletesebb bemutatása történik, majd az adatbázis-kezelő rendszerek felépítésének ismertetésére következik. Ezután az adatmodellek normalizálása, a normálformák jellemzése, valamint a relációs algebrai műveletek áttekintésére kerül sor.

Az adatbázisok ma már az üzleti élet minden területén elterjedtek. A pénzügyi, kereskedelmi folyamatokat adatbázisok alkalmazásai segítik. Megtalálhatóak a kereskedelemben is, ahol a vásárlók és az ügyfelek számára szolgáltatnak adatokat. Adatbázisokat használnak az egészségügyben és a tudományos kutatásokban is.

Adat

Adatnak neveznek olyan feldolgozott információt, amelyet érdemesnek tartanak valamilyen formában megőrizni. Ma már leginkább elektronikus módon tárolják az adatokat.

Adatbázis

Az *adatbázis* (*DB - Database*) lényegében nem más, mint hosszú ideig megőrzendő adathalmaz, vagyis egy adott formátumban és rendszerben tárolt adatok együttese. Az adatbázis szervezettsége lehetővé teszi a tárolt adatokhoz való hozzáférést és ezek módosítását. Hallgatólagosan feltételezzük, hogy az adatbázis egy számítógép háttértárában van, és ott tartósan megmarad akkor is, ha a gépet kikapcsolják, vagy az adatbázist tartalmazó adathordozót lekapcsolják a számítógépről.

Az adatbázis-kezelés kétféle adatbázistípust különböztet meg: *operációs adatbázist és analitikus adatbázist*. Sok vállalat, szervezet használ operációs adatbázisokat. Ezeket elsősorban jellemzi az elektronikus tranzakció-feldolgozás, tehát az adatok begyűjtése, módosítása és karbantartása napi rendszerességgel történik. Az ilyen adatbázisban tárolt adatok dinamikusak, ami azt jelenti, hogy folyamatosan változnak, és mindig a friss állapotot tükrözik. Az üzletek, gyárak, klinikák, kórházak ilyen operációs adatbázist alkalmaznak, mivel adataik állandó változásban vannak.

Az analitikus adatbázisokat elsősorban elektronikus elemzési feldolgozásra használják. Ilyenkor az a fontos, hogy az adatok időbeli változását nyomon lehessen követni. Ezek akkor hasznosak, ha trendeket kell elemezni, statisztikákat kell készíteni, vagy stratégiai döntéseket kell hozni. Az ilyen adatbázisok statikus adatokat tartalmaznak, ami azt jelenti, hogy tartalmuk soha (vagy csak nagyon ritkán) változik. Az analitikus adatbázisok pillanatnyi állapotot tükröznek. Kémiai laboratóriumok, geológiai vállalatok, piacelemző cégek dolgoznak analitikus adatbázisokkal.

6.2. Adatmodellek

Adatmodell

Az adatmodell az adatok szervezésének logikai képe, amely alapján kell megvalósítani az adatok fizikai megjelenítését. A valós világ részeit logikailag strukturált formában leíró adatmodellek mindegyike egyed típusok összességének tekinthető. A különböző adatmodellek megegyeznek abban, hogy a valós világ

részei leírhatók az egyed-tulajdonság-kapcsolat (entity-attribute-relationship) hármas segítségével. Az adatmodellek alapelemeit a koncepcionális adatmodellben határozzák meg. Ebből lehet megalkotni a logikai adatmodellt, és ezt a fizikai adatmodellben lehet megvalósítani.

Az adatmodellek két ok miatt különböznek egymástól

- az egyedtípus tulajdonságtípusainak különböző kezelésében (az egyedtípusok belső szerkezetének eltéréseiből)
- a kapcsolatok kezelésének különbözőségében.

Az adatmodellek közötti lényeges különbség az adatok közötti kapcsolatok tárolásában van. Az adatmodellek között tapasztalható különbségek alapján több típust különböztet meg a szakirodalom:

- hierarchikus adatmodell,
- hálós adatmodell (kétszintű hálós adatmodell, CODASYL-hálós adatmodell),
- relációs adatmodell,
- kiterjesztett adatmodell.

Alapvetően csak három klasszikus adatmodell, a hierarchikus, a hálós és a relációs adatmodell ismert. A kétszintű hálós és a CODASYL-hálós adatmodell között kicsi a különbség. A kiterjesztett adatmodell a CODASYL-hálós és a relációs adatmodellek előnyös tulajdonságait egyesíti.

A hálós és a hierarchikus adatmodelleket az első kereskedelmi adatbázisrendszerekben használták az 1960-as évek végén és az 1970-es években. A relációs adatmodellek később ezeket kiszorították.

6.2.1 Hierarchikus adatmodell

A hierarchikus adatmodellen alapuló adatbázisok az 1960-as évek vége felé kezdtek megjelenni. A hierarchikus modell volt a legelső az adatbázis-kezelőkben és egyben a leginkább korlátozott. A hierarchikus adatbázis szerkezete gráfelméleti szempontból fa-gráfok összessége.

Az adatokat ebben a modellben hierarchikus módon tárolják, és egy lefelé fordított fával ábrázolják. A fa gyökere ennek az adatbázisnak egy táblája, a többi tábla a gyökérből kiinduló ágak mentén helyezkedik el.

A hierarchikus adatbázis szülő-gyermek típusú kapcsolatokból épül fel. Ebben a kapcsolatban a szülőtáblához egy vagy több gyermektábla tartozhat, a gyermektábláknak azonban csak egy szülőjük lehet. Grafikusan úgy jelölik, hogy egy nyíl mutat a szülőtől a gyermekre. A táblákat mutatókkal kapcsolják össze. A hierarchia tetején áll az ősszüelő, amit másképpen egy fa-gráf gyökerének neveznek. A felhasználó az adatok olvasását mindig a gyökérben kezdi, és innen halad lefelé a kívánt adatig. Az utódok nélküli szülőket a fa-gráf leveleinek (leaf) nevezik (6.1. ábra)

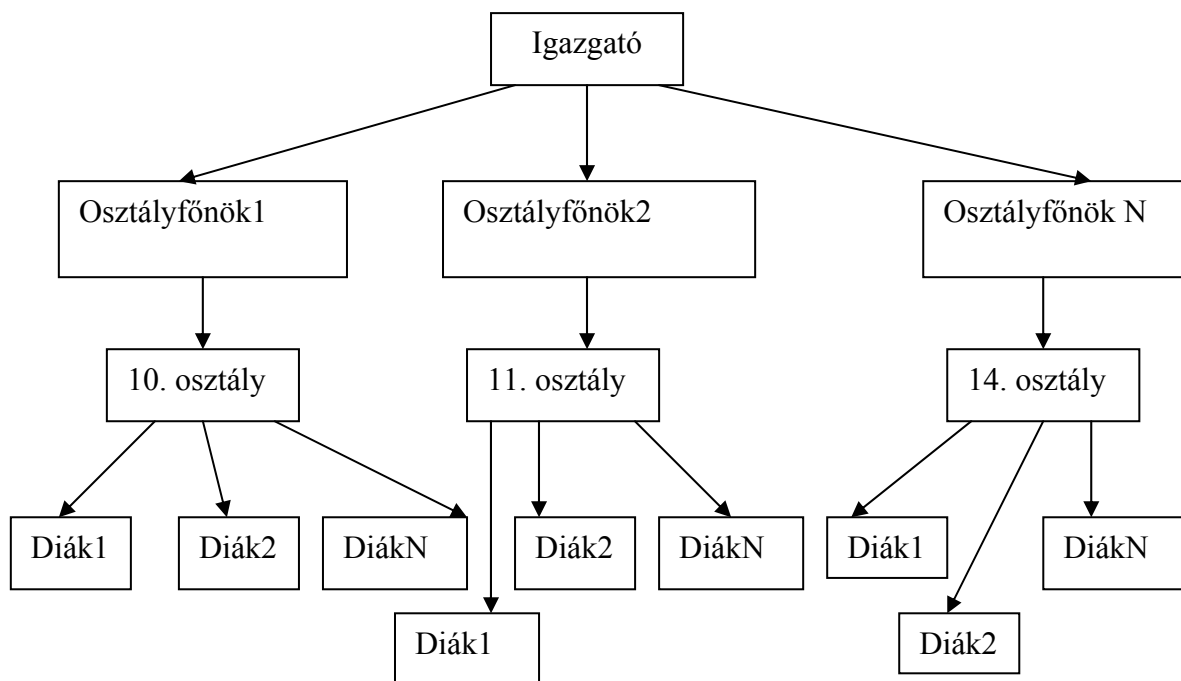
Az egy gyökérhez tartozó fát adatbázis-rekordnak (database record) neveznek. Helyesebb, adatbázisrekord-típusnak nevezni. Az adatbázisrekordok típusainak összessége képezi az adatbázis sémáját. A sémában minden gyökértípusból csak egy van. Az adatbázisban minden adatbázisrekordnak vannak előfordulásai. Az adatbázis – a gráfok terminológiája szerint – egy erdő.

Az adatmodell jellemzői

- Az adatbázis több egymástól független fából állhat.
- A fa csomópontjaiban és leveleiben helyezkednek el az adatok.
- A közöttük levő kapcsolat szülő-gyermek kapcsolatnak felel meg.
- 1:N típusú kapcsolatokat kezel. Az adatszerkezet egyik típusú adata a hierarchiában alatta lehelyezkedő egy vagy több más adattal áll kapcsolatban.
- A hierarchikus adatmodellben nem ábrázolhatunk M:N típusú kapcsolatokat.
- Hátránya, hogy az adatok elérése csak egyféle sorrendben lehetséges, a tárolt hierarchiának megfelelő sorrendben.

A hierarchikus adatbázisok használatának előnye, hogy a felhasználó nagyon gyorsan hozzájuthat az adatokhoz, mivel a táblák közvetlenül kapcsolódnak egymáshoz. További előny a hivatkozási-épség automatikus megvalósulása. Ekkor ugyanis teljesül, hogy a gyermektábla egy rekordja a szülő tábla egy létező rekordjához kapcsolódik, és a szülő táblából történő törlés automatikusan kihat a gyermektáblákra is.

A hierarchikus adatbázisok megfelelő megoldásnak bizonyultak a szalagos tárolóeszközöket használó nagygépek esetén, nagyon népszerűek voltak az 1970-es években. Ezek az adatbázisok gyors és közvetlen hozzáférést biztosítanak az adatokhoz és számos esetben jól használhatóak.

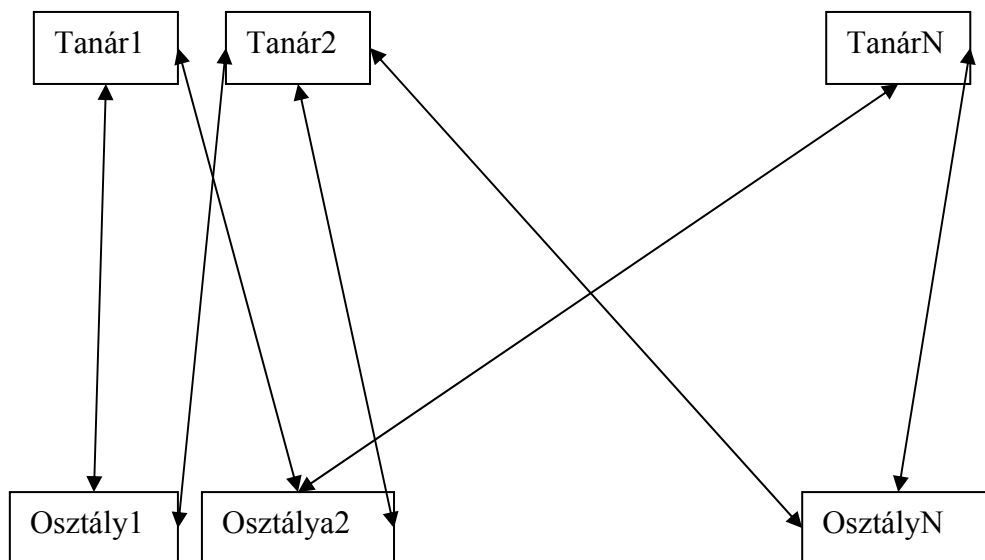


6.1. ábra: Egy hierarchikus adatmodell

6.2.2 Hálós adatmodell

A hálós adatmodellek esetében az egyes azonos vagy különböző összetételű adategységek (egyed típusok, adattípusok) között a kapcsolat egy gráffal ábrázolható.

A gráf csomópontok és ezeket összekötő élek rendszere, melyben tetszőleges két csomópont között akkor van adatkapcsolat, ha azokat él köti össze egymással. Egy csomópontból tetszőleges számú él indulhat ki, de egy él csak két csomópontot köthet össze (6.2. ábra).



6.2. ábra: Egy hálós adatmodell

Hálós modellek jellemzői

- Minden adategység tetszőleges más adategységgel lehet kapcsolatban.
- A adatmodellben M:N típusú és 1:N típusú kapcsolatok is leírhatók.
- Az adatbázisba fixen beépített kapcsolatok következtében csak a tárolt kapcsolatok segítségével bejárható adat-visszakeresések oldhatók meg hatékonyan.
- Hátrányuk, hogy szerkezetük merev, módosításuk nehézkes.

6.2.3 Kétszintű hálós adatmodell

A kétszintű hálós adatmodellt az 1960-as évek végétől kezdték alkalmazni. Az 1:N jellegű kapcsolatokat nyilakkal jelölik. A modellben fő- és alrekordok vannak. A főrekord-típustól az alrekord-típusig vezető 1:N jellegű kapcsolatokat láncolási útnak nevezik. Ebben a modellben egy rekord lehet több szülő gyermeke is. Egy láncolási útvonalon levő rekordtípus nem lehet ugyanazon útvonalnak a főrekordja is. Az eredmény egy olyan modell, amelynek csak két szintje van.

Minden főrekord-típus rendelkezik egyedi azonosítóval, elsődleges kulccsal. Egy főrekord-típus kulcsának szerepelnie kell a vele kapcsolatban lévő mindegyik alrekord-típusban is (másodlagos attribútumként).

6.2.4 CODASYL-hálós adatmodell

A CODASYL-hálós adatmodellt 1971-ben publikálták. Az 1:N jellegű kapcsolatokat redundancia nélkül lehet ábrázolni. A kétszintű hálót és a CODASYL-hálót együttesen hálós adatmodellnek nevezi.

A CODASYL-hálós adatmodell a tulajdonos rekord és a tagrekord terminológiát használja. A tulajdonosrekord és a tagrekord közötti kapcsolatot halmaznak (set) nevezik. A halmaznak egyedi névvel kell rendelkeznie. A halmaz az 1:N jellegű kapcsolatokat jeleníti meg. A modell bármely rekordja lehet a halmaznak tulajdonos- és tagrekordja is. Egy halmaz a tulajdonosrekordból a tagrekordra mutató nyíllal jeleníthető meg. Az 1:1 jellegű

kapcsolatokat ugyanúgy lehet ábrázolni, mint az 1:N jellegűeket. Két rekordtípus között több halmaz is megengedett.

6.2.5 Relációs adatmodell

A relációs adatmodell az 1980-as évektől kezdve a legelterjedtebb adatmodell. A modellben az adatokat táblázatok soraiban ábrázolják. Ebben a modellben nincsenek előre definiált kapcsolatok az egyes adategységek között. A kapcsolatok létrehozásához szükséges adatokat tárolják többszörösen. A relációs modell a 6.3. és a 6.4. fejezetekben részletesen ismertetésre kerül.

6.2.6 Objektumorientált adatmodell

A relációs adatbázis kezelő rendszer (RDBMS – Relation Database Management System) programjait általános üzleti alkalmazásokban elterjedten használják. Léteznek azonban olyan problémák is (számítógépes tervezés, földrajzi információs rendszerek, multimédiás tároló-rendszerek), amelyek támogatására nem alkalmasak. A probléma megoldására két adatbázis modell jelent meg: az *objektum központú adatbázis* és az *objektum-relációs adatbázis*.

Az *objektumközpontú* másként az *objektum-orientált adatmodell* rendelkezik az objektumközpontú programozási nyelvek összes tulajdonságával. Az adatbázis működéséről az adatbázis-fejlesztő gondoskodik, az adatok kezelésére vonatkozó műveleteket lehetővé teszi az objektumközpontú adatbázis-programozási szoftveren belül. Az objektumorientált adatbázis modellnek nincs elméleti alapja, mint a relációs modellnek.

Az *objektum-orientált adatmodell*, más néven a *bővített relációs adatmodell* többféle objektumközpontú elemekkel és tulajdonságokkal egészíti ki a relációs adatmodellt (osztályok, betokozás, öröklődés). A fő szempont az összetettebb adattípusok kezelésének lehetővé tétele. Az adatmodell fejlődése még nem érte el azt a szintet, hogy komolyabb adatbázis-alkalmazásokban lehessen felhasználni. Az adatmodell folyamatos fejlesztés alatt áll.

Az adatmodellezés a hagyományos módszertanok sajátossága. Az objektumorientált módszertanokban az adatmodellezést a komplexebb objektummodellezés váltja fel. Az objektummodellezés sok tekintetben az adatmodellezés kiterjesztése. A 6.1. táblázat a hasonlóságot mutatja a kétfajta modellezés között.

6.1. táblázat: A hagyományos adatmodellezés és az objektummodellezés összehasonlítása

Adatmodellezés	Objektummodellezés
Egyed-előfordulás	Objektum
Egyedtípus	Osztály
Egyedtípus attribútumai	Osztály attribútumai
-	Osztályra értelmezhető műveletek
Egyedek kapcsolatai	Objektum kapcsolatai
Főtípus-altípus kapcsolat (általánosítás és specializáció)	Öröklődési kapcsolat

Az *objektumok* lehetnek tárgyak, személyek, jelenségek, események, viszonyok is. Szorosan véve az objektumok a konkrét egyed-előfordulásoknak felelnek meg. Az azonos objektumok alkotnak egy osztályt, ahogyan az egyed-előfordulások egyed típust.

Az *osztály* egyrészt az azonos típusú objektumok halmaza, másrészt ehhez rendelve definiálható az osztályba tartozó objektumok közös szerkezete.

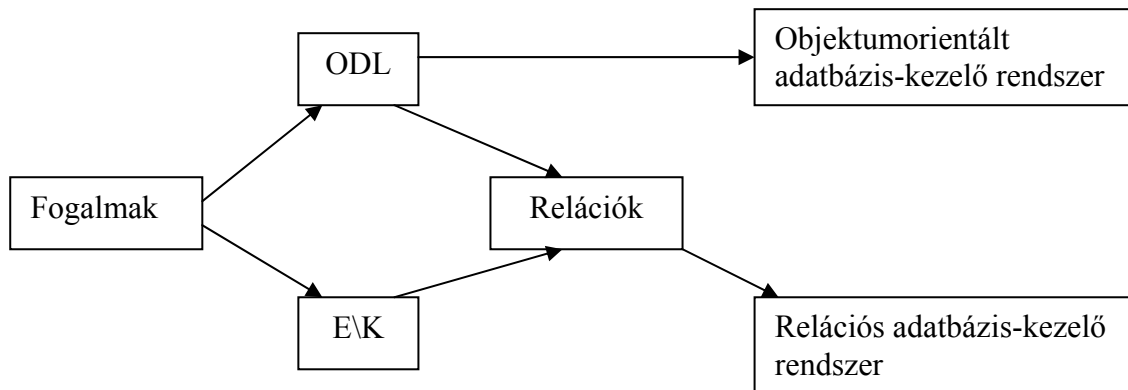
Az *osztály attribútumai* akár többértékűek is lehetnek, a típusuk pedig maga is valamilyen komplex osztály lehet.

Az objektumok ugyanúgy *alkotnak kapcsolatokat*, mint az egyedeket. Ezeket a kapcsolatokat típusszinten az osztályok kapcsolataként lehet definiálni. Az objektumok kapcsolatai is jellemezhetők a fokukkal, az opcionális/kötelező vagy stabil/instabil minőségükkel. Szokás megkülönböztetni *öröklődési kapcsolatokat*, *aggregációs kapcsolatokat* és *egyszerű ún. asszociációkat*. A kapcsolatok az osztálykapcsolat diagramon ábrázolhatók.

Az *osztályok kapcsolatai* között kiemelt szerepet játszanak az öröklődési kapcsolatok. Egy öröklődési kapcsolatban beszélhetünk alosztályról, illetve főosztályról. Az *alosztály* (a speciálisabb osztály) objektumhalmaza a *főosztály* (az általánosabb osztály) objektumhalmazának részhalmaza. Az öröklődési kapcsolatok arról kapták a nevüket, hogy az alosztály megörökli a főosztály attribútumait és műveleteit. Az alosztály szintjén csak a főosztály attribútumain felüli azon plusz attribútumokat kell definiálni, amik speciálisan csak az alosztályba tartozó objektumokat jellemzik.

Az *öröklődési kapcsolatok* szerint az osztályok hierarchiát alkotnak (esetleg, ha többszörös öröklődés is megengedett, több hierarchia-hálónak kapcsolódnak össze). A hierarchia tetején álló osztályba minden objektum beletartozik, és ennek szintjén olyan attribútumokat és műveleteket lehet definiálni, amik minden objektumra értelmezhetők. Az alacsonyabb szinteken lévő osztályok az objektumok egyre kisebb (specifikusabb) halmazát képviselik és a főosztályokra jellemzőkön kívül, további attribútumokkal és műveletekkel rendelkeznek.

Az adatbázis-tervezés objektum orientált megközelítése az ODL (Object Definition Language), mely illeszkedik a szabványos objektum orientált adatbázis rendszerekhez. Az ODL egy szabvány, amely segítségével adatbázisok struktúráját specifikálni lehet az objektumorientált terminológiával. Az ODL elsődleges célja, hogy támogassa az adatbázisok objektumorientált tervezését, és ennek a tervnek a közvetlen transzformálását objektumorientált adatbázis-kezelő rendszerek deklarációiba (OODBMS). Az ilyen rendszerek elsődleges nyelve a C++ vagy a Smalltalk, így az ODL-t ezen nyelvek egyikére kell transzformálni (6.3. ábra).



6.3. ábra: ODL konvertálása egy objektumorientált adatbázis-kezelő rendszer deklarációiba

6.3. Relációs adatmodell

A relációs adatmodell napjaink legelterjedtebb adatmodellje. A relációs adatmodellt megalkotó Codd az IBM cégnél dolgozott, s abban az időben az IBM igen sok energiát ölt már bele a hálós adatmodellen alapuló adatbázis kezelő rendszer (DBMS) kialakítására. Mint vezető adatbázis technológiai cég, az IBM nem engedhette meg, hogy ne vegye figyelembe ezt az újszerű ötletet. A cég beindított egy kísérleti projektet, amelynek célja a relációs adatmodell megvalósíthatóságának a vizsgálata és egy minta relációs adatbázis kezelő rendszer (RDBMS) kialakítása volt. A beindított projekt a System/R elnevezést kapta, ahol az R betű a relációs (relational) jelölésre utal.

Az adatmodell alapjait 1970-ben fektette le Codd az "*A Relational Model of Data for Large Shared Data Banks*" cikkében. A cikk elsődleges célja volt, hogy bebizonyítsa, létezik más alternatíva is az akkor elterjedő hálós adatmodell mellett, amely ráadásul matematikailag megalapozott eszközöket, fogalmakat használ, így pontosabb, egzaktabb leírást, kezelést tesz lehetővé. Az adatmodell a matematika két ágára, a halmazelméletre és az elsőrendű predikátumkalkulusra épít. Az adatmodell neve a reláció szóból ered, ami egy halmazelméleti fogalom.

A megtervezett modellben egyszerű, könnyen megtanulható leírási módot sikerült megvalósítani. Egyszerűségének következtében gyorsan népszerűvé is vált a felhasználók körében, és sok implementációja született meg a személyi számítógépek piacán is. Az elméleti megalapozottság a kutatók, a szakemberek szimpátiáját is kiváltotta, s ez az adatmodell számos új fejlesztési projekt alapját képezi. Az adatmodell mindenki számára fontos előnye az egyszerűség mellett a modell rugalmassága.

A System/R projekt két fázisban zajlott le. Az 1970-es évek közepére kifejlesztettek egy prototípus relációs adatbázis kezelő rendszert (RDBMS), amely még egy felhasználós környezetben működött. Az első részben a kutatások a fizikai elérési módszerek kidolgozására, a kezelő nyelv kialakítására, és a műveletek optimalizálására irányultak. A projekt második fázisában kibővítették a rendszert több felhasználós platformra, így már egy igazi, gyakorlatban is alkalmazható RDBMS rendszert hoztak létre. A projekt 1976-ban zárult le.

A System/R sikereire felfigyeltek más szoftverfejlesztők is. Ennek következtében a 70-es évek végére kialakult néhány új önálló társaság, amelyek a piacon is eladható RDBMS rendszerek kifejlesztésébe kezdtek bele.

Az első termékként megjelenő RDBMS rendszer 1979-ben jelent meg. Ez egy PDP-n futó ORACLE rendszer volt. Napjainkban az ORACLE a világ egyik legnagyobb RDBMS

forgalmazó cége, nyeresége napjainkban is drasztikusan fokozódik. Az ORACLE rendszer mellett, más cégek is jelentkeztek nemsokára saját RDBMS termékeikkel, többek között megemlíthetők az INFORMIX, INGRES, DB2, SYSDATABASE és RDB.

6.3.1 A relációs adatbázis objektumai

A relációs adatbázis az adatokat *relációkban* (*kapcsolatokban*) tárolja, amelyek a felhasználó számára *táblák* formájában jelennek meg. A táblában egy sort *rekordnak* neveznek, a rekord *mezőiből* áll össze (6.4. ábra).

Ügyfelek

Ügyfél azon.	Ügyfél vezetéknéve	Ügyfél keresztnéve	Város	többi mező
9001	Kovács	János	Debrecen	...
9002	Nagy	István	Szeged	...
9003	Kiss	József	Budapest	...
9004	Kocsis	Klára	Eger	...
9005	Búza	Éva	Miskolc	...

←————— mezők —————→

↑
rekordok
↓

6.4. ábra: Táblaszerkezet

Tábla

A táblák az adatbázis legfontosabb szerkezetei, minden tábla egyetlen, jól meghatározott tárgyat ír le. A rekordok és a mezők sorrendje lényegtelen, minden tábla tartalmaz legalább egy olyan mezőt, amely egyedi módon azonosítja a tábla rekordjait, ezt elsődleges kulcsnak nevezzük. Azokat a táblákat, amelyek a tárolt adatok alapján információt szolgáltatnak, adattábláknak nevezik. Az ilyen táblákban tárolt adatok általában dinamikusak, mivel módosíthatók és többféle módon feldolgozhatók.

Mező

A *mező* az adatbázis legkisebb szerkezete, amely a tábla tárgyának egy jellemzőjét adja meg. A mezők tárolják a tényleges adatokat. Egy jól megtervezett adatbázisban minden mező egy vagy több értéket tartalmaz, és a mező neve utal a tárolt értékre.

Rekord

Az adatbázis egy *sora*, amelyet *rekordnak* is neveznek, a tábla tárgyának egy egyedi példányát írja le. A rekord a teljes mezőkészletet magában foglalja, függetlenül attól, hogy az adott mezők tartalmaznak-e értékeket. Az elsődleges kulcsként használt mező azonosítja a tábla sorait, vagyis a rekordokat.

Kulcsok

A *kulcsok* olyan mezők, amelyek különleges szerepet töltenek be a táblában. Egy tábla számos különféle típusú kulcsot tartalmazhat, a legfontosabbak az *elsődleges* és a *másodlagos kulcsok*. Az elsődleges kulcs olyan mező, vagy mezőcsoport, amely egyedi módon azonosítja a rekordokat a táblán belül. Ha az elsődleges

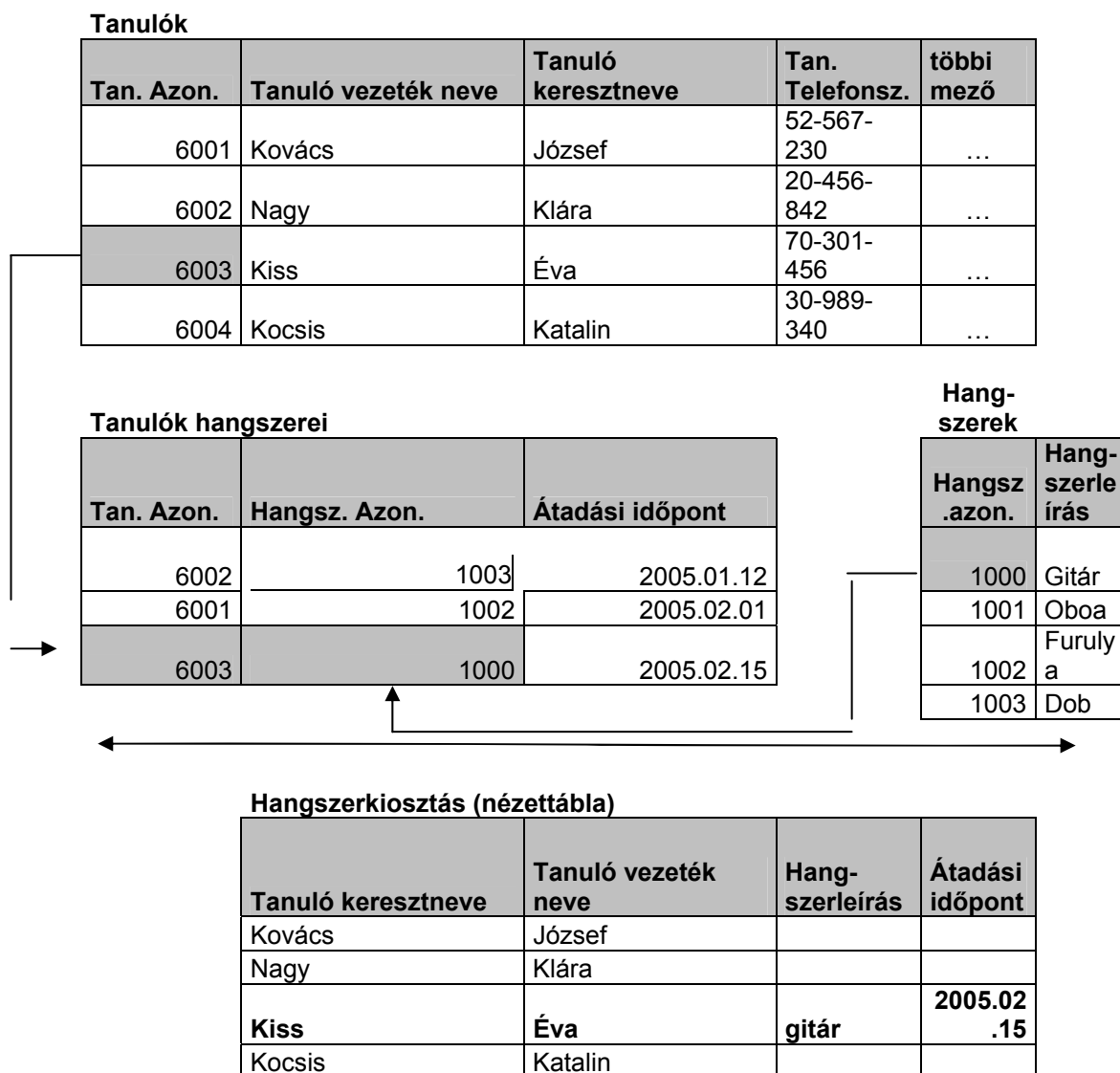
kulcs több mezőből áll, akkor *összetett elsődleges kulcsnak* nevezik. Az elsődleges kulcs a tábla legfontosabb kulcsa.

- Az elsődleges kulcs értéke egy rekordot azonosít az adatbázisban.
- Az elsődleges kulcs mező egy táblát azonosít az adatbázisban.
- Az elsődleges kulcs táblaszintű épséget biztosít, és segít a táblák összekapcsolásában.
- Az adatbázis minden táblájának kell, hogy legyen elsődleges kulcsa.

Nézettábla

A *nézettábla* egy virtuális tábla, amely az adatbázis egy vagy több táblájának mezőiből áll, és a nézettáblát felépítő táblákat együttesen alaptábláknak nevezzük. A nézettáblák tartalmukat más táblákból nyerik. Ezek a táblák lehetővé teszik, hogy többféle nézőpontból lehessen vizsgálni az adatbázis tartalmát.

A következő példa mutatja, hogy a Tanulók, a Tanulók hangszerei, és a Hangszerek táblából készül Hangszerkiosztás nézettábla (6.5. ábra).



6.5. ábra: Példa nézettáblára

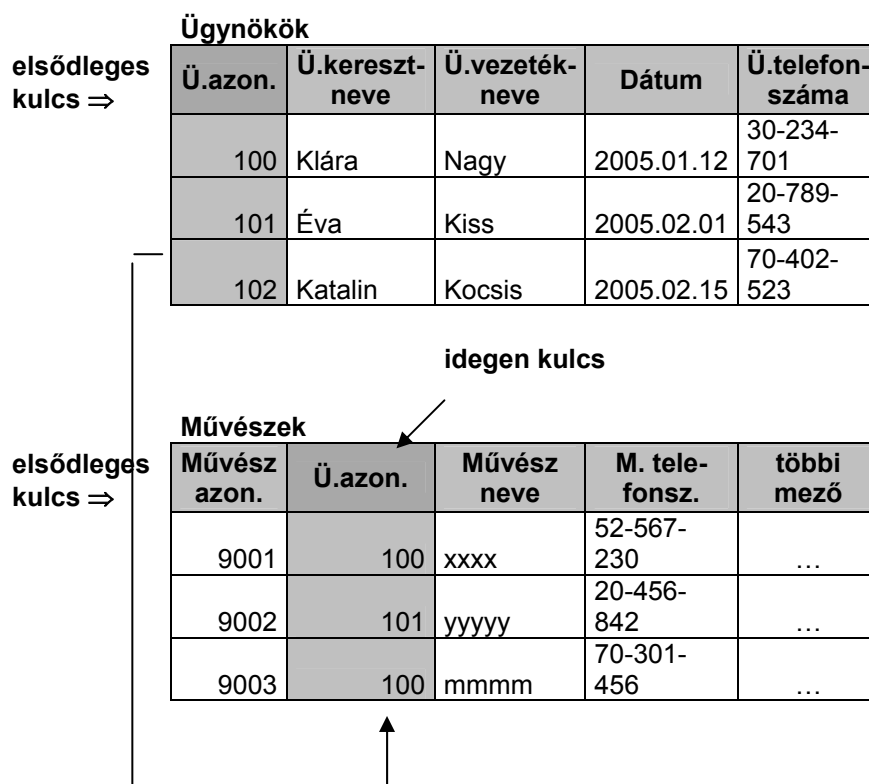
Idegen kulcs

Két tábla közötti kapcsolat létesítéséhez az egyik tábla elsődleges kulcsát be kell építeni egy másik tábla szerkezetébe, ahol ez az érték idegen kulccsá válik. Az idegen kulcs kifejezés abból ered, hogy a második táblának is van elsődleges kulcsa, így az első táblából átvett elsődleges kulcs idegen a második tábla számára (6.6. ábra).

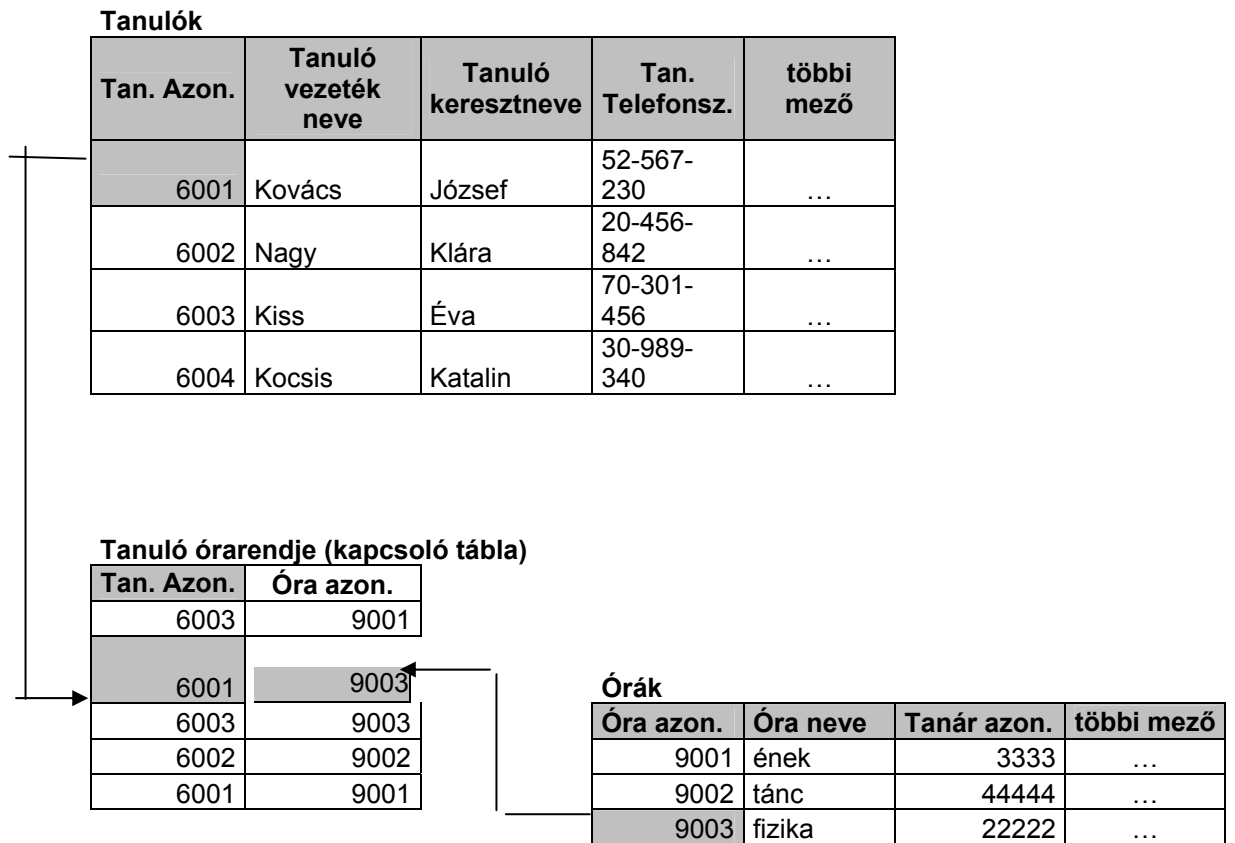
Kapcsolatok

Ha két tábla rekordjai valamilyen értelemben összetartoznak akkor a két tábla kapcsolódik egymáshoz. A táblák közötti kapcsolatok elsődleges és másodlagos kulcsokkal, és kapcsolótábla segítségével valósíthatók meg.

A 6.7. ábra két tábla összekapcsolását mutatja kapcsolótábla segítségével.



6.6. ábra: Példa elsődleges és idegen kulcsokra

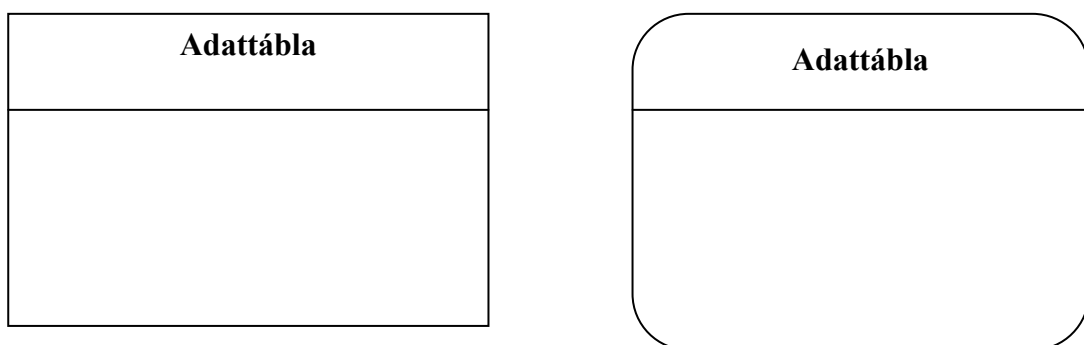


6.7. ábra: Példa két tábla összekapcsolására kapcsolótábla segítségével

Kapcsolattípusok

A táblák közötti kapcsolatoknak három típusa van: egy-egy, egy-sok és sok-sok.

Elfogadott szokásos jelölés szerint a táblákat téglalapokkal, a részhalmaz táblákat lekerekített sarkú téglalapokkal jelölik, és feltüntetik a tábla nevét is az ábrán. A 6.8. ábra mutatja a jelöléseket.

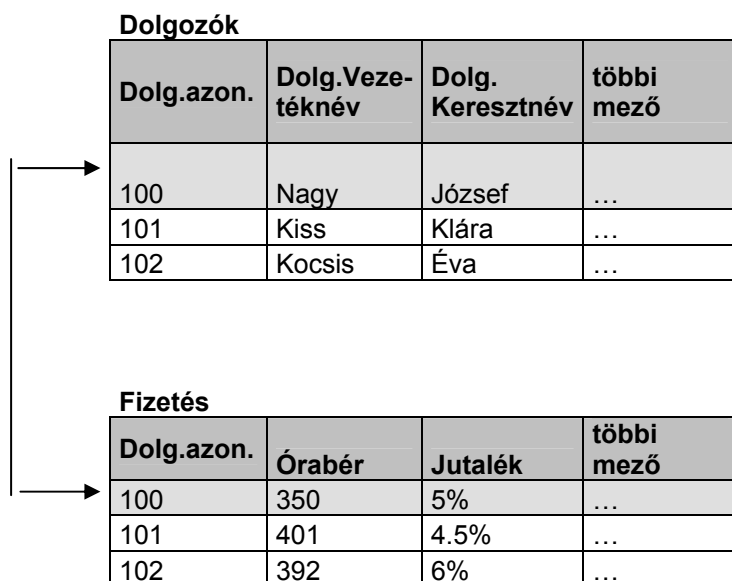


6.8. ábra: Adattábla és részhalmaz tábla jelölése

Egy-egy kapcsolatok

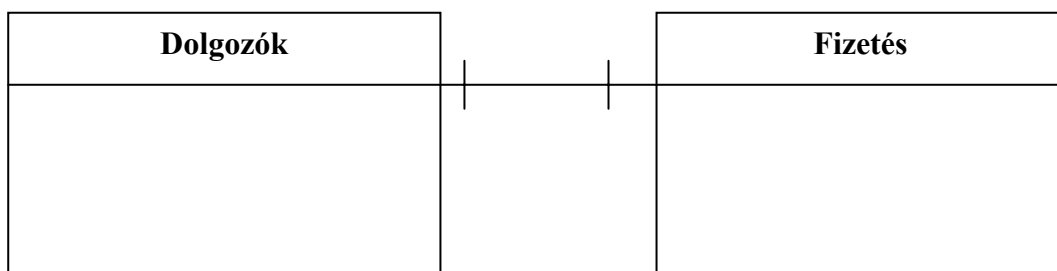
Két tábla közötti egy-egy kapcsolat esetében az első tábla egy rekordjához a második táblából csak egy rekord tartozik, és a második tábla minden rekordja az

első táblában csak egy rekordnak feleltethető meg. A kapcsolat úgy jön létre, hogy az első tábla elsődleges kulcsát beillesztik a második tábla szerkezetébe, ahol idegen kulcs lesz (6.9. ábra).



6.9. ábra: Példa egy-egy kapcsolatra

A következő ábra szemlélteti általánosságban az egy-egy kapcsolatot. A két táblát összekötő vonalra merőleges vonal a **Dolgozók** tábla mellett azt jelzi, hogy a **Fizetés** tábla egyes rekordjaihoz csak egy rekord tartozik a **Dolgozók** táblában. A **Fizetés** tábla melletti merőleges vonal azt jelöli, hogy a **Dolgozók** tábla egyes rekordjaihoz csak egy rekord tartozik a **Fizetés** táblában (6.10. ábra).



6.10. ábra: Egy-egy kapcsolat vázlata

Egy-sok kapcsolatok

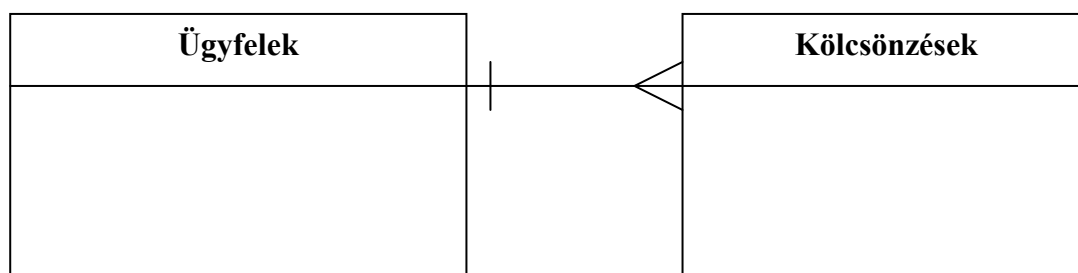
Egy-sok kapcsolat akkor valósul meg két tábla között, ha az első tábla egy rekordjához a második táblából több rekord is tartozhat. A második tábla egy rekordjához azonban az első táblából csak egy rekord kapcsolódhat. Egy-sok kapcsolatot úgy jön létre, hogy az első tábla elsődleges kulcsát beépítik a második táblába, ahol ez idegen kulcs lesz.

Példa egy-sok kapcsolatra (6.11. ábra).

Ügyfelek				Kölcsönzések	
Ügyfél azon.	Vásárló vezetéknév	Vásárló Keresztnév	többi mező	Ügyfél azon.	Film azon.
801	Kiss	István	...	801	1234
802	Kocsis	József	...	802	1345
803	Búza	Klára	...	803	2340
804	Kalocsai	Éva	...	803	5673
805	Kovács	Katalin	...	803	3879
				805	5034
				804	9832

6.11. ábra: Példa egy-sok kapcsolatra

Az egy-sok kapcsolat diagram látható a 6.12. ábrán. A két táblát összekötő vonalra merőleges vonal azt jelzi, hogy a **Kölcsönzések** egyes rekordjaihoz csak egy rekord tartozik az **Ügyfelek** táblában. A „varjúláb” azt jelöli, hogy az **Ügyfelek** egyes rekordjaihoz több rekord is tartozhat a **Kölcsönzések** táblában.



6.12. ábra: Egy-sok kapcsolat diagramja

Sok-sok kapcsolatok

Sok-sok kapcsolat van két tábla között, ha az első tábla egy rekordjához a második táblából több rekord is kapcsolódhat, és a második tábla egy rekordja az első táblából több rekorddal is összekapcsolható (6.13. ábra).

Kereszthivatkozási táblákkal alakítható ki ilyen kapcsolat. A kereszthivatkozási tábla segíti a két tábla rekordjainak az összekapcsolását. A kereszthivatkozási a kapcsolatban álló táblák elsődleges kulcsaiból hozható létre, ezek az elsődleges kulcsmezők együtt adják a kereszthivatkozási tábla összetett elsődleges kulcsát, másrészt idegen kulcsok is. Ha a sok-sok kapcsolat nem megfelelően van megadva, akkor határozatlan kapcsolatról van szó.

Diákok

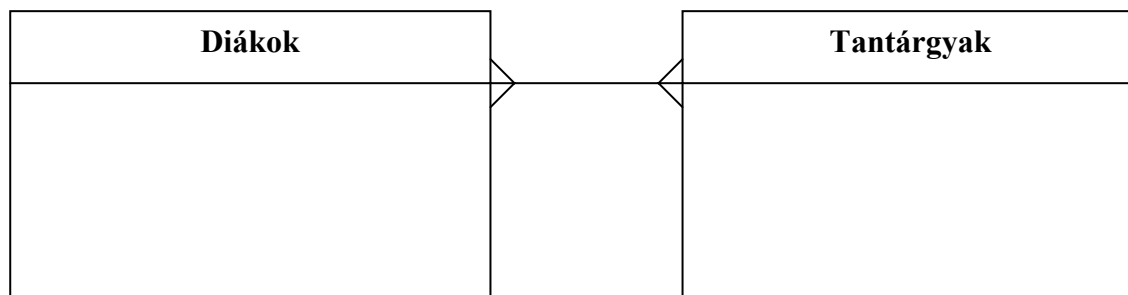
Diák azon.	Diák vezéknév	Diák keresztnév	többi mező
501	Kovács	István	...
502	Nagy	József	...
503	Kiss	Klára	...
504	Kocsis	Éva	...
505	Búza	Katalin	...

Tantárgyak

T.azon	Tantárgy neve	Kredit	Terem	többi mező
8001	Analízis	5	110	...
8002	Geometria	4	121	...
8003	Fizika	5	201	...
8004	Hálózatok	3	202	...
8005	Kémia	3	313	...

6.13. ábra: Határozatlan sok-sok kapcsolat

A sok-sok kapcsolat diagramot mutatja a 6.14. ábra. A Diákok táblánál levő „varjúláb” azt jelöli, hogy a Tantárgyak egyes rekordjaihoz több rekord is tartozhat a Diákok táblában. A Tantárgyak melletti „varjúláb” azt jelöli, hogy a Diákok tábla egyes rekordjaihoz több rekord is tartozhat a Tantárgyak táblában.



6.14. ábra: Sok-sok kapcsolat diagramja

A részvétel típusai

A táblák részvétele a kapcsolatban lehet *kötelező* vagy *nem kötelező*. Ha A és B két egymással kapcsolatban álló tábla, akkor

- Az A tábla részvétele *kötelező* a kapcsolatban, ha az A táblának legalább egy rekordot tartalmaznia kell, mielőtt a B táblába felvennénk az első rekordot.
- Az A tábla részvétele a kapcsolatban *nem kötelező*, ha a B táblába úgy is felvehetünk rekordot, hogy az A táblában még egyetlen rekord sincs.

A következő példában, ha az új ügyfelek felvétele csak akkor lehetséges, ha létezik hozzájuk ügynök, akkor az Ügynökök tábla részvétele kötelező. Ha ügynökök nélkül is

felvehetünk ügyfeleket a táblába, akkor az Ügynökök tábla részvétele a kapcsolatban nem kötelező (6.15. ábra).



6.15. ábra: Ügynökök és Ügyfelek tábla

A részvétel mértéke

A részvétel mértéke határozza meg a legkisebb és legnagyobb rekordszámot, amellyel egy tábla a vele kapcsolatban álló tábla egy rekordjához kapcsolódhat.

Ha A és B két egymással kapcsolatban lévő tábla, akkor meg kell adni, hogy legalább hány B-beli rekordnak kell kapcsolódnia, és legfeljebb hány B-beli rekord kapcsolódhat A egyetlen rekordjához. Ha az A tábla egy rekordjához a B tábla egy, de legfeljebb 10 rekord kapcsolódhat, akkor a B tábla részvételének mértéke a kapcsolatban 1, 10.

Adatépség

Az adatok épsége az adatbázisban tárolt adatok érvényességére, következetességére és pontosságára vonatkozik. Az adatbázis tervezés során négyféle adatépségre kell figyelni.

- *Táblaszintű épség* (egyedépség): egy rekord kétszer nem kerülhet be a táblába, legyen olyan mező, amely egyedi módon azonosítja a rekordokat.
- *Mezőszintű épség* (tartomány épség): minden mező értékei érvényesek, következetesek, pontosak legyenek.
- *Kapcsolat szintű épség* (hivatkozási épség): a táblák közötti kapcsolatok hibátlanok legyenek, a táblák rekordjai összhangban maradjanak az adatok beszúrása, módosítása, törlése esetében.
- *Működési szabályok*: Korlátozzák az adatbázis tartalmát. Például meghatározhatják egy mező lehetséges értékeit, a táblák részvételének típusát és mértékét a kapcsolatokban.

Relációséma

A relációnév és a reláció attribútumai együtt adják a *relációsémát*. A reláció neve után zárójelben vannak az attribútumok, vesszővel elválasztva.

Adatbázisséma

A relációsémák összessége az *adatbázisséma*. Egy relációhoz vagy relációk összességéhez tartozó adatokat az adott relációsémához vagy adatbázissémához tartozó előfordulásnak nevezik.

A *relációséma* és a *reláció-előfordulás* között lényeges különbség van. A séma a reláció nevét és attribútumait tartalmazza és viszonylag állandó. Az előfordulás a relációsorainak a halmaza és az előfordulás gyakran változhat.

A relációk nem állandóak, többször is változhatnak. A változások egy része a reláció soraira vonatkozik, mint például új sorok beszúrása, létező sorok megváltoztatása, vagy sorok törlése. Adott reláció sorainak halmazát *reláció-előfordulásnak* nevezik. Az adatbázis rendszerek általában csak a relációnak azokat a sorait kezelik, amelyek aktuálisan vannak a relációban. Ezt a reláció-előfordulást *aktuális előfordulásnak* nevezik.

A relációséma megváltoztatása kevésbé általános. Lehetnek olyan esetek, amikor attribútumokat kell felvenni, vagy törölni. Az adatbázissémák változtatása általában költséges, ha sok ezer sor mindegyikében meg kell változtatni a komponenseket. Ha új attribútum felvétele történik, nem is biztos, hogy a táblasorok új komponenseihez megfelelő értékeket lehet rendelni.

6.3.2 Relációs adatbázis-kezelő rendszer

A relációs adatbázis-kezelő rendszer (RDBMS) olyan szoftver, amely alkalmas relációs adatbázisok létrehozására, módosítására és kezelésére.

Az 1970-es évek eleje óta számos RDBMS program született. Ezek sokféle hardveren, operációs rendszeren és programozási környezetben működtek. Ma már ezek a programok nélkülözhetetlen részévé váltak életünknek.

A relációs adatbázisok megjelenésekor az RDBMS programok nagygépes rendszereken futottak. A 70-es évek elején két RDBMS volt ismert: a System/R, amelyet az IBM cég fejlesztett ki, és az INGRES (*Interactive Graphics Retrieval System*). E két modell nagymértékben hozzájárult a relációs modell általános elterjedéséhez.

Az 1980-as években számos kereskedelmi RDBMS-t fejlesztettek ki nagygépes rendszerekhez. Ilyen volt az ORACLE és a DB2.

Az 1980-as évek első felében a személyi számítógépek egyre nagyobb mértékben elterjedtek. Ezzel egy időben megjelentek a PC-re írt RDBMS programok: DBASE, FOXPRO, PARADOX. Az 1990-es évek végén egyre több felhasználó kezdett adatbázisokat használni.

Felmerült az adatok megosztásának szükségessége, vagyis a sok felhasználó számára elérhető, központosított adatbázis. A gyártók kifejlesztették az ügyfél-kiszolgáló RDBMS programokat. Ezekben a rendszerekben az adatok egy adatbázis-kiszolgálóként működő gépen kapnak helyet, a felhasználók a saját gépeiken futó programokkal (ügyfél) férnek hozzá ahhoz. Az adatbázis-fejlesztő ügyfél-kiszolgáló RDBMS programokkal hozza létre és tartja karban az adatbázist, illetve az ahhoz kapcsolódó végfelhasználói programokat. Az adatbázis-kiszolgáló gondoskodik az adatok épségéről és biztonságáról. Az ügyfél-kiszolgáló RDBMS programokat hosszú ideje sok helyen alkalmazzák, ha nagymennyiségű megosztott adatokat kell kezelni. A legerjedtebbek a MICROSOFT SQL SERVER 2000 és az ORACLE9I APPLICATION SERVER.

6.4. Adatbázis-kezelő rendszerek

Az adatbázisokhoz tartozik egy programrendszer, neve *adatbázis-kezelő rendszer* (Database Management System - DBMS). Ezzel lehet az adatok tárolását megszervezni, az adatbázisban tárolt adatokat kezelni: adatokat megkeresni, kiválasztani, karbantartani, módosítani, új adatokat beilleszteni, vagy egyes adatokat törölni.

Ezeket a rendszereket szokás egyszerűen *adatbázisrendszereknek* nevezni. Egy adatbázis-kezelő rendszerrel hatékonyan készíthető, kezelhető az adathalmaz, az adatok hosszú ideig, biztonságosan megőrizhetőek. Ezek a rendszerek a legbonyolultabb, legösszetettebb programok közé sorolhatóak. Egy *adatbázis-kezelő rendszerrel* szembeni igények a következők:

1. Legyen lehetőség új adatbázisok létrehozására, és az adatok logikai struktúráját, vagyis az adatbázis sémáját, egy speciális nyelven lehessen megadni. *Adatdefiníciós nyelvnek* nevezik ezt a speciális nyelvet. Angol neve Data Definition Language, rövidítve: DDL.
2. Legyen lehetőség megfelelő nyelv segítségével az adatok módosítására és lekérdezésére. Ezt a nyelvet *adatmanipulációs nyelvnek* vagy *lekérdező nyelvnek* nevezik. Angol neve: Data Manipulation Language, rövidítve: DML.
3. Megvalósítható legyen nagy mennyiségű adat hosszú időn keresztül tárolása. Az adatok védelme biztosítható legyen az illetéktelen hozzáférésektől és a meghibásodásoktól. Az adatok biztonsága garantálható legyen. Az adatbázis módosításakor és a lekérdezéseknél hatékony adathozzáférést tegyen lehetővé.
4. Korrekt módon lehessen kezelni több felhasználó egy időben történő adathozzáférését, az egyes felhasználók műveletei ne legyenek hatással a többi felhasználóra, az egyidejű adat hozzáférések miatt az adatok ne váljanak hibássá vagy következtelenné.

Az adatbázis tartalma időben változik. Az adatbázis kifejezés alatt a pillanatnyilag tárolt információra lehet gondolni. Az adatbázis-kezelő szó az adatbázis-kezelő programrendszer szó rövidítése. Az adatbázis-kezelő programjait működtetőket *felhasználóknak*, *user*-eknek, nevezik. Azokat, akik felelősek az adatbázis-kezelő programjainak karbantartásáért, új programok megírásáért, ezeknek az adatbázis-kezelőbe történő beillesztéséért, az információ épségéért, *adatbázis-adminisztrátornak* hívják.

Néhány adatbázis-kezelő rendszer

Banki rendszerek

Adatelemek: ügyfelek nevei, címei, folyószámlák és hitelszámlák egyenlegei, ügyfelek és számlák közötti kapcsolatok. Ilyen kapcsolat például: kinek melyik számlához van aláírási jog. Gyakori a számlák egyenlegének lekérdezése és a módosítások, amelyek a számlára vonatkozó befizetést vagy kifizetést jelentenek. Gyakori, hogy egyidejűleg több banki ügyintéző és bankjegykiadó automatát használó ügyfél végez lekérdezést vagy módosítást. Alapvetően fontos, hogy az egyidejűleg végzett műveletek miatt ne vesszenek el egy automatánál elvégzett műveletek adatai. Hibák nem megengedhetőek. Ha a pénzt kiadta az automata, akkor a pénzfelvételt is rögzítenie kell a banknak, ha közben áramszünet lesz. Másrészt nem megengedhető, hogy a pénzfelvételt rögzítsék, de a pénzt ne adja ki az automata áramszünet miatt. Az ilyen műveletek helyes kezelése nem egyszerű.

Vállalati nyilvántartások

Az első számítógépes alkalmazások közül jó néhánynak az volt a feladata, hogy egy vállalat különböző nyilvántartásait kezelje. Az adatok kapcsolódhattak az eladásokhoz, kimenő és bejövő számlákhoz, vagy a dolgozókhoz. A lekérdezések olyan jelentések kinyomtatásából állnak, mint például a kinn lévő számlák, a dolgozók havi fizetése. Az eladások, vásárlások, számlakibocsátások, számlakifizetések, dolgozók felvétele, elbocsátása, előléptetése mindezek az adatbázis módosítását jelentik.

Repülőgép-helyfoglalási rendszer

Ebben az esetben az adatelemek a következők lehetnek:

- Egy vevőnek egy járatra szóló helyfoglalása, ülőhely sorszáma.
- A járatra vonatkozó információk: indulás és érkezés időpontja, melyik repülőtérről indul és hová érkezik a repülőgép, melyik repülőgép szállítja az utasokat.
- A jegyárakra, az igényekre és a még szabad jegyekre vonatkozó információk.

A leggyakoribb lekérdezések: egy adott városból egy másikba mely járatok indulnak egy időintervallumban, mely ülőhelyek szabadok, mennyi a jegy ára. Jellegzetes adat módosítások lehetnek: egy vevő helyfoglalása egy járatra, egy ülőhely kiadása egy vevőnek. Egy időpillanatban sok különböző jegyeladással foglalkozó ügynökség érheti el az adatok egy-egy részét. Az adatbázis-kezelő rendszernek lehetővé kell tennie az egyidejű hozzáféréseket, de meg kell akadályozni, hogy egy ülőjegyet kétszer adjanak el egyidejűleg. Továbbá meg kell akadályozni, a felvitt adatok elvesztését abban az esetben, ha a rendszer váratlan hiba folytán leállna.

Első adatbázis-kezelő rendszerek

Az első adatbázis-kezelő rendszerek használatakor célszerű volt, hogy a felhasználók az adatokat olyan vizuális formában ábrázolják, ahogyan azok tárolva vannak. Ezek a rendszerek különböző adatmodelleket használtak az adatbázisban tárolt információk szerkezetének ábrázolásához. A legfontosabb ezen modellek közül a hierarchikus adatmodell, amely egy fa szerkezettel ábrázolja az adatokat, és a hálós adatmodell, amely egy gráffal ábrázolja az adatokat. Ez utóbbit a 60-as évek végén szabványosították egy CODASYL jelentésben (Committee on Data Systems and Languages – Adatrendszerekkel és Nyelvekkel Foglalkozó Bizottság). Ezek a modellek ma már csak történeti érdekességként említhetők.

Ezekkel a korai modellekkel és rendszerekkel a gond az volt, hogy nem támogattak semmilyen magas szintű lekérdezőnyelvet. Például a CODASYL lekérdező nyelvnek olyan utasításai voltak, amelyek csak azt engedték meg a felhasználónak, hogy adatelemről adatelemre mozogjon az elemek között meglévő mutatókból álló gráf mentén. Ilyen programok megírása meglehetősen nagy erőfeszítést igényelt még egyszerű lekérdezések esetében is.

Az IBM volt az első cég, amelyik relációs és a relációs modell előtti modelleket támogató adatbázis-kezelő rendszereket is árusított. Röviden: a relációt mint táblát képzelhetjük el. A tábla oszlopaiban vannak az attribútumok, amelyek az oszlopban szereplő értékek jellegét mutatják. Az attribútum-nevek alatti sorokban vannak a reláció sorai.

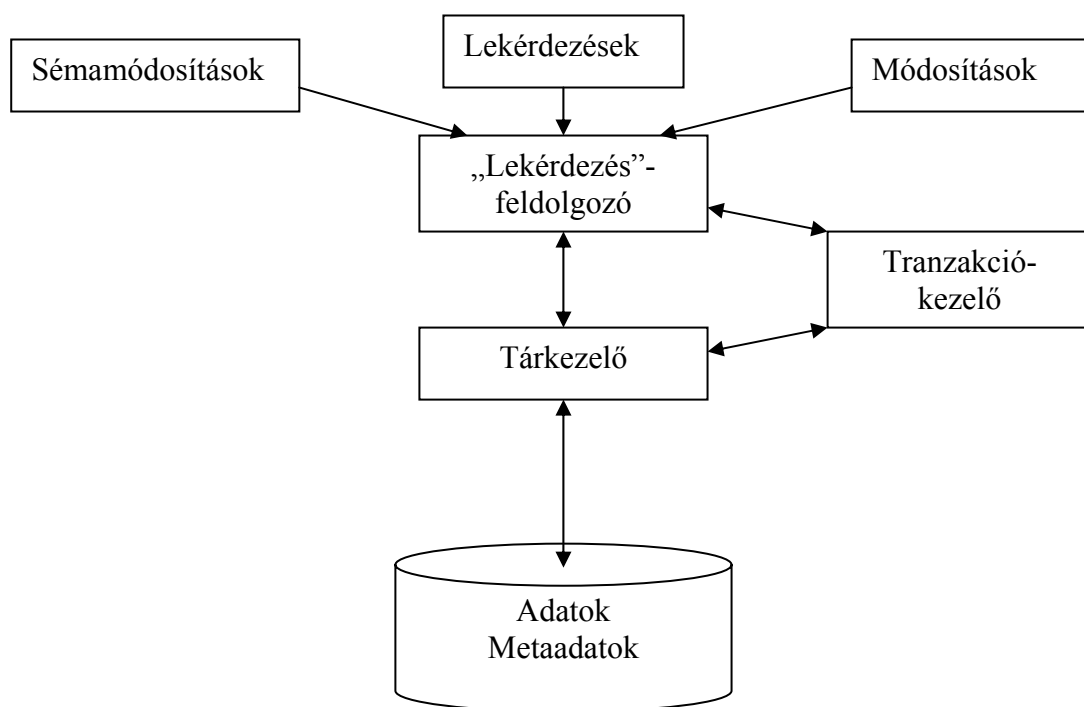
Később újabb és újabb cégek alakultak, amelyek relációs adatbázis-kezelők megvalósításával és árusításával foglalkoztak. Ma már ezek a cégek közül néhány a világ legnagyobb szoftverkereskedői közé tartozik.

6.5 Adatbázis-kezelő rendszerek felépítése

Ebben a szakaszban felvázoljuk, hogyan néz ki általában egy adatbázis-kezelő rendszer felépítése. Az 6.16. ábra bemutatja az adatbázis-kezelő rendszerek legfontosabb részeit.

Az adatbázis nemcsak adatokat, hanem úgynevezett **metaadatokat** is tartalmaz, ami az adatok szerkezetét írja le. A metaadatok között szerepelnek a relációk nevei, a relációk attribútumainak neve, az attribútumok adattípusai (például egész vagy 25 hosszúságú karakterlánc).

Az adatbázis-kezelő rendszerek gyakran indexeket használnak az adatok elérésére. Az index olyan adatstruktúra, amely lehetővé teszi, hogy az adatelemeket gyorsan megtaláljuk, ha ismerjük a hozzájuk tartozó értékek bizonyos részét. A leggyakrabban előforduló index olyan, amelyik segítségével egy reláció azon sorait kereshetjük meg, amelyekben az egyik attribútum értéke adott. Az indexek a tárolt adatok közé tartoznak, olyan információk, hogy mely attribútumokra léteznek indexek, a metaadatok részét képezi.



6.16. ábra: Egy adatbázis-kezelő rendszer felépítése

Lekérdezés feldolgozó: kezeli a lekérdezéseket, az adatok és a metaadatok módosítására vonatkozó kéréseket, az adatbázis-műveleteket egyszerű utasításokra bontja, a lekérdezések végrehajtásának optimális sorrendjét kialakítja és átadja a tárkezelőnek.

Tárkezelő: elvégzi a kért információk beolvasását a tároló helyről és a kért adatmódosításokat. A tárkezelő két része: fájl-kezelő és a puffer-kezelő. A fájlkezelő nyilvántartja a fájlok lemezen való elhelyezkedését, az adatállomány blokkjainak olvasását/írását végzi. Puffer-kezelő a memória munkaterületeit kezeli. Az adatblokkokat a lemezzről a fájlkezelő segítségével olvassa be, és kiválaszt egy memória területet, ahol az adatblokkokat tárolja.

Tranzakció kezelő: felelős a rendszer sérthetlenségéért. Kapcsolatban van a lekérdezés feldolgozóval és a tárkezelővel is. Biztosítja, hogy az egy időben futó lekérdezések és módosítások ne ütközzenek össze egymással, rendszerhiba esetén se vesszenek el adatok. Az adatmódosítások naplózásával biztosítható, hogy rendszerhiba esetén a módosítások újra végrehajthatóak legyenek, amelyek lemezzre írása eredetileg nem volt sikeres.

A *tranzakció* olyan műveletek egy csoportja, amelyeket egymás után egy egységként kell végrehajtani. Az adatbázis-kezelő rendszerek gyakran sok tranzakció egyidejű végrehajtását engedik meg. A tranzakciók helyes lefutása a következő négy elvárás teljesülését jelenti:

Atomosság (atomiság): Megköveteli, hogy a tranzakció vagy teljes egészében hajtsódjon végre, vagy semmi ne hajtsódjon végre belőle.

Következetesség: Egy adatbázisban a „következetes állapotok” fogalma azt jelenti, hogy az adatok megfelelnek bizonyos elvárásoknak. Például egy repülőgép-helyfoglalási rendszerben következetességi feltétel lehet az, hogy egyetlen ülőhelyet se rendeljenek hozzá két különböző utashoz. Amikor az utasok áthelyezése történik az ülőhelyek között, akkor rövid időre a következetességi feltétel megsérül, és a tranzakció-kezelőnek kell biztosítania, hogy a tranzakciók befejeződése után az adatbázis ismét következetes állapotba kerüljön, vagyis kielégítse az összes következetességi feltételt.

Elkülönítés: Amikor két vagy több tranzakció egyidejűleg fut, azok kihatását el kell különíteni egymástól. Ez azt jelenti, hogy semmiféle olyan eredmény vagy kihatás nem következhet be az adatbázisban, amit két tranzakció egyidejű futása okozott, és ami nem fordult volna elő, ha a két tranzakció egymás után fut le.

Tartósság: Ha egy tranzakció befejeződött, akkor annak eredménye nem vesztethet el rendszerhiba esetén sem, még akkor sem, ha a rendszer közvetlenül a tranzakció befejezése után hibásodik meg.

A fenti négy feltétel megvalósító technikák: a zárolás, a naplózás és a tranzakció érvényesítése.

Zárolás: A tranzakció-kezelő zárolhatja a tranzakció által elérni kívánt adattételt. Amíg egy tranzakció zárolva tart egy tételt, addig a többi tranzakció nem érheti el azt.

Naplózás: A tranzakció-kezelő az összes megkezdett tranzakciót, a tranzakciók által az adatbázisban végzett módosításokat, és a tranzakciók végét feljegyzí egy naplóban. A napló mindig olyan tároló eszközre íródik, ami nem érzékeny az esetleges áramkimaradásra. Ilyenek például a lemezek. A napló minden esetben azonnal a lemezre íródik. Az összes művelet naplózása fontos a tartósság biztosításában.

Tranzakciók érvényesítése: A tartósság és az atomosság érdekében a tranzakciók egyfajta „puhatolózás” jelleggel kerülnek végrehajtásra, ami azt jelenti, hogy az adatbázisbeli módosítások kiszámításra kerülnek, de ténylegesen az adatbázisban még nem történnek meg. Amikor a tranzakció készen áll a befejezésre, hogy érvényesítse az elvégzett munkát, a módosítások a naplóba kerülnek. Először ezek a naplóbejegyzések íródnak lemezre, és csak ezután történik meg az adatbázis tényleges módosítása. Így ha a rendszer össze is omlik, akkor is látható lesz a naplóból, amikor a rendszer helyreáll, hogy a változtatásokat még el kell végezni az adatbázisban. Ha a rendszer még az előtt omlik össze, hogy az összes

módosításokat a naplóba jegyezte volna, akkor a tranzakció nyugodtan visszagörgethető.

Az adatbázis-kezelő rendszer inputjai: lekérdezések, módosítások, sémamódosítások. A lekérdezések az adatokra vonatkozó kérdések, amelyek két különböző módon jöhetnek létre:

1. Létrejöhetnek *általános lekérdező-interfészen* keresztül. Ekkor a rendszer megengedi a felhasználónak, hogy SQL lekérdezéseket adjon meg, amelyeket a lekérdezés-feldolgozó kap meg és végrehajt.
2. Létrejöhetnek *alkalmazói program interfészen* keresztül. Az adatbázis-kezelő rendszerek megengedik a programozónak, hogy olyan programot írjon, amelyik az adatbázis-kezelőnek szóló hívásokon keresztül lekérdezi az adatbázist. Ezen az interfészen keresztül nem lehet tetszőleges kérdéseket feltenni, de amit le lehet kérdezni, az általában egyszerűbben kérdezhető le, mintha meg kellene írni a lekérdezést SQL-ben.

Az input *módosító műveletek* az adatok módosítására szolgálnak. Ezek is kiadhatók egy általános interfészen vagy egy alkalmazás-interfészen keresztül.

Sémamódosításokat általában csak az arra illetékes személyek adhatják ki, akiket adatbázis-adminisztrátornak neveznek. Ők megváltoztathatják az adatbázis sémáját, vagy új adatbázist hozhatnak létre.

Az adatbázis-kezelő rendszerek általában a kliens-szerver felépítést követik, ahol a legfontosabb adatbázis-komponensek a szerver oldalon futnak, a kliens pedig a felhasználóval való kommunikációt kezeli.

6.6. Normalizálás. Normálformák

Az adatmodellezés során vizsgálni kell a valós jelenségeket (*egyed*), azok tulajdonságait (*attribútum*) és összefüggéseit (*kapcsolat*). Nemcsak elfogadható adatbázis-szerkezetet kell készíteni, hanem az optimális adatmodellt is célszerű lehet megtalálni.

Az egyedek optimális tulajdonságainak kialakításában a *normalizálás* matematikai eljárás segít. Az egyedek legjobb *normálformája* (*NF*) több lépésben alakítható ki. A normálformák egymásba skatulyázhatók. Azaz ha az egyed 3NF, akkor már szükségszerűen 2NF alakú is.

A normalizálás olyan matematikai eljárás, amelynek egyetlen célja a *redundanciák* kiszűrése a relációs táblákból.

Redundancia

Redundanciáról akkor beszélünk, ha valamely tényt, vagy a többi adatból levezethető mennyiséget, többször tárolunk az adatbázisban, vagyis adatismétlődés fordul elő.

A redundancia, a szükségtelen tároló terület lefoglalása mellett, komplikált adatfrissítési és karbantartási műveletekhez vezet, amelyek az adatbázis *inkonzisztenciáját* okozhatják.

Inkonzisztencia

Egy adatbázist akkor neveznek inkonzisztensnek, ha egymásnak ellentmondó tényeket tartalmaz.

A tervezés során meg kell határozni egy olyan optimális adatstruktúrát, ami a lehető legkevesebb adatismétlődést tartalmaz.

A normalizálás lényegében adattáblázatokat szétbontó műveletek sorozata, amelynek eredményeként egymással kapcsolatban álló, az eredetnél kisebb tárolási igényű táblázatok állnak elő. Az egyedek belső struktúráját a tulajdonságai közötti sajátos viszonyok alapján lehet kialakítani úgy, hogy az átfedések szempontjából kedvező adatmodell keletkezzen.

Összefoglalva a normalizálás előnyei

- csökken a tárolási igény,
- megszűnnek a bővítési, törlési és módosítási anomáliák (rendellenességek),
- áttekinthetőbb lesz az adatbázis.

Azt az attribútumot, amely minden egyes táblázati sorban eltérő értéket vesz fel, *kulcsjelöltnek* (*candidate key*) nevezik. A kulcsjelöltek közül ki kell választani az azonosítóként alkalmazottat. Ezt *elsődleges kulcsnak* (*primary key*) hívják, a többi jelöltet *alternáló kulcsnak* (*alternate key*) mondják. Az elsődleges kulcs, mint azonosító lehet összetett is. Ha egy tábla elsődleges kulcsa másik táblában is megjelenik, akkor abban a másikban *idegen kulcsnak* (*foreign key*) nevezik. A hivatkozás integritás azt mondja ki, hogy az idegen kulcs csak olyan értéket vehet fel, ami elsődleges kulcsként már létező érték a kapcsolódó főlérendelt táblában.

Funkcionális függés

Az E egyed B tulajdonsága akkor és csak akkor funkcionálisan függ az egyed A tulajdonságától, ha az E egyed minden egyes előfordulásában az A értéke minden időpillanatban csakis egy B értékkel társul. A funkcionális függést (*functional dependency*) FD-vel rövidítik, jele: „ $\rightarrow$ ”.

AUTÓ (**rendszer**, autó típus, szín, törzsszám, ...)

A fenti példában látható, hogy a **rendszer**től funkcionálisan függ az **autó típus**, a **szín** és a **törzsszám**. Jelöléssel:

rendszer $\rightarrow$ (autó típus, szín, törzsszám)

A funkcionális függés *reflexív*: azaz $A \rightarrow A$;
additív: azaz ha $A \rightarrow B$ és $A \rightarrow C$, akkor $A \rightarrow B, C$.

A funkcionális függés módosított megfogalmazása

Az E egyedben az A tulajdonság akkor és csak akkor határozza meg funkcionálisan a B-t, ha az utóbbi minden előfordulásában ugyanazt az értéket veszi fel, amikor az A értéke is ugyanaz.

A funkcionális függéshez mindig *fordított irányú függetlenség* tartozik. Ha $A \rightarrow B$, akkor $B \not\rightarrow A$. Vagyis ha A meghatározza B-t, akkor B nem határozhatja meg A-t. Ha a fordított irányú függés is fennállna, akkor *kölcsönös függésről* (*mutual dependency*) lenne szó,

rövidítve *MD*, jele: $\leftrightarrow$. Ha két tulajdonság között egyik irányba sem lehet megállapítani függést, akkor *kölcsönös függetlenségről* van szó.

Összetett függés

Gyakori eset, hogy a *FD* bal oldala több tényezőből áll, ezt összetett függésnek nevezik.

Erős függés, gyenge függés

Ha a *FD* minden baloldalához kell, hogy tartozzon jobb oldal, akkor a függést erősnek, ha nem, akkor gyengének mondják.

rendszer → autó típus *erős függés*

rendszer → Casco *gyenge függés*, mert nincs minden autónak Casco biztosítása.

Teljes függés, részleges függés

A függés akkor teljes (full), ha a függő a teljes összetett meghatározótól függ, és ha abból bármelyik tagot kiemeljük, a függés megszűnik. A függés akkor részleges (partial), ha az összetétel valamelyik tagját elhagyva a függés továbbra is fenn áll.

Nem normalizált egyed

Az egyed ismétlődő ismeretet tartalmaz, ha van olyan tulajdonsága, amely funkcionálisan független az azonosítójától. Ilyenkor az egyed nem-normalizált, másképpen *0NF*.

Előnormalizálást akkor kell végezni, ha az egyednek vannak olyan adatai, amelyek egy név alatt több értéket vesznek fel. Ha az egyedben van olyan tulajdonság, amely nem függ a kulcstól, akkor az egyed ismétlődő ismeretet tartalmaz. A következő táblázat egy nem normalizált egyed mutat, amely *ismétlődő csoportot* tartalmaz (6.17. ábra).

SZEMÉLY

Törzsszám		Név	Nyelv	Vizsgaév	Vszint
AAA	...	Kovács L.	angol	1984	Felső
			francia	1985	Közép
			olasz	1975	Felső
BBB	...	Szabó P.	olasz	1986	Felső

6.17. ábra: Ismétlődő csoportot tartalmazó egyed, *0NF*

A táblázat első három sora egy egyed-előfordulást alkot. Egy *Törzsszám* értékhez több *Nyelv* érték is tartozhat, nem áll fenn a *Törzsszám* → *Nyelv* függés, tehát a *Nyelv* független a *Törzsszámtól*. Az egyednek vannak olyan adatai, amelyek egy név alatt több értéket vesznek fel, tehát ismétlődő csoportot tartalmaz. Megfordítva: Ha az egyedben van olyan tulajdonság, amely nem függ a kulcstól, akkor az egyed ismétlődő ismeretet tartalmaz.

A nem-normalizált *0NF* alakú egyedeket *NINF* (*not first normal form*) alakúnak is hívják. Ezeket az egyedeket legalább *első normálformájúra* (*INF*) kell alakítani. Az ismétlődés megszüntetése úgy történik, hogy az alapegyedből kiemeljük a többszörös értékű tulajdonságokat (*Nyelv*, *Vizsgaév*, *Vszint*), és ezt a sort kiegészítjük az alapegyed kulcsával (*Törzsszám*), hogy felé kapcsolatot lehessen teremteni. Az új egyed kulcsa ilyenkor mindig

összetett (Törzsszám+Nyelv). Az ismétlődő tulajdonságok kiküszöbölését mutatják a következő táblázatok (6.18. ábra).

SZEMÉLY			
Törzsszám		Név	
AAA	...	Kovács L.	
BBB	...	Szabó P.	

NYELVTUDÁS			
Törzsszám	Nyelv	Vizsgaév	Vszint
AAA	angol	1984	felső
AAA	francia	1985	közép
AAA	olasz	1975	felső
BBB	olasz	1986	felső

6.18. ábra: Az ismétlődő tulajdonságok kiküszöbölése

Első normálforma

Az egyed akkor és csak akkor van legalább első normálformában (*1NF*), ha minden nem-kulcs tulajdonságtípusa funkcionálisan függ az azonosítótól.

KÁR	
Kárszám	Dátum
23000	2000.02.15
34000	2001.05.23
45000	2002.04.12

KÁR\KOCSI

Kárszám	Rendszám	Típus	Tulajdonoskód	Kárösszeg
23000	ABC 134	Lada	11111	X
23000	BCD 265	BMW	22222	Y
34000	DEF 896	Lada	33333	Z
45000	ABC 134	Lada	11111	Q

6.19. ábra: Egyed típusok 1NF alakban

A két egyed egyike sem tartalmaz ismétlődő tulajdonságot, azaz tulajdonságaik mindegyikét meghatározzák a kulcsok. A kiemelt szürke sorok fizikai redundanciát mutatnak. Ahányszor kárt szenved az autó, annyiszor rögzítik a típusát, ez felesleges (6.19. ábra).

Részleges függés

Az E egyed típus C tulajdonsága akkor és csak akkor függ részlegesen az A+B összetett azonosítótól, ha a C-t az A vagy a B is meghatározza.

A normalizálási gondokat *anomáliáknak* (visszásságoknak) nevezik. Ilyen anomáliák:

- tárolási anomália,
- frissítési anomália,

- törlési anomália,
- beviteli anomália.

Normálforma-dekompozíció

Normálforma-dekompozíciónak nevezik azt az eljárást, amelynek során az eredeti egyedtypust a rossz függés mentén felbontják, és a redundanciát okozó tételt egy másik egyedtypusba teszik.

Az előző modellből kiszűrve a részleges függést, a következő áll elő (6.20. ábra).

KOCSI

Rendszám	Típus	Törzsszám
ABC 134	Lada	11111
BCD 265	BMW	22222
DEF 896	Lada	33333
XYZ 999	Fiat	99999

KÁR\KOCSI

Kárszám	Rendszám	Kárösszeg
23000	ABC 134	X
23000	BCD 265	Y
34000	DEF 896	Z
45000	ABC 134	Q

6.20. ábra: A részleges függés megszüntetése

Második normálforma

Az egyed akkor és csak akkor van legalább $2NF$ alakban, ha minden nem-kulcs tulajdonsága teljes függéssel függ az azonosítójától.

Ha az egyed $1NF$ alakú, vagyis nem tartalmaz ismétlődést, akkor vizsgálni kell a részleges függést. Erre csak akkor van szükség, ha az egyed azonosítója összetett, mert részleges függés csak ekkor léphet fel. A részleges függések a megfelelő tulajdonságok eltávolításával szüntethetők meg.

Tranzitív függés

Az E egyed nem-kulcs C tulajdonsága (Férőhely) akkor és csak akkor tranzitíven függ az egyed A kulcsától (Rendszám), ha azt meghatározza az azonosítótól függő B tulajdonság (Típus) is.

KOCSI

Rendszám	Típus	Tulajdonos	Foglalkozás	Telephely	Főhatóság	Férőhely
ABC 134	Lada	Szabóné	tanárnő	Debrecen	-	5 fő
BCD 265	BMW	AB Kft	-	Szeged	-	5 fő
DEF 896	Lada	Kovács P	Tanár	Pécs	-	5 fő
FGH 333	Polski	XY Rt	-	Debrecen	M	4 fő

6.21. ábra: $2NF$ alakú KOCSI egyed

Ebben a KOCSI egyedben nincs ismétlődő tulajdonság és annak azonosítója nem összetett, így legalább második normál formában van. Viszont tartalmaz tranzitív függést (6.21. ábra). (Rendszám $\rightarrow$ Típus és Típus $\rightarrow$ Férőhely így Rendszám $\rightarrow$ Férőhely).

Harmadik normálforma

Az egyed akkor és csak akkor van legalább **3NF** alakban, ha minden nem-kulcs tulajdonsága függ a teljes azonosítótól, és csakis attól függ.

KOCSITÍPUS

Típus	Férőhely
BMW	5 fő
Lada	5 fő
Polski	5 fő

KOCSI

Rendszám	Típus	Tulajdonos	Foglalkozás	Telephely	Főhatóság
ABC 134	Lada	Szabóné	tanárnő	Debrecen	-
BCD 265	BMW	AB Kft	-	Szeged	-
DEF 896	Lada	Kovács P	Tanár	Pécs	-
FGH 333	Polski	XY Rt	-	Debrecen	M

6.22. ábra: Harmadik normálformájú egyedek

Mindkét egyed **3NF**. A két új egyed nem az eredeti kulcsrészét képező adatán kapcsolódik egymáshoz, hanem egy leíró kapcsoló tulajdonságán át (*Típus*) (6.22. ábra).

Az egyed akkor van **3NF**-ban, ha minden leíró tulajdonsága függ a kulcsától (**1NF**), csak a kulcsától (**2NF**) és semmilyen más tulajdonságától, csak a kulcsától (**3NF**).

A **3NF** alakú egyedben nincs ismétlődő tulajdonság (**0NF**), nincs részleges függés (**1NF**), és nincs tranzitív függés (**2NF**).

Az előzőekből látható, hogy a normalizálás *dekompozícióval* végezhető el. A nem megfelelő formájú egyedeket több egyedre bontja szét ez a művelet. Azt a műveletet, amikor oszlopokat emelünk ki a relációból, *kivetítésnek* (*projekciónak*) nevezik. Az *összekapcsolás* (*join*) műveletben két relációt azok közös oszlopai alapján egy harmadikba egyesítenek. A normalizálás kivetítések és összekapcsolások sorozata. A dekompozíciót úgy kell elvégezni, hogy veszteségmentes legyen. Ez a művelet *reverzibilis*, azaz megfordítható, vagyis az eredmény relációból mindig visszaállíthatók az eredetiek. A dekompozíciónak független relációkat kell eredményeznie.

Az eddig bemutatott normalizálások viszonylag könnyen érthetőek és egyszerűen elvégezhetőek. Problémák akkor adódnak, ha az egyedben több kulcsjelölt illetve több kulcsjellegű tulajdonság van.

Boyce-Codd normálforma

Az egyed akkor és csak akkor van Boyce-Codd normálformában, ha minden meghatározó tulajdonsága egyben kulcsjelölt is.

Az egymásba skatulyázás alapján minden **BCNF** alakú egyed egyben **3NF** formájú is. Elméletileg a **BCNF** forma jobb, mint a **3NF**. Gyakorlatban azonban nem mindig igaz ez. Vannak olyan esetek, amikor a **BCNF** több nehézséget okoz, mint amennyit megszüntet.

TANÍTÁS (diák, tárgy, tanár)

Ez **3NF** van, de nincs **Boyce-Codd normálformában**. A diák+tárgy $\rightarrow$ tanár, tanár $\rightarrow$ tárgy, diák+tanár $\rightarrow$ tárgy függések alapján három meghatározó van. Közülük a diák+tárgy és a diák+tanár kulcsjelölt, mert tőlük függ a harmadik tulajdonság is. A tanár viszont olyan, amiktől nem függ a reláció összes többi tétele, ezért nem lehet kulcsjelölt.

Atomi egyed

Atominak nevezzük az egyedet, ha több kulcsjelöltet tartalmaz, de nem bontható le egymástól független egyedekre.

A normalizálási folyamatban ha ezen a ponton van független lebontása az egyednek, akkor a szétbontást el kell végezni. Ha nincs ilyen, akkor mérlegelni kell, hogy az eredeti egyed maradjon változatlanul, vagy el lehet végezni egy olyan megbontást, amelyik kevés bajjal jár.

Többértékű függés

Az E egyed B tulajdonsága akkor és csak akkor többértékű függéssel (*multivalued dependency*) függ az A tulajdonságtól, ha az adott A értéknek megfelelő B értékek készlete csak az A-tól függ és független az egyed C tulajdonságától. Röviden: *MVD*, jelölése: „ $\rightarrow$ ” „”.

TANFOLYAM

Kurzus	Tanár	Téma
Fizika	Zöld	Mechanika
Fizika	Zöld	Optika
Fizika	Barna	Mechanika
Fizika	Barna	Optika
Fizika	Fekete	Mechanika
Fizika	Fekete	Optika
Matematika	Fehér	Algebra
Matematika	Fehér	Geometria

6.23. ábra: Tanfolyam tábla (BCNF alak)

Ez a tábla **BCNF** alakban van, mert *csupakulcs* reláció, vagyis minden tétele az összetett kulcs része. A Tanár többértékű függéssel függ a Kurzus tulajdonságtól, mert a Kurzus értékhez (Fizika) adott Tanár értékthalmaz (Zöld, Barna, Fekete) tartozik és a Téma nem határozza meg a Tanárt (minden témát több tanár oktathat). A többértékű függés mindig párosával jár. Ha fenn áll az $A \rightarrow B$ függés, akkor léteznie kell az $A \rightarrow C$ függésnek is. Ha a kurzustól több értékűen függ a tanár, akkor a témának is ugyanúgy kell viselkednie, és megfordítva. Ezért a többértékű függőséget csak olyan egyeden lehet értelmezni, amely legalább három tulajdonságot tartalmaz (6.23. ábra).

A funkcionális függés a többértékű függés olyan speciális esete, amelyben a függő értékészlet egyetlen tétel. A többértékű függőség mindig páros, így a következő jelölés alkalmazható: $A \rightarrow B \mid C$.

Negyedik normálforma

Az E egyed **4NF** alakban van akkor és csak akkor, ha az abban lévő bármilyen $A \rightarrow B$ többértékű függőség (MVD) egyben $A \rightarrow B$ funkcionális függőség (FD) is.

Az előző ábrán két **MVD** van, a **Kurzus** többértékűen meghatározza a **Tanárt**, és attól többértékűen függ a **Téma** is. Ha ki van jelölve egy **Kurzus** érték, akkor meghatározható a **Tanárok** készlete, függetlenül a **Téma** értékektől. Fordítva ugyanez igaz a **Témára** is. A többértékű függés karbantartási anomáliát okoz. Ezért a **Tanfolyam** egyedet fel kell bontani a két többértékű függés mentén, vagyis két külön egyedtípust kell létre hozni.

Az előző ábrán a **Kurzus** $\rightarrow B$ **Tanár** többértékű függés nem volt funkcionális függés, így a **TANFOLYAM** egyed csak **BCNF** alakú. A következő ábra két egyede már **4NF** alakú, mert csupa kulcs relációk, és nincs bennük többértékű függés (6.24. ábra).

KURZUS TANÁRAI

<i>Kurzus</i>	<i>Tanár</i>
Fizika	Zöld
Fizika	Barna
Fizika	Fekete
Matek	Fehér

KURZUS TÉMÁI

<i>Kurzus</i>	<i>Téma</i>
Fizika	Optika
Fizika	Mechanika
Matek	Algebra
Matek	Geometria

6.24. ábra: **4NF** alak

Kapcsolásfüggés

Az E egyed akkor és csak akkor tesz eleget a kapcsolásfüggésnek, ha $X, Y, \dots, Z$ kivetítéseinek az összekapcsolásával helyreállítható úgy, hogy $X, Y, \dots, Z$ az E egyed tulajdonságainak a részhalmazai.

Ötödik normálforma

Az egyed akkor és csak akkor van **5NF** alakban, ha a kulcsjelöltjei között nem áll fenn kapcsolásfüggés. (Ami azt is jelenti, hogy nincsen közöttük többértékű és funkcionális függés sem.)

Az **5NF**-nek nincs gyakorlati értelme, ez csak a teljesség miatt került megemlítésre.

6.7. Relációs műveletek

Relációs algebra

A relációs algebra módszert ad arra hogyan építhetők új relációk a régiekből. A relációs algebrai kifejezések alapjait a relációk képezik, mint operandusok. A reláció megadható a nevével, vagy a sorainak egy listájával. A relációkra vagy egyszerűbb relációs algebrai kifejezésekre alkalmazva az operátorokat egyre

bonyolultabb kifejezések építhetők fel. A lekérdezés egy relációs algebrai kifejezés.

A relációs algebrai műveletek csoportjai a következők

- A relációkra alkalmazhatók a *hagyományos halmazműveletek* /egyesítés, metszet, különbségképzés/. A halmazműveletek csak azonos szerkezetű relációk között hajthatók végre.
- A reláció egyes részeit eltávolító műveletek: „*kiválasztás*” bizonyos sorok kihagyását, „*vetítés*” bizonyos oszlopok kihagyását jelenti.
- Két reláció sorait kombináló műveletek: „*Descartes-szorzat*”, amely az összes lehetséges módon párosítja a relációk sorait. „*Összekapcsolási*” műveletek: szelektíven párosítják össze két reláció sorait.
- „*Átnevezés*” művelet, amely nem változtatja meg a reláció sorait, de megváltoztatja a reláció sémáját, vagyis az attribútumok neveit és / vagy a reláció nevét.

Egyesítés (unió) $\cup$

Az unió művelete azonos szerkezetű két vagy több reláció között végezhető el. $A \cup B$, az A és B relációk egyesítése, olyan elemek halmaza, amelyek vagy az A-ban vagy a B-ben vannak, vagyis legalább az egyik relációnak elemei. Minden elem egyszer szerepel az egyesítésben, még akkor is ha benne van A-ban és B-ben is.

Metszet $\cap$

A metszet művelete azonos szerkezetű két vagy több reláció között végezhető el. $A \cap B$, az A és B relációk metszete, csak azokat a reláció sorokat tartalmazza, amelyek az A-ban és a B-ben is benne vannak.

Különbség $\setminus$

$A \setminus B$, az A és B relációk különbsége, olyan reláció sorokat tartalmaz, amelyek az A-ban benne vannak, de B-ben nincsenek. $A \setminus B \neq B \setminus A$.

A halmazműveletek relációkra történő alkalmazásakor a következő feltételeknek teljesülniük kell:

- Az A és B relációk sémájának ugyanazt az attribútum halmazt kell tartalmaznia.
- Az A és B relációk attribútumainak sorrendjének azonosnak kell lennie, azaz a relációk oszlopainak sorrendjének kell azonosnak lennie.

Vetítés

A vetítés operátorral az A relációból olyan új reláció hozható létre, amelyik csak az A bizonyos oszlopait tartalmazza. A vetítés eredménye egy attribútum halmaz, amelyet egy rendezett listával lehet jelölni.

Kiválasztás

A kiválasztás operátor az A relációból olyan új relációt hoz létre, amelyik az A sorainak csak egy részét tartalmazza. Olyan sorok kerülnek be az eredménybe, amelyek az A attribútumaira vonatkozó feltételeket teljesítik. Az eredmény reláció sémája azonos az eredeti A reláció sémájával, és az attribútumokat is azonos sorrendben tartalmazza.

A feltételes kifejezés hasonló, mintha programozási nyelven írnánk. A feltételben lévő operandusok konstansok vagy az A reláció attribútumai.

Descartes-szorzat

Az A és B relációk Descartes-szorzata (egyszerűen szorzata) olyan párok halmaza, amelyek első eleme az A tetszőleges eleme, a második eleme a B halmaz tetszőleges eleme. A Descartes-szorzat jelölése: $A \times B$. Mivel A és B relációk, a szorzat eredménye is reláció. A relációk elemei sorok, a Descartes-szorzatuk olyan hosszabb sort jelent, amelyben az alkotó sorok mindegyik komponense megjelenik. Az A minden sora mellé minden lehetséges módon melléírjuk B sorait. Az A komponensei sorrendben előbb szerepelnek a B komponenseinél.

Az $A \times B$ eredményreláció sémája az A és a B reláció sémájának egyesítése. A közös attribútumokat meg kell különböztetni. A.G illetve B.G jelöli a közös attribútumokat, amelyek az A illetve a B relációban szerepeltek.

Természetes összekapcsolás

Az A és B természetes összekapcsolásán azt értjük, hogy A-nak és B-nek csak azokat a sorait párosítjuk össze, amelyek értékei megegyeznek az A és B sémájának közös attribútumain. Ha $G_1, G_2, \dots, G_n$ olyan attribútumok, amelyek megtalálhatóak az A és B sémájában is, akkor az A egy a sorának és B egy b sorának a párosítása csak akkor történhet meg sikeresen, ha az a és b értékei megegyeznek az összes attribútumban.

Az összekapcsolás jelölése: $A \bowtie B$. Az összekapcsolt sor megegyezik az a sorral az A reláció összes attribútumán, és megegyezik a b sorral B összes attribútumában. Sikeres összekapcsoláskor az a és b sorok megegyeznek a közös attribútumokban.

Théta-összekapcsolás

Az A és B relációknak a G feltételre vonatkozó théta-összekapcsolásának jelölése: $A \bowtie_G B$. A művelet eredménye az alábbiak szerint adódik:

- Meghatározzuk A és B szorzatát.
- A szorzatból kiválasztjuk azokat a sorokat, amelyek eleget tesznek a G feltételnek.
- Az eredmény sémája az A és B sémájának egyesítése. Ha jelölni akarjuk, hogy az attribútumok melyik sémából származnak, akkor használjuk az „A.” és „B.” jelöléseket az attribútumok előtt.
-

Átnevezés

A relációra alkalmazható olyan operátor is, amely átnevezi az A relációt. Ez így jelölhető: $\rho_{C(G_1, G_2, \dots, G_n)}(A)$. Az eredmény reláció neve C, sorai megegyeznek az A soraival és attribútumainak nevei rendben $G_1, G_2, G_3, \dots, G_n$. Ha csak a reláció neve változik, az attribútum nevek nem, akkor ezt így jelölhető: $\rho_C(A)$.

Ellenőrző kérdések:

1. Mit értünk az adat és az adatbázis kifejezéseken?
2. Milyen adatmodellek ismertek?
3. Jellemezze a hierarchikus adatmodellt!
4. Jellemezze a hálós adatmodellt (kétszintű hálós adatmodell, Codasy1-hálós)!
5. Jellemezze az objektum-orientált adatmodellt!

6. A relációs adatmodell fejlődése.
7. Jellemezze a relációs adatmodellt!
8. Mit értünk táblán, mezőn, rekordon, nézettáblán, kulcson, idegen kulcson?
9. Milyen kapcsolattípusok vannak a relációs adatmodellben? Jellemezze ezeket!
10. Hogyan ábrázolhatók a kapcsolattípusok a relációs adatmodellben?
11. Ismertesse a relációs adatbázis kezelő szoftverek fejlődését!
12. Milyen elvárások vannak az adatbázis kezelő rendszerrel szemben?
13. Az adatbázis kezelő rendszerek felépítése. Az egyes elemek feladatait ismertesse!
14. Mit értünk metaadatok alatt?
15. Mit értünk tranzakción és tranzakció helyes lefutásán?
16. Milyen „technikák” biztosítják a tranzakció helyes lefutását?
17. Mit értünk redundancián, inkonzisztencián, normalizáláson, normálformán?
18. Mit jelent a funkcionális függés?
19. Mit jelent 0NF, 1NF, 2NF, 3NF, BCNF, 4NF, 5NF?
20. Mit jelent a részleges függés?
21. Mit értünk tranzitív függésen?
22. A relációs algebrai műveleteket sorolja fel! Mit jelentenek az egyes műveletek?

Irodalomjegyzék

- Bakó Sándor (1999): Adatbáziskezelés, Pedellus Novitas Kft, Debrecen.
- Békéssy András - Demetrovics János (2005): Adatbázis-szerkezetek, Akadémiai Kiadó, Budapest.
- Hallasy Béla (2002): Adatmodellezés, Nemzeti Tankönyvkiadó Rt, Budapest.
- Hector Garcia-Molina – Jeffrey D. Ullman – Jennifer Widom (2001): Adatbázisrendszerek megvalósítása, Panem Könyvkiadó Kft, Budapest.
- Jeffrey D. Ullman – Jennifer Widom (1998): Adatbázisrendszerek, Alapvetés, Panem Könyvkiadó Kft, Budapest.
- Michael J. Hernandez (2004): Adatbázis-tervezés, A relációs adatbázisok alapjairól földi halandóknak, Kiskapu Kiadó, Budapest.
- Rajtik János (2004): Adatbázis-kezelés MS ACCESS 97, Pedellus Tankönyvkiadó Kft, Debrecen.

7. RELÁCIÓS ADATBÁZIS KEZELŐ RENDSZEREK A GYAKORLATBAN

Ebben a fejezetben először az Access adatbázis kezelő program rövid ismertetésére kerül sor. Ezt követi a relációs adatbázis kezelő rendszerek szabványos lekérdező nyelvének a legfontosabb utasításainak összefoglalása.

Az Access adatbázis-kezelő rendszer

7.1. Adatműveletek, adattábla kezelése

A relációs adatbázis az adatokat *relációkban* tárolja, amelyek a felhasználó számára *táblák* formájában jelennek meg. Egy tábla egy témakörhöz tartozó adatokat rendezett formában tárolja. A tábla sorokból, oszlopokból áll, jól meghatározott szerkezete van, egyedi névvel lehet rá hivatkozni. A tábla oszlopait mezőknek nevezik, amelyeknek neve, típusa, mérete van. A tábla sorai a rekordok, amelyek ugyanazokból a mezőkből állnak.

Az egyes mezőkhöz rendelhető fontosabb adattípusok.

Szöveg: legfeljebb 255 karakter hosszú karakterlánc, alapértelmezett hosszúsága 50 karakter.

Feljegyzés: legfeljebb 65535 karakternyi szöveg.

Szám: numerikus adatok tárolására.

Dátum/idő: 100-tól 9999-ig terjedő években dátum és időpont tárolására szolgál.

Pénznem: szimbólum, amely az érték mellett szerepel.

Számláló: automatikusan generált egyedi sorszám, értéke egyesével növekszik új rekordok felvételénél.

Igen/nem: logikai érték.

OLE objektum: más alkalmazásban (például Word, Excel) létrehozott objektum tárolására használható (szöveges dokumentum, kép, hang, stb.).

Hiperhivatkozás: például URL cím. Hossza legfeljebb 65535 karakter, nem indexelhető.

A következő mezőtulajdonságokat lehet megadni.

Mezőméret: a szövegmező mérete 1 és 255 karakter között lehet.

Formátum: az adatok megjelenítési és kinyomtatási módja.

Tizedes helyek: a tizedes helyek száma, a szám és a pénznem típusoknál.

Beviteli maszk: formázó karakterek megjelenítésére szolgál, amelyek biztosítják, hogy a bevitt adatok megfeleljenek a megadott maszknak.

Cím: a mező átnevezése végezhető el ezzel. Az eredeti név helyett ez jelenik meg az adattáblában, a lekérdezésben, az űrlapon és a jelentésben.

Alapértelmezett érték: új rekord hozzáadásakor automatikusan ez az érték jelenik meg a mezőben.

Érvényességi szabály: a mezőbe bevihető adatok körét korlátozó kifejezés.

Érvényesítési szöveg: az adatmezőben érvénytelen adat beírásakor megjelenő figyelmeztető üzenet.

Kötelező: a mezőbe mindig kell megadni adatot, nem hagyható üresen.

Nulla hosszúság engedélyezése: **Szöveg** vagy **Feljegyzés** típusú mezőknél megengedett a nulla hosszúságú karakterlánc is.

Indexelt: az adott mező értékeire gyorsan kereshető lesz a tábla.

Új érték: Számláló típusú mezőnél megadható, hogy növekvően vagy véletlenszerűen kapja a mező az értékét új rekord hozzáadásakor.

Adatműveletek

Táblához új rekord hozzáadása

Minden tábla tartalmaz egy látszólagos üres rekordot a tábla alján. Ezt a * karakter jelzi a tábla bal szélén a rekordkijelölő sávban. Ha ebbe a rekordba elkezdődik az adatbevitel, akkor alatta automatikusan megjelenik egy újabb utolsó rekord. Az adattábla mezőibe bevitt adatokat később meg lehet változtatni. Az elkezdett módosítások az ESC billentyűvel visszavonhatóak.

Táblában rekord mentése

Az Access a táblában egy rekord bevitele vagy módosítása után minden változtatást automatikusan elment, amikor a következő rekordra lépés történik, vagy az adatlap bezárásra kerül. Ha rekord változtatásait egy mező tartalmának megváltoztatása után azonnal menteni kell, akkor a SHIFT+ENTER billentyűket kell megnyomni vagy a **Rekordok\Rekordok mentése** funkciót kell választani.

Táblában rekord törlése

A táblában a rekord törléséhez ki kell jelölni a törlendő rekordot, ez a rekordkijelölő sávba való kattintással történik. Egyszerre több rekordot is ki lehet jelölni, ha az egér balgombját lenyomva tartva mozgatjuk azt a rekordkijelölő sávban. A törlés elvégezhető például a **Szerkesztésmenü Rekord törlése** funkcióval, vagy az eszköztár **Rekord törlése** ikonnal is. Hasonlóan törölhető oszlop (mező) is.

Másolás, áthelyezés

Ezek a műveletek teljesen hasonlóan elvégezhetők, mint az Excelben.

Tábla rendezése

A táblában először ki kell jelölni azt a mezőt, amely szerint rendezni kell. Majd a **Rekord menü Rendezés** funkciót kell választani, ahol ki kell választani, hogy növekvő vagy csökkenő sorrendű legyen a rendezés.

Ha a rendezést bonyolultabb feltétel szerint kell elkészíteni, akkor a **Rekordok menü Szűrő parancs Szűrés** kijelöléssel menüpontot kell választani. A feltételnek nem megfelelő rekordok megkeresése a **Rekordok menü Szűrő parancs Szűrés kizárással** funkcióval kezdeményezhető.

Az űrlappal történő szűréskor egyszerre több mező tartalmára is megadható feltétel vagy egy mezőre több feltételt lehet beállítani. A **Rekordok menü Szűrő parancs Szűrés űrlappal** menüpont választása után megjelenő adatlapon az egyes mezőkhöz tartozó szűrőértékeket legördülő menüből lehet kiválasztani. Az összes feltétel beállítása után a **Szűrő menü Szűrés parancs rendezés** funkció választásával megkaphatóak a táblában a feltételeknek megfelelő rekordok. Még összetettebb feltétel adható meg az **Irányított szűrés**i

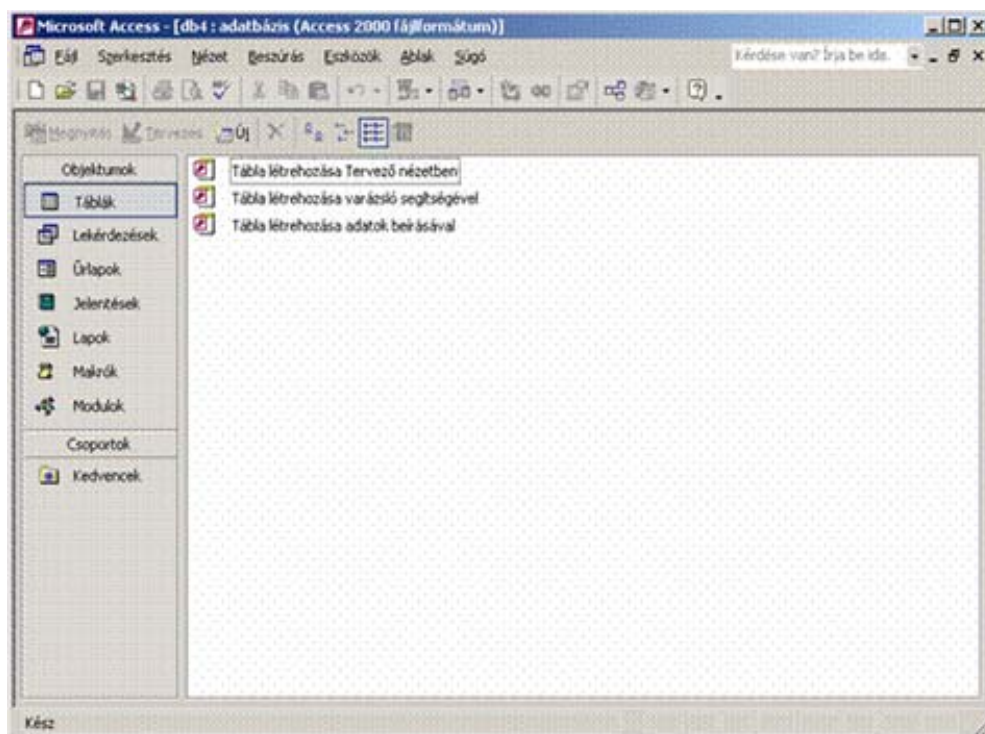
eljárással. Ez a Rekordok menü Szűrő parancs Irányított szűrés funkciórarendezés menüponttal kezdeményezhető.

A kiszűrt vagy a rendezett rekordok helyett újból az összes rekord megjelenítését a Rekordok menü Rendezés törlése menüpontból lehet kérni.

7.2. Adatbázis tervezése (relációs adatbázis tervezése). Adatbázis létrehozása

Adatbázis létrehozása

Új adatbázis létrehozása vagy meglévő adatbázisfájl megnyitása után megjelenik az Adatbázis ablak (7.1. ábra).



7.1. ábra: Adatbázis ablak

Az ablak bal oldalán választható ki az objektumtípus, utána a jobb oldalon megjelenő elemek közül lehet választani. Az ablak tetején szereplő **Megnyitás**, **Tervezés**, **Új** gombok értelemszerűen használhatók a kiválasztott objektumtípusra.

Az egyes adatbázis-objektumok létrehozására általában három lehetőség van.

- *Automatikus*: ekkor az Access automatikusan elkészíti az objektum szokásos, legcélszerűbb változatát.
- *Varázsló segítségével*: a szokásos varázsló-technikával végigvezet a tervezési folyamaton.
- *Tervező nézetben* teljesen manuális a tervezés, itt a rendszer valamennyi lehetősége elérhető.

Az Access program kezelése többnyire kézenfekvő, ezért a továbbiakban az egyes részek ismertetése nem teljes részletességgel történik.

A tábla létrehozása a **varázslóval** a leggyorsabb. A **varázslóban** lehet választani üzleti és **személyes** mintatáblákból és mintamezőkből. Ezek közül kiválaszthatók a szükségesek, át is nevezhetők. Az elsődleges kulcs is megadható.

A tábla létrehozása az **adatok beírásával** lehetőség kiválasztásakor 20 sorból és 10 mezőből álló üres táblázat jelenik meg, amelynek a neve **Tábla1**, a mezők nevei: **Mező1**, ... , **Mező10**. A mező nevek tetszőlegesen módosíthatók.

Tervező nézetben a tábla szerkezetét lehet megadni, adatokkal nem tölthető fel a tábla. A **Tervező nézet** választása után három oszlopos táblázat jelenik meg. Az első oszlop a mező neve, második a mező adattípusa, a harmadik oszlop a mező leírása. A táblázat alatt az adattípusoktól függő alapértelmezett mezőtulajdonságokat lehet megadni (7.2. ábra).

Mezőnév	Adattípus	Leírás
AKO	Szöveg	Árukód
ANE	Szöveg	Áru megnevezése
MEGYS	Szöveg	Mennyiségi egység
AFK	Szám	ÁFA kulcs
NYAR	Szám	Nyilvántartási ár

Általános	Megjelenítés
Mezőméret	50
Formátum	
Beviteli maszk	
Cím	
Alapértelmezett érték	
Érvényességi szabály	
Érvényesítési szöveg	
Kötelező	Nem
Nulla hosszúság engedély	Igen
Indexelt	Igen (nem lehet azonos)
Unicode-tömörítés	Igen
IME-mód	Nem beállított
IME-mondatmód	Nincs konverzió

A mezőnév legfeljebb 64 karakter hosszú lehet a szóközöket is beleértve. Az F1 billentyű lenyomására megjelenik a súgó a mezőnévről.

Tervező nézet. F6 = Ablakok közötti váltás. F1 = Súgó.

7.2. ábra: Tábla létrehozása Tervező nézetben

Tábla szerkesztése

Ha már elkészült tábla szerkezetét kell megváltoztatni, akkor a tábla kijelölése után a **Tervezés** gombot kell választani. A **Tervező nézetben** ismertetett három oszlopos táblázat jelenik meg. A **mezőnév** oszlopban kell kijelölni a módosítandó mezőt, az ablak alján a **mezőtulajdonságok** és a hozzá tartozó értékek láthatók. Ekkor van lehetőség az adott mező típusának a megváltoztatására. Az adattípus beállítása után megadhatók a mezőtulajdonságok.

Táblák összekapcsolása

Jól megtervezett adatbázis esetében minden tábla csak egyféle dologról tartalmaz adatokat. Ha a táblákat együtt kell kezelni, akkor össze kell kapcsolni azokat. Az összekapcsolás után a rekordok úgy viselkednek, mintha egyetlen táblában lennének. Az összekapcsolás feltétele, hogy a különböző táblákban legyenek

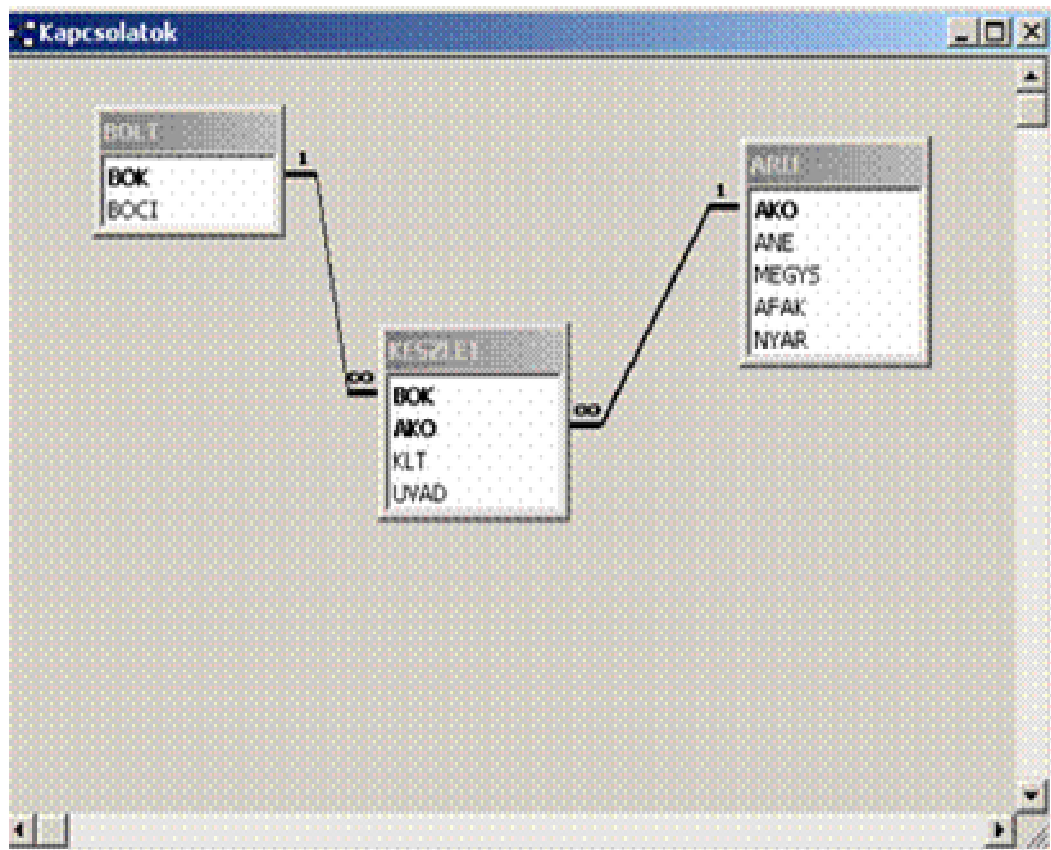
azonos mezők, amelyekkel elvégezhető az összekapcsolás. Ez a **kulcs mező**. A kulcs mezőnek az összekapcsolandó két táblában azonos adattípusúnak, azonos méretűnek és azonos mezőtulajdonságúnak kell lennie. Az egyik táblában a kulcs mező elsődleges kulcs, a másikban idegen kulcs.

Elsődleges kulcs létrehozása

A tervező ablak bal szélén ki kell választani a kulcs sorát, és az eszköztár **Elsődleges kulcs** gombjára kell kattintani. Összetett (több mezőből álló) kulcs esetén **Ctrl** lenyomva tartásával több sort tudunk egyszerre kijelölni.

Kapcsolat létrehozása

Kapcsolat létrehozása az **Eszközök** menü **Kapcsolatok** menüpontból kezdeményezhető. Megjelenik a **Tábla hozzáadása** panel, ahol ki kell választani azokat a táblákat, amelyek részt vesznek a kapcsolatok kialakításában, majd a kapcsolatot biztosító mezőt az elsődleges táblából egérrel át kell húzni a másik tábla megfelelő mezőjéhez. A **Létrehozás** gomb megnyomásával befejeződik a kapcsolatkészítés (7.3. ábra).



7.3. ábra: Kapcsolatok az Access-ben

A kapcsolat jellemzőinek módosítását vagy törlését a kapcsolati vonalra kattintva lehet elvégezni (7.4. ábra).



7.4. ábra: Kapcsolatok szerkesztése ablak

A táblák közötti kapcsolatok típusai

Egy-egy (1:1) kapcsolat, egy-sok (1:N) kapcsolat, sok-sok (N:M) kapcsolat. Ezek a kapcsolatok részletesen ismertetésre kerültek a 6. fejezetben.

Az Access *elsődleges táblának* nevezi az 1:N kapcsolatok 1-oldalán álló táblát, *illesztő-táblának* az N:M kapcsolatot megvalósító táblát, és *illesztésnek* az összekapcsolás műveletet.

Adatok aktualizálása

Az Adatbázis ablakban ki kell választani a táblát, majd **Megnyitás** gomb. Módosítás közben a rendszer ellenőrzi a kulcsfeltételeket. Az üres sztring megadását a rendszer általában Null (definiálatlan) értéknek tekinti.

7.3. Lekérdezések. Szűrések a lekérdezésben

A lekérdezés elkészítési céljai: mezők kiválasztása, rekordok kiválogatása, adatok törlése, több táblából adatok kigyűjtése, adatok csoportosítása, számítások végzése. A lekérdezés eredményeként létrejövő táblázatban csak a kért mezők és tartalmuk jelenik meg.

Lekérdezés típusának megadása

- választó lekérdezés,
- keresztáblás lekérdezés,
- táblakészítő lekérdezés,
- frissítő lekérdezés,
- hozzáfűző lekérdezés,

- törlő lekérdezés.

Lekérdezés elkészíthető Tervező nézetben vagy varázslóval.

Lekérdezés készítése Tervező nézetben

Az Adatbázis-ablakban a lekérdezés objektumtípust, majd az Új gombot, és végül a Tervező nézetet kell választani. A Tábla hozzáadása ablakban ki kell jelölni a lekérdezéshez szükséges táblákat. Utána megjelenik a Tervező ablak, amelynek beállítási lehetőségei.

- A Mező és Tábla sorokban a lekérdezéshez szükséges mezők adhatók meg.
- A Rendezés sorban az adott mező szerinti rendezettség írható elő.
- A Megjelenítés sorban jelölhető, hogy az adott mező megjelenjen-e az eredménytáblában, vagy például csak a feltételek kiértékeléséhez szükséges.
- A Feltétel sorban az adott mezőre vonatkozó feltétel adható meg. A mező nevét itt nem kell újra kiírni, feltétel lehet például "> 100".

A lekérdezés végrehajtása az Adatbázis ablakban kettős kattintással, vagy a Megnyitás gombbal történik.

Választó lekérdezés

A leggyakrabban használt lekérdezés típus. A lekérdezés készítéséhez felhasználható egy vagy több tábla. A lekérdezés eredménye egy változó tartalmú adattábla, tehát a tábla tartalma annak megfelelően módosul, ahogyan az input táblák változnak. Két változata van: az Azonosakat kereső és a Nem egyezőket kereső lekérdezés. A választó lekérdezés tervező nézetében a gyorsmenüből a Tulajdonságok pontot választva a lekérdezés jellemzőit lehet beállítani (7.5. ábra).

A lekérdezés tulajdonságainak értelmezése:

Leírás: a lekérdezés céljának rövid leírása, maximum 255 karakter.

Összes mező a kimenetre: ha ezt Igen-re állítják, akkor a lekérdezés összes input tábláinak összes mezője megjelenik a lekérdezés eredményeként. A Nem választása esetén csak a kiválasztott mezők tartalma jelenik meg.

Csúcsérték: beállítható, hogy a lekérdezés feltételének megfelelő rekordok közül az első hány darab, a rekordok első hány százaléka vagy az összes jelenjen meg.

Egyedi értékek: ha Igen-re állítják, akkor a lekérdezésben szereplő azonos értékek közül csak egy jelenik meg.

Egyedi rekordok: ha Igen-re állítják, akkor a lekérdezésben csak azok a rekordok jelennek meg, amelyek minden mezőtartalomban különböznek a többitől.

Futási engedélyek: a beállított Felhasználóé vagy Tulajdonosé értéknek megfelelően az adathozzáférési jog vagy a felhasználóé vagy a tulajdonosé.

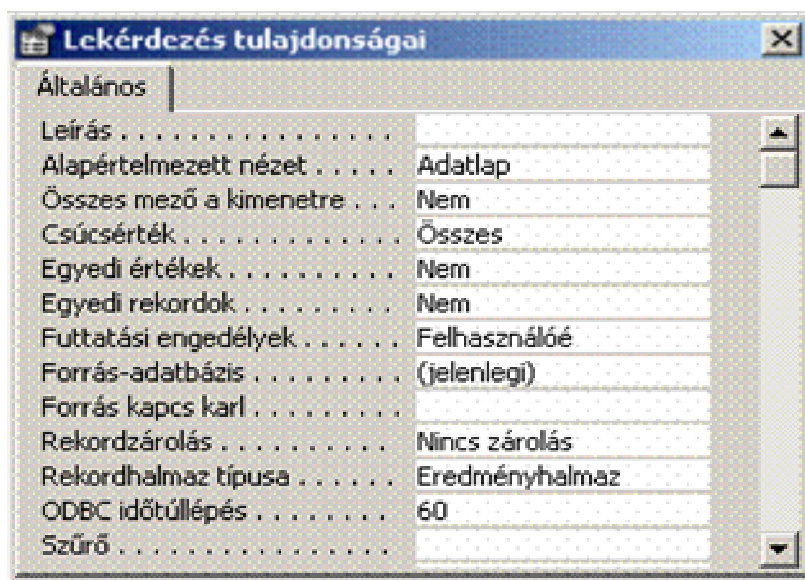
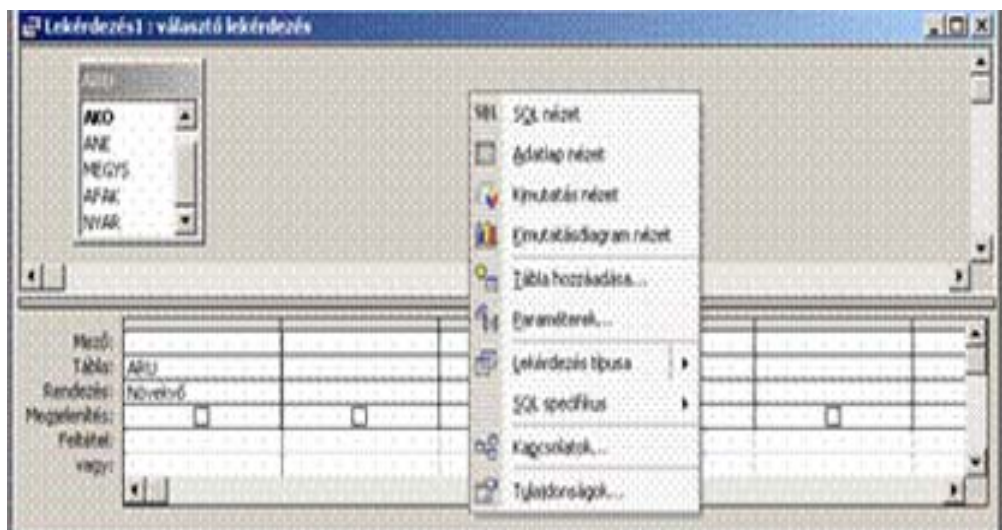
Forrás adatbázis: a külső adatbázisnak az elérési útját tartalmazza, amelynek az adatait a lekérdezés tartalmazza.

Forrás kapcs karl: a külső adatbázis létrehozásához felhasznált alkalmazás nevét jelentő karakterlánc.

Rekordzárolás: több felhasználós környezetben a lekérdezés futtatása idejére adatvédelmi okokból különböző érvényességi körben zárolni kell a rekordokat. A zárolt rekord módosítása másik felhasználó által nem lehetséges.

ODBC időtúllépés: beállítható, hogy nyílt adatbázishoz kapcsolódva hány másodperc után jelezzen időtúllépést az Access.

Szűrő: szűrőfeltétel adható meg, hasonlóan az SQL WHERE záradékban megszokott formában. A WHERE kulcsszót nem kell használni.



7.5. ábra: Választó lekérdezés

Keresztábrás lekérdezés

Ez a lekérdezés ugyanazokat az adatokat tartalmazza, mint a választó lekérdezés, de az adatokat nemcsak vízszintesen, hanem függőlegesen is csoportosítja. A választó lekérdezés tulajdonságain kívül ebben a lekérdezésben szerepel még az oszlopfejléc. *Oszlopfejléc:* beállítható, hogy a lekérdezés eredmény táblájában milyen oszlopok és milyen sorrendben jelenjenek meg.

Táblakészítő lekérdezés

Ez a lekérdezés egy vagy több táblából új táblát hoz létre. A lekérdezés Tulajdonság ablaka a Választó lekérdezéshez képest a következőket tartalmazza:

Céltábla: az új, létrehozandó tábla neve.

Cél-adatbázis: annak a külső (Access) adatbázisnak az elérési útvonala és neve, amelyikbe átkerülnek a rekordok.

Cél kapcs karl: Access adatbázis esetén nem használják. Nem Access adatbázis esetén itt lehet megadni az adatbázis típusát.

Törlő lekérdezés

Az adattáblából ezzel törölhetők a feleslegessé vált rekordok. Csak teljes rekordokat lehet törölni, egyes mezőket nem. Figyelni kell a kapcsolt adattáblákra, mert törölődhetnek a másik adattábla rekordjai is, ha be van állítva a Hivatkozási integritás és a Kapcsolt mezők kaszkádolt törlése.

Hozzáfüző lekérdezés

Ezzel a lekérdezéssel egy vagy több tábla rekordcsoportjait lehet egy másik tábla végéhez hozzáfüzni. Így az adatok újragépelése nélkül lehet adatokat bevinni a táblába egy már létező táblából. Alkalmazható csak az adott feltételnek megfelelő rekordok hozzáfüzésére is.

Frissítő lekérdezés

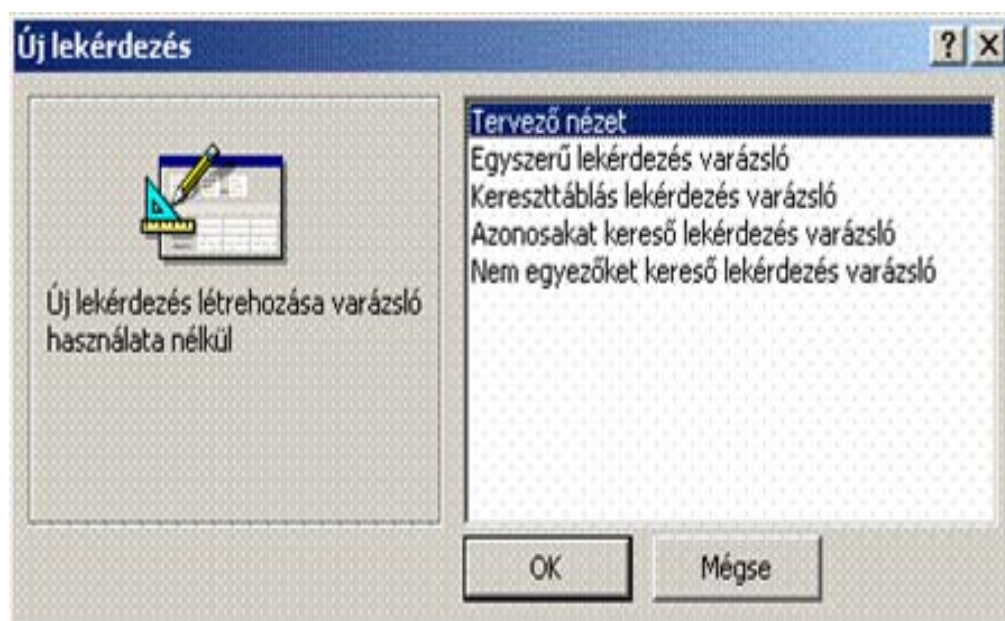
Olyan esetekben célszerű ezt alkalmazni, amikor sok rekord egy vagy több mezőjét ugyanúgy kell módosítani. Például egy termék árának megadott százalékkal történő megváltoztatása.

Lekérdezések többféle módon hozhatók létre.

- Lekérdezés panelen az Új gombot választva.
- Beszúrás\Lekérdezés menü funkció választásával.

Lekérdezések létrehozása

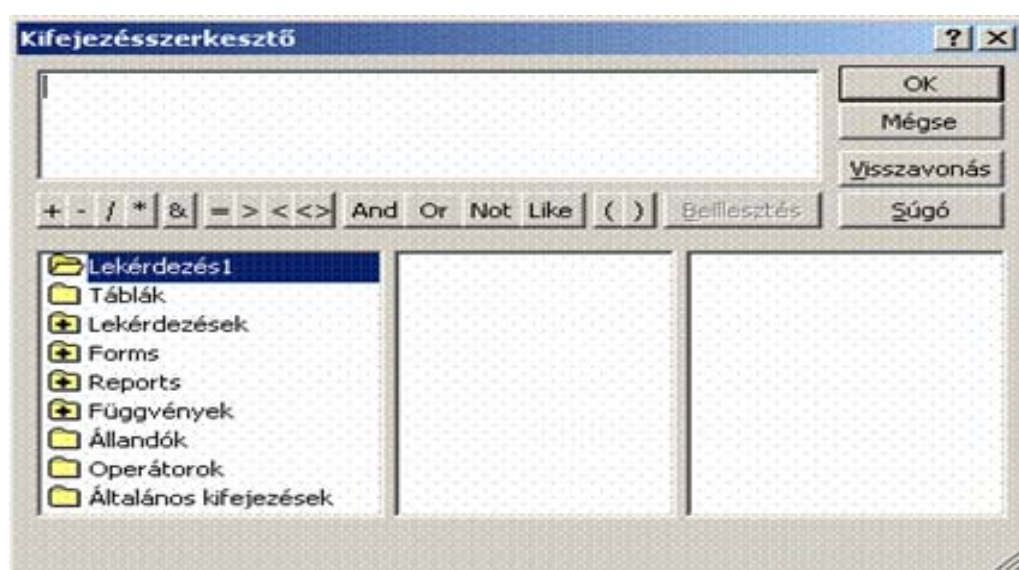
Az Új lekérdezés ablakban lehet választani, milyen módon készüljön a lekérdezés (7.6. ábra). A varázslók használatával gyorsabban hozhatók létre lekérdezések, de végül még tervező nézetben meg kell adni a feltételeket.



7.6. ábra: Új lekérdezés ablak

Lekérdezés módosítása

A lekérdezés módosításának oka lehet, ha például az eredmény halmazban nem szerepel valamelyik szükséges mező. A módosítások elvégezhetők a **Lekérdezés Tervező nézetében** megjelenő **tervezőrács** használatával. Ez egy grafikus eszköz, a neve **QBE (Query By Example; magyarul: lekérdezés minta alapján)**. A tervezőrács mutatja a lekérdezés hatására létrejövő eredményhalmaz oszlopait. Itt megadható, hogy a kijelölt adattáblákból mely mezők tartalma szereljen az egyes oszlopokban. A tervezőrács tartalma a 7.5. ábrán látható. A tervezőrács utolsó két sorában a feltétel szerkesztéséhez alkalmazható a **Kifejezésszerkesztő**, amely a **Tervezőrács Feltétel** sorából a gyorsmenüből elérhető (7.7. ábra).



7.7. ábra: Kifejezésszerkesztő ablak

7.4. Űrlapok, jelentések készítése

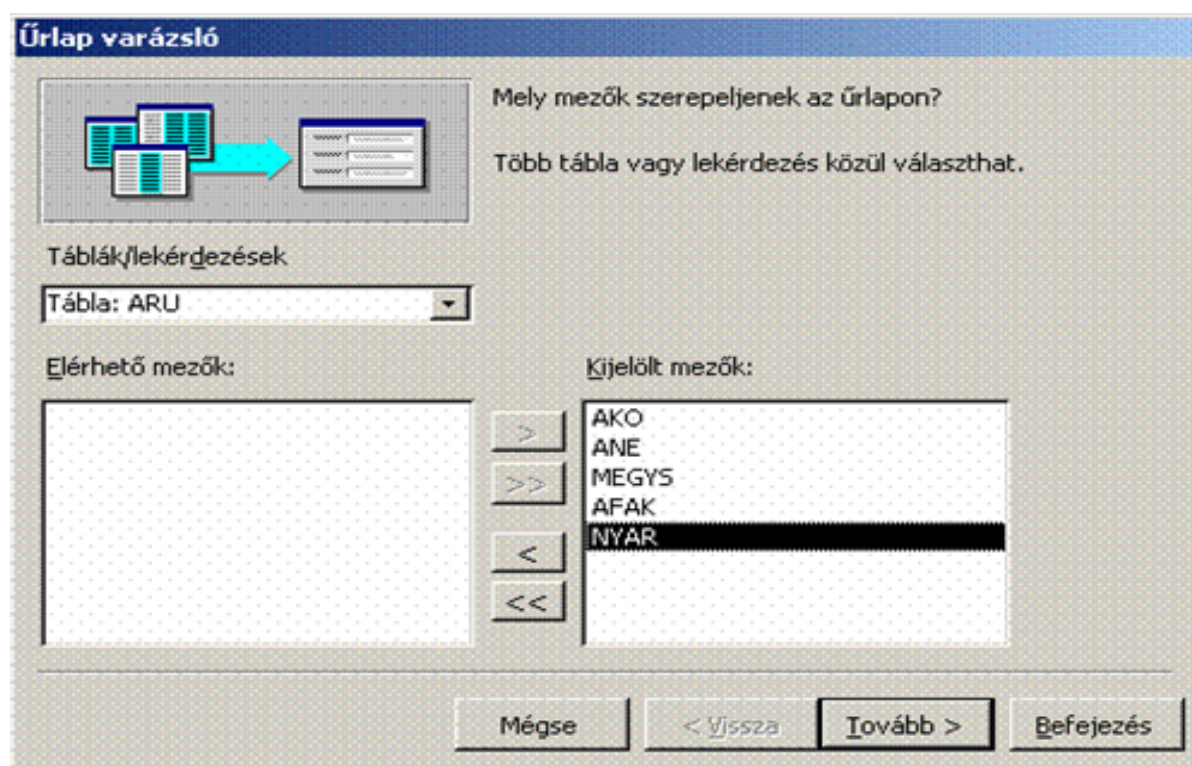
Űrlap létrehozása

Az űrlap létrehozása előtt meg kell tervezni, hogy mi legyen annak tartalma és milyen szakaszban. Az Accessben az űrlapnak öt szakasza van.

- *Űrlapfejléc:* tartalmazhatja az űrlap címét, rövid kezelési tájékoztatást vagy parancsgombot. Az űrlap kinyomtatásakor ez csak a legelső oldal tetején jelenik meg.
- *Oldalfejléc:* a nyomtatott oldalak fejrészét tartalmazza.
- *Törzs:* adatbeviteli mezőket, vezérlőelemeket tartalmaz.
- *Oldallábléc:* a nyomtatott oldalak alján megjelölő adatok, például: lapszám, dátum, stb.
- *Űrlaplábléc:* vezérlőelemek, információk, amelyek csak egyszer, az űrlap kitöltése után jelennek meg.

Az űrlap létrehozható Tervező nézetben vagy varázsló alkalmazásával.

Az Űrlap varázsló ablakban kiválasztható, hogy mely táblából, lekérdezésből készüljön az űrlap, és milyen mezőket tartalmazzon (7.8. ábra). A második lépésben megadható, milyen legyen az űrlap szerkezete. A harmadik lépésben lehet megválasztani a stílust. A varázsló negyedik lépésében megadható az űrlap címe, és lehet választani az űrlap megnyitása, megtekintése, adatbevitel, valamint az űrlap tervének módosítása között. A varázsló működésének befejezése után elkezdhető az adatok bevitele a kiválasztott formátumban (7.9. ábra).



7.8. ábra: Űrlap varázsló ablak

7.9. ábra: Adatbeviteli űrlap

Az űrlap létrehozásának másik módja **Tervező nézet** választása. Ekkor a képernyőn megjelenik az üres űrlap, a méretezést segítő vonalzó. Az űrlapon a különböző vezérlő elemek elhelyezési igazítását egy rács segíti, amit a **Nézet** menüben lehet ki- illetve bekapcsolni. Az eszköztárban lévő **mezőlista** gombbal, vagy a **Nézet\mezőlista** funkcióval lehet valamely adattáblához tartozó mezőket elhelyezni az űrlapon. Az űrlap szerkesztéséhez felhasználhatók az **eszközkészlet** funkciói. A **vezérlőelem** olyan objektum, amely adatokat jelenít meg, műveletet hajt végre, vagy szebbé teszi az űrlapot. A vezérlőelemek formátuma nagy részletességgel szabályozható.

Jelentések

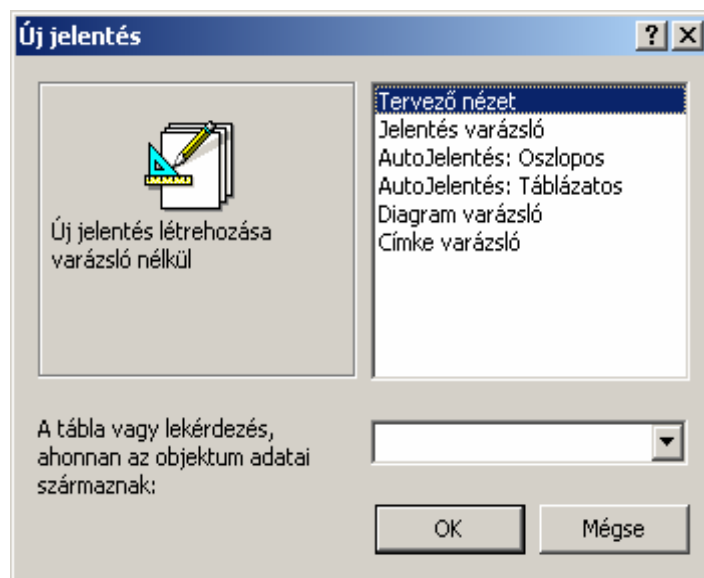
Az adatok elemzésére vagy nyomtatott formában való megjelenítésére jelentések alkalmazhatók. Így például kinyomtatható egy jelentés, amely csoportosítja az adatokat, és összegzéseket végez, míg egy másik jelentés különböző módon formázott adatokat tartalmazhat borítékcímkék nyomtatásához vagy körlevelekhez.

Jelentést akkor alkalmaznak, ha az adatbázisban tárolt adatokat kell kinyomtatni. Lehetőség van táblák és űrlapok kinyomtatására is, de igazán esztétikus eredmény jelentések készítésével érhető el.

Jelentés készítése

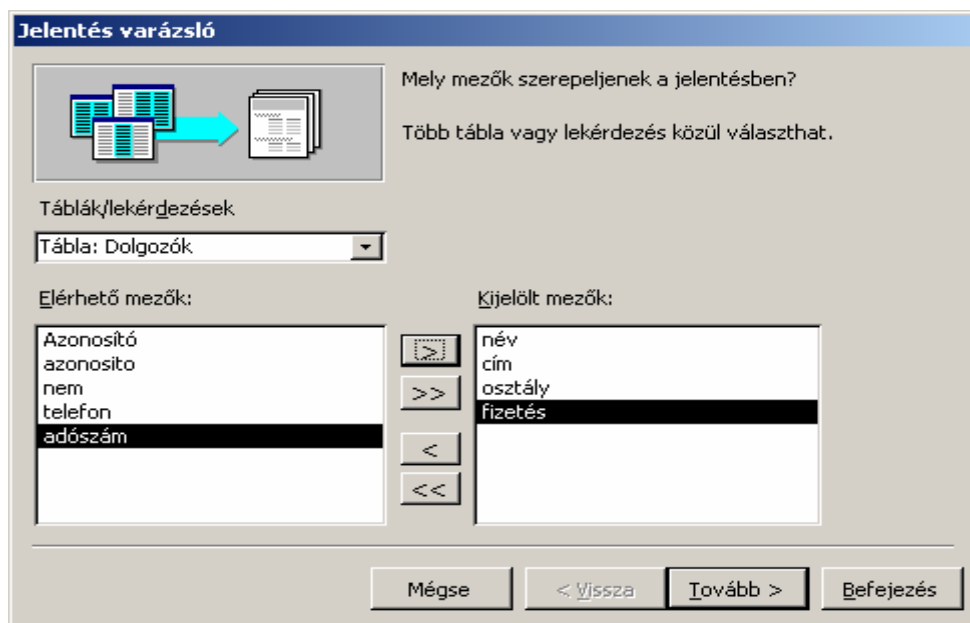
A jelentéseket mindig érdemes **varázslóval** készíteni, mert szinte tetszőleges kimutatás elkészíthető vele. Ezért a következőkben a **varázsló** lépései kerülnek ismertetésre.

A példában egy olyan jelentés készítése követhető nyomon, amelyben osztályonként megjelennek a *dolgozók nevei, címei, és fizetéseik, osztályonként kiemelve az átlagfizetést és az ott dolgozók számát*. A jelentés készítés elkezdéséhez az egérrel kattintani kell az adatbázis-ablak **Objektumok** paneljében a **Jelentések** gombra, majd meg kell nyomni az **Új** gombot. A felkínált lehetőségek közül a **Jelentés varázsló** kiválasztható (7.10. ábra).



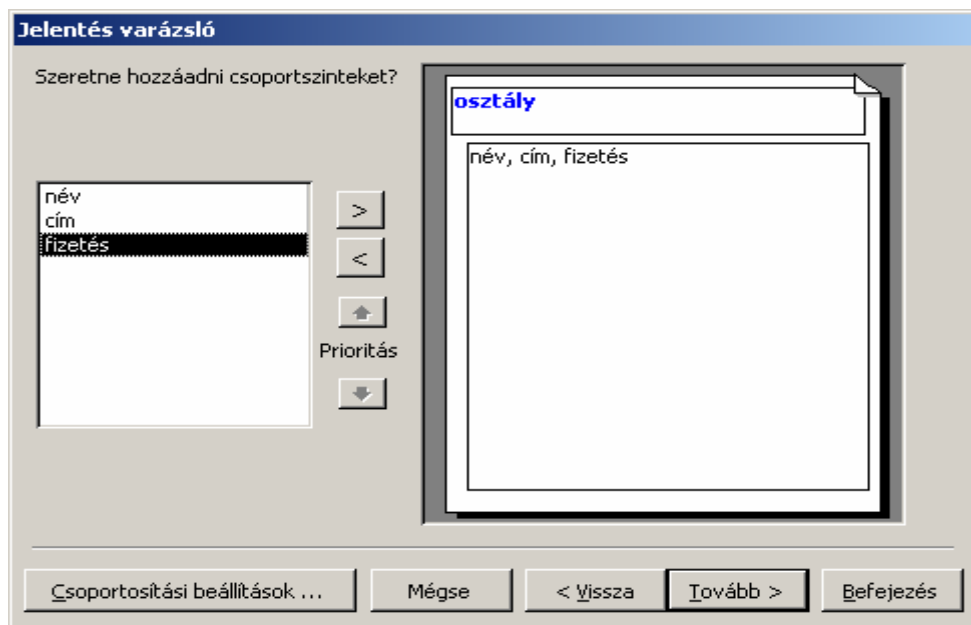
7.10. ábra: Új jelentés készítése ablak

1. lépés: Itt kell beállítani, hogy melyik tábla illetve lekérdezés, annak pedig mely mezői tartalmazzák a jelentésben szerepeltetni kívánt adatokat (7.11. ábra).



7.11. ábra: Jelentés varázslóval táblák, mezők kiválasztása

2. lépés: Ebben a lépésben jelölhetők ki a csoportszintek, vagyis mely mező (illetve mezők) alapján történik a rekordok csoportosítása. A csoportokra bontás több szinten is történhet, vagyis egy csoporton belül alcsoportok hozhatók létre, amennyiben több mező is ki lett jelölve (7.12. ábra).

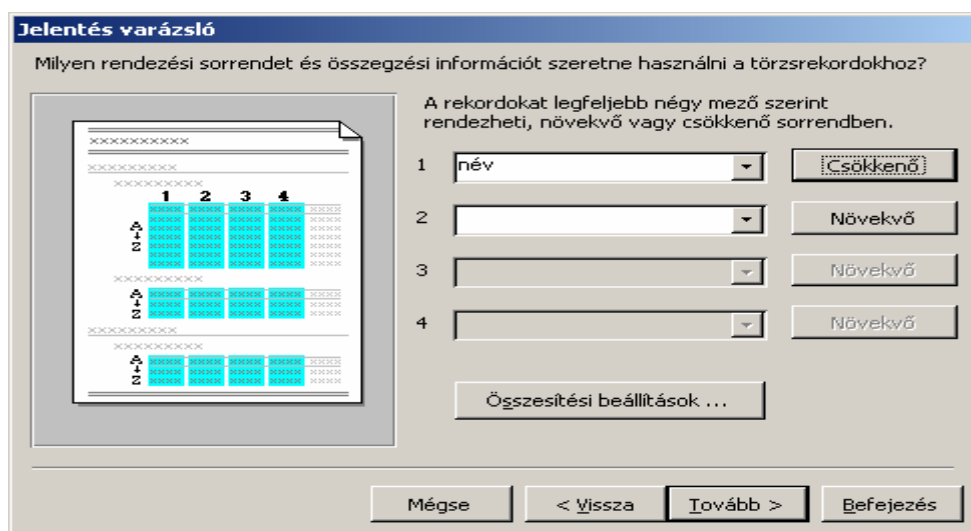


7.12. ábra: Jelentés varázsló csoportosítási szintjeinek megadása

A Csoportosítási beállítások gomb segítségével bonyolultabb csoportosítási szempontok is beállíthatók, mint pl. a nevek kezdőbetűjeként csoportosíthatók a dolgozók vagy 10000 Ft-onként a fizetésük alapján. A példában *osztályonkénti csoportosítás* látható.

3. lépés: Most azt kell beállítani, hogy egy csoporton belül mely mező szerint legyenek sorba rendezve a rekordok. A példában *név szerinti növekvő sorrendbe rendezés* történt (7.13. ábra).

Az Összegzési beállítások... gomb segítségével *csoportonkénti minimum, maximum, átlag, összeg* jeleníthető meg a jelentésben a kiválasztott mező értékei alapján (7.14. ábra).



7.13. ábra: Jelentés varázslóval rendezési sorrend megadása

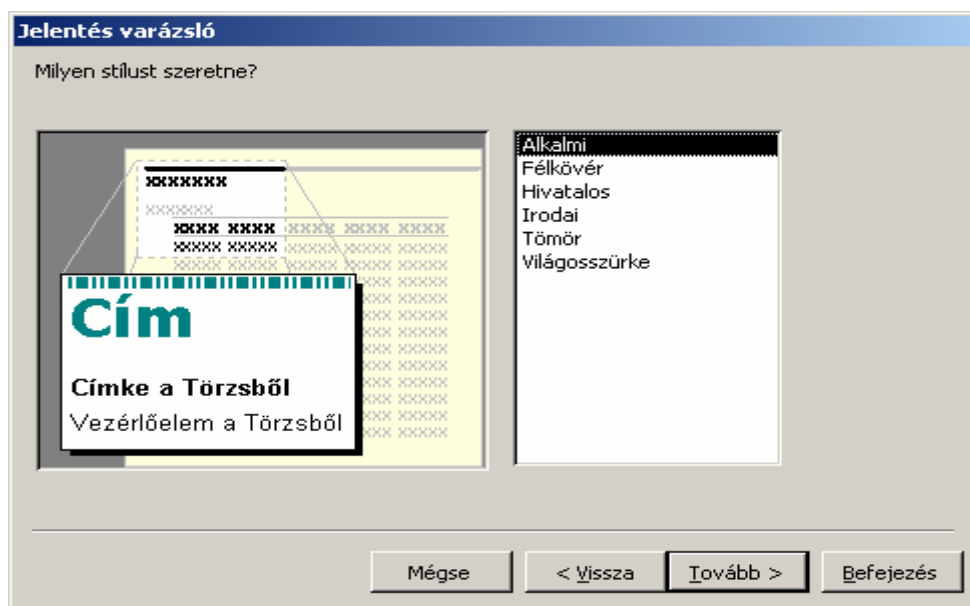
7.14. ábra: Összesítő beállítások kiválasztása

Amennyiben a Megjelenítés-nél a Csak Összesítés opció van kijelölve, a törzsrekordok mezői nem jelennek meg, csak a *csoportátlag* (illetve *összeg*, *minimum*, vagy *maximum* - amelyik éppen be van állítva). A példában a Törzs és Összesítés opció választása történt, melyben az egyes rekordok adatai is láthatóak és a csoportátlag is minden csoport végén.

4. lépés: Ebben a lépésben a megjeleníteni kívánt adatok elrendezése és a lap tájolása állítható be (7.15. ábra).

7.15. ábra: Adatok elrendezése és laptájolás megadása

5. lépés: Ebben a lépésben a jelentés stílusa választható ki a felkínált 6 fajta stílus közül (7.16. ábra).



7.16. ábra: Jelentés stílus kiválasztása

6. lépés: Ez az utolsó lépés, itt már csak a jelentés címét kell meghatározni, ez kerül a jelentés tetejére.

Jelentés szerkesztése

Az elkészített jelentés Tervező nézetben utólag átalakítható, ezzel azonban tipikusan csak kisebb átalakításokat végzünk, nagyobb átalakítások helyett új jelentést készítünk.

A jelentés részei Tervező nézetben (7.17. ábra)

Jelentésfej/jelentésláb: A jelentés legelején ill. legvégén jelenik meg csak (értelemszerűen) egyszer.

Oldalfej/Oldalláb: Minden oldal tetején ill. alján megjelenő rész.

Csoportfej/Csoportláb: Minden csoport elején ill. végén megjelenő rész. Ha több csoportszint szerepel, akkor minden csoportnak van saját csoportfej és csoportláb része (A példában csak egy csoportszint van: az osztály, így **Osztályfej** és **Osztályláb** rész található a jelentésben). A csoportfejben célszerű elhelyezni a csoport alapjául szolgáló mező értékeit megjelenítő beviteli mezőt, a csoportlábban pedig a csoportra vonatkozó összegző, átlag, minimum, maximumszámításokat tartalmazó beviteli mezőket.

Törzs: A törzs rész minden rekordnál megjelenik, így itt érdemes megjeleníteni a nem csoportszintű mezőket.

Jelentésfej									
Dolgozók									
Oldalfej									
osztály			név			cím			fizetés
osztály fejléc									
OSztály									
Törzs									
			név			cím			fizetés
osztály lábléc									
="Összegzés:" & "osztály" & " & " & [osztály] & " (" & Count(") & " & If(Count(")=1;"törzsrekord";"törz									
Avg									=Avg([fizeté
Oldalláb									
=Now()									
=[Page] & " / " & [Oldalak] & " oldal"									
Jelentésláb									

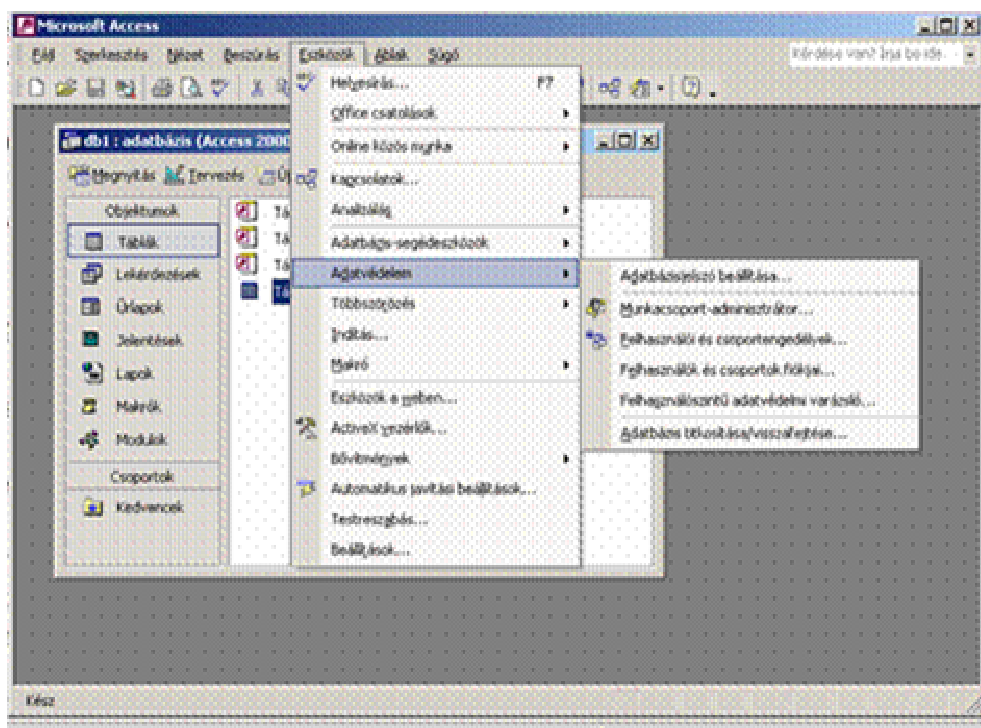
7.17. ábra: Jelentés részei tervező nézetben

7.5. Jogosultság, biztonság, adatvédelem³⁷

Az Access-ben az adatvédelemmel foglalkozó szolgáltatások az Eszközök menü Adatvédelem menüpontból érhetők el (7.18. ábra).

Adatvédelem jelszóval

Az adatvédelem egyik lehetősége, hogy jelszóval lehessen hozzáférni az adatbázishoz. A jelszó beállítását csak akkor lehet elvégezni, ha a Fájl menü Megnyitás parancsnál a kizárólagos jelölő négyzet be van jelölve.



7.18. ábra: Adatvédelmi lehetőségek

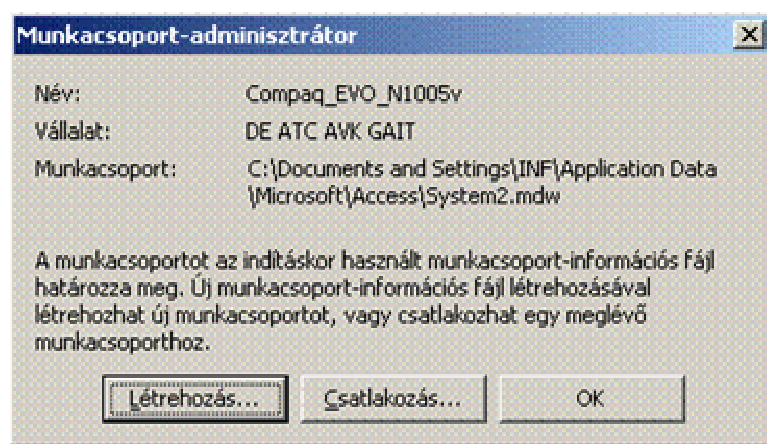
³⁷ Az adatvédelem és adatbiztonság kifejezéseket gyakran rosszulértelmezik. Az adatvédelem jogi, az adatbiztonság pedig számítástechnikai kifejezés.

Az Eszközök menü Adatvédelem parancs Adatbázis jelszó beállítása funkcióval megadható a jelszó. Ha későbbiekben a védett adatbázist kell megnyitni, akkor először a megjelenő párbeszédablakba be kell írni az érvényes jelszót. A jelszó módosítása úgy történik, hogy először törölni kell a régi jelszót, és utána újat kell megadni.

Adatok felhasználói szintű védelme

A felhasználónak már az Accessbe való belépésekor azonosítania kell magát, és csak a beállított jogosultságok alapján férhet hozzá az egyes objektumokhoz.

A közös adatokat használókat *munkacsoportnak* nevezi az Access program. Telepítéskor két csoport jön létre, a rendszergazdák (adminisztrátorok) és a felhasználók csoportja. Létre lehet hozni új felhasználókat, illetve meg lehet szüntetni felhasználókat (7.19. ábra).



7.19. ábra: Munkacsoport adminisztrátor ablak

Az Eszközök menü Adatvédelem parancs Munkacsoport adminisztrátor menü funkcióval lehet munkacsoportot létrehozni. A Létrehozás gomb megnyomása után meg kell adni a nevet, a szervezet nevét és a munkacsoport kódját.

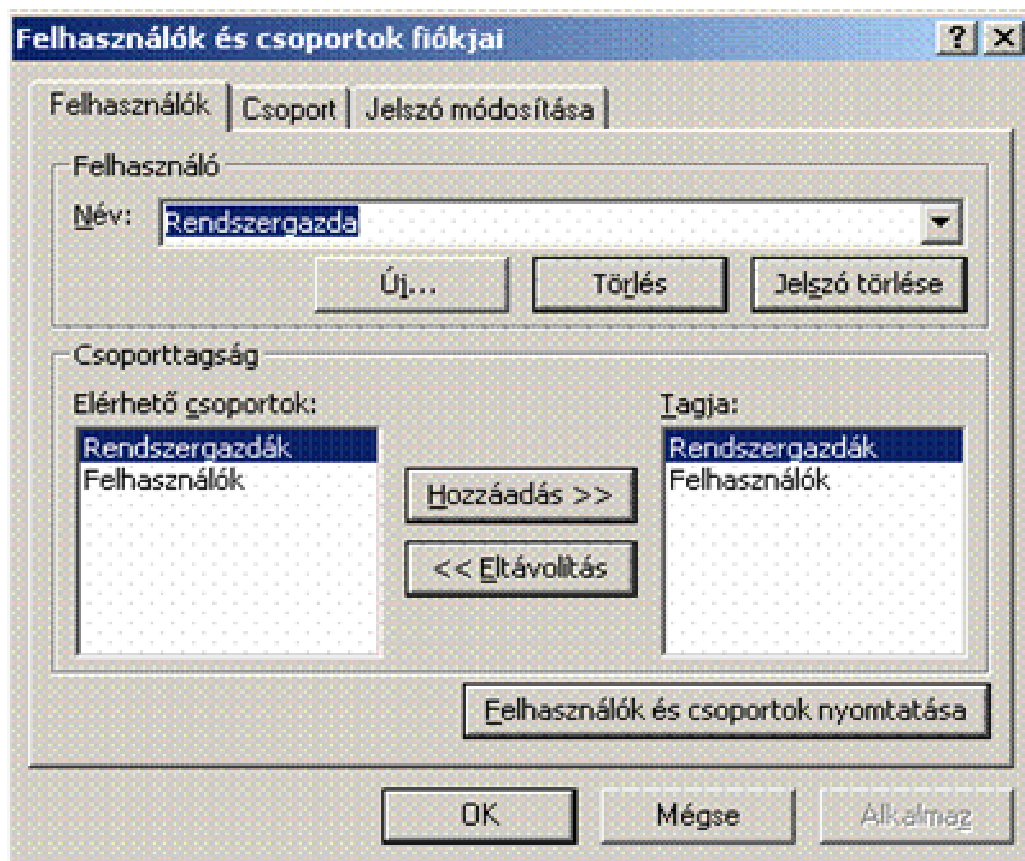
Ha olyan adatbázist kell használni, amelyik rendelkezik az előzőekben megadott védelemmel, akkor először csatlakozni kell a hozzá beállított munkacsoport fájlhoz. Ez az Eszközök menü Adatvédelem parancs Munkacsoport adminisztrátor funkció választása után a Csatlakozás gombot kell megnyomni (7.19. ábra).

Az adatvédelem használatához be kell kapcsolni a bejelentkezési eljárást. Ehhez a rendszergazda felhasználói jelszavát kell megváltoztatni.

A jelszó módosítása az Eszközök menü Adatvédelem parancs Felhasználók és csoportok fiókjai funkció választásával történik. A Jelszó módosítása fül kiválasztása után meg kell adni az új jelszót, az ellenőrzést és OK. Ezután az Access program elindítása után a megadott munkacsoport-hoz csak a bejelentkezési párbeszédablak adatainak helyes kitöltése után lehet kapcsolódni. Ezután meg lehet adni a felhasználókat és a csoportokat az Eszközök menü Adatvédelem parancs Felhasználói adatvédelmi varázslóval.

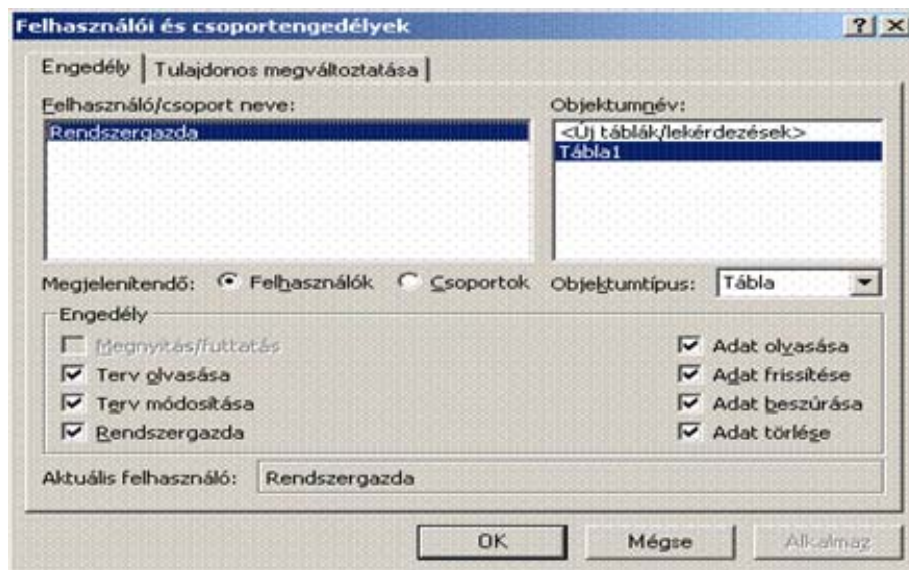
Az adatvédelem jó használatához meg kell adni az adatbázis felhasználóit és felhasználói csoportjait. Ez megtehető az Eszközök menü Adatvédelem parancs Felhasználók és csoportok fiókjai funkció választása után. A csoport

fület, majd az Új gombot kiválasztva, új csoportot lehet létrehozni. A Felhasználó fül, majd az Új gomb választása után, új felhasználót lehet definiálni. Ha már vannak felhasználói csoportok és felhasználók, akkor a felhasználókat csoportokba lehet sorolni. Ez a Hozzáadás illetve az Eltávolítás gombokkal történhet (7.20. ábra).



7.20. ábra: Felhasználók és csoportok fiókjai ablak

A felhasználóknak és a csoportoknak különböző engedélyeket lehet adni, amelyekkel hozzáférhetnek az adatbázis-objektumokhoz. Ezt az Eszközök menü Adatvédelem parancs Felhasználói csoport engedélyek funkció választása után lehet megtenni. Az engedélyeket lehet megadni külön a csoportoknak és a felhasználóknak. A Felhasználói és csoportengedélyek párbeszéd ablakban ki kell választani a Felhasználó\csoport nevét, az objektumot és engedélyeket (7.21. ábra). A Felhasználói és csoportengedélyek párbeszéd ablakban a Tulajdonos megváltoztatása párbeszéd ablakban van lehetőség az objektumok tulajdonosának a megváltoztatására.



7.21. ábra: Felhasználói és csoportengedélyek

7.6. Adatbázisok strukturált lekérdező nyelvének, az SQL-nek, ismertetése

A relációs adatbázis kezelő rendszerek egy SQL-nek nevezett szabványos nyelv segítségével kérdezik le és módosítják az adatbázist (*SQL – Structured Query Language – Strukturált lekérdező nyelv*). A legtöbb adatbázis kezelő kiterjeszti a szabványos SQL nyelvet olyan parancsokkal és utasításokkal, amelyek nem kompatibilisek az SQL szabvánnyal, így csak egy-egy adott RDBMS-ben működnek.

Az SQL fejlődésének főbb állomásai

- 1970: Codd definiálta a relációs adatmodellt és leírta később az SQL nyelv fő kritériumait
- 1974: SEQUEL nyelv leírása (Structured English QUery Language), az IBM cég fejlesztése
- 1979: ORACLE
- 1981: ORACLE 2 (SQL alapú RDBMS nagygépre), INGRES, IBM SQL/DS
- 1982: IBM DB2 (SQL alapú RDBMS nagygépre)
- 1986: ANSI SQL szabvány
- 1988: SQL SERVER DB2-höz
- SQL szolgáltatások (DBASE IV)
- SQL alapú adatbázis kezelők
- MS SQL SERVER
- 1992: SQL 2 szabvány
- SQL 3

SQL utasításainak két fő csoportja

- DDL (Data Definition Language): adatstruktúra definiáló utasítások
- DML (Data Manipulation Language): adatokon műveletet végző utasítások

SQL egységei

- adatbázis,

- tábla (table, reláció),
- sor (row),
- oszlop (column),
- elemi adat.

SQL nyelv szintaxisa

- *Kisbetű és nagybetű* a nyelv alapszávaiban egyenértékű.
- *Utasítások* sorfolytonosan írhatók, lezárás pontosvesszővel.
- *Változó* nincs, csak tábla- és oszlopnevekre lehet hivatkozni. Kifejezésben hivatkozás egy tábla adott oszlopára: tábla.oszlop (ha a tábla egyértelmű, akkor elhagyható).
- *Alias név*: név AS másodnév (egyes implementációkban AS elhagyható).
- *Szövegkonstans*: 'szöveg'
- *Dátum*: DATE '1968-05-12'. Egyes rendszerek az SQL szabványtól eltérő konvenciót alkalmaznak, például 13-NOV-94 (ORACLE), 02/15/1994 (DBASE).
- *Idő*: TIME '15:31:02.5' (óra, perc, másodperc).
- *Stringek konkatenációja*: + (XBASE), || (ORACLE).
- *Relációjelek*: =, <=, >=, !=, <>
- *Logikai műveletek*: AND, OR, NOT. Egy logikai kifejezés értéke ISMERETLEN (UNKNOWN), ha benne NULL érték szerepel.

Az utasítások szintaxisának leírásánál az elhagyható részleteket szögletes zárójelben jelölik.

Adattípusok (rendszerenként eltérők lehetnek)

- CHAR(n) n hosszúságú karaktersorozat,
- VARCHAR(n) legfeljebb n hosszúságú karaktersorozat,
- INTEGER egész szám (röviden INT),
- REAL valós (lebegőpontos) szám, más néven FLOAT,
- DECIMAL(n[,d]) n jegyű decimális szám, ebből d számú tizedesjegy,
- DATE dátum,
- TIME idő,
- LOCIGAL logikai (igen/nem).

Az adattípushoz "DEFAULT érték" megadásával alapértelmezett érték definiálható. Ha ilyet nem adunk meg, az alapértelmezett érték NULL.

Következőkben alkalmazott jelölések

- NAGYBETŰ kulcsszó
- *dőlt kisbetű* változó
- kisbetű érték
- aláhúzás alapértelmezés(default)
- { } opcionális elem
- | vagylagos elemek
- ... „és így Tovább” (folytatás)
- ∇ szóköz

Parancsok szintaxisa

PARANCS_KULCSSZÓ∇paraméterek;
Az SQL paraméterei általában kulcsszavak.

Paraméterek $\Rightarrow$ KULCSSZÓ $\nabla p_1, p_2, \dots, p_n$
 $p_1, p_2, \dots, p_n \Rightarrow$ a paraméter lista elemei (értékek)

7.6.1. Adatbáziskezelő SQL utasítások

Adatbázis létrehozása

CREATE DATABASE adatbázisnév;

Adatbázis megnyitása

{START} DATABASE adatbázisnév;

Információ egy adatbázisról

SHOW DATABASE;

Adatbázis bezárása

CLOSE DATABASE;

Adatbázis törlése

DROP DATABASE adatbázisnév;

7.6.2. Táblakezelő SQL utasítások

Tábla létrehozása

```
CREATE TABLE táblanév  
  ( oszlopnév adattípus [feltétel],  
    ... ...,  
    oszlopnév adattípus [feltétel]  
    [, táblaFeltételek]  
  );
```

1. Példa

```
CREATE TABLE hallgató (  
  név CHAR(30),  
  cím VARCHAR(255),  
  nem CHAR(1),  
  születésnap DATE  
);
```

A **név** és a **cím** típusa karakter. A különbség a kettő között, hogy a **név** fix hosszúságú karakter sorozat, amelynek végén a fel nem használt helyek üres karakterek. A **cím** változó hosszúságú karaktersorozat, amelyek maximális hossza 255 karakter. A **nem** attribútumai 'N' (nő) vagy 'F' (férfi), vagyis egyetlen karakter. A **születésnap** attribútuma DATE, vagyis dátumtípusú.

2. Példa

Az OSZTÁLY (osztálykód, osztálynév, vezAdószám) tábla létrehozása.

```
CREATE TABLE Osztály  
  ( osztálykód CHAR(3) PRIMARY KEY,
```

```
osztálynév CHAR(20),  
vezAdószám DECIMAL(10)  
);
```

A tábla módosításakor a definiált kulcsfeltételek automatikusan ellenőrzésre kerülnek. PRIMARY KEY és UNIQUE esetén ez azt jelenti, hogy a rendszer nem enged olyan módosítást, illetve új sor felvételét, amely egy már meglévő kulccsal ütközne.

Tábla törlése

```
DROP TABLE táblanév;
```

Hatására a séma és a hozzá tartozó adattábla törlődik.

3. Példa

```
DROP hallgató;
```

Tábla módosítása

```
ALTER TABLE táblanév  
[ADD (újelem, ..., újelem)]  
[MODIFY (módosítás, ..., módosítás)]  
[DROP (oszlop, ..., oszlop)];
```

újelem: egy "oszlopnév adattípus [feltétel]", vagy egy "táblafeltétel", mint a CREATE TABLE utasításban.

módosítás: "oszlopnév adattípus [feltétel]".

Oszlopok törlését nem minden rendszer engedi meg.

4. Példa

```
ALTER TABLE Dolgozó ADD (szüldátum DATE);  
ALTER TABLE Dolgozó MODIFY (lakcím VARCHAR(60));
```

Táblához új attribútum hozzáadása

```
ALTER TABLE táblanév ADD (oszlop1, {oszlop2,...});
```

5. Példa

```
ALTER TABLE hallgató ADD telefon CHAR(15);
```

Így a hallgató relációnak öt attribútuma lesz, az első négyet az 1. Példa mutatja, az ötödik a telefon attribútum, amely 15 hosszúságú karakter sorozat.

Táblában attribútum törlése

```
ALTER TABLE táblanév DROP (oszlop1, {oszlop2,...});
```

6. Példa

```
ALTER TABLE hallgató DROP cím;
```

Alapértelmezés szerinti érték megadása

Egy attribútum és az adattípus definiálásakor választható a DEFAULT kulcsszó és egy megfelelő érték. Az érték lehet vagy nullérték vagy egy konstans.

7. Példa

ALTER TABLE hallgató ADD telefon CHAR(15) DEFAULT 'titkos';

Indexek létrehozása

Az indexek kezelése nem része általában a szabványnak, de valamilyen formában minden RDBMS támogatja. Az index egy olyan adatstruktúra egy reláció *A* attribútumára, amely lehetővé teszi az olyan sorok gyors megkeresését, amelyeknek az *A* attribútuma bizonyos rögzített értéket vesz fel. Az indexek jól használhatóak olyan lekérdezésekben, amelyekben az *A* attribútum értéke egy konstanssal kerül összehasonlításra. Például $A = 5$ vagy $A \leq 5$. A következő utasítással hozható létre az index.

CREATE [UNIQUE] INDEX indexnév ON tábla(oszloplista);

Ha UNIQUE szerepel, akkor az indextábla nem tartalmazhat két azonos indexkulcsú rekordot.

8. Példa

CREATE INDEX DolgInd ON Dolgozó(név,osztálykód);

Index megszüntetése

DROP INDEX indexnév;

9. Példa

DROP INDEX ÉvIndex;

Táblába új sor felvétele

INSERT INTO táblanév [(oszloplista)] VALUES (értéklista);

Ha oszloplista nem szerepel, akkor valamennyi oszlop értéket kap a CREATE TABLE-ben megadott sorrendben. Egyébként, az oszlopnév-listában nem szereplő mezők NULL értéket kapnak.

10. Példa

INSERT INTO Dolgozó (név, adószám)
VALUES ("Tóth Aladár", 1111);

Táblában sor(ok) módosítása

UPDATE táblanév
SET oszlop = kifejezés, ..., oszlop = kifejezés
[WHERE feltétel];

Az értékadás minden olyan soron végrehajtódik, amely eleget tesz a WHERE feltételnek. Ha WHERE feltétel nem szerepel, akkor az értékadás az összes sorra megtörténik.

11. Példa

UPDATE Dolgozó
SET lakcím = "Debrecen, Rózsa u. 5."
WHERE név = "Kocsis István";

Táblában sor(ok) törlése

DELETE FROM táblanév

[WHERE feltétel];

A fenti utasítás hatására azok a sorok törlődnek, amelyek eleget tesznek a WHERE feltételnek. Ha a WHERE feltétel elmarad, akkor az összes sor törlődik (de a séma megmarad).

12. Példa

```
DELETE FROM Dolgozó
WHERE név = "Kocsis István";
DELETE FROM Osztály;
```

Az adatbázis lekérdezésére a SELECT utasítás szolgál, amely egy vagy több adattáblából egy *eredménytáblát* állít elő. Az eredménytábla a képernyőn listázásra kerül, vagy más módon használható fel.

A SELECT utasítás alapváltozata

```
SELECT [DISTINCT] oszloplista
FROM táblanévlista
[WHERE feltétel];
```

A felsorolt táblák Descartes-szorzatából szelektálásra kerülnek a feltételnek eleget tévő sorok, majd ezekből projekcióval kiválasztódnak az eredménytábla oszlopai. A DISTINCT opciót akkor kell kiírni, ha az azonos sorokból csak egyet kell megtartani.

Ha oszloplista helyén * karakter van, ez valamennyi oszlop felsorolásával egyenértékű. A SELECT legegyszerűbb változatával adattábla listázása érhető el.

```
SELECT * FROM T;
```

Projekció megvalósítása

```
SELECT [DISTINCT] A1,...,An FROM T;
```

13. Példa

```
SELECT DISTINCT szerző, cím FROM Könyv;
```

Szelekció megvalósítása

```
SELECT * FROM T WHERE feltétel;
```

14. Példa

```
SELECT * FROM Könyv WHERE kivétel<2000.01.01;
```

Természetes összekapcsolás

Legyen az a feladat, hogy elő kell állítani például az ÁRU(cikkszám,megnevezés) és VÁSÁRLÁS(cikkszám,mennyiség) táblák természetes összekapcsolását.

```
SELECT Áru.cikkszám, megnevezés, mennyiség
FROM Áru, Vásárlás
WHERE Áru.cikkszám = Vásárlás.cikkszám;
```

Alias nevek

A SELECT után megadott **oszloplista** valójában nem csak oszlopneveket, hanem tetszőleges kifejezéseket is tartalmazhat, és az eredménytábla oszlopainak elnevezésére alias nevek adhatók meg.

15. Példa

A RAKTÁR(cikkszám, név, egységár, mennyiség) táblából egy ÖSSZES(áru, érték) tábla létrehozása.

```
SELECT név AS áru, egységár*mennyiség AS érték FROM Raktár;
```

7.6.3. Összesítő függvények

Egy oszlop értékeiből egyetlen érték létrehozása (például átlag). Általános alakban.

függvénynév ([DISTINCT] oszlopnév)

Ha DISTINCT szerepel, akkor az oszlopban szereplő azonos értékeket csak egyszer kell figyelembe venni. A számításnál a NULL értékek figyelmen kívül maradnak. Az összesítő függvények a következők lehetnek.

- AVG: átlagérték,
- SUM: összeg,
- MAX: maximális érték,
- MIN: minimális érték,
- COUNT: elemek száma. Ennél a függvénynél az oszlopnév helyére * is írható, amely valamennyi oszlopot együtt jelenti.

16. Példa

Az eredménytábla egyetlen elemből áll, amely az átlagfizetést adja.

```
SELECT AVG(fizetés) FROM Dolgozó
```

A fizetések összege.

```
SELECT SUM(fizetés) FROM Dolgozó
```

A Dolgozó tábla sorainak száma, vagyis a dolgozók száma.

```
SELECT COUNT(*) FROM Dolgozó
```

Csoportosítás

(GROUP BY, HAVING)

Ha a tábla sorait csoportonként kell összesíteni, akkor a SELECT utasítás a GROUP BY oszloplista alparanccsal bővítendő. Egy csoportba azok a sorok tartoznak, melyeknél az oszloplista értéke azonos. Az eredménytáblában egy csoportból egy rekord lesz. Az összesítő függvények csoportonként hajtódnak végre.

17. Példa

A DOLGOZÓ táblából osztályonként az átlagfizetés kiszámolása.

SELECT osztkód, AVG(fizetés) FROM Dolgozó
GROUP BY osztkód;

A SELECT után összesítő függvényen kívül csak olyan oszlopnév feltüntetésének van értelme, amely a GROUP BY-ban is szerepel.

A GROUP BY által képezett csoportok közül válogatni lehet a HAVING feltétel alparancs segítségével. Csak a feltételnek eleget tevő csoportok kerülnek összesítésre az eredménytáblában.

18. Példa

Ki kell válogatni azokat a dolgozókat, akiknek az átlagfizetése > 80000.

SELECT osztkód, AVG(fizetés) FROM Dolgozó
GROUP BY osztkód
HAVING AVG(fizetés) > 80000;

Eredménytábla rendezése

ORDER BY oszlopnév [DESC], ..., oszlopnév [DESC]

Ez a SELECT utasítás végére helyezhető, és az eredménytáblának a megadott oszlopok szerinti rendezését írja elő. Alapértelmezés szerint a rendezés növekvő sorrendben történik. Ha fordított rendezésre van szükség, a DESC (descending) kulcsszó írandó a megfelelő oszlopnév után.

19. Példa

Dolgozók és fizetéseik listája az osztálykódok növekvő, ezen belül a fizetések csökkenő sorrendjében.

SELECT osztkód, név, fizetés FROM Dolgozó
ORDER BY osztkód, fizetés DESC;

A SELECT utasítás általános alakja

A SELECT utasítás az alábbi alparancsokból állhat.

SELECT [DISTINCT] oszloplista	projekció
FROM táblanévlista	Descartes-szorzat
[WHERE feltétel]	szelekció
[GROUP BY oszloplista]	csoportosítás
[HAVING feltétel]	csoport-feltétel
[ORDER BY oszloplista];	rendezés

Az egyes alparancsok megadási sorrendje az angol nyelv szabályait követi, végrehajtási sorrendjük viszont az alábbi.

1. FROM	Descartes-szorzat
2. WHERE	szelekció
3. GROUP BY	csoportosítás
4. HAVING	csoport-szelekció
5. SELECT	projekció

6. ORDER BY rendezés

Ha egy SELECT utasítás WHERE vagy HAVING feltételében olyan logikai kifejezés szerepel, amely SELECT utasítást tartalmaz, ezt *alkérdésnek* vagy *belső SELECT-nek* is nevezik.

20. Példa

Az alábbi utasítás azon dolgozók listáját adja, amelyek fizetése kisebb, mint az átlagfizetés.

```
SELECT név, fizetés FROM Dolgozó  
WHERE fizetés < ( SELECT AVG(fizetés) FROM dolgozó );
```

Ebben a példában az alkérdést elég csak egyszer kiértékelni, hiszen a Dolgozó tábla minden egyes sorára ugyanaz lesz az eredmény.

Egy adatbázisban általában kétféle adatra van szükségünk.

- alapadatok: tartalmuk aktualizáló műveletekkel módosítható;
- származtatott adatok: az alapadatokból generálhatók.

A virtuális tábla (*nézettábla*, *view*) *nem tárol adatokat*. Tulajdonképpen egy transzformációs formula, amelyet úgy lehet elképzelni, mintha ennek segítségével a tárolt táblák adatait lehetne látni egy speciális szűrőn keresztül.

Nézettáblák alkalmazási lehetőségei

- Származtatott adattáblák létrehozása, amelyek a törzsadatok módosításakor automatikusan módosulnak (pl. összegzőtáblák).
- Bizonyos adatok elrejtése egyes felhasználók előtt (adatbiztonság vagy egyszerűsítés céljából).

Nézettábla létrehozása

```
CREATE VIEW táblanév [(oszloplista)] AS alkérdés;
```

A SELECT utasítás eredménytáblája alkotja a nézettáblát. "Oszloplista" megadásával a nézettábla oszlopainak új név adható. A CREATE VIEW végrehajtásakor a rendszer csak letárolja a nézettábla definícióját, és majd csak a rá való hivatkozáskor generálja a szükséges adatokat.

A nézettáblák általában ugyanúgy használhatók, mint a tárolt adattáblák, vagyis ahol egy SQL parancsban táblanév adható meg, ott rendszerint a nézettábla neve is szerepelhet.

Ha a nézettábla tartalma módosul, akkor a módosítás a megfelelő tárolt táblákon hajtódik végre, és természetesen megjelenik a nézettáblában is. Alapelv, hogy egy SQL rendszer csak akkor engedi meg a nézettábla módosítását, ha azt a rendszer korrekten és egyértelműen végre tudja hajtani a tárolt táblákon. Nem lehet módosítani a nézettáblát, ha definíciója

- DISTINCT opciót,
- FROM után egynél több táblanevet,
- GROUP BY alparancsot tartalmaz.

Ha egy módosítható nézettáblába új rekord kerül felvételre, akkor az alaptáblának a nézettáblában nem szereplő oszlopaiba szükségképpen NULL lesz.

Ha a CREATE VIEW utasítás végén a WITH CHECK OPTION záradék szerepel, akkor a rendszer nem engedi meg a nézettábla olyan módosítását, amely nem tesz eleget a leválogatási feltételnek. Például a következő utasítás nem engedi meg az osztálykód módosítását, vagy 'A01'-től különböző osztálykód felvitelét.

```
CREATE VIEW Dolg2 AS
  SELECT adószám, név, lakcím, osztálykód FROM Dolgozó
  WHERE osztálykód='A01' WITH CHECK OPTION;
```

7.6.4. Hozzáférési jogok, jogosultságok kezelése az SQL-ben

Hozzáférési jogok adományozása

```
GRANT jogosultságok ON adatbáziselemek TO felhasználók
[WITH GRANT OPTION];
```

Jogosultságok

SELECT: lekérdezés engedélyezése.

ALTER: struktúramódosítás engedélyezése (ALTER TABLE).

INSERT[(oszlopok)], UPDATE[(oszlopok)], DELETE: tábla módosítás engedélyezése a megfelelő utasítással. Oszlopok megadása esetén az engedély csak az adott oszlopokra vonatkozik.

REFERENCES: külső kulcs hivatkozás engedélyezése az adatbáziselemre.

ALL PRIVILEGES: az összes adományozható jogosultság.

Adatbáziselem: amelyre a jogosultság adományozása történik.

Felhasználó: akinek a jogosultságot adományozzák.

WITH GRANT OPTION: továbbadományozási jog adása.

Jogosultság visszavonása

```
REVOKE jogosultságok ON adatbáziselemek FROM felhasználó
[CASCADE];
```

CASCADE: a visszavont jogosultság alapján továbbadományozott jogosultságok is visszavonásra kerülnek, feltéve, hogy ugyanazt a jogot az illető más forrásból nem szerezte meg.

Egy SQL utasítást csak akkor hajt végre a rendszer, ha a felhasználó a végrehajtáshoz szükséges valamennyi jogosultsággal rendelkezik.

21. Példa

```
GRANT SELECT ON Dolgozó TO Kovács, Tóth;
GRANT UPDATE(lakcím) ON Dolg1 TO Horváth WITH GRANT OPTION;
REVOKE SELECT ON Dolgozó FROM Tóth;
```

Ellenőrző kérdések:

1. Milyen objektum típusok érhetők el az Access Adatbázis ablakában?
2. Mit tartalmaz és milyen szerkezetű a tábla?
3. Mire használhatók a lekérdezések és az űrlapok?

4. Milyen eszközökkel védhetők az Access-ben az adatbázisok?
5. Milyen engedélyek adhatók a felhasználóknak és a csoportoknak az Access-ben?
6. Milyen adattípusok adhatók meg az Access-ben?
7. Milyen mezőtípusok vannak az Access-ben és mik a jellemzőik?
8. Milyen mező tulajdonságok adhatók meg a táblákban?
9. Milyen típusú lekérdezések készíthetők?
10. Mi a lényege a keresztáblás lekérdezésnek?
11. Hogyan kezdődik a lekérdezés létrehozása?
12. Milyen grafikus eszköz segíti a lekérdezés tervezői munkát?
13. Milyen szakaszok különíthetők el az űrlapon?
14. Mire alkalmazható az Access jelentéskészítője?
15. Hányféleképpen készíthetők az Access-ben jelentések?
16. A Jelentés Varázsló alkalmazásával hogyan lehet elkészíteni egy jelentést?
17. Milyen szerkezetűek az Access jelentések?
18. Milyen résznyelvek alkotják az SQL-t?
19. Milyen adattípusok vannak az SQL-ben?
20. Milyen SQL utasítással lehet táblát létrehozni, módosítani, törölni? Adjon meg példákat!
21. Milyen SQL utasítással lehet a táblát lekérdezni? Adjon meg egy példát lekérdezésre!
22. Milyen SQL utasítással lehet indexet létrehozni, törölni? Adjon meg példákat!
23. Milyen SQL utasítással lehet táblába új sort felvenni, sort módosítani, sort törölni? Adjon meg példákat!
24. Milyen SQL utasítással lehet adatbázist megnyitni, létrehozni, lezárni, törölni? Adjon meg példákat az említett műveletekre!
25. Milyen összesítő függvényeket lehet alkalmazni az SQL SELECT utasításban?
26. Hogyan végezhetők el csoportosítások az SQL SELECT utasításban?
27. Hogyan végezhető el az eredmény táblák rendezése az SQL-ben?
28. Hogyan lehet létrehozni nézettáblát SQL-ben?
29. Milyen SQL utasítással lehet jogosultságokat adományozni, illetve visszavonni?

Irodalomjegyzék

- Bakó Sándor (1999): Adatbáziskezelés, Pedellus Novitas Kft, Debrecen.
- Békéssy András - Demetrovics János (2005): Adatbázis-szerkezetek, Akadémiai Kiadó, Budapest.
- Hallasy Béla (2002): Adatmodellezés, Nemzeti Tankönyvkiadó Rt, Budapest.
- Hector Garcia-Molina – Jeffrey D. Ullman – Jennifer Widom (2001): Adatbázisrendszerek megvalósítása, Panem Könyvkiadó Kft, Budapest.
- Jeffrey D. Ullman – Jennifer Widom (1998): Adatbázisrendszerek, Alapvetés, Panem Könyvkiadó Kft, Budapest.
- Michael J. Hernadez (2004): Adatbázis-tervezés, A relációs adatbázisok alapjairól földi halandóknak, Kiskapu Kiadó, Budapest.
- Rajtik János (2004): Adatbázis-kezelés MS ACCESS 97, Pedellus Tankönyvkiadó Kft, Debrecen.

8. SZÁMÍTÓGÉP-HÁLÓZATOK

8.1 Céljai, elemei

A **számítógép-hálózatok** alatt az egymással kapcsolatban lévő önálló számítógépek rendszerét értjük. A hálózat mai célja, hogy a felhasználó ne egy számítógéppel, hanem a hálózat erőforrásaival kerüljön kapcsolatba. A számítógép-hálózatok kialakításának céljai a következők:

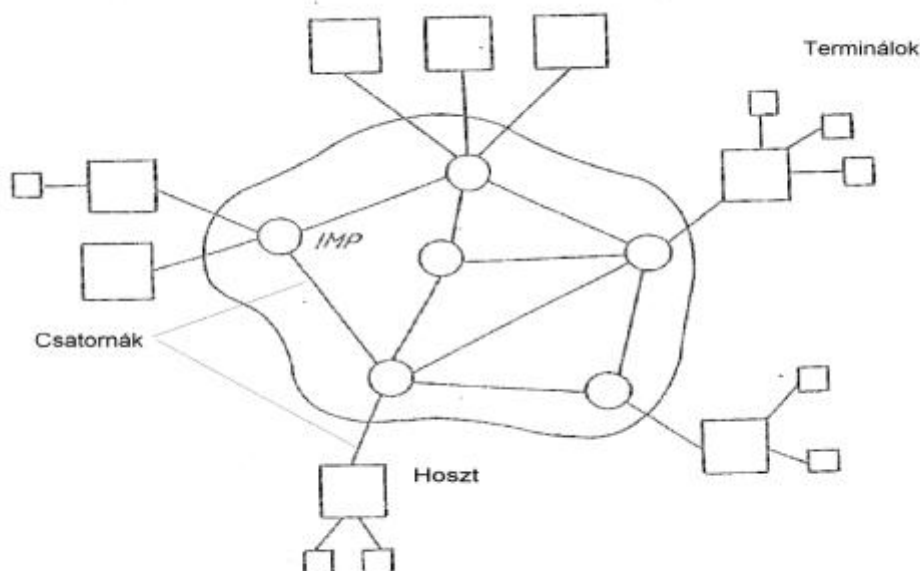
- **Erőforrás megosztás:** az eszközök (nyomtatók, nagy kapacitású tárolók), programok, adatok a felhasználók számára azok fizikai helyétől függetlenül a célfelhasználók számára elérhetők legyenek.
- **Nagyobb megbízhatóság:** alternatív erőforrások alkalmazása (pl. fájlok több gépen való tárolása, egyszerre több CPU alkalmazása).
- **Takarékosság:** A kis számítógépek sokkal jobb ár/teljesítmény aránnyal rendelkeznek, mint a nagyobbak (egy erőforrásgép kb. 10-szer gyorsabb, viszont ezerszer drágább, mint egy PC). **Kliens-szerver** modell: minden felhasználónak (kliens) saját PC-je van, az adatokat egy vagy több, közösen használt szerveren tárolják.
- **Skálázhatóság:** annak a biztosítása, hogy a rendszer teljesítményét a terhelés növekedésével fokozatosan növelni lehessen újabb szerverek, kliensgépek hozzáadásával (nem pedig az erőforrásgépet kell kicserélni).
- **Kommunikáció, hozzáférés távoli információkhoz.** A hálózati rendszer kommunikációs közegként is használható. Ez a terület a leggyorsabban fejlődő ág. Ide tartozik az elektronikus levelezés, a hálózati telefon és a videó átvitel. Elérhetővé válnak a központi adatbázisok. Ezek az adatbázisok sok helyről lekérdezhetők, és sok helyről tölthetők.
- **Központi programtárolás és kiszolgálás.** A programokat nem feltétlenül kell a saját gépünkön tárolni. Elképzelhető, hogy a programok csak a futtatás idejére töltődnek le a gépünkre, és a használatért eseti bérleti díjat fizetünk. Saját hálózatokon ennek a megoldásnak az az előnye, hogy mindig az aktuális verzió fog futni a felhasználó gépén anélkül, hogy ezzel külön kellene foglalkoznunk. A korszerű vállalati rendszerekben általában helyi programtárolás van, de futtatás előtt a rendszer ellenőrzi, hogy a példány megegyezik-e a szerveren tárolttal. Ha szükséges frissíti a programot és letölti a hiányzó komponenseket.

8.1.1 Hálózati struktúrák

A hálózati struktúrák és a fogalmak meghatározásában a mai napig érvényesek az ARPA (Advanced Research Project Agency) által kidolgozott elvek.

- o **Hosztoknak** (host) nevezzük azokat a gépeket, amelyekben a felhasználó program fut.
- o **Kommunikációs alhálózat** (communication subnet) köti össze a hosztokat. Az alhálózat az összeköttetést biztosító csatornákból és kapcsolóképekből áll. A csatornát (channels) szokás még vonalnak, áramkörnek, vagy több vonal esetén trónknek is nevezni. A hoszt nem része az alhálózatnak!
- o **A kapcsológép** (Interface Message Processor) az interfész üzenet feldolgozó gép. Feladata, hogy a bemenetére kerülő adatot meghatározott kimenetre

kapcsolja. Fizikailag ez lehet egy speciális gép (pl. router), de lehet egy számítógép része is (pl. hálózati kártya).



8.1. ábra: **Kommunikációs alhálózat**

Egy IMP-hez egy hoszt is tartozhat, de egy IMP-hez több hoszt is kapcsolódhat. Az egy IMP – több hoszt a nagytávolságú hálózatokra jellemző.

8.1.2 Hálózati hardver

A számítógép-hálózatok osztályozásának szempontjai közül két fontos szempont az átviteli technológia és a hálózat mérete (kiterjedése) szerinti osztályozás.

I. Az átviteli technológia szerinti két fő csoport a következő:

Adatszóró hálózatok. Egyetlen kommunikációs csatorna, ezen osztozik a hálózat összes gépe. Ha bármelyik gép elküld egy üzenetet, azt az összes többi gép megkapja. Ekkor a címzési eljárás ugynevezet csoportcímzés.

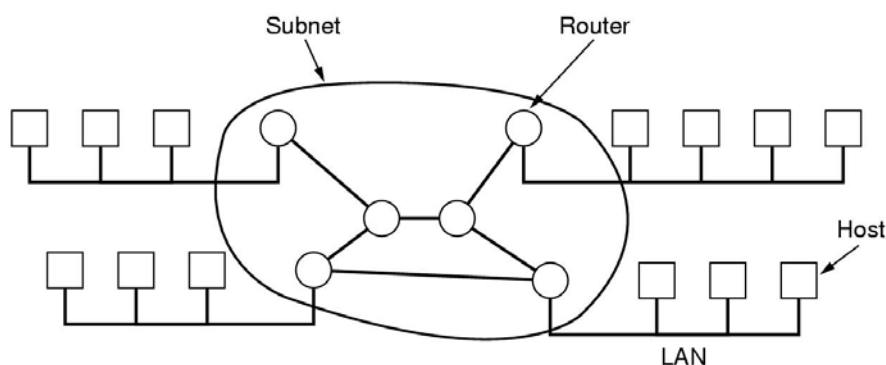
Kétpontos hálózatok. Ebben az esetben a számítógépek párosával kapcsolódnak egymáshoz, vagyis pont-pont kapcsolattal rendelkeznek. Ilyen hálózati kapcsolat esetén két számítógép közötti kapcsolatban az adatok, üzenetek küldése több útvonalon is lehetséges. Az ilyen hálózatok esetében fontos szerep jut a forgalomirányítási algoritmusoknak.

II. Méret szerint osztályozva:

Lokális hálózatok (Local Area Network, LAN). Az ilyen helyi hálózatok egy intézményen, vállalaton stb. belül működnek.

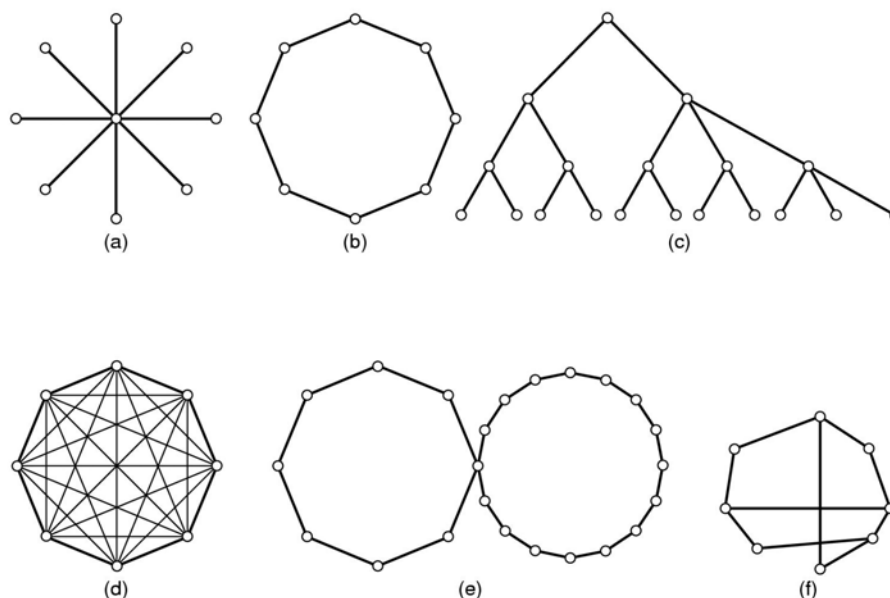
Nagyvárosi hálózatok (Metropolitan Area Network, MAN). Lényegében a lokális hálózatok nagyobb változata, és általában hasonló technológiára épül.

Nagy kiterjedésű hálózatok (Wide Area Network, WAN). Egész országra, illetve földrészre kiterjedő hálózatok. Részei a hosztok (host) és az őket összekapcsoló kommunikációs alhálózat (communication subnet) vagy röviden alhálózat. Az alhálózat feladata az üzenetek továbbítása a hosztok között. Az alhálózat részei az átviteli vonalak (más néven áramkörök, csatornák vagy trónkók) és a kapcsolóelemek. A kapcsolóelemek olyan speciális számítógépek, amelyeket két vagy több átviteli vonal összekapcsolására használnak (nincs egységes elnevezés, a továbbiakban mi routernek nevezzük).



8.2. ábra: A hosztok és az alhálózat közötti kapcsolat

A routerek tárolják, majd a megfelelő kimeneti csatorna szabaddá válása esetén továbbítják a csomagot. Az ilyen hálózatok szokásos elnevezései: **tárol-és-továbbít (store-and-forward)**, **két pont közötti (point-to-point)** vagy **csomagkapcsolt (packet-switched)**. Szinte az összes nagy kiterjedésű hálózat ilyen típusú.



(a) Csillag. (b) Gyűrű. (c) Fa. (d) Teljesen összekötött. (e) Egymást metsző gyűrűk (f) Szabálytalan.

8.3. ábra: Router kapcsolódási topológiák

A nagy kiterjedésű hálózatok másik nagy csoportja a műholdas vagy földi rádiós rendszerek. Ezek adatszóró rendszerek.

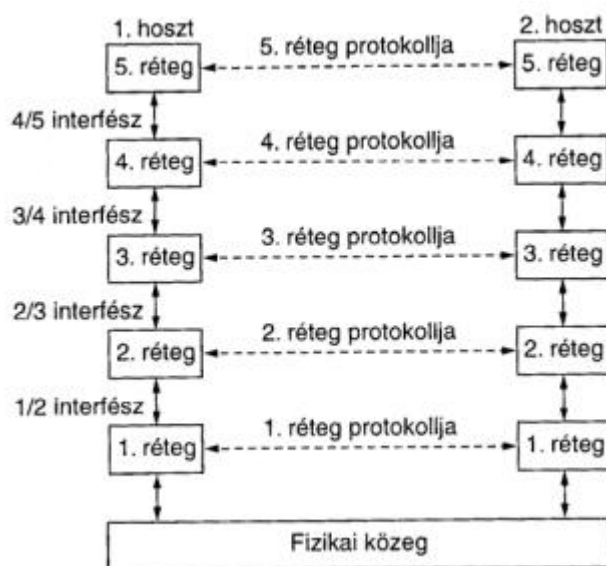
Összekapcsolt hálózatok. Egymástól különböző, sokszor nem kompatibilis hálózatok összekapcsolása, mely általában egy **átjárónak** nevezett (**gateway**) számítógép segítségével történik. Elnevezés: **internetwork**, **internet** (ilyen az Internet is).

8.1.3 Hálózati szoftver

A hálózati szoftver alatt azokat a szoftvereket értjük, melyek a hálózat működtetésével kapcsolatosak. A hálózati operációs rendszerek nem tartoznak a hálózati szoftver fogalomkörébe, bár a működéshez elengedhetetlenül szükségesek.

A számítógépek párbeszédének írott és íratlan szabályait együttesen protokollnak nevezzük. A protokoll egy megállapodás, amit az egymással kommunikáló felek párbeszédének szabályait rögzíti. A hétköznapi életben is vannak protokoll szabályok, amiknek betartása megkönnyíti, megsértése megnehezíti a kommunikációt. Az eltérő szabályok értelmezési nehézségeket okozhatnak. (Egy bolgár előre-hátra mozgatja a fejét, az abban a környezetben „nem”-et jelent, a nálunk megszokott „igen” helyett.)

Annak érdekében, hogy csökkentsék a hálózatok bonyolultságát, a legtöbb hálózatot strukturálják, **rétegekbe (layer)** vagy **szintekbe (level)** szervezik. Minden réteg az alatta levőre épül. Az egyes rétegek célja, hogy a felette levőknek szolgáltatásokat nyújtson oly módon, hogy közben a szolgálatok implementálásának részleteit azok elől elrejtse. Minden réteg csak a szomszédos réteggel van kapcsolatban. Az egyes rétegek párbeszédének szabályait a réteg-protokollok tartalmazzák. Példaként nézzünk egy 5 – rétegű hálózatot:



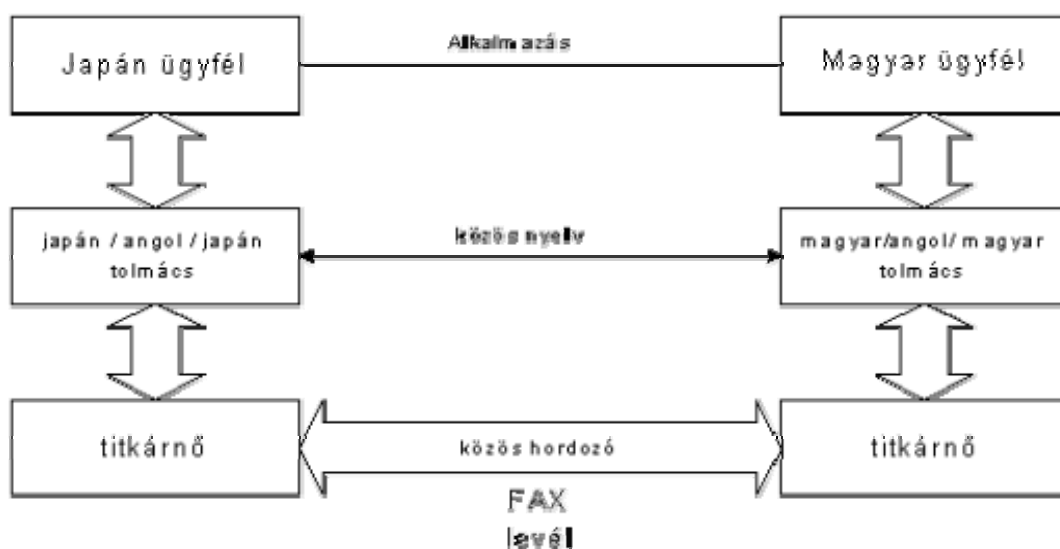
8.4. ábra: Réteg szemléletű kommunikáció

Minden réteg formálisan a vele azonos szintű réteggel társalog. Az adat valójában végighalad az alatta lévő összes rétegen, de ezt a rendszer elfedi. Az első réteg alatt van a fizikai közeg.

Az egyik gép n -edik rétege párbeszédet folytat a másik gép n -edik rétegével. A párbeszéd írott és íratlan szabályait az n -edik réteg **protokolljának (protocol)** nevezzük. Minden egyes réteg az alatta levő rétegnek vezérlőinformációkat és adatokat ad át egészen a legalsó rétegig, ami már a kapcsolatot megvalósító fizikai közeghez kapcsolódik. Az ábrán a virtuális kommunikációt szaggatott, a fizikai kommunikációt pedig folytonos vonalak jelölik.

Az egymással szomszédos (egymás alatt lévő) rétegek között interfész (interface) található. Az interfész definiálja a rétegek közötti elemi műveleteket, és azokat a szolgáltatásokat amit nyújt, illetve használ. A rendszerek tervezésének kritikus része az interfészek definíciója. A definíciónak világosnak és egyértelműnek kell lenni, hogy egy réteg a funkciók megváltozása nélkül tervezhető és kicserélhető legyen. Célszerű, ha az interfészt úgy választjuk meg, hogy az átadandó információ minimális legyen. A rétegek száma is kompromisszum. Ha kevés a rétegszám, bonyolult lesz a megvalósítás. Túl sok réteg pedig lassítja a rendszert, mert a sok interfész programja időt igényel.

A rétegszemlélet megértéséhez vizsgáljuk meg az emberi kommunikációt.



8.5. ábra: Az emberi kommunikáció modellje

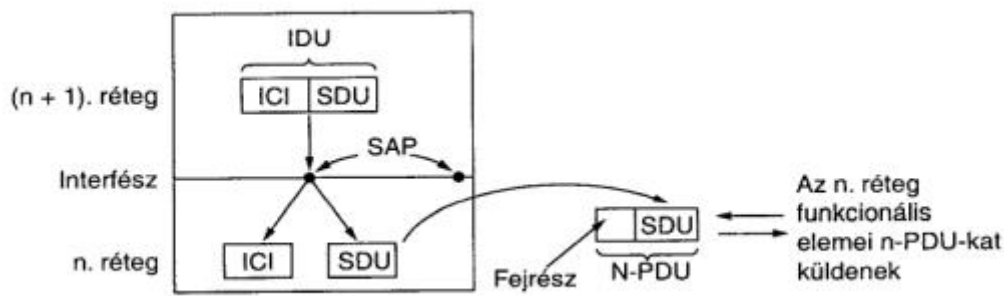
A rétegek és protokollok halmazát hálózati architektúrának nevezzük.

Mivel egy hálózatban tipikusan kettőnél több számítógép van, így szükség van egy olyan mechanizmusra, ami a küldőt és a fogadót azonosítja. Sok alkalmazásban szükség lehet arra, hogy a célállomások egy csoportját azonosítsuk. Minden rétegben kell lennie egy olyan mechanizmusnak, amely az üzenet küldőjét és vevőjét azonosítja. Meg kell továbbá határozni az adatok továbbításának a szabályait. Vannak olyan rendszerek, amelyek az adatokat csak egy irányban szállítják (**szimplex átvitel**), amelyek az adatokat időben váltakozva mindkét irányban szállítják (**fél-duplex átvitel**) és amelyek az adatokat egyszerre mindkét irányban szállítják (**duplex átvitel**).

A rétegek feladatának pontos leírásához definiált fogalmak kellenek. A pontos leírást általában a szabványok tartalmazzák, a „hétköznapi” használatban ezek bürokratikusnak hatnak, és nehézkesek.

Az n . réteg szolgáltatásokat nyújt az $n+1$. réteg számára. A szolgáltatások a szolgálat elérési pontokon keresztül hozzáférhetők. Minden szolgálat elérési pont valójában egy cím. A

telefon hálózaton pl.: egy szolgálat elérési pont a fali csatlakozó. A csatlakozót a telefonszám azonosítja. A rétegben lévő aktív elemeket entitásoknak hívják. Entitás pl. egy áramkör, vagy egy szoftverfolyamat. Az $n+1$. réteg kommunikációja úgy valósul meg, hogy átad a SAP-on keresztül egy IDU-t az n . rétegnek. Az n . réteg az alatta lévő rétegeken keresztül cserél SDU-kat a vele azonos szinten lévő réteggel. Az n . réteghez tartozó SDU-t továbbítása érdekében egy entitás feldarabolhatja kisebb egységekre. Ezeket az adategységeket hívjuk PDU-nak. Az n -PDU az n . réteghez tartozó Protocol Data Unit. PDU lehet például egy csomag.



ICI	Interface Control Information.	(interfész vezérlő információ)
IDU	Interface Data Unit	(interfész adatelem)
SAP	Service Access Points	(szolgálat elérési pontok)
SDU	Service Data Unit	(szolgálati adatelem)
PDU	Protocol Data Unit	(protokoll adatelem)

8.6 ábra: Rétegek kapcsolatai

A rétegek két különböző **szolgálatot** nyújthatnak a felettük levő rétegek számára:

- **Összeköttetés alapú szolgálat:** A szolgálatot igénybe vevő felhasználó először létrehozza az összeköttetést, majd felhasználja, végül lebontja (telefon).
- **Összeköttetés nélküli szolgálat:** Minden egyes üzenet rendelkezik egy teljes címmel, és minden üzenet az összes többitől független útvonalon továbbítódik (levél).

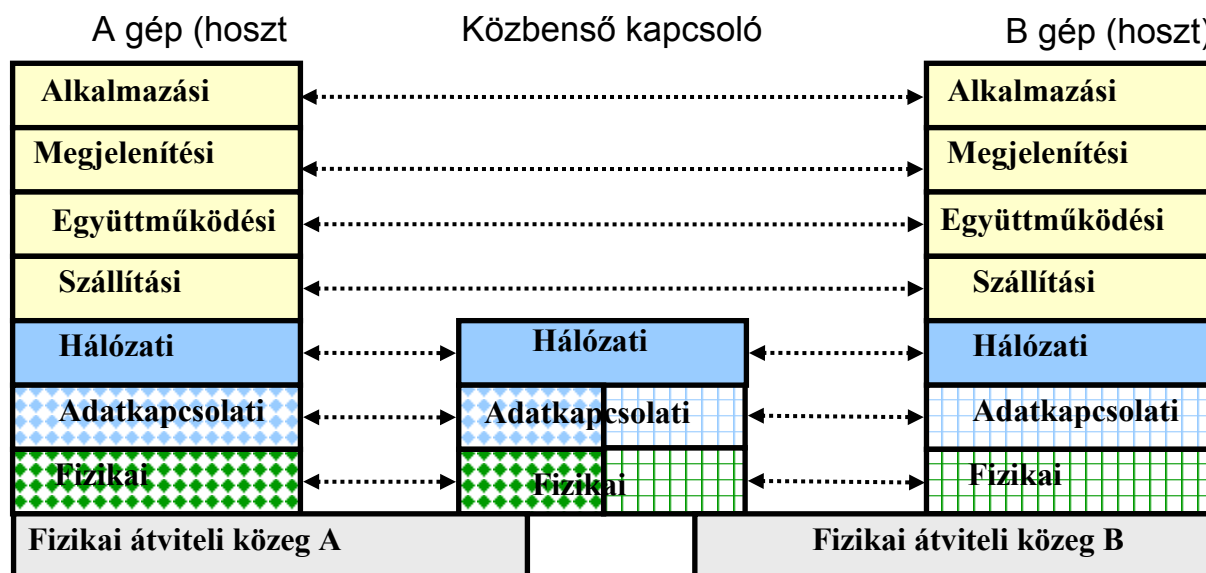
A szolgálatokat olyan szolgálatprimitívek, azaz elemi műveletek halmazával írhatjuk le, amelyek a szolgálatokat elérhetővé teszik a felhasználók számára. A protokollok olyan szabályhalmazok, melyek megadják, hogy a társ entitások az adott rétegen belül milyen formátumban, milyen jelentéssel küldenek egymásnak csomagokat, kereteket, üzeneteket. A szolgálat definiálja az objektumon végrehajtható műveleteket, de nem mondja meg hogyan kell ezeket implementálni. A protokoll a szolgálat implementációja. A protokollok cserélhetők, ha a szolgálat változatlan marad.

8.2 Hivatkozási modellek

A gyakorlatban előforduló hálózati architektúrák valamennyien megfelelnek a rétegekből álló modellnek. Különbségek a rétegek számában, és a megvalósított funkciókban vannak.

8.2.1 Az OSI hivatkozásai modell

Az Open System Interconnection az ISO (International Standards Organization) ajánlása. A nyílt rendszerek összekapcsolásával foglalkozik. A nyílt rendszerek olyan rendszerek, amelyek képesek más rendszerekkel való kommunikációra. Az OSI modellnek hét rétege van.



8.7. ábra: Az OSI rétegmodell

Fizikai réteg: Feladata a bitek továbbítása a kommunikációs csatornán olyan módon, hogy az adó oldali bitet a vevő is helyesen értelmezze (a 0-at 0-nak, az 1-et, 1-nek). Megoldandó feladatok, kérdések: a fizikai közeg, és az információ tényleges megjelenési formája, egy bit átvitelének időtartama, egy vagy kétirányú a kapcsolat, hogyan épüljön fel egy kapcsolat és hogyan szűnjön meg, milyen legyen az alkalmazott csatlakozó fizikai, mechanikai kialakítása?

Adatkapcsolati réteg: Feladata adatok megbízható továbbítása az adó és fogadó között. Az adatokat **adatkeretekké (data frame)** tördeli, ellátja kiegészítő cím, egyéb és ellenőrző információval, ezeket továbbítja, majd a vevő által visszaküldött nyugtakereteket (acknowledgement frame) véve ezeket feldolgozza. Felmerülő problémák: hogyan jelezzük a keretek kezdetét és a végét, mi történjék akkor ha egy keret elvész, mi történjék akkor ha a nyugtakeret vész el, mi legyen akkor, ha az adó adási sebessége jelentősen nagyobb, mint a vevőké?

Hálózati réteg: A hálózati réteg a kommunikációs alhálózatok működését vezérli. A két végpont közti kapcsolat lebonyolítása és a torlódás elkerülése a feladata. Eltérő lehet a hálózatok címezési módszere, különbözhetnek a maximális csomagméreteik és protokolljaik is. E problémák megoldásáért, azaz a heterogén hálózatok összekapcsolásáért a hálózati réteg a felelős. Üzenetszórásos hálózatokban az útvonal-kiválasztási mechanizmus igen egyszerű, így a hálózati réteg általában vékony, sokszor nem is létezik

Szállítási réteg: Feladata a hosztok közötti átvitel megvalósítása (itt már végpontok közötti összeköttetésről van szó, ld. ábra). A szállítási réteg alapvető feladata az, hogy adatokat fogadjon a viszonyrétegtől, kisebb darabra vágja szét azokat (ha szükséges), majd adja tovább

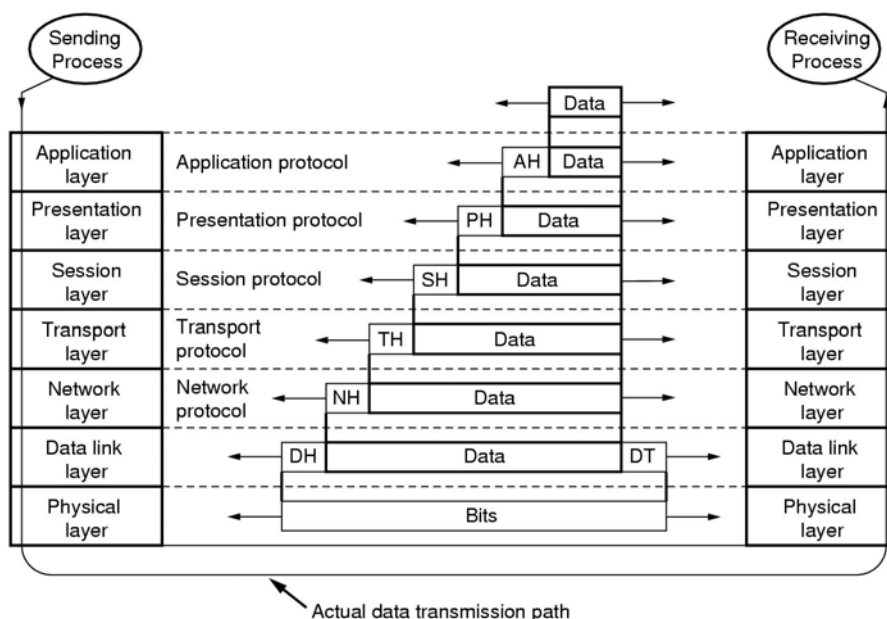
a hálózati rétegnek és biztosítsa, hogy minden darab hibátlanul megérkezzen a másik oldalra. Továbbá, mindezeket hatékonyan kell végrehajtania, ráadásul oly módon, hogy a viszonyréteg elől el kell fednie a hardvertechnikában elkerülhetetlenül bekövetkező változásokat.

Viszony réteg (más néven **együtműködési réteg**): A különböző gépek felhasználói viszonyt létesítenek egymással, mint például bejelentkezés egy távoli operációs rendszerbe, állománytovábbítás két gép között. Felelős a kapcsolat felépítéséért, és a két végpont kommunikációjának végén a kapcsolat lebontásáért. A viszonyréteg egyik szolgáltatása a párbeszéd szervezése. A viszonyok egyidőben egy- és kétirányú adatáramlást is lehetővé tehetnek. A viszonyréteg egy másik szolgáltatása a szinkronizáció.

Megjelenítési réteg: Tipikus feladatai: az adatok szabványos módon történő kódolása, tömörítés, titkosítás. A réteg a továbbított információ szintaktikájával és szemantikájával foglalkozik. A továbbított információ általában nem egy véletlenszerű bit halmaz, hanem szám betű, dátum, stb. A különböző rendszerekben eltérő módon ábrázoljuk az adatokat. A különböző rendszerek kommunikációja érdekében a lokális szintaktikát átalakítjuk egy globális (absztrakt) szintaktikává, amit minden rendszer a saját lokális szintaktikájává alakítva tud felhasználni.

Alkalmazási réteg: Az alkalmazási réteg széles körben igényelt protokollokat tartalmaz. A rétegben a felhasználói program vezérli a működést. A programok között számos olyan is van, melyekre széles körben van igény, általános megoldásuk célszerű, vagy csak így működőképes a dolog. Gyakori feladat a fájl átvitel (file transfer). Az állománytovábbításhoz sorolható az elektronikus levelezés, távoli adatbázisok elérése. Itt megoldandó az eltérő fájlrendszerek kezelése, szövegsorok kezelése, stb. (Egy UNIX szerverről le tudunk másolni egy fájlt egy DOS-t használó gépre. Ez a szint a felhasználói programok szintje (e-mail, fájl átvitel, távoli bejelentkezés, stb.).

A tényleges átvitel függőleges irányban történik, de az egyes rétegek úgy működnek, mintha vízszintes irányban továbbítanák az adatokat.



8.8. ábra Adatátvitel a rétegmodell szerint

8.2.2 A TCP/IP hivatkozási modell

Az Internet hivatkozási modellje. Két legjelentősebb protokolljáról a TCP (Transmission Control Protocol) és az IP (Internet Protocol) kapta a nevét. Kifejlesztésekor a fő célkitűzés az volt, hogy mindaddig, amíg a két végpont működőképes, addig a kapcsolat megszakítás nélkül működjön akkor is, ha a köztük lévő hálózat egy része megsemmisül (háborús csapásmérés, terrorakció, sőt, akár műszaki hiba). A hálózat legyen alkalmas a fájl átviteltől a beszéd átvitelig minden feladatra.

A megoldás egy 4-rétegű modell, amely jelentősen különbözik az OSI ajánlástól (amely több mint 10 évvel korábbi). A TCP/IP és az OSI rétegek megfeleltetése vázlatosan az alábbi:

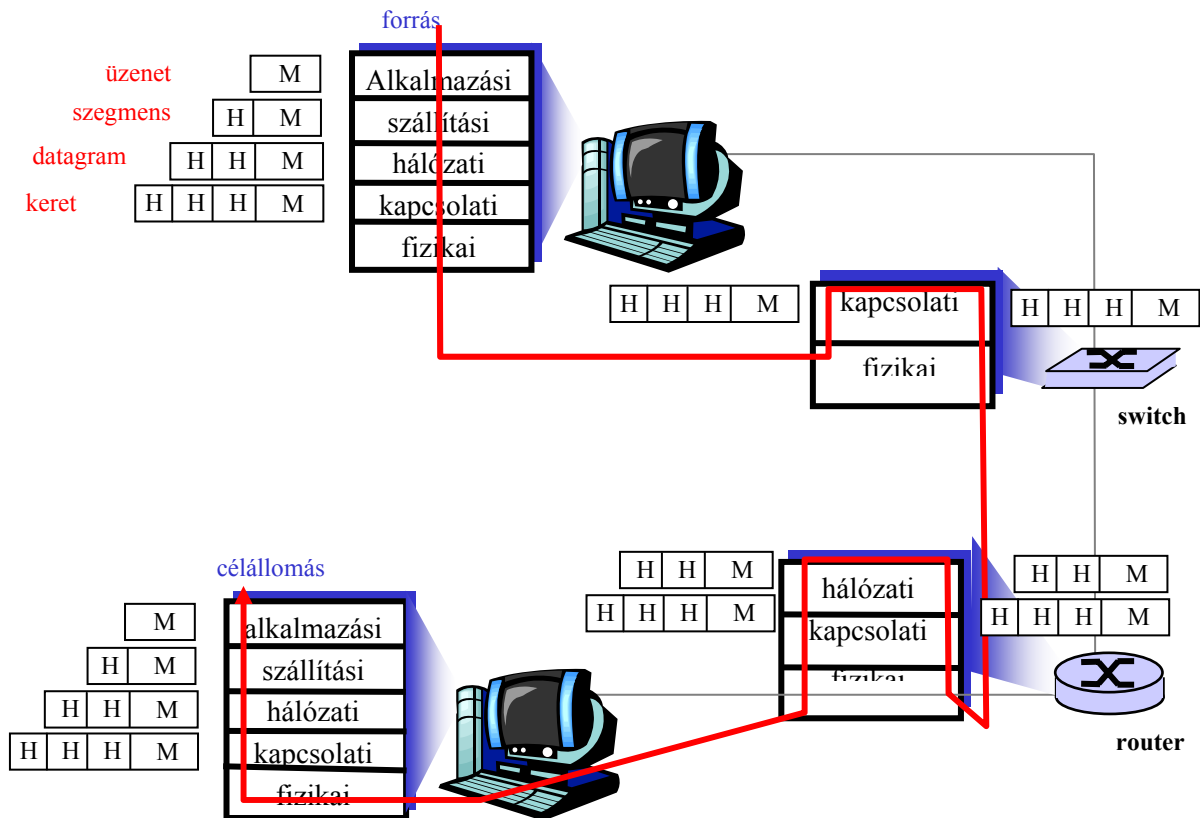
Alkalmazási réteg	Alkalmazási réteg
Megjelenítési réteg	
Viszony réteg	
Szállítási réteg	Szállítási réteg
Hálózati réteg	Internet réteg
Adatkapcsolati réteg	Hoszt és hálózat közötti réteg
Fizikai réteg	

8.9 ábra: TCP/IP hivatkozási modell

A megfeleltetés nem pontos. A rétegek feladatai azonban erősen hasonlóak.

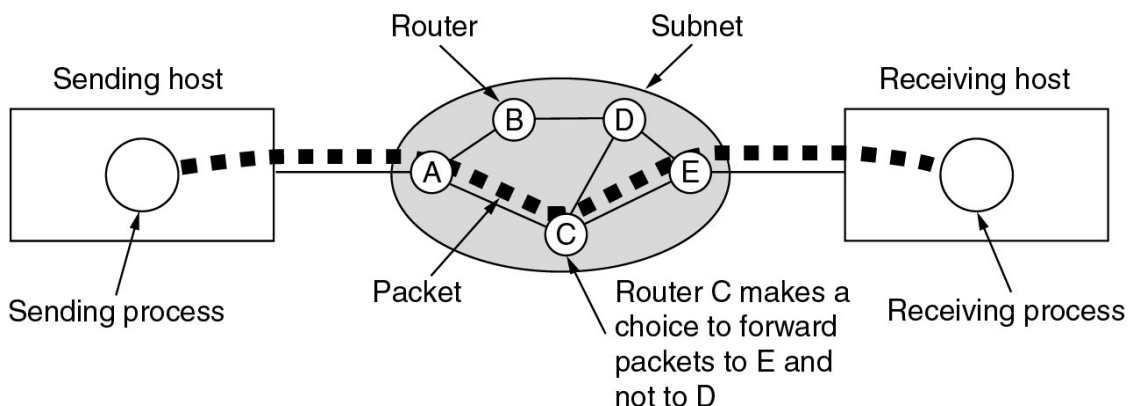
- o A hálózat elérési réteg nagyjából az adatkapcsolati és fizikai réteg feladatát látja el. Valójában kevésbé definiált, és csak azt írja elő, hogy a hálózat alkalmas legyen IP csomagok továbbítására.
- o Az internet réteg feladata, hogy a hoszt bármilyen hálózatba csomagokat tudjon küldeni, és onnan csomagokat fogadni. A csomagok nem biztos, hogy az elküldés sorrendjében érkeznek meg. A sorrend helyreállítása a felsőbb rétegek feladata. Az Internet réteg meghatároz egy protokollt és egy csomagformátumot. Ez az Internet Protocol (IP).
- o A szállítási réteg az Internet réteg felett helyezkedik el. Lehetővé teszi a társentítések közti párbeszédet. A szállítási rétegben létrehozhatunk megbízható, összeköttetés alapú protokollokat, ilyen az átvitel vezérlő protokoll (Transmission Control Protocol, TCP), vagy datagram jellegű protokollt, mint az UDP (User Datagram Protocol). A TCP a beérkező adatfolyamot feldarabolja, átadja az Internet rétegnek. A célállomáson a TCP összegyűjti a csomagokat, és adatfolyamként továbbítja az alkalmazási rétegnek. A TCP végzi a forgalomszabályozást is. Az UDP egy összeköttetés mentes, nem megbízható protokoll. Nem tartalmaz csomag sorba-rendezést sem. Általában az olyan kliens-szerver alkalmazásokban használatos, ahol a gyors válasz fontosabb a garantált válasznál (címfeloldás, beszéd, videó). Pl.: egy címfeloldási kérésre nem kapunk választ, akkor megismételjük a kérést. Lényegtelen, hogy miért nincs válasz.
- o Alkalmazási réteg. TCP/IP modellben nincs viszony és megjelenési réteg. A szállítási réteg felett van az alkalmazási réteg. Ez tartalmazza az összes magasabb szintű protokollt. Néhány megvalósított protokoll:

- o - Elektronikus levelezés (SMTP)
- o - Domain Name Service (DNS)
- o - HTTP (World Wide Web)
- o - FTP (fájl transzfer)



8.10. ábra: Adatátvitel a TCP/IP modellben

Összehasonlítás: Az OSI modell kifejezetten alkalmas a számítógépes hálózatok elemzésére. Ezzel szemben az OSI protokollok nem lettek népszerűek. A TCP/IP-re viszont ennek pont az ellentéte igaz: a modell gyakorlatilag nem létezik, a protokollok viszont rendkívül elterjedtek.

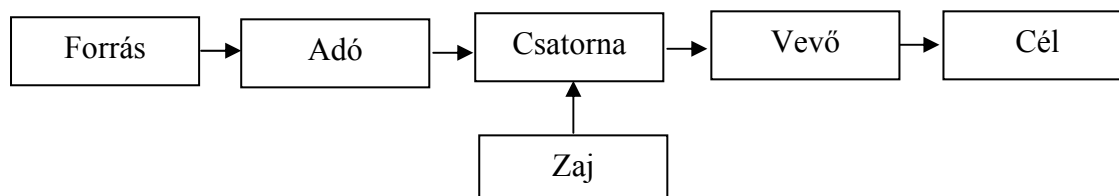


8.11. ábra: Csomagtovábbítás a hálózaton

8.3 A hálózat fizikai megvalósítása

8.3.1 A fizikai réteg

A csatornán történő információátvitel során az adó megváltoztatja a csatorna fizikai közegének valamilyen tulajdonságát, ami a közegen továbbterjed, és a vevő ezt a fizikai közegváltozást érzékeli. Megváltoztathatjuk az áramkörben a feszültséget, az áramot, a frekvenciát, a fázisszöveget. Jelhordozó lehet a fény intenzitása, vagy akár a felszálló füst megszakítása is (indián füstjelek). Az adatátvitel távolságát a jelek gyengülése, és a csatorna zaja befolyásolja.



8.12. ábra: Adatátviteli modell

Kommunikáció általános modellje.

A csatorna legfontosabb jellemzői:

- sávszélesség
- zaj
- kódolási eljárás.

Sávszélesség alatt általában az átvitt legmagasabb és legalacsonyabb frekvencia különbségét értik, ahol a frekvencia átviteli függvény 3 dB-el csökken. A valós rendszerekben a sávszélességet műszaki eszközökkel korlátozzák, az alsó és felső határfrekvenciánál meredek levágás van. Jó közelítéssel azt mondhatjuk, hogy a sávon belül van jel, a sávon kívül nincs jel. A **Sávszélesség** analóg rendszerben: a jel frekvencia tartománya (pl. beszéd: 300Hz-3300Hz), digitális rendszer: maximális információ átviteli sebesség (bit/s). A fizikai réteg további fontosabb jellemzői a következők:

Vonal: fizikai összeköttetés.

Csatorna: Két fél közötti kommunikációs kapcsolat.

Vonalmegosztás: egy vonalon több csatorna (pl. kábel TV)

Megvalósítási lehetőségei:

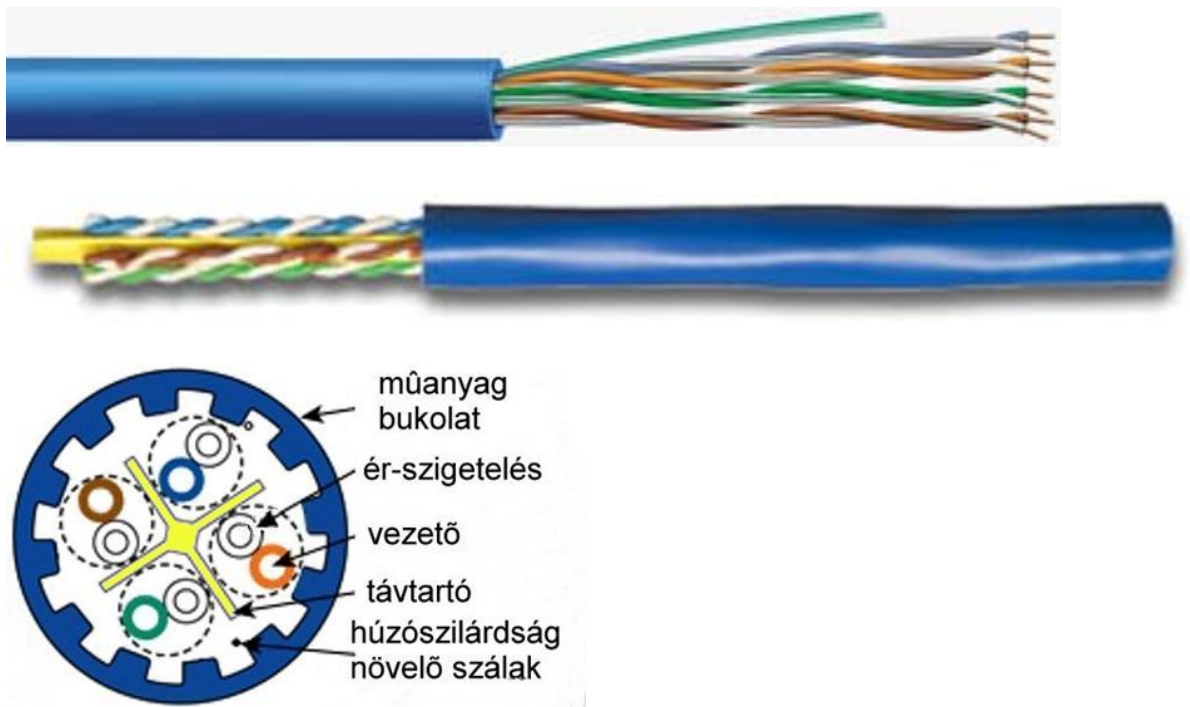
- **Multiplexelés** (frekvenciaosztás, időosztás)
- **Csomagkapcsolás:** az információ kisebb adagokra bontása, egy vonalon különböző gépek csomagjai haladhatnak, tárol-továbbít elv, csomagokban cím információ van elhelyezve.
- **Vonalkapcsolás:** az adatvezeték a kommunikálni szándékozó adó, illetve vevő kapja meg. Útvonal kialakítása kapcsolóközpontokon keresztül. Tényleges fizikai kapcsolat, viszont a kapcsolat létrehozásához idő kell.

A fontosabb fizikai átviteli közegek a következők:

Csavart érpár

Két spirálszerűen egymás köré tekert szigetelt rézhuzal. A két eret azért sodorják össze, hogy csökkentsék a kettő közötti elektromágneses kölcsönhatást. Elnevezések: UTP, STP – ((Un)shielded Twisted Pair). Akár 100 Mbit/s –os átviteli sebességet is el lehet érni.

Az Ethernet hálózatokban 10BaseT néven specifikálták. Két sodort érpár az adás és a vétel számára. 100m-es maximális szegmenshossz.



8.13. ábra: Csavart érpár

Az érpárok egymásra hatása csökkenthető, ha az érpárok távolságát növeljük. A kábel erek között távtartó lehet, mint az alábbi ábrán látható. Az ábrán bemutatott kábel CAT-6e kategóriájú, UTP, 750MHz-ig használható.

Koaxiális kábel

Jobb árnyékolás, mint a csavart érpárnál, ezért nagyobb átviteli sebesség és nagyobb szegmenshossz.



8.14. ábra: Koaxiális kábel

I. Alapsávú koaxiális kábel – digitális átvitelre, 50 Ω -os. Akár 1-2 Gb/s-os átviteli sebességet is elérhetünk. Leggyakrabban lokális hálózatok kialakítására alkalmazzák.

Ethernet hálózatokban az alapsávú koaxiális kábelek két típusa ismert az ún. vékony (10Base2) és a vastag (10Base5). A típusjelzésben szereplő 2-es és 5-ös szám az Ethernet hálózatban kialakítható maximális szegmenshosszra utal: vékony kábelnél ez 200 méter, vastagnál 500 méter lehet. A vékony koaxnál BNC csatlakozókat, míg a vastag változatnál ún. vámpír csatlakozókat alkalmaznak.

II. Szélessávú koaxiális kábel – szabványos kábel TV-s analóg átvitel, 75 Ω -os. Egy kábelben több csatorna, egymástól független, többféle kommunikáció. AD – DA átalakítások. Kevésbé alkalmas digitális átvitelre, mint az alapsávi (tehát egycsatornás) kábel, viszont nagy előnye, hogy már igen nagy mennyiségben telepítettek ilyeneket.

Üvegszálak kábel

Laboratóriumi körülmények között már a 100 Gb/s-os sebességet is elérték. A fényvezető szálak adatátviteli rendszernek három fő komponense van: a fényforrás (LED vagy lézerdíóda), az átviteli közeg és a fényérzékelő (fotodíóda vagy fototranzisztor).

Fény terjedés:

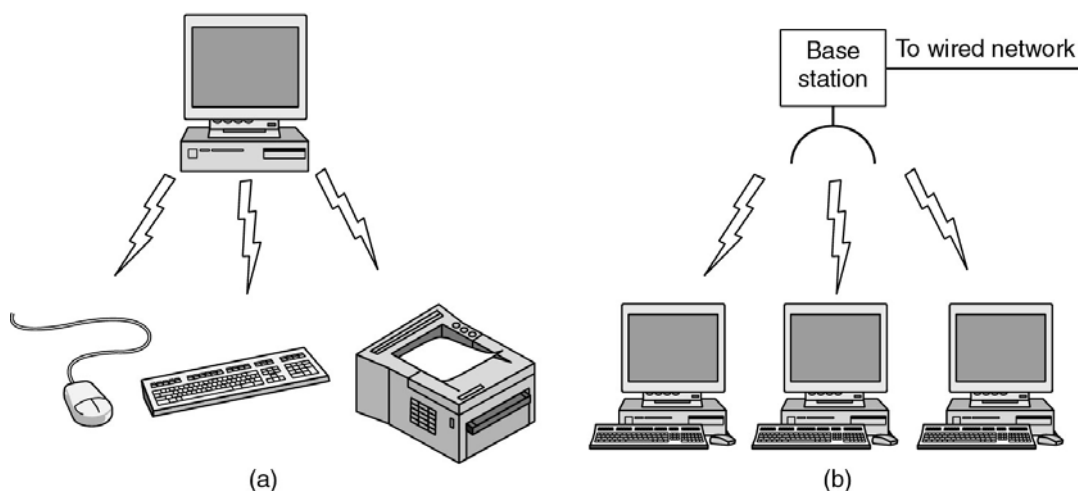
- teljes visszaverődés (többmódusú szál, több különböző szögű fénysugár),
- a szál átmérőjét néhány hullámhossznyira lecsökkentjük (8-10 μm , egymódusú szál, drágább, nagyobb távolságra használható).

A fényvezető kábelben általában több szálat fognak össze.

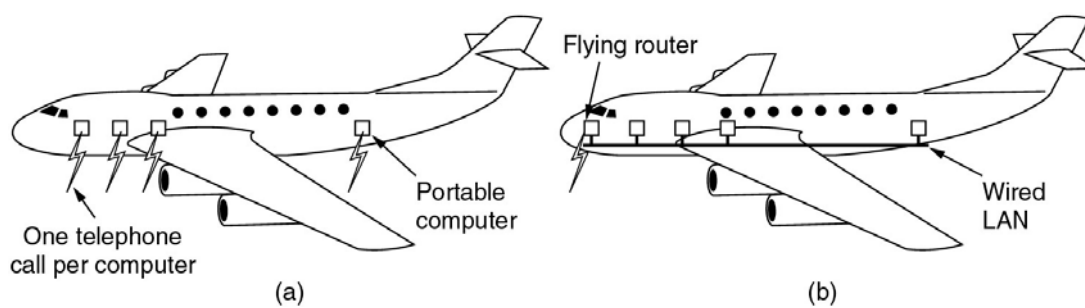
Nagyobb a sáv szélessége, kisebb a csillapítása, mint a rézvezetéknek. Nem érzékeny az elektromágneses zavarokra. Vékony, könnyű. Nehéz lehallgatni.

Vezeték nélküli átviteli közegek

- **Rádiófrekvenciás átvitel.** Mikrohullámú tartományban (100 MHz felett). Egyenes vonal mentén terjed (ismétlők kb. 50 km-enként), jól fókuszálható (parabolaantenna).
- **Infravörös** (10^{12} - 10^{14} Hz). Elsősorban kistávolságú adatátvitel esetén (pl. TV távirányító). Olcsó, könnyen előállítható, viszonylag jól irányítható, viszont óriási hátrány, hogy szilárd testeken nem képes áthaladni (de így alkalmasak lehetnek épületen belüli lokális hálózatok átviteli rendszerének szerepére is).
- **Látható fénycsugár átvitel.** Pl. két épület lokális hálózatát az épületek tetejére szerelt lézerek segítségével kapcsoljuk össze. Igen nagy sáv szélesség, nagyon olcsó (viszont az időjárás befolyásolhatja).
- **Műholdas átvitel.** Geostacionárius műholdak. A műholdakon lévő transzponderek a felküldött mikrohullámú jeleket egy másik frekvencián felerősítve visszasugározzák (3,7...4,4 GHz le, 5,9...6,4 GHz fel). Jelentős késleltetés (250-300 ms).



8.15. ábra: (a) Bluetooth konfiguráció (b) Wireless LAN



8.16. ábra: (a) Személyi mobil számítógép (b) Repülő vezetékes LAN

Az adatátviteli közegeken megvalósított átvitel típusa lehet az

- o Analóg és
- o Digitális átvitel

Analóg átvitel

A múltat teljes egészében az analóg átvitel jellemezte (telefon, rádió, televízió). A kialakított kommunikációs infrastruktúra is döntően analóg. Az analóg telefonvonalakat (előfizetői hurok) még évtizedekig fogják használni adatátvitelre. Egy szakszerűen telepített Cat5 UTP kábel 11 nagyságrenddel jobb (3-4 nagyságrendnyi sebességkülönbség, 7-8 nagyságrendnyi hibaaránybeli különbség) mint egy telefonkábel.

Modem

A modem (modulátor-demodulátor) a digitális információt a telefonvonalon való átvitel céljából analóggá alakítja, majd a másik oldalon vissza. A telefonvonal egy szinuszos váltakozójelet visz át. A modem a bináris jel vezérlésével ezt modulálja, majd a modulált analóg jelből a bináris jelet visszaállítja (demodulálja). A modem szabványok három területet ölelnek fel: modulációs, hibajavító és adattömörítő protokollok.

Digitális átvitel

A folyamatos jelek helyett 0-kból és 1-ekből álló sorozatok haladnak a vonalakon.

Előnyök:

- Hibákra érzékenyebb (csak két állapotot kell jól megkülönböztetni).
- Tetszőleges jel átvihető (hang, kép, ...)
- A jelenlegi vonalakon jóval nagyobb adatátviteli sebességet lehet elérni.
- Olcsóbb (nem szükséges az analóg hullámformát pontosan helyreállítani).

Az átvitel során mindig biteket viszünk át, de mivel eleinte szövegátvitelt valósítottak meg, ezért az átvitt információ egysége a bitesoport volt, amely a szöveg egy karakterét kódolta. Az ilyen, bitesoportokat átvivő módszert szokták **karakterorientált átviteli** eljárásnak nevezni. A hálózatok elterjedésével a szöveges jellegű információk mellett más jellegű információk átvitele is szükségessé vált, ezért a bitesoportos átvitel helyett a tetszőleges bitszámú üzenetátvitel került előtérbe, ezek a **bitorientált** eljárások.

8.3.2 Hálózat felépítés (topológiájuk)

A hálózat felépítését, topológiáját a kábelek elrendeződése, a csomópontok fizikai elhelyezkedése határozza meg. Ez a "hálózat alakja".

- **Sín:** a hálózatnak van egy gerince(BackBone - közös adatátviteli vonal), amihez az összes csomópont csatlakozik. A gerinc mindkét vége ellenállással van lezárva, a rendszer elemei sorba vannak fűzve egy kábelre. Olcsó, kevés kábel kell hozzá. Hiba esetén az egész hálózat működésképtelen lesz.
- **Csillag:** a csomópontok egy közös elosztóba (hub) vannak bekötve. A csillag topológiánál ilyen elosztók gyűjtik össze egy-egy gépcsoport jeleit és továbbítják a központ felé. A csillag topológia előnye az, hogy egy új elosztó beépítésével újabb és újabb gépcsoportokat lehet a rendszerhez kapcsolni. Nem üzenetszórásos (ponttól-pontig). Szakadás esetén megbízhatóbb, viszont sok kábel kell hozzá ezért drágább.
- **Gyűrűs:** a csomópontokat közvetlenül egymáshoz csatlakoztatják, soros elrendezésben, így azok egy zárt hurkot alkotnak. Az üzenetek fogadása egy alkalmas csatoló eszköz segítségével történik. Előre történő huzalozása nehézkes, új csomópont hozzáadása, vagy elvétele megbonthatja a hálózatot. A biztonság kedvéért 2 kábellel is összeköthetik a gépeket. Az adatáramlásnak meghatározott iránya van. Amíg az adatot nem mentik le, addig a gyűrűben kering, tárolódik. Nagy a kockázat, az adatok sérülhetnek, elveszhetnek. Ezt elkerülendő a címzettnek mielőbb le kell menteni és nyugtázni, hogy ne keringjen a végtelenségig.
- **Busz:** egyetlen kábelre felfűzött gépek alkotják a hálózatot. Gyakori a helyi hálózatokban, mivel olcsó a kialakítása. Hátránya, hogy a kábel megbontása, azaz bármely gép kiemelése a hálózathoz a hálózat működésképtelenségét eredményezi.
- **Vegyes (fa):** A busz topológia fa topológiává egészíthető ki, amelyben a többszörös buszágak különböző pontokon kapcsolódnak össze, így alkotva egy fastruktúrát. Meghibásodás esetén csak a csomópont és a hozzátartozó gyökerek esnek ki.
- **Hierarchikus:** az előző formák vegyes alkalmazása.

8.3.3 A hálózatok részelemei

A hálózatok részelemei a hálózat típusától, felépítésétől függenek. Csoportosításuk a következő:

- **számítógépek:** amelyeket össze kívánunk kötni. Ezek a gépek önállóan is, vagy terminálként is működhetnek.
- **vezérlő elektronikák;**

hálózati kártyák: olyan vezérlő egység, amely a számítógépbe építve a hálózat és a gép kapcsolatát biztosítja. Típusát meghatározza a hálózati architektúra és a kábelezés. Sok fajta hálózati kártya kapható, de árban, megbízhatóságban és minőségben eltérnek egymástól.

Kapcsolat



Optikai szál



Koax kábel

Interfészek



Ethernet kártya



Wireless kártya

Switch-ek/router-ek

Router



Switch



8.17. ábra: Néhány hálózati eszköz

- **HUB:** passzív hálózati eszköz, mely a szegmensek kapcsolatát biztosítja. Mivel jelerősítést nem végez, az előírt kábelhosszt nem léphetjük túl.
- **repeater:** olyan elektronikus eszköz, amely az adatátvitel során, a csillapítás következtében torzult jelek felismerését, helyreállítását és újraidőztetését végzi.
- **router:** egy intelligens eszköz, amely meghatározza a hálózaton áramló adatsomagok útvonalát.
- **bridge:** azonos architektúrájú, de különböző protokollok segítségével működő hálózatok össze kapcsolását teszi lehetővé.
- **switch:** olyan szerkezeti elem, amely útvonalszegmensek időleges egymáshoz rendelésével épít fel kommunikációs útvonalat.
- **modem:** olyan eszköz, mely telefonvonalon keresztül teszi lehetővé az adatátvitelt.
- **csatlakoztási felületek:** a hálózati kártyán, ill. a kábelezésen kialakított elemek típusa, mely lehet BNC, UTP, stb.

- **átviteli közeg;**
 - **kábel:** adatátvitelt biztosító közeg. Típusa a hálózat architektúrájától függ. Leggyakrabban a sodrott érpárú - UTP csatlakozójú -, vagy a koaxiális - BNC csatlakozójú - típusú kábelezést alkalmazzák. Figyelem ez a koax kábel nem azonos a TV készülékekhez használt koax kábelrel.
 - **T elágazó:** a koax kábelezésnél használt csatlakozó, melyet a soros topológiájú hálózatokban a hálózati kártyákra csatlakoztatva használunk.
 - **lezáró:** a soros topológiájú hálózatokban a hálózat gerincvezetékének két végpontját kell lezárunk ezzel az 50 ohmos elemmel.
 - **patch panel:** UTP -s hálózatok esetén alkalmazott segéd tábla, amely a felhasználók gépei felől bejövő kábelek rendezését végzi.
 - **patch kábel:** ún. sodrott érpárú, UTP csatlakozóval ellátott kábel, mely a fali csatlakozó és a számítógép hálózati csatlakozója (kártya) közötti összeköttetést biztosítja.
- **vezérlő egység:** ide tartoznak a hálózatot vezérlő számítógépek (szerver) - több is lehet egy hálózatban.
- **működtető program:** a hálózat működését, az őt működtető program, a hálózati operációs rendszer szervezi, vezérli, ill. határozza meg.

A felsorolt eszközök közül természetesen nem mindenre van szükség minden esetben. Azt, hogy éppen mire van szükség az alkalmazott eszközökből a kiépített architektúrától, kábelezéstől, a hálózat bonyolultságától függően kell megválasztani.

8.4 Hálózatok összekapcsolása

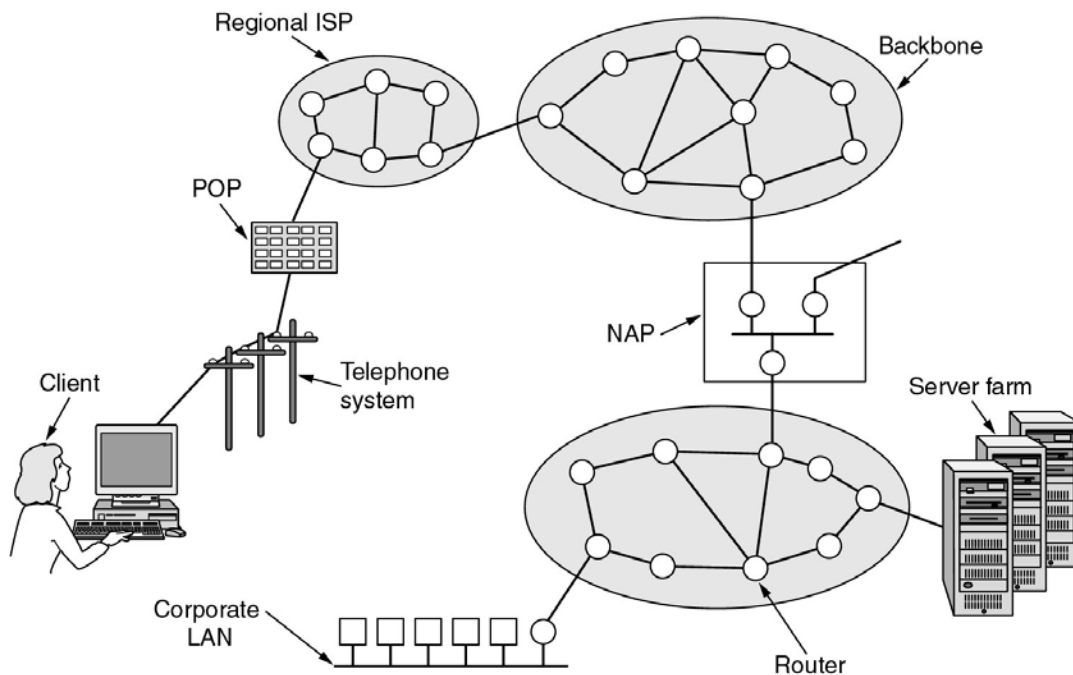
Hálózatok, hálózatrészek összekapcsolására szolgáló eszközök:

- **Ismétlők (repeater):** Egyedi biteket másolnak hálózati (kábel) szegmensek közt. Alacsony szintű eszközök (fizikai réteg), amelyek csak erősítik vagy újragerálják a gyenge jeleket. A hosszú kábelek meghajtásához szükséges áramot szolgáltatják, így hálózat által átfogott távolság növelhető.
- **Hidak (bridge):** Adatkapcsolati kereteket tárolnak és továbbítanak LAN-ok között (2. réteg). Megvalósítja az adatkapcsolati réteg funkcióit (pl. ellenőrző összeg vizsgálata). Kisebb változtatásokat is végrehajthatnak a kereten, mielőtt továbbítanak a másik hálózaton.
- **Többprotokollós routerek:** Eltérő hálózatok között továbbítanak csomagokat. Elvükben hasonlóak a hidakhoz, kivéve, hogy a hálózati (3.) rétegben működnek. Az egyik vonalukon veszik a beérkező csomagot és egy másik vonalon továbbítják őket, mint ahogy azt minden router teszi, de a vonalak különböző hálózatokhoz tartozhatnak és különböző protokollokat használhatnak.
- **Szállítási átjárók (transport gateway):** A szállítási (4.) rétegben teremtenek kapcsolatot két hálózat között.
- **Alkalmazási átjárók:** Egy alkalmazás két részét kapcsolják össze az alkalmazási rétegben. (Pl. levelezési átjárók.)

Hozzáférés az Internethez (kapcsolt vonali, illetve közvetlen vonali hozzáférés)

1. **közvetlen kábelkapcsolat:** helyi hálózatoknál a leggyakoribb forma. Olcsó, könnyen kivitelezhető és a fenntartása nem jár költségekkel.

2. **nagy távolságú kábelezés:** ritkán alkalmazzák. A kábelezés bonyolultsága miatt igen költséges.
3. **optikai kábelezés:** a nagy távolságú kábelezéshez hasonló tulajdonságokkal rendelkezik. Költséges, de megbízható és rendkívül gyors adatátvitelt tesz lehetővé.
4. **mikrohullámú lánc:** olyan földrajzi viszonyok között alkalmazzák, ahol a kábelezés nem megoldható. Költséges ugyan, de gyors adatátviteli forma.
5. **műholdas kapcsolat:** a mikrohullámú lánc műholdas változata. A már meglévő műholdas rendszerekhez igen könnyen hozzákapcsolható új végpontok kialakításával.
6. **ISDN vonal:** viszonylag gyors, de hosszú idejű adatátvitelnél költséges a használata, hiszen a szolgáltatást biztosító telefontársaságok a normál telefonvonalhoz hasonlóan kezelik, ill. számlázzák. Általában napi 3 - 4 órás kapcsolattartás alatti alkalmazható.
7. **bérelt vonali összeköttetés:** ott alkalmazzák, ahol fontos az állandó kapcsolat. Sebessége a kiépítésnél meghatározott sáv szélesség függvénye. Fenntartási költsége az adatátviteltől független, állandó.
8. **kapcsolt vonali összeköttetés:** egyszerűbben, a telefonvonalon modemeken keresztül létesített adatátvitel. Mivel a kapcsolat normál telefonvonalon történik, az adatátvitel minden egyes percéért fizetni kell.



8.18. ábra: Hálózatok Internet hálózatba kapcsolása

A tűzfalak Védik a lokális hálózatot és a kimenő forgalmat is szűrik.

Elemei:

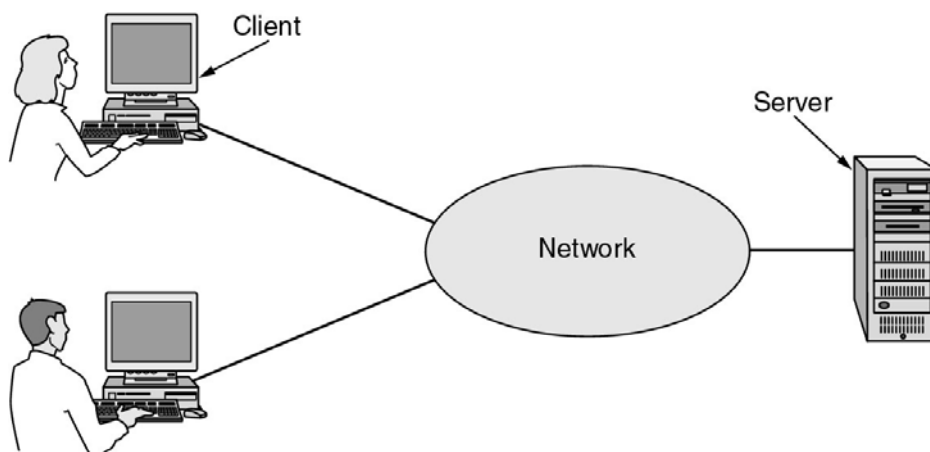
- **Csomagszűrő (packet filter):** Egy szabályos router pár külön feladatkörrel ellátva. Minden kimenő vagy bejövő csomagot megvizsgál, a bizonyos feltételeket kielégítő csomagokat továbbítja, amelyek nem mennek át a teszten, azokat eldobja. A rendszeradminisztrátor által konfigurált táblázatok vezérlik. Pl. le tudja tiltani a bejövő Telnet szolgáltatást igénylő csomagokat.

- **Alkalmazási átjáró:** Alkalmazási szinten működik. Pl. levelezési átjáró, amely minden bejövő vagy kimenő üzenetet átvizsgál, és a vírust tartalmazó leveleket eldobja.

8.5 Kliens-Szerver modellek

A számítógépes hálózatokban a felhasználók közötti kommunikáció egyik megoldási módja a szerver-kliens architektúra amelyen a kommunikációban résztvevő két fél nem egyenrangú módon vesz részt, hanem dedikált szerepeket (szerver illetve kliens) töltenek be. A kommunikációt jellemzően a kliens kezdeményezi, mégpedig azzal a céllal, hogy valamilyen műveletet vagy lekérdezést végeztessen el a szerverrel. A szerver a kérést megkapva elvégzi a megfelelő lépéseket, majd az eredményt a kliens felé továbbítja. Bár az adatok a kommunikáció során értelemeszerűen mindkét irányban áramolhatnak, a műveleteket elvégzését mindig a kliens kezdeményezi, és mindig a szerver hajtja végre - ennek megfordítására nincs mód.

Azokat az architektúrákat, amelyekben mindkét fél kölcsönösen kezdeményezheti művelet végrehajtását a másikon, egyenrangúnak szokás hívni.



8.19. ábra: A kliens-szerver architektúra

Sok számítógépes rendszer és hálózati szolgáltatás működik ezen architektúrával. Ilyen modelleket találunk több Internet szolgáltatás működésében.

Ellenőrző kérdések:

1. Melyek a számítógépes hálózatok legfontosabb jellemzői?
2. Sorolja fel miért előnyös a számítógépeket hálózatba kapcsolni!
3. Mi az a hoszt?
4. Magyarázza meg, hogy mit takar az IMP fogalma!
5. Határozza meg a protokoll fogalmát!
6. Mik azok a hálózati rétegek?
7. Mi a hálózati architektúra?
8. Milyen rétegekből épül fel az OSI modell?
9. Mi a fizikai réteg feladata?
10. Mi az adatkapcsolati réteg szerepe, és milyen alrétegei vannak?

11. Mit biztosít a hálózati réteg?
12. Mi a különbség az üzenet és a csomagkapcsolás között?
13. Ismertesse a csavart érpáras átviteli közegeket!
14. Mire használhatók a koaxiális kábelek?
15. Ismertesse az optikai kábeles átviteli közegeket!
16. Milyen elven működik a vezeték nélküli adatátvitel?

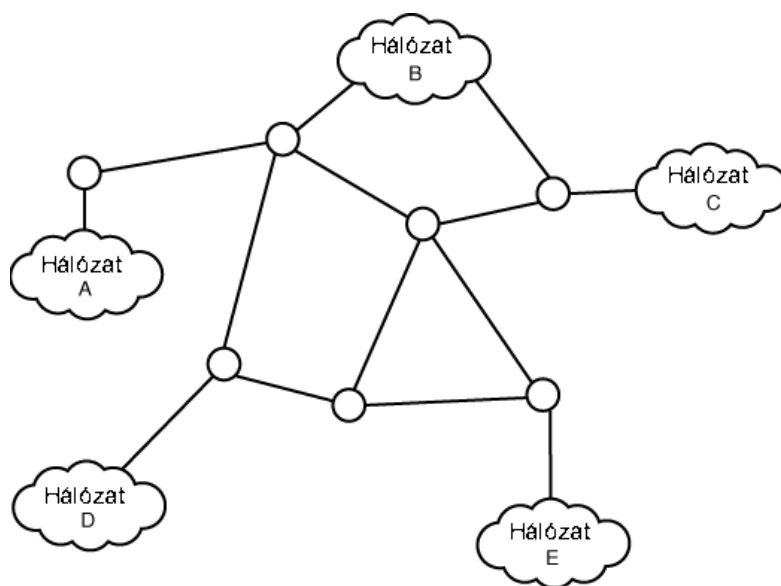
Irodalomjegyzék

- Davies D.W. – Barber D.L. A: *Számítógép-hálózatok*. Műszaki Könyvkiadó, Budapest, 1978.
- Füstös J.: *World Wide Web - Bevezetés a hálózati információszolgáltató rendszer tervezésébe és használatába* Szak Kiadó Kft, Bicske, 1996.
- Hargittai P. – Kaszanyiczki L.: *Internet haladóknak* LSI Oktatóközpont. Budapest
- HegedüsG. –Herdon M. – Kovács Gy. – Némedi J.: *Számítógép-hálózatok* (Számalk, 2002)
- Kónya L.: *Számítógép-hálózatok* (INOK Kiadó Budapest, 2006)
- Magó Zs. – Nagy S.: *Hálózati felhasználói ismeretek* (Számalk, 2003)
- Martin J. - Chapman, K.: *Lokális hálózatok* Novotrade, Budapest, 1992.
- Tanenbaum A. S. : *Számítógép – hálózatok* Novotrade, Budapest, 1992.
- Tanenbaum, A.: *Computer Networks. 3rd Edition.* (Prentice-Hall International, Inc., 1996)
- Számítógép-hálózatok. 2. kiadás.* (Panem Kiadó, Budapest, 2004)

9. A TCP/IP PROTOKOLL ÉS AZ INTERNET

9.1. A TCP/IP protokoll

A TCP/IP protokoll készlet a különböző operációs rendszerekkel működő számítógépek, illetve számítógép-hálózatok közötti kapcsolat létrehozására szolgál. Ez egy olyan protokollkészlet, amelyet arra dolgoztak ki, hogy hálózatba kapcsolt számítógépek megoszthassák egymás között az erőforrásaikat. A TCP és az IP a legismertebb, ezért az egész családra a TCP/IP kifejezést használják. Segítségével különálló számítógép-hálózatok hierarchiája alakítható ki, ahol az egyes gépeket, illetve helyi hálózatokat nagy távolságú vonalak kötik össze (9.1. ábra)

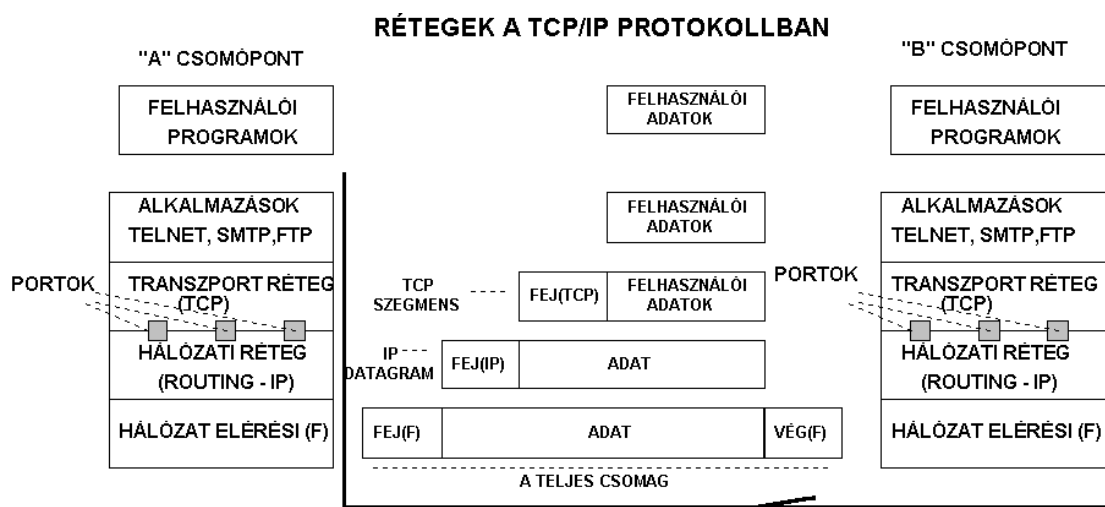


9.1. ábra: Helyi hálózatok összekötése nagy távolságú vonalakkal

Tipikus hálózati feladat a levelezés megvalósítása, amit protokoll (SMTP- Simple Mail Transfer Protocol) szabályoz. A protokoll az egyik gép által a másiknak küldendő parancsokat definiálja, például annak meghatározására, hogy ki a levél küldője, ki a címzett, majd ezután következik a levél szövege. A protokoll feltételezi továbbá, hogy a kérdéses két számítógép között megbízható kommunikációs csatorna létezik. A levelezés, mint bármely más alkalmazási rétegbeli protokoll, a küldendő parancsokat és üzeneteket definiálja. A tervezésekor a TCP/IP-t vették alapul, azaz azzal együtt használható. A TCP a felelős azért, hogy a parancsok biztosan elkerüljenek a címzethez. Figyel arra, hogy mi került át, és ami nem jutott el a címzethez, azt újraadja. Amennyiben egy rész, pl. az üzenet szövege, túl nagy lenne (meghaladja egy datagramm, vagyis egy üzenetben küldhető csomag méretét), akkor azt a TCP széttördeli több datagrammra, és biztosítja, hogy azok helyesen érkezzenek célba. Mivel a fenti szolgáltatásokat jó néhány alkalmazás igényli, ezért ezeket nem a levelezés, hanem egy külön protokoll tartalmazza. Az egész TCP tulajdonképpen nem más, mint rutinok olyan gyűjteménye, amelyet a különböző alkalmazások vesznek igénybe, hogy megbízható hálózati kapcsolatot építsenek ki más számítógépekkel. A TCP hasonlóképpen alapul az IP szolgáltatásokon. Habár a TCP szolgáltatásait sok alkalmazás igényli, vannak olyanok, amelyeknek nincs rájuk szükségük. Persze léteznek olyan szolgáltatások, amelyeket minden alkalmazás megkíván. Ezeket

szedték egybe az IP-be. Ugyanúgy, ahogy a TCP, az IP is egy rutinyűjtemény, de ezt a TCP-t nem használó alkalmazások is elérhetik. A különböző protolloknak ezt a szintekbe rendezését rétegezésnek nevezik. Ennek megfelelően az alkalmazási programok (mint például a levelezés), a TCP, illetve az IP külön réteget alkotnak, amelyek mindegyike az alatta lévő réteg szolgáltatásait használja. A TCP/IP alkalmazások általában a következő négy réteget veszik igénybe:

- alkalmazási protollok (pl. levelezés);
- a TCP-hez hasonló protollok, amelyek rengeteg alkalmazás számára biztosítanak szolgáltatásokat;
- IP, amely a datagrammok célba juttatását biztosítja;
- a felhasznált fizikai eszközök kezeléséhez szükséges protollok (pl. Ethernet)



9.2. ábra: Fejrészekkel történő kiegészítés a rétegekben

Az alapfeltevés az, hogy nagyszámú különböző hálózat áll egymással összeköttetésben átjárók (gateway) segítségével. Ezeken a hálózatokon lévő bármely számítógépet vagy erőforrást a felhasználónak el kell tudnia érni. Az adatsomagok esetleg több tucat hálózaton is keresztülmehetnek mielőtt a célállomásra érkeznének. Az ezt megvalósító útvonal-választásnak természetesen láthatatlannak kell maradnia a felhasználó számára, abból ő mindössze egy Internet címet kell, hogy ismerjen. Ez egy olyan számnégyes, mint például a 128.6.4.194, ami tulajdonképpen egy 32 bites számot reprezentál. A felírás 4 darab 8 bites decimális szám formájában történik.

9.2. Címzési rendszer

Ha egy hálózat számítógépei a kommunikációhoz a TCP/IP protokollt használják, minden számítógép minden adaptere (hálózati kártyája) egyedi azonosítóval rendelkezik, mely egyedi azonosítók alapján a számítógépet az IP protokoll megtalálja a hálózatban. A számítógéphez rendelt azonosítót IP-címnek (IP address) nevezzük, mert az IP protokoll alapvető feladata, hogy a TCP szállítási szintű csomagokat a fejrészben megadott című állomáshoz továbbítsa, akár nagy kiterjedésű hálózaton keresztül is.

A címzési rendszer kialakításánál azt a valóságos tényt vették figyelembe, hogy a címzés legyen hierarchikus: azaz vannak hálózatok, és ezen belül gépek (hosztok). Így célszerű a címet két részre bontani: egy hálózatot azonosító, és ezen belül egy, a gépet azonosító címre.

A hálózati csomópontok IP-címe 32 bites szám, amelyet a leggyakrabban az úgynevezett pontozott tízes formában (dotted decimal form) írunk le, azaz négy darab 0 és 255 közötti decimális számmal, például 193.255.67.4. Ebben a formában a 32 bites IP-címet 8 bitenként konvertáljuk tízes számrendszerbe, és az egyes 8 bites szakaszokra gyakran külön is hivatkozunk.

Az IP-cím két részből áll: az első a csomópontot tartalmazó helyi hálózatot azonosítja, a másik a hálózaton belül a csomópontot. Az, hogy az IP-címből hány bit a hálózat és hány a csomópont azonosítója, elsősorban attól függ, hogy az összekapcsolt hálózatok rendszerében mennyi hálózatra, illetve hálózatonként mennyi csomópont van szükség. A hálózatazonosító az összekapcsolt hálózatok között, a csomópontazonosító a hálózaton belül egyedi. Ha a hálózat az Internethez csatlakozik, a hálózatazonosítónak az egész Interneten belül egyedinek kell lenni. Ezért az Internethez csatlakozó hálózatok azonosítóit (a számítógépek IP-címeinek első néhány - 8, 16, vagy 24 - bitjét) külső szolgáltató határozza meg. Ezt központilag az InterNIC (Inter-Network Information Center végzi különböző régiók különböző szervezeteinek bevonásával. Az IP-címeket, címtartományokat és így a hálózatokat különböző osztályba sorolják. A címzési rendszerben 4 címzési osztályt alakítottak ki. (9.3. ábra)

8				8				8				8				HÁLÓZAT				HOSZT									
0		HÁLÓZAT (7)						HOSZT (24)										"A"		0 - 126				0.0.1 - 255.255.254					
1		0		HÁLÓZAT (14)						HOSZT (16)										"B"		128.0 - 191.255				0.1 - 255.254			
1		1		0		HÁLÓZAT (21)						HOSZT (8)						"C"		192.0.0 - 223.255.255				1 - 254					
1		1		1		0		TÖBBSZÖRÖS CÍM (28)										"D"											
1		1		1		1		FENNTARTVA (RESERVED) (28)										"E"											

IP CÍMFORMÁTUMOK

9.3. ábra: Az IP címformátumok

A cím négy bájtot szokásos - közéjük pontokat írva - a bájtok decimális megfelelőjével leírni.

Az első három címforma a következő: a.) 128 hálózatot hálózatonként 16 millió hoszttal (A osztályú cím), b.) 16 384 hálózatot 64 K-nyi hoszttal (B osztályú cím), illetve c.) 2 millió hálózatot, (amelyek feltételezhetően LAN-ok), egyenként 254 hoszttal azonosít. Az utolsó előtti címforma (D osztályú cím) többszörös címek (multicast address) megadását engedélyezi, amellyel egy datagram egy hosztcsoporthoz irányítható. Az utolsó címforma (E) fenntartott.

Az IP-címbe a hálózat és a csomópont azonosítója az alhálózati maszk (netmask) segítségével választható szét, ezért amikor egy hálózati csomópontot konfigurálunk, az IP-cím mellett az alhálózati maszkot is meg kell adni. Az IP protokoll számára az IP-cím és az alhálózati maszk csak együtt értelmes, mert az IP-cím mindig két részből áll. Az alhálózati maszk hiányában a csomópont nem tudja

meghatározni az őt tartalmazó hálózat címét, amely az útválasztáshoz elengedhetetlen.

Az alhálózati maszk is 32 bites szám, amelyben 1-esek jelzik a hálózat, 0-k a csomópont azonosítójának IP-címbeli helyét. Az alhálózati maszk 1-esekből álló sorozattal kezdődik, és 0-sorozattal ér véget.

Példa alhálózati maszk használatára:

IP-cím: 196.225.15.4
Alhálózati maszk: 225.225.255.0

Kettes számrendszerben:

IP-cím: 11000100 11100001 00001111 00000100
Alhálózati maszk: 11111111 11111111 11111111 00000000

A két szám között az ÉS (AND) műveletet bitenként elvégezve a hálózat címét kapjuk:

11000100 11100001 00001111 00000000
(Tízes számrendszerben: 196.225.15.0)

Ha az ÉS műveletet a cím és az alhálózati maszk inverze között végezzük el, az állomás hálózaton belüli címét kapjuk.

A címzésnél bizonyos címtartományok nem használhatók.

A 127-el kezdődő címek a "loopback" (visszairányítás) címek, nem használhatók a hálózaton kívül, a hálózatok belső tesztelésére használható.

A hoszt címrészbe csak 1-eseket írva lehetséges az adott hálózatban lévő összes hosztnak üzenetet küldeni (broadcast). Például a 195.13.2.255 IP címre küldött üzenetet a 193.13.2 című hálózatban lévő összes gép megkapja.

Ha a hoszt címrésze 0, az az aktuális hálózatot jelöli. Például a saját gépről 0.0.0.0 címre küldött üzenet a saját gépre érkezik.

Az Internetben a rétegeknek megvan az egyedi azonosítója a címzéshez:

Réteg	Címzési módszer
Alkalmazási	Hoszt neve, portja
Internet	IP cím
Hálózatalérési	Fizikai cím

Amikor egy program adatokat küld a TCP/IP-hálózaton keresztül, az elküldendő adatokhoz mellékeli a saját és a címzett IP-címét is. Ha a címzett címében a hálózat azonosítója más, mint a küldőt tartalmazó hálózat címe, a címzett csak útválasztón (útválasztókon) keresztül érhető el.

Ezért a küldő számítógépen futó IP protokollnak először azt kell megállapítania, hogy az elküldendő csomag címzettje helyi hálózatban van-e, ezt pedig a következőképpen teszi.:

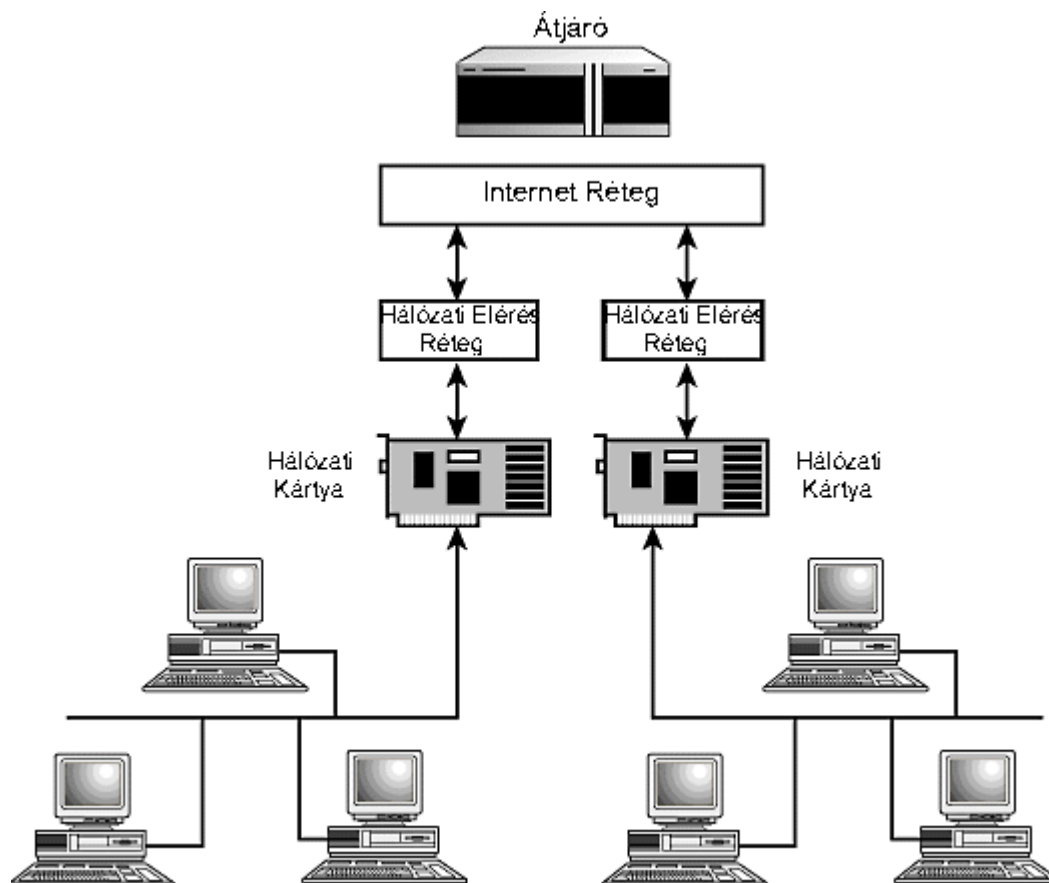
- a küldő IP-címéből a küldő alhálózati maszkja segítségével előállítja a hálózatazonosítót (éppugy, mint a fenti példában),
- a címzett IP-címéből a küldő alhálózati maszkjával előállítja a hálózati címet (a címzett alhálózati maszkjával nem rendelkezik),
- a kapott két számot összehasonlítja.

Ha a két szám egyezik, megkeresi a helyi hálózatban, ha pedig nem, a csomagot az alapértelmezés szerinti átjárónak (amely nem más, mint egy útválasztó berendezés) küldi el.

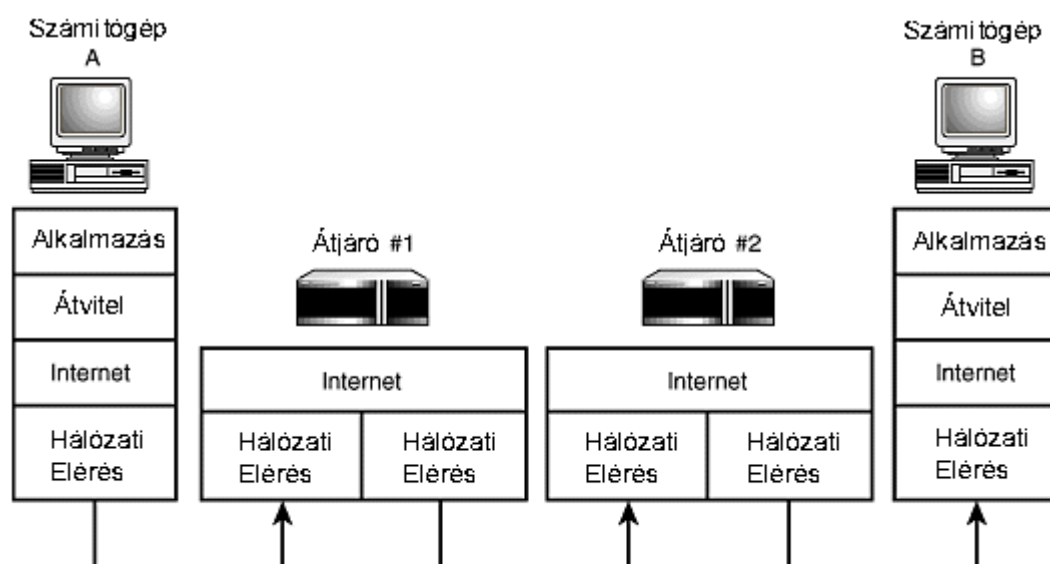
Az IP-cím nem a számítógépet, hanem annak csak a hálózati illesztőjét azonosítja. Ha a számítógépben több hálózati kártya van minden illesztőnek külön IP-címet kell adni.

A számítógépeket alacsonyabb (fizikai, adatkapcsolati) szinten nem az IP-cím azonosítja, hiszen a sok közül ez csak egyetlen (bár kétségtelenül a legelterjedtebb) megállapodás a számítógépek címezésére. Azonban minden hálózati hardverelemnek az egész világon egyedi azonosítója van: ez a hálózatkártya-azonosító (NetCard ID) vagy hardvercím (hardware address). Egy hálózati kártya tehát a vele elektromosan összekapcsolt más hálózati kártyának célzottan jeleket tud küldeni a címzett kártya hardvercíme alapján, az IP-cím alapján azonban nem. Feladat tehát, hogy a címzett állomás eléréséhez az IP-címhez meg kell találni az adott IP-címmel rendelkező hálózati kártya hardvercímét. Ez a művelet a címfeloldás (address resolution). A címzett állomásnak az IP-cím alapján való megtalálása a hálózatban az IP protokoll feladata.

Az útválasztás az a művelet, amelynek során a rendszer egy helyi hálózat valamely számítógépétől az adatcsomagokat különböző vonalszakaszokon keresztül eljuttatja azokhoz a címzettekhez is, amelyek nem részei a helyi hálózatnak. A TCP/IP protokollt használó rendszerek számára az a helyi hálózat amelynek csomópontjai azonos hálózati címet használnak. Ha egy csomag elküldésekor a címzett csomópont IP-címében a hálózati cím más, mint a küldőé, az IP protokoll megpróbálja a csomagot egy útválasztóhoz (router) továbbítani, amelynek az a feladata, hogy a kapott csomagot továbbítsa a címzett hálózat felé. Az útválasztó olyan berendezés, amelynek több hálózati csatlója van, és mindegyik más (helyi) hálózathoz csatlakozik. Az útválasztó csomagokat fogad az egyes, hozzá csatlakozó hálózatok számítógépeitől, és továbbítja őket egy másik hálózati csatlóján. Az, hogy melyik hálózati csatlót kell használni a csomag elküldéséhez, a memóriájában lévő útválasztási tábla (routing table) alapján dönti el. A tábla bejegyzései hálózatok felé vezető útvonalakat (route) képviselnek. Két hálózat összekapcsolását mutatja a 9.4. ábra, két gép közötti kapcsolatot pedig a 9.5. ábra.



9.4. ábra: Két hálózat összekapcsolása átjáróval



9.5. ábra: Két gép közötti kapcsolat átjárókkal

9.3. A Domén Név Rendszer (DNS – Domain Name System)

A számítógépek IP-címeit nehéz megjegyezni és könnyű elgépelni. Természetes tehát a felhasználóknak az az igénye, hogy a számítógépeket az IP-címek

helyett könnyen olvasható és megjegyezhető nevek megadásával érjük el. Azonban a TCP/IP protokollkészlet használata esetén a számítógépeket csak az IP-cím alapján lehet elérni, név alapján nem. Ezt a műveletet névfeloldásnak (name resolution) nevezzük.

A névfeloldás alkalmazásával az Interneten lévő szolgáltató gép vagy valamelyik csomópont eléréséhez a számítógépeket csomópontnévvel (host name) is megadhatjuk. A csomópontnév tetszőleges, legfeljebb 256 karakterből álló szöveg lehet. Az Interneten az úgynevezett teljes tartománynévvel (Fully Qualified Domain Name – FQDN) hivatkozhatunk rá. A tartománynév pontokkal (.) tagolt csomópontnév (host name), amelynek egyes részei a számítógépet tartalmazó szervezetet, illetve a számítógép helyét határozzák meg. Minden csomópontnévhez egyetlen IP-cím tartozik, de egy csomóponthoz (azaz IP-címhez) több név is rendelhető.

Míg az egyes hosztokat a hosztcímük egyértelműen meghatározzák, addig a hosztokat több felhasználó használja, tehát a hozzájuk kapcsolódó felhasználókat is meg kell különböztetnünk egymástól. Erre azok felhasználói neve (login- vagy felhasználónév), vagyis az adott hoszton egyedi azonosító-név szolgál.

Egy személy Internet elérhető levelezési (E-mail) címe tehát két főrészből áll, és a következő alakú:

felhasználónév@hoszt.aldomén.domén

9.4. Internet szolgáltatások

Az Interneten, mivel eltérő felépítésű hálózatokat kötnek össze, szükséges az Interneten folyó kommunikáció közös szabványainak kidolgozása, amelyet az RFC (Request for Comments) dokumentumok tartalmazzák. A szabványok közös alapjául a UNIX operációs rendszerben megvalósított megoldások szolgáltak, mivel elsőként ilyen operációs rendszerű gépeket kötöttek össze. Az Internet lényegesebb alkalmazási protokolljai a következők:

- SMTP Simple Mail Transfer Protocol egy alkalmazási protokoll, amely a hálózati felhasználók egymással való kommunikációját teszi lehetővé, elektronikus levelezés formájában, az elektronikus levelek továbbításának „nyelvét” meghatározva.
- TELNET Terminál emuláció segítségével a saját gépét terminálnak használva egy távoli hosztra felhasználóként lehet bejelentkezni.
- FTP File Transfer Protocol A fájl átviteli eljárás segítségével a felhasználónak lehetővé teszi az általános könyvtár és fájlműveletek végrehajtását a saját gépe és egy távoli hoszt lemezegysége között. Pl.: fájlokat vihet át, törölhet, átnevezhet fájlokat.
- GOPHER Hierarchikusan felépített információban kereső protokoll
- HTTP Hypertext Transfer Protocol

A következőkben ezen protokollok és ezeket használó szolgáltatások közül a legfontosabbakat tárgyaljuk.

9.4.1. Elektronikus levelezés

Az egyik legalapvetőbb szolgáltatás az elektronikus levelezés. Ez az alkalmazás az SMTP (Simple Mail Transfer Protocol -- egyszerű levéltovábbítási protokoll) –re épül. A levelezés, illetve a levelezést megvalósító protokoll működését a következőkben tekintjük át. Tegyük fel, hogy a TOPAZ.RUTGERS.EDU nevű számítógép szeretné az alábbi üzenetet elküldeni.

Date: Sat, 27 Jun 87 13:26:31 EDT

From: hedrick@topaz.rutgers.edu

To: levy@red.rutgers.edu

Subject: meeting

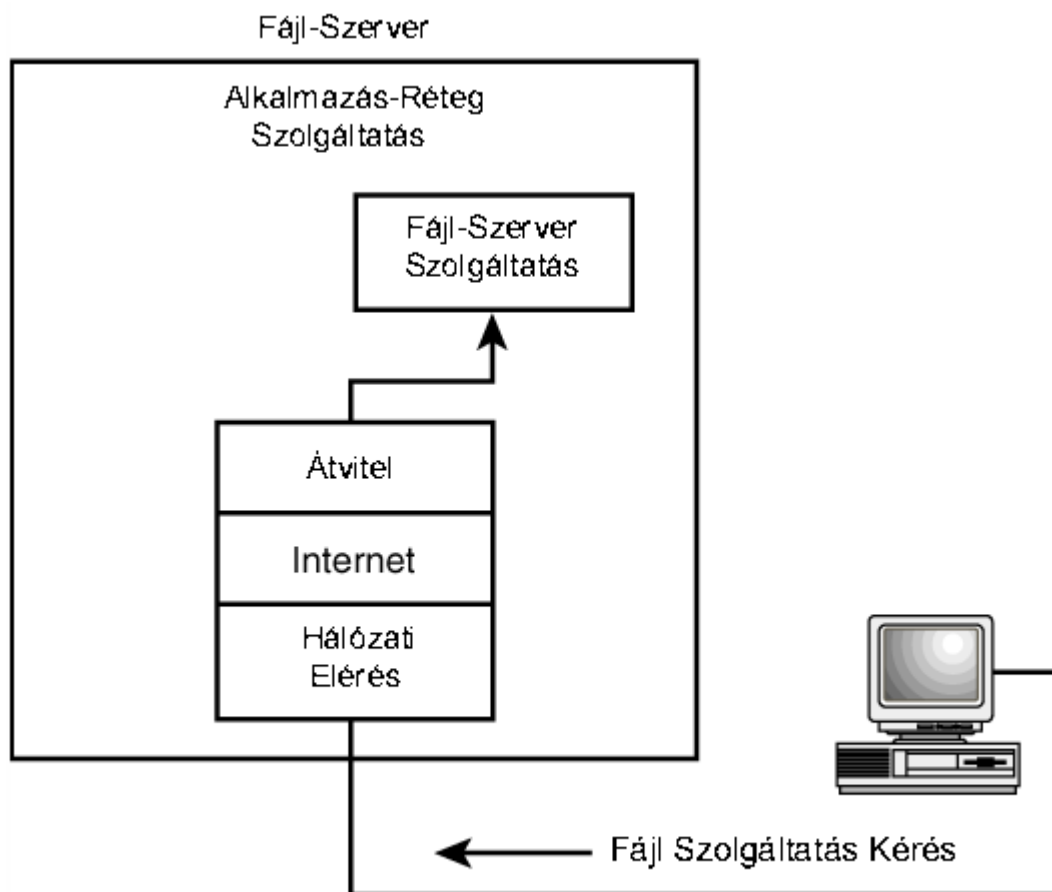
Menjünk holnap vacsorázni!

Az üzenet formátumát egy Internet szabvány (RFC 822) írja le. A szabványban megfogalmazódik, hogy az üzenetet ASCII karakterekként kell továbbítani. Az üzenet szerkezetének az alábbiak szerint kell kinéznie: fejléc sorok, aztán egy üres sor, majd az üzenet szövege következik. Végül a fejléc sorok szintaxisát definiálja részletesen: általában egy kulcsszó, majd egy érték.

A fenti üzenet címzettje LEVY@RED.RUTGERS.EDU. Kezdetben ez úgy nézett ki, hogy csak a címzett nevét és a gépet írták bele: "személy és gép". A szabványok fejlődése azonban ezt sokkal rugalmasabbá tette. Ma már más rendszerek üzeneteinek a kezelésére is vannak előírások (ami persze "magától értetődő"). Ezzel lehetővé válik az Internetbe be nem kapcsolt gépek miatti automatikus átirányítás (forwarding): például az üzenetek egy sor rendszer számára egy központi (mail server) géphez kerülnek. Egyáltalán nem szükséges tehát, hogy létezzen a RED.RUTGERS.EDU névvel jelölt számítógép. A névkiszolgálókat úgy is be lehet állítani, hogy az üzenetek címzettet jelentő mezőjébe tanszékeket írunk, és minden egyes tanszék üzeneteit egy megfelelő számítógéphez irányítjuk. Az is lehetséges, hogy a @ jel előtti részbe ne egy felhasználónak a nevét írjuk, hanem valami mást. Egyes programokat fel lehet készíteni az üzenetek feldolgozására, ezáltal válnak lehetővé a leveleken keresztüli parancsküldés, mint például a levelező listákra való feliratkozás. A levelezési listák, illetve az olyan általános nevek, mint "postmaster" vagy "operator" kezelésére is felkészült a rendszer.

9.4.2. Állományok átvitele - FTP - File transfer protokoll

Az FTP protokoll a hálózatban lévő gépeken megtalálható fájlok átvitelére használható. Használata az elektronikus levelezéssel szemben már folyamatos hálózati kapcsolatot igényel. Adatátviteli sebesség igénye is jelentősebb, hiszen elfogadható időn belül kell átvinnünk esetleg több száz kilobájtnyi adatot. Néhány kbit/s-os átviteli sebesség már elfogadható. A szolgáltatás szintén szerver-kliens modellen alapul, azaz egy szolgáltató szerver és a felhasználó gépe közötti fájlok átvitelét biztosítja (9.6. ábra).



9.6. ábra: Az FTP kliens-szerver modell

Az FTP protokoll két átviteli módban működhet: ascii és binary. Az előbbi, mivel 7 bites kódokat használ, szövegállományok átvitelére alkalmas, az utóbbi bármilyen általános fájlra. Fontos továbbá, hogy egyes rendszerek (pl. Unix) különbséget tesznek kis és nagybetűk közt, erre fel kell figyelni, ha korábban nem ehhez szoktunk, ráadásul mivel Unix alatt a „kep1.png” és a „Kep1.png” két külön fájl, Windows alatt pedig ugyanaz, felülírhatjuk saját fájljainkat, ha nem figyelünk kellőképpen. Azaz a fájl nevében tetszőlegesen lehetnek kis és nagybetűk.

A felhasználó általában akkor tud egy távoli gépről/gépre másolni, ha a távoli gépen is rendelkezik felhasználói jogosultsággal (account-tal).

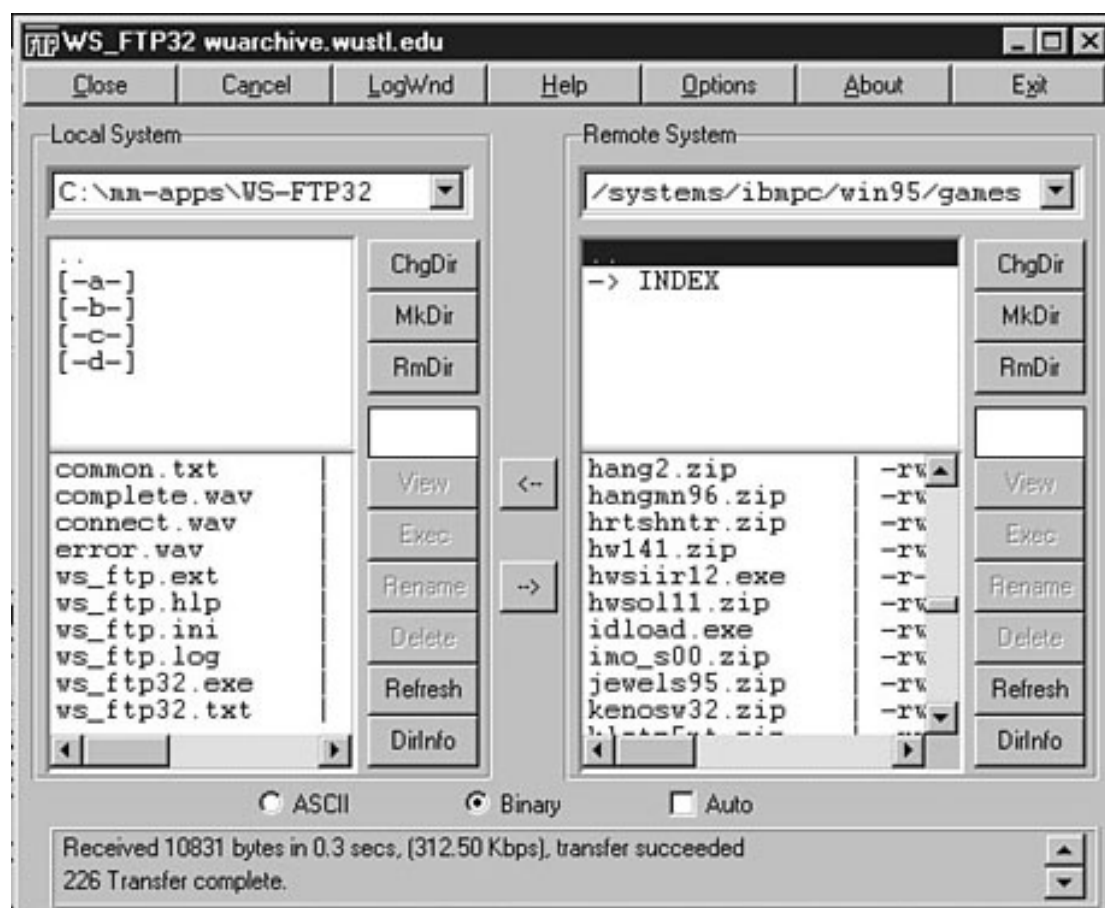
A kapcsolat egy FTP programmal lehetséges, ott kell megadni a célgép nevét, ami egy Internet cím. Ha a kapcsolat létrejött, a rendszer kéri az azonosítót és a jelszót. Ha a belépés sikeres, akkor a következő legalapvetőbb parancsokat használhatja:

- dir paranccsal listázhatja a célgép könyvtárszerkezetét,
- cd paranccsal válthat a könyvtárak között,
- get paranccsal hozhat le fájlokat a távoli gépről,
- mget-tel egyszerre többet
- put paranccsal tölthet fel fájlt a távoli gépre,

- mput-tal egyszerre többet.
- Az ascii és binary üzemmódok közt az asc illetve bin paranccsal lehet váltani.

Vannak mindenki számára elérhető ún. nyilvános elérésű gépek, amelyekre természetesen nem kell account-tal rendelkezni, ez az ún. anonymous ftp. Az ilyen gépekre bejelentkezve bejelentkező (login) névként az "anonymous" szót kell begépelni. A rendszer ekkor arra kér, hogy jelszóként a saját email-címünket adjuk meg, ez sokszor gyakorlatilag nem kötelező, kizárólag statisztikai célt szolgál. Ezek után a távoli gépet, pontosabban annak nyilvánosan elérhető könyvtárait láthatjuk, és az összes fenti FTP parancs használható.

A Windows operációs rendszerekben alkalmazhatunk kényelmes grafikus felületet a fájlok átvitelére. Erre példát mutat a következő 9.7. ábra.



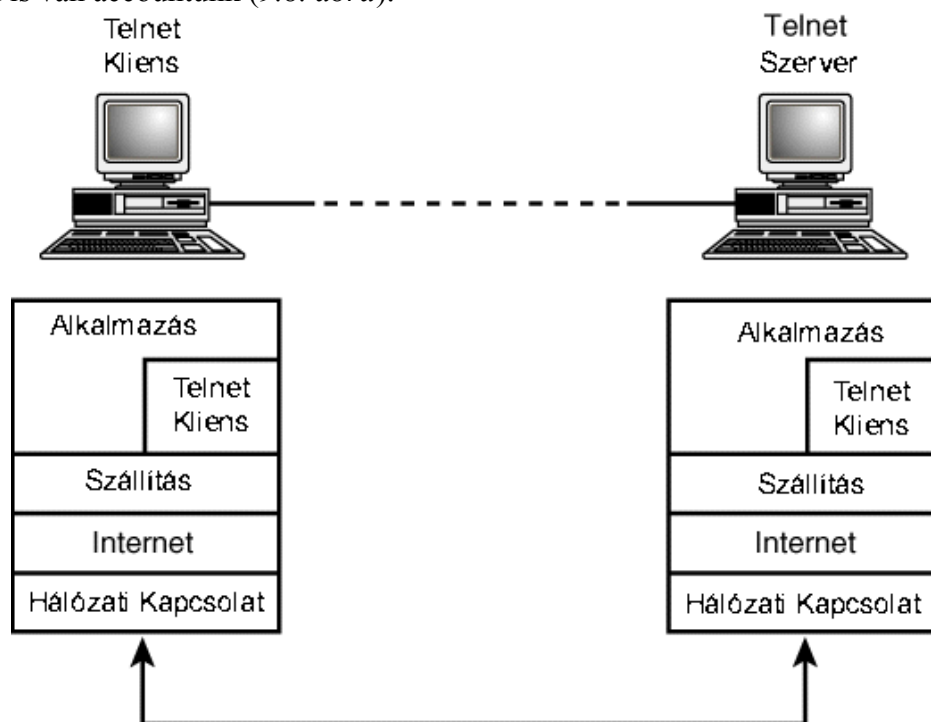
9.7. ábra: Az FTP kliens program felhasználói felülete

Azok részére, akik csak Email kapcsolattal rendelkeznek, létezik a levéllel történő off-line FTP, az FTPMAIL. Ennek az a lényege, hogy vannak olyan hálózatra kötött számítógépek amelyek az FTPMAIL szerver programot futtatják, ez fogadja a leveleket, és feldolgozza a bennük a FTP-vel elérni kívánt gép címét és az FTP parancsokat tartalmazó utasításokat. Az FTPMAIL program végrehajtja a kijelölt FTP kapcsolatot, letölti a megadott fájlt, UUENCODE-olja, majd elküldi levélben a feladónak. Ez egy nem túl kényelmes, de jól használható módszer fájlok letöltésére,

ha nincs más mód. Természetesen ehhez pontosan ismerni kell a letöltendő fájl pontos útvonalát is.

9.4.3. TELNET

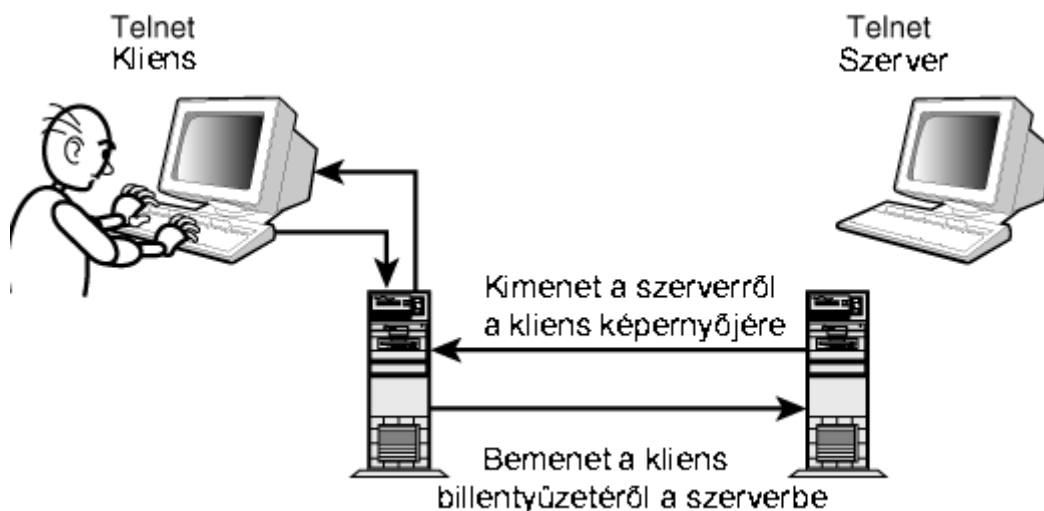
Egy távoli gépre úgy lehet belépni, mintha egy terminálja előtt ülnénk. Azaz a TELNET a gépek közti távoli bejelentkezést lehetővé tevő protokoll neve. Ez is folyamatos (on-line) hálózati kapcsolatot igényel, és sebességigénye hasonló az FTP-hez, (persze csak ha azt szeretnénk, hogy egy leütött billentyű ne 10 másodperc múlva jelenjen meg...). TELNET-tel csak akkor tudunk egy másik gépre belépni, ha azon a gépen is van accountunk (9.8. ábra).



9.8. ábra: A TELNET kliens-szerver szolgáltatás

Bejelentkezés után a rendszer úgy viselkedik, mintha ott ülnénk a távoli gép előtt, azaz a távoli gép operációs rendszerének konvenciói érvényesek, parancsainkat a TELNET protokoll adja át a távoli gép operációs rendszerének, és a távoli operációs rendszer hajtja végre (9.9. ábra). Így a távoli gépen programokat futtathatunk, megnézhetjük az odaérkezett leveleinket, stb.

Ezen lehetőség a hálózati gépek biztonságának egy sebezhető pontja. Ha ugyanis egy távoli gépre rendszeradminisztrátori jogokkal tudunk belépni (felhasználói név: root, a jelszót automatikus próbálkozási módszerrel "kitaláljuk"), akkor a géppel mindent megtehetünk. Az ilyen behatolás módot nyújt arra is, hogy a távoli gépet felhasználva (a TELNET-et ott elindítva) lépjünk be egy "kényesebb" gépre. Ez utóbbi behatolás felderítésekor a behatoló címe az erre használt gép címe, és ha az oda történő behatolás nyomait eltüntetjük, akkor nem lehet kideríteni a kényesebb gépre behatolót.



9.9. ábra: A kliens-szerver kommunikáció

9.4.4. A World Wide Web

A WWW az Internet világban forradalmi változást hozott. Hatására az Internet akadémiai, kutatói hálózatból üzleti és hobby hálózattá vált, szerepet kapott a szórakoztatás világában, a tájékoztató médiák körében, a pénzforgalom és kereskedelem, a reklám világában, az üzleti alkalmazások motorjává vált. Hatása akkora, hogy sokan, mikor az Internet kifejezést meghallják, csakis a WWW világra gondolnak.

A WWW koncepciójában a már jól ismert kliens-szerver koncepció mellett három - tulajdonképpen eddig szintén ismert -paradigma fonódik össze. Ezek a hypertext paradigmája, a hypertext utalások kiterjesztése IP hálózatokra gondolat és a multimédia paradigmája.

A hypertext paradigma lényege olyan szövegmegjelenítés, melyben a lineáris vagy a hierarchikus rendszerű, rendezett szöveg olvasás korlátja megszűnik. Elektronikus szövegek lineáris olvasásához elegendő egy egyszerű szövegnézegető (viewer). Már a legegyszerűbb szövegszerkesztő is megfelel, melynek segítségével előre, hátra lapozhatunk a szövegben, sőt, egy esetleges kereső (search) funkcióval már-már átléphetünk egy szinttel feljebb, közelíthetjük a rendezett szövegek olvasásához. A rendezett olvasást biztosítanak a szótárprogramok, adatbázis lekérdezők. A hypertext jellegű rendszerekben a szövegdokumentumokban valamilyen szövegrészekhez rögzítettek kapcsolódó dokumentumaik is. A megjelenítő valamilyen módon kiemelten jeleníti meg ezeket a szövegrészeket. Ezek a kiemelt részek utalások (kapcsolatok, linkek) más dokumentumokra, más szövegekre, szövegrészekre. A hypertext böngésző nem csak kiemelten jeleníti meg a szövegrészeket, hanem lehetőséget ad azok kiválasztására is (pl. mutatóval rákattinthatunk). A kiemelt rész kiválasztásával az utalt, a hivatkozott (linked) dokumentum betöltődik a nézegetőbe, folytatható az olvasás, természetesen itt ugyancsak lehetnek utalások, akár közvetlenül, akár közvetetten már előzőleg nézegetett dokumentumra is. Az így biztosított információs rendszer jellegzetesen hálós szerkezetű. Léteznek hypertext szövegeket létrehozó, azokat kezelni tudó információs rendszerek, bár jelentőségük a WWW terjedésével egyre szűkebb.

A hypertext IP hálózatra való kiterjesztése megszünteti azt a korlátozást, hogy az utalások csak ugyanarra a helyszínrre, számítógéprendszerre vonatkozhatnak. Egy-

egy kapcsolódó dokumentum helye a hálózaton "akárhol" lehet, ha az utalások megfelelnek az Uniform Resource Locator (URL) szabványnak.

Végül a multimédia paradigma megszünteti a szövegekre való korlátozást: nemcsak hypertext háló, hanem hypermédia háló alakulhat ki. Hivatkozott dokumentum lehet kép, hanganyag, mozgókép, adatfájl, szolgáltatás stb. is. Ráadásul a kép dokumentumokban könnyű elhelyezni további utalásokat is, onnan tovább folytatható a láncolás.

URL (Uniform Resource Locator – Egységes forrásazonosító)

Az egységes forrásazonosító megadja a megjelenítő program számára, hogy az adott szövegrészhez képhez, grafikához kapcsolt dokumentumot milyen módszerrel lehet megjeleníteni, milyen típusú kapcsolatot kell felépíteni illetve hogy ez a forrás hol, az Internetre kapcsolt gépek közül melyiken található. Ilyen azonosítás a következő:

`http://helios.date.hu:70/web/inf/index.htm`

A kapcsolt állomány az index.htm nevet viseli a helios.date.hu gépen lévő web/inf nevű könyvtárban. A kiszolgáló a HTTP protokollal érhető el, amely a Web-szolgáltatáshoz az alapértelmezésként szereplő 80-as port helyett a 70-es portot használja. Az URL a következő információkat tartalmazza:

A protokollt, amelyet az adott forrás eléréséhez használunk. Ezt az URL első tagja adja meg. Ilyen protokollok például az FTP, HTTP, GOPHER stb.

Annak a kiszolgálónak az Internet-nevét (domain név) vagy címét (IP cím) amelyen az adott forrás található. Ez az információ két perjellel (//) kezdődik és egy (/) zárja le.

A kiszolgáló portjának a számát. Ha ez nem szerepel, akkor a megjelenítő-program az általánosan használt alapértelmezést feltételezi. Ha nem a http alapértelmezése szerinti 80-as port címet használják akkor ezt az URL-ben a kiszolgáló nevéhez vagy címéhez kettősponttal (:) kapcsolva kell megadni.

A forrás helyét a kiszolgáló lemezegységének hierarchikus állományrendszerében (könyvtár/fájlnév).

Egy adott HTML-kapcsolaton belül az azonos könyvtárban lévő állományok eléréséhez nem kell a teljes keresési útvonalat megadni. Ha egy dokumentumot elértünk a rendszeren, ez már bizonyos információkat szolgáltat a következő kapcsolat felépítéséhez. Így a szomszédos állományok eléréséhez elegendő egy rész-URL alkalmazása, ami az aktuális dokumentumhoz viszonyítva relatív kapcsolódást biztosít. Azonos könyvtárban lévő dokumentumok esetén elég csak először a teljes URL-t megadni, utána már elég a többi fájlnak csak a nevét megadni. A `http://helios.date.hu/` URL esetén a megjelenítő-program a megadott kiszolgáló főkönyvtárát keresi. A WWW szerver konfigurálásakor megadható, hogy ilyen esetben melyik legyen az a HTML dokumentum, amelyet a kiszolgáló elküld a felhasználónak. Ez lehet pl. üdvözlés, vagy információ a szolgáltatásokról, más URL megadása, tartalomjegyzék, hibaüzenet.

A WWW kiszolgálót futtató gépen a felhasználók a saját könyvtárukban lévő, a rendszer konfigurálásakor definiált speciális nevű alkönyvtárban mindenki számára hozzáférhető, személyes HTML dokumentumokat hozhatnak létre. Ezekre a

könyvtárakra való hivatkozás a ~ karakterrel kezdődik, és a könyvtári hivatkozás a felhasználó neve. A ~ karakter azt jelzi a kiszolgáló számára, hogy ez nem egy szokásos alkönyvtár, hanem az adott felhasználó alkönyvtárában kell az állományokat keresni. Például a „nagy” felhasználói névhez tartozó személyes dokumentumok a `http://helios.date.hu/~nagy/` URL segítségével érhetők el. A kiszolgáló konfigurálásakor meg kell adni annak az alkönyvtárnak nevét, amelyben a felhasználók létrehozhatják az ilyen személyes dokumentumaikat. Ez a könyvtárnév a kiszolgáló konfigurációs állományában (a UNIX rendszereknél általában a `/etc/httpd.conf`) megtalálható (pl. `public_html`, `wwwhomepage`).

Ugyancsak a rendszer létrehozása során definiálható annak az állománynak a neve, amely a rendszerbe való belépéskor, illetve a saját könyvtárak címezésekor megjelenik a felhasználók képernyőjén. Ezt a HTML dokumentumot általában `welcome.html` vagy `index.html` névvel látják el.

A HTTP protokoll

A WWW kliensek a böngészőprogramok, a tallózók. Képesek a Hyper Text Markup Language (HTML) direktívaival kiegészített szövegek megjelenítésére, bennük az utalásokhoz rendelt szövegrészek kiemelt kezelésére, a kiemelt szövegek kiválasztására. Képesek bizonyos kép dokumentumok megjelenítésére, ezekben kiemelések kiválasztására, hangfájlok, videók lejátszására, vagy közvetlenül, vagy valamilyen segédprogram aktiválásával. A szerverek pedig képesek szöveg-, kép-, hang- és videó fájlokat megkeresni saját fájlrendszerükben, és azokat elküldeni a kliensnek megjelenítésre. A kliens és szerver között üzenetváltások jellegzetesen négy lépéses forgatókönyv szerint történnek a Hypertext Transfer Protocol (HTTP) szabályozása alatt.

- Az első lépés a kapcsolat-létesítés (connection): ezt a kliens kezdeményezi, hozzá legfontosabb információ a szerver azonosítója.
- A második lépésben a kliens kérelmet (request) küld a kapcsolaton a szervernek, ebben közli, hogy milyen protokollal, melyik dokumentumot kéri (nem részletezzük, de az átviteli eljárás, a method is paramétere a kérelemnek).
- Ezután a szerver megkeresi a kért dokumentumot és válaszol (response): a kapcsolaton leküldi a kért dokumentumot.
- Végül a kapcsolat lezárul (close). Mindezek után a kliens felelőssége, hogy mit is csinál a leküldött dokumentummal.

Mindenesetre ideiglenesen tárolja a saját memóriájában és/vagy fájlrendszerén, és a dokumentum fajtájától függően megjeleníti azt, esetleg elindítva külső lejátszót, annak átadva dokumentumot közvetve jeleníti meg, lehetőséget ad a felhasználónak végleges lementésre stb. Már a programozás kérdéskörébe tartozik, hogy ha olyan dokumentumot kap a böngésző, melyet közvetlenül nem tud megjeleníteni, lejátszani (futtatni), milyen segédprogramot hívjon meg a megjelenítésre. A felhasználó a MIME szabványoknak megfelelő lejátszókat beállíthat, rendszerint a böngésző konfigurációs menüjében.

A HTTP ügyfél-kiszolgáló protokollt hypertext dokumentumok gyors és hatékony megjelenítésére tervezték. A protokoll állapotmentes, vagyis az ügyfélprogram több kérést is küldhet a kiszolgálónak, amely ezeket a kéréseket egymástól teljesen függetlenül kezeli, és minden dokumentum elküldése után le is

zárja a kapcsolatot. Ez az állapotmentesség biztosítja, hogy a kiszolgáló mindenki számára egyformán elérhető és gyors.

9.4.5. A WWW alkalmazások fejlesztésének eszközei

A HTML

A dokumentumok logikai struktúráját a HTML (Hypertext Markup Language) jelölései segítségével lehet szabályozni. A HTML arra készült, hogy segítségével a dokumentumok szokásos, sorban egymás utáni olvasása helyett, a szövegben elhelyezett kapcsolatok alapján az egész dokumentum könnyebben legyen áttekinthető és elolvasható. Segítségével logikusan szervezett és felépített dokumentumokat lehet készíteni, mivel a nyelv alkalmas logikai kapcsolatok létrehozására a dokumentumon belül és dokumentumok között, amit a dokumentum olvasója kezelhet. A dokumentum fogalmát itt általánosabban kell értelmeznünk: ezek objektumok, amelyek lehetnek: szöveg, kép(grafika), hang (zene), de akár mozgókép (film) is.

A fenti módon szervezett szöveget hypertextnek hívjuk. A folyamatos, sorokba rendezett szöveg végigolvasása helyett a kereszthivatkozásokat követve könnyen el lehet menni a szöveg egy más részére, megnézni más információkat, azután visszatérni, folytatni az olvasást, azután megint egy másik bekezdésre ugrani. Ilyen szerkezetűek a Microsoft Windows súgója, illetve a Windows alatt futó programok súgói. Amennyiben a szöveg mellett más objektum is megjelenik, akkor hipermediáról beszélünk.

A hálózaton az objektumok, illetve ezek részei közötti kapcsolatok magába a szövegbe épülnek be megjelölt szavak és grafikus elemek formájában. Amikor egy ilyenre a felhasználó az egérrel rákattint, a rendszer automatikusan létrehozza a kapcsolatot, és a kapcsolt objektumot megjeleníti a képernyőn (vagy ha hang, lejátssza). Lényeges, hogy a kapcsolt objektum is tartalmazhat további kapcsolásokat különböző objektumokhoz, amelyek elvileg a hálózaton bárhol lehetnek. A WWW úgy is tekinthető, mint egy dinamikus információ tömeg, amelyben a hypertext segítségével kapcsolatok (linkek) vannak. Ennek eredményeként adott információ a hálózat bármely pontjáról megszerezhető, illetve ugyanahhoz az információhoz több úton is el lehet jutni a különböző kapcsolatokon keresztül.

A HTML formátumú fájl valójában egy szöveges fájl, szintén szöveges (olvasható) vezérlőkódokkal. Ezek a vezérlőkódok < és > jelek között szerepelnek, és a szöveg megjelenését, formátumát, például a betűk nagyságát, formáját, stb. jelölik. A szöveg egyéb dokumentumokra vagy a dokumentum más részeire való hivatkozásokat is tartalmazhat amit a vezérlőkódok segítségével adhatunk meg linkek formájában. Ezek a linkek — amelyek a megjelenítéskor általában kék színű, aláhúzott szövegekként, vagy kék keretes ikonokként jelennek meg — hypertext alakúvá teszik a dokumentumot. A legtöbbször minden egyes link hivatkozás egy másik HTML oldalra, ami a Világhálózat bármely pontján lehet.

Az XML

Az XML (Extensible Markup Language) egy leíró nyelv, a strukturált információkat tartalmazó dokumentumok számára. A strukturált információk kétféle dolgot tartalmaznak: egyrészt tartalmat (szöveg, képek) másrészt információkat a tartalom struktúrájáról (például, hogy az adott helyen lévő szöveg a fejléc lábléc vagy

fejezetcím). A leíró nyelv pedig egy mechanizmus arra, hogy ezeket a struktúrákat azonosítsuk a dokumentumban. Az XML specifikáció azt definiálja, hogy milyen módon írható le egységesen a dokumentum. A "dokumentum" szó mögött nem a hagyományos értelemben vett dokumentumot kell érteni, hanem más XML adatformátumok sokaságát. Ilyenek lehetnek például vektor grafikák, E-commerce tranzakciók, matematikai egyenletek, stb. Az XML rövid idő alatt az Internet egyik alapvető építőelemévé vált. A világon egyre több vállalat használja különböző e-business alkalmazásoknál.

A PHP

Hivatalos nevén "PHP: Hypertext Preprocessor", azonban már régen kinőtte ezt az utótagot. Mára már a PHP a legelterjedtebb tartalomgenerátor a HTML oldalakhoz, a PHP-t használó weboldalak száma több millióra tehető. A népszerűség oka abban keresendő, hogy a nyelv (amint azt a neve is jelzi) kezdettől fogva a HTML oldalakba ágyazásra lett tervezve, a fejlesztőkörnyezetek is eleve úgy vannak kialakítva, hogy Web szerverhez kapcsolódnak, és a programot ezen keresztül futtatják, az eredményt pedig weboldalként jelenítik meg. A széleskörű használat következményeként rengeteg kiegészítése készült, adatbázis kezeléstől képkonvertáláson át grafikus kezelőfelületig rengeteg mindent tudunk készíteni a segítségével. A nyelvhez leírást és sok fontos kiegészítést találhatunk a <http://www.php.net/> weboldalon. A HTML-be ágyazottságból kifolyólag alapvetően weboldalak forrásába írunk PHP programot, így meg kell különböztetnünk a dokumentum egyéb részeitől. Ez többféle módon is megtehető:

A JAVA programozási nyelv

Mint említettük, a WWW böngészőkkel egységes, felhasználóbarát felületet kapott a WWW, ezzel részben az Internet is. A programozás eszközeit - korlátozottan - igénybe lehet venni. A CGI (Common Gateway Interface) programokkal, mint például a PHP, melyek a szerver oldalon futnak, bizonyos feladatokat megoldhatunk, bizonyos alkalmazásokat készíthetünk, vagy készíthetnek számunkra.

A Sun Microsystem fejlesztői felismerve az eddigi programnyelvek korlátozásait egy teljesen új programnyelvet dolgoztak ki a WWW programozáshoz, a Java nyelvet. Ezzel párhuzamosan a WWW tallózók fejlesztői olyan böngészőt készítettek, amelyik a Java nyelven írt programokat képes értelmezni és futtatni. Az ilyen tallózók Java virtuális gépként viselkednek. A HTML dokumentumokban a Java programokra való hivatkozások ugyanúgy találhatók meg, mint a más, pl. kép hivatkozások, és a dokumentum letöltése során akár ezek is letöltődnek. Az a tény, hogy a program nem a szerver oldalon fut (mint a CGI programoknál történik), hanem letöltődik a böngészőhöz és a böngésző hajtja azt végre. Ennek egyik előnye az, hogy tehermentesíti a szerveret, esetlegesen a hálózatot. Másik, talán még nagyobb előny, hogy nem kell a különböző Operációs rendszerekhez, géptípusokhoz illeszteni az alkalmazást, a "szabványos" Java kódot a Java virtuális gép, a böngésző végre tudja hajtani, a böngésző feladata az adott hardver, operációs rendszer adottságaihoz való illesztés. Hátrány is jelentkezik azonban, elsősorban biztonsági kérdések merülnek fel a Java alkalmazások (APPLET) futtatásánál. Miután a helyi gépen futtatunk, akár bizonytalan eredetű programokat, külön gondot kellett fordítani arra, hogy ne legyen lehetséges vírus- vagy féregprogramokat készíteni a Java nyelv segítségével. Ennek következtében a Java programcskák nem képesek a számukra kijelölt területen túllépni, maguk a böngészők pedig külön kérésünkre további biztonsági szintként nem

fogadnak Java alkalmazásokat (amivel el is veszítjük a programozhatóságot). A Java nyelv könnyen megtanulható, különösen C++ ismeretek birtokában.

A Java az Internet közvetlen tartozékának tűnik. Javában viszonylag egyszerű alkalmazásokat írni, az Interneten keresztül más gépek felé adatokat továbbítani, grafikákat, interaktív weboldalakat, felületeket létrehozni. Természetesen minden más programozói feladat is megoldható, amire a klasszikus programozási nyelvek képesek. De vajon mi a kapcsolat a korábbi nyelvekkel? Végül is nem szokás új programozási nyelvet fejleszteni anélkül, hogy ne használnának fel korábbi nyelvekkel kapcsolatos tapasztalatokat.

A Java erősen támaszkodik a C++ nyelvre. Ennek oka a C++ objektumorientáltságában, a gyorsaságában, és a teljesítményében keresendő, de a nyelv korábbi jelentősége az Interneten sem elhanyagolható. Ugyanakkor a Java nyelvet megtisztították rengeteg szükségtelen dologtól, ami a C++ nyelv használatát megnehezítette. Ez a tisztítás a nemcsak a programozók tehermentesítését szolgálja, hanem a helyesen működő programok fejlesztését is garantálja. Egy Internet-nyelv esetén ez különösen fontos, mivel szakszerűtlenül programozott alkalmazások, amelyek nem megfelelően működnek vagy összeomlanak, lefagynak, nagy kockázatot jelentenek egy számítógép hálózatban.

Ezek után tekintsük át mik azok a dolgok, amik leginkább jellemzik a nyelvet:

- **Egyszerű.** A Java egy olyan programozási nyelv, amelynek szintaktikája a C++ mintáját követi. Ez az a nyelv, amellyel ma a legtöbb objektumorientált szoftvert fejlesztő programozó dolgozik. A Sun mérnökei a nehezen érthető, és bizonyos esetekben fölöslegesen bonyolult dolgoktól megtisztították a nyelvet.
- **Objektumorientált.** A Java objektumorientált nyelv. Ez egy olyan fejlesztési módszert jelent, amelyben újrafelhasználható adatobjektumok megfelelő összekapcsolásával hozzuk létre a kívánt programot.
- **Biztonságos.** Mivel a Java appletek a felhasználó gépén futnak a hálózatról való letöltés után, ezért lényeges, hogy a letöltött kód ne tartalmazzon hibákat és vírusokat.
- **Rendszerfüggetlen és hordozható.** A Java egyik legnagyobb erősségét az a képessége jelenti, hogy ugyanaz a kód különböző számítógép-platformokon is futtatható. A fejlesztőknek nem kell a már megírt programot minden olyan platformra külön-külön átírni, lefordítani és hibamentesíteni, amelyen futtatni szeretnék. Bármely gép, amelyen van Java interpreter (értelmező), képes a Java appletek futtatására. A Javat nem érdekli, hogy milyen operációs rendszer van a gépen.
- **Párhuzamosságot támogató.** A Java lehetőséget ad arra, hogy a végrehajtás egyszerre több szálon fusson (multitasking). Ez rendkívül fontos tulajdonság egy Webet megcélzó programnyelvtől, mert így jobb interaktív tulajdonságok, és nagyobb valósidejű teljesítmény érhető el.

Appletek és programok

A programozók a Java segítségével önálló programokat is írhatnak, amelyek C++ nyelven programozók programjaihoz hasonlítanak, továbbá olyan appleteket is készíthetnek, amelyek egy böngészőn belül futtathatók. A legtöbb Java kód, amellyel találkozunk, böngészőben futtatható applet, és nem önálló program.

- **Appletek.** Az applet olyasmit jelent, mint “kis alkalmazás”, ami alatt a következőt kell értenünk: az appletek nem önálló programok, hanem mindig egy meghatározott környezetet igényelnek, amiben egyáltalán képesek létezni és végrehajtódni. Ezt a környezetet a WWW böngészők jelentik, amelyeknek természetesen “Java-képesnek” kell lenniük. Ma már minden modern böngésző biztosítja ezt. Ha egy Web dokumentumot egy applettel szeretnénk gazdagítani, akkor egy hivatkozást kell elhelyeznünk a HTML-dokumentumban az appletre. Ha egy Internet-felhasználó ezek után kapcsolatba lép a dokumentummal, akkor a böngésző először magát a dokumentumot tölti le, majd mikor a felhasználó arra a helyre jut, ahol az appletnek meg kell jelennie, a böngésző automatikusan végrehajtja azt. A dokumentum olvasóinak az applet úgy jelenik meg, mint az oldal szerves része, így nekik - bizonyos körülmények között – egyáltalán nem tűnik fel, hogy a háttérben épp egy program fut.
- **Alkalmazások (programok)** Ellentétben az appletekkel, az alkalmazások teljes értékű önálló programok, amelyek végrehajtásához nincs szükség böngészőre vagy más speciális környezetre. (leszámítva a Java-értelmezőt). Az önálló programok futtatásához a Java interpretert kell használnunk, ez egy olyan speciális program, amely a Java virtuális kódját processzorspecifikus bináris utasításokká fordítja. Az így futtatott alkalmazásoknak olyan képességeik vannak, amelyekkel az appletek nem rendelkeznek, például fájlműveleteket tudnak végezni.

9.4.6. Valós idejű kapcsolattartás (IP telefon)

Ma már több mint fél évtizedes, világméretű konvergencia-folyamat figyelhető meg a vállalati számítógépes hálózatok, illetve a telefonrendszerek között. A korszerű megoldás, az IP-telefonrendszerek piacának meghatározó technológiája a Voice over IP (VoIP).

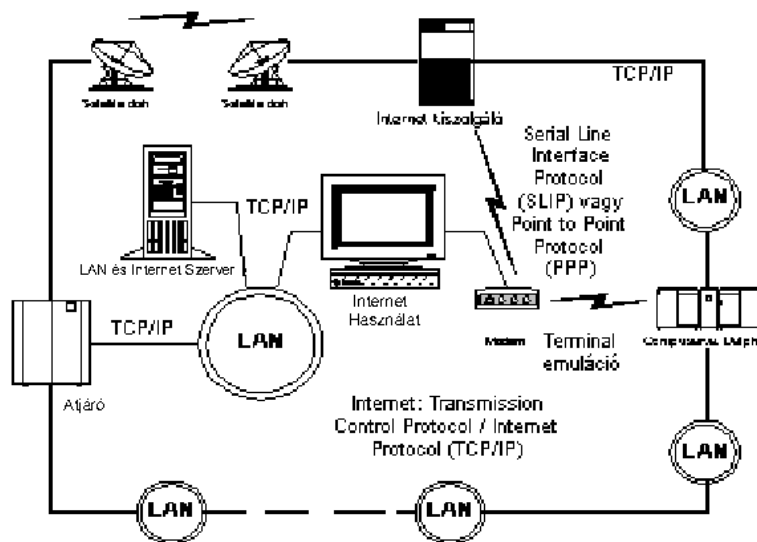
A folyamat mozgatója az informatikai és távközlési rendszerek költséghatékonyabb üzemeltetése, rugalmasabb bővítési lehetősége, illetve az informatikai rendszer magasabb integrálási szintje, amely a teljes informatikai rendszerre nézve kedvezőbb birtoklási költséget (TCO) eredményez a hagyományosan különálló informatikai és telefonrendszerekkel összehasonlítva.

Az IP-telefonrendszer alkalmazásában a nagyvállalatok és a kormányzat jár az élen, ahol az üzemeltetési költség és az informatikai rendszer integrálása alapvető szempont az üzleti folyamatok támogatásában. További fellendülés várható a piacon az IP-telefonok árának várható csökkenése és a nyilvános VoIP-telefonszolgáltatások elterjedésével, mivel a két rendszer közötti átjárás biztosítása további költségelőnyökkel jár.

Az IP-telefonrendszerek telefonszolgáltatásai megegyeznek a korszerű digitális rendszerek szolgáltatásaival, sőt a számítógép és a telefonrendszer integrálásában (CTI, Computer Telephony Integration) meghaladják azokat. Az IP-telefonok könnyen kezelhető, speciális funkciói segítségével a felhasználók jobban ki tudják használni – a digitális alközponti rendszerekben is sokszor meglévő, de nehézkesen használható – telefonszolgáltatásokat. A hang és adat egy hálózaton belüli kombinációja további új és hatékony integrált alkalmazásokra ad lehetőséget, amelyek elősegítik a vállalat munkatársainak mobilitását, növelik hatékonyságukat, az ügyfél-elégedettséget, mindez pedig pozitívan hat a nyereségre.

Az IP-telefonrendszer számos előnnyel rendelkezik a használat, üzemeltetés és a fenntartás területén a hagyományos alközponti rendszerekkel összehasonlítva. Közös infrastruktúra használata miatt a rendszer fenntartási és beruházási költsége alacsonyabb, kisebb személyzettel magasabb szolgáltatási színvonal biztosítható, beleértve a vezeték nélküli telefonok, adatterminálok és notebookokat is. Az IP-telefonrendszer központilag menedzselhető, így a telefonmellékek kiosztása, változása és az egyéb napi üzemeltetési feladatok hatékonyabban elvégezhetők. Hatékonyabban használja a sáv szélességet és több távközlési szolgáltatóhoz kapcsolódva a legkedvezőbb tarifa kiválasztása alapján képes irányítani a kimenő hívásokat, amelyek távközlési költségek csökkenését eredményezik. Az IP-telefonrendszerhez hatékonyan integrálható a számítógépes, illetve a korszerű internet-, intranetes alkalmazásokhoz, segítségével elérhetők az internet- és intranet-alkalmazások azokon a munkahelyeken is, ahol nincs számítógép.

9.5. Kapcsolódás az Internetre



9.10. ábra: Az Internet hálózat felépítése, kapcsolódási lehetőségek

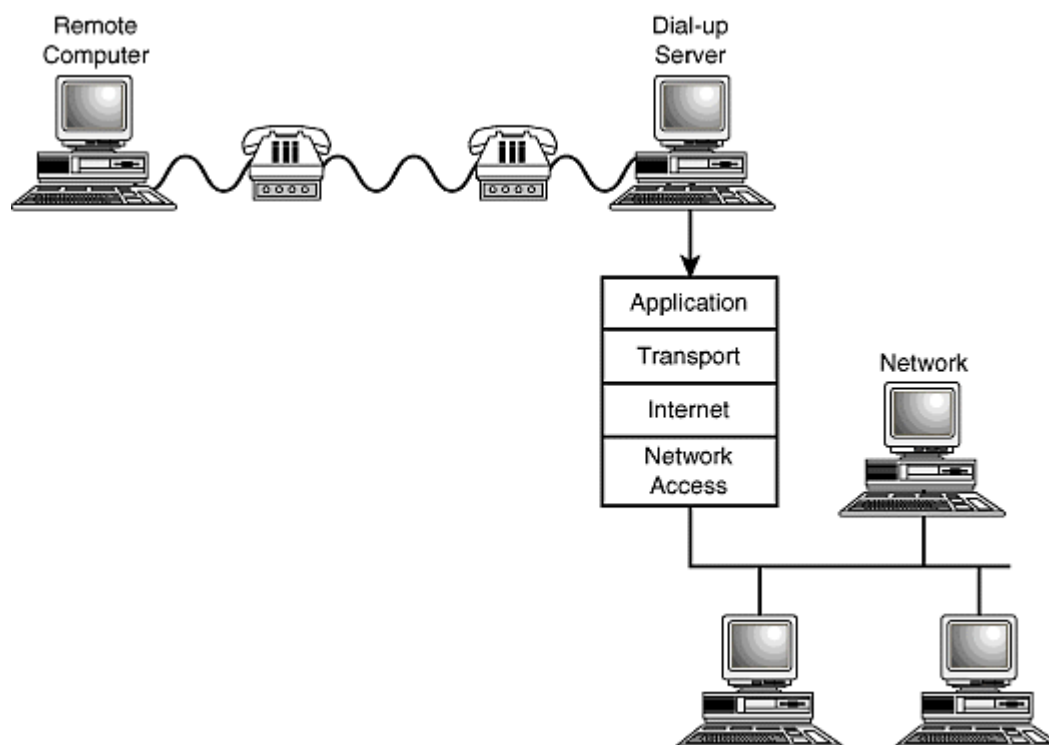
Az Internet felépítést a következő ábrán láthatjuk. A legfontosabb része a nagy adatátviteli sebességű, általában optikai kábelekből, és műholdas kapcsolatokról álló gerinchálózat (bone), amely az ide kapcsolódó hálózatok információit szállítja.

A csomagokat routerek irányítják a különféle útvonalakon. Azonban kevés felhasználónak adatik meg a gerincire csatlakozás közeli lehetősége, általában a "főúttól messze", mellékutak mentén, vagy csak egy kis ösvény végén laknak. Ez a hasonlat itt azért is találó, mert valóban tükrözi az adatátviteli sebesség csökkenését, amit például egy telefonos kapcsolat jelenthet. A felhasználó által elérhető adatátviteli sebességet a gerincig vezető alhálózatok adatátviteli sebessége közül a legkisebb fogja meghatározni.

9.5.1. Telefonvonalon való kapcsolódás

A megfelelő hálózati teljesítmény eléréséhez csak nagyteljesítményű gépekkel lehet a gerincvonalakra csatlakozni. Az átlagos felhasználók ezért a helyi hálózati kapcsolataikat használhatják fel, míg egyéni felhasználók számára az Internet

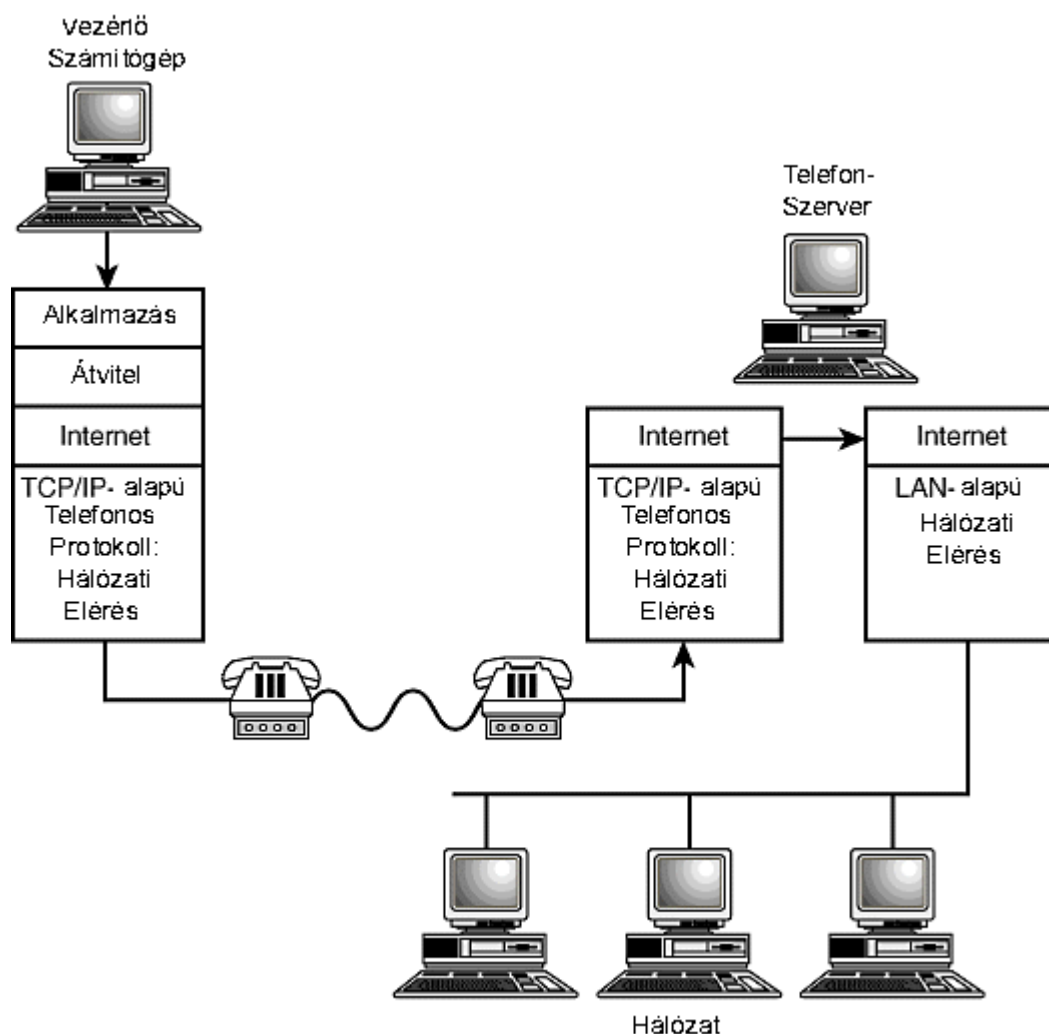
szolgáltatók (providerek) által üzemeltett nagyteljesítményű gépeken keresztül való csatlakozás a megoldás. Ennek megfelelően a következő kapcsolódási megoldások lehetségesek:



9.11. ábra Telefonvonalon való kapcsolódás

Hálózati kapcsolódás. Feltétel: a helyi hálózaton a TCP/IP protokoll használata. Egy routeren keresztül az Internetre küldött csomagok eljuthatnak a célokig.

SLIP/PPP kapcsolat. Telefonvonalon keresztüli kapcsolódás (9.11. ábra). Ilyenkor egy modem és a telefonvonalon TCP/IP szerű kapcsolatot megvalósító SLIP/PPP (SLIP - Serial Line Interface Protocol, PPP - Point to Point Protocol) protokoll szükséges. Számítógépünk a vonal másik végén egy Internetre kapcsolódó kiszolgáló számítógépen keresztül egy IP címet hordozó hálózatra kapcsolt géppé válik. On-line szolgáltatón keresztül (terminál emulációval) az Internetre kapcsolódó gépen fut az a program, amelyet a telefonvonalon keresztül a számítógépet terminálként használva kezelünk (9.12. ábra).



9.12. ábra: Terminálszerver megoldás

A felhasználót általában nem ez, hanem az elérhető szolgáltatások érdeklik. A szolgáltatások alapvetően két csoportba sorolhatók: közvetlen hálózati kapcsolatot nem igénylő (off-line) szolgáltatás - ilyen a levelezés-, és azt igénylő (on-line) szolgáltatások. Ennek megfelelően is több megoldás lehetséges:

A legegyszerűbb szolgáltatás a levelezés: ez lényegében hálózati kapcsolatot nem igényel. Általában egy Internet szolgáltató számítógépén elhelyezett postaládát használunk: ennek tartalmát modemcsatlakozáson keresztül kezelhetjük; UUCP- (Unix to Unix Copy) protokoll segítségével. Unix-ot futtató géppel modemcsatlakozáson keresztül kapcsolódunk a szolgáltató gépére és a leveleket egy menetben fel-, illetve letöltjük; Shell-számlát nyitunk: terminálként (vagy a szolgáltató speciális szoftverén keresztül) bejelentkezünk a szolgáltató gépére, és arról böngésszük a hálózatot; SLIP vagy PPP számlát nyitunk, amelyen keresztül gyakorlatilag minden böngésző, levelező és kommunikációs Internet-alkalmazást futtathatunk.

Ha megvan a lehetőség rá, beköthetjük helyi hálózatunkat az Internetbe. TCP/IP-t és az Internet-segédprogramokat telepítünk a hálózaton, majd a LAN-t valamilyen hálózati kapcsolattal (X.25, ISDN-, vagy sebességű bérelt telefonvonal) routeren keresztül rácsatlakoztatjuk az Internetre.

9.5.2. ADSL

Az ADSL technológia az eddigi hagyományos modemmel elérhető sebességnél lényegesen nagyobb (512-4000 kbit/s) letöltési sebességet kínál. Az új technológia telefonköltség nélküli, korlátlan internetezést tesz lehetővé a hagyományos telefonvonalon, mint átviteli közegen, a telefon vagy fax párhuzamos használata mellett.

Az ADSL az *Asymmetric Digital Subscriber Line* angol szavak rövidítése, jelentése: aszimmetrikus digitális előfizetői vonal. Egy olyan új technológia, amely a hagyományos telefonvonalat, (csavart réz érpárt) rendkívül gyors internetezésre alkalmas, nagy sáv szélességű digitális vonallá alakítja át. Az aszimmetria az adatkommunikáció két irányának eltérő sebességére utal - a legtöbb internetező számára fontos letöltési irány itt sokkal gyorsabb, mint az általában alig használt feltöltési.

Az ADSL alapú Internet elérés elsősorban azok számára megfelelő eszköz, akik a letöltési irányban (például szörfölés, fájlok letöltése) igényelnek nagy sáv szélességet és sebességet, a másik irány sáv szélessége kevésbé fontos.

9.5.3. Wi-Fi

Wi-Fi (úgy is, mint **WiFi**, **Wifi** vagy **wifi**), a **Wireless Fidelity** rövidítéséből – az IEEE által kifejlesztett vezeték nélküli mikrohullámú kommunikációt megvalósító szabvány. Irodákban, nyilvános helyeken (repülőtér, étterem, stb.) megvalósított vezeték nélküli helyi hálózat, aminek a segítségével a látogatók saját számítógépükkel kapcsolódhatnak a világhálóra.

A WiFi a WECA (Wireless Ethernet Compatibility Association) bejegyzett márkanéve, és a korábban IEEE 802.11b-nek nevezett szabvány közérthetőbb, könnyebben megjegyezhető márkanéve, valamint az ilyen eszközök kompatibilitásának is jelölése. Bármelyik gyártótól is szerezzük be az ilyen eszközeinket, működni fognak egymással.

Az WiFi-nek megfelelő eszközök olyan hálózati eszközök, amelyek segítségével rádiós adatátviteli összeköttetést tudunk megvalósítani. Ezek az eszközök a 2400 Mhz-es frekvencia sávban működnek néhányszor 10mW-os adóteljesítménnyel. A WiFi eszközök segítségével akár 11Mbps sebességet (a rendszer sebessége jelentősen függ a vételi viszonyoktól, ha nem megfelelő a rádió kapcsolat, a rendszer automatikusan visszakapcsol kisebb sebességre) is el tudunk érni, ami megfelel egy hagyományos 10 Mbps vezetékes hálózat sebességének.

Rádiós kapcsolatoknak két típusa van, az ad-hoc és a strukturált. Ad-hoc módban a hálózati kártyák közvetlenül egymással kommunikálnak, míg strukturált módban egy központi egységen (Access point) keresztül tartják a kapcsolatot. Az ad-hoc mód előnye, hogy kis gépszámnál (max 5-10 gép) nem szükséges a központi egység beszerzése. Strukturált módban lényegesen több, akár 64-256 gép is kapcsolódhat egy központi egységhez. Ha több központi egységet összekapcsolunk lehetőségünk van roamingra is, tehát a kiépített hálózaton belül bárhol lehetünk, sőt akár mozoghatunk is, mindig on-line maradunk.

Rádiós hálózatonál mindig felmerül az adatbiztonság kérdése. A WiFi eszközök tartalmazzák a WEP-et (wireless equivalency protocol), ami a 40 bites titkosítást jelent - ami vezetékes hálózatoknál megszokott biztonságot nyújt. Akinek ez nem elég kis többlet költséggel 128 bites titkosítást is választhat.

9.5.4. Wi-MAX

Amilyen jelentős sikereket ért el a Wi-Fi a korábbi évben, legalább olyan komoly sikerekre számíthat a WiMax a közeljövőben.

A WiMAX (Worldwide Interoperability for Microwave Access) hatvanhét céget tömörít magába, többek között az Alcatel, Siemens és az Intel jelentette be terveit az együttműködésre és közös munkára, hogy fejlett WiMAX eszközöket és bázisállomásokat építsenek. Az Intel a WiMax-ra egyébként a Wi-Fi egyfajta kiegészítéseként tekint.

Nem a ma használatos vezeték nélküli hálózatok (WLAN) felváltására hozták létre, hanem a kiegészítésükre. Ez nagy mértékben kiterjeszti majd a jelenlegi vezeték nélküli IP-hálózatok (Wi-Fi) alkalmazási körét, a védett ill. szabad frekvenciasávokban történő üzemeltethetőségnek, a közvetlen rálátást nem igénylő egyedülálló átviteli jellemzőknek és a garantált szolgáltatási minőséget biztosító technológiának köszönhetően.

A WiMAX révén városnyi területeket is össze lehet majd kötni, vagyis mindenhová eljuthat majd a szélessávú internethozzáférés. Míg a 802.16-os szabvány (a WiMAX) akár 50 kilométeres körzetben is adhat hálózati hozzáférést, a Wi-Fi (WLAN szabvány) csak 100 méteres körzetet képes ellátni. Ennek a technológiának az a további - és igen nagy - erénye, hogy nem kell hozzá közvetlen rálátás az átjátszókra, s ezzel voltaképpen többet ígér minden ma használatos szélessávú, vezeték nélküli kapcsolatnál, mivel a végpontokban nem kell majd kültéri antenna.

A WiMax hálózatok legfeljebb 70 Mbit/s sebességű adatátvitelre képesek. A WiMax mind a DSL (Digital Subscriber Lines), mind pedig a kábelnetes megoldásoknál olcsóbb lehet, hiszen esetében nem kell kábeleket lefektetni, a vezeték nélküli infrastruktúra kiépítése pedig rendkívül olcsó.

9.5.5. Mobil Internet

Az internet jóvoltából erősen megnőtt és felértékelődött az információk iránti igény. Sokan úgy gondolják, hogy ha bármikor és a világon bárhol hozzáférhetnek a világhálón a szükséges adatokhoz, ha egy másik földrésről is hozzákapcsolódhatnak cégük információrendszeréhez, akkor legyen módjuk erre utazás vagy tárgyalás közben is.

Egy bekábelezett helyi hálózat (LAN) optimális megoldás az irodaházakban dolgozók többségének, de mi legyen azokkal, akik napközben sokszor változtatják a helyüket – például értekezletek, bemutatók miatt? És a tárgyalásra érkező vendégek? Ők hova "csatlakozzanak", ha friss információkra van szükségük a megbeszélések alatt?

Egyszerű és kézenfekvő a megoldás. Vezeték nélküli, rádiós hálózattal kell helyettesíteni a hagyományos összeköttetést mindenütt, ahol erre igény van. Az ötlet egyáltalán nem új, hiszen a mobiltelefonok már hosszú évtizedek óta ezekre az elvekre támaszkodnak, csak az általuk közvetített információk típusa – alapján beszéd, kisebb hányadban rövid szöveges információ, napjainkban pedig egyre inkább kép és film – erősen eltér az IT-világ vezeték nélküli rendszerének kívánalmaitól.

Ha a frekvenciákat most figyelmen kívül hagyjuk, akkor a gyártóktól és a konkrét berendezéstípusoktól függően különböző szempontok szerint csoportosíthatjuk a mobil rendszereket: vizsgálhatjuk őket a használati mód, illetve a rendszerben részt vevők száma alapján.

Kapcsolat két pont között

A használati mód sajátosságára jó példa a Bluetooth. Alapjában kis távolságú, eszközök közötti kapcsolat fenntartására jó, például mobiltelefonhoz vezeték nélküli fejbeszélő illesztésére vagy a telefon és a noteszgép közötti adatcserére.

Gyakorlatilag a 2,5G-s és a 3G-s GSM rendszerek elterjedéséig a mobiltelefonok is efféle üzemmódban voltak használatosak, hiszen ha valakit felhívtunk vagy SMS-t küldtünk neki, akkor egy másik készülékkel kapcsolódtunk össze – esetleg a világ két pontja között. Mostanra – a GPRS, majd az UMTS rendszere révén – egyre inkább átalakul a GSM telefonok használati módja, hiszen ezeken a már elfogadható átviteli sebességű rendszereken adatbázisokhoz, internethez is csatlakozhatunk a telefonnal.

A láthatatlan hálózat

A másik csoport nem eszköz–eszköz kapcsolatot tart fenn. Az ebbe tartozó eszközök terminálként illeszthetők egy meglévő hálózathoz. A PDA vagy a noteszgép (és nemsokára a mobiltelefon is) szerencsés esetben éppen úgy viselkedik és használható egy WLAN-szolgáltatás által lefedett területen, mintha egy kiépített vezetékes hálózathoz csatlakoztattuk volna. A lefedett területen perifériák – nyomtatók, projektorok, adatkiszolgálók – is telepíthetők ideiglenesen, bekábelezés nélkül. Az ilyen nyomtatót egyébként nemcsak a többi, a rendszerbe belépett vezeték nélküli számítógép használhatja, hanem a vezetékes hálózat felhasználói is.

Ha felhasználók száma alapján osztályozzuk a vezeték nélküli hálózatokat, akkor három fő csoportot érdemes megkülönböztetnünk. Az első a személyes méretű hálózat; nevezzük ezt WPAN-nak (Wireless Personal Area Network – vezeték nélküli személyi hálózat). A WPAN kis teljesítményű és kis hatótávolságú. Ide tartoznak: a Bluetooth, a különböző otthoni rádiós interfészek és persze az IrDA, vagyis az infravörös interfész is.

Ellenőrző kérdések:

1. Milyen rétegei vannak az Internet hálózatnak, és ez hogyan viszonyul az OSI modellhez? Mi az egyes rétegek feladata?
2. Mi a TCP protokoll feladata?
3. Mi az IP protokoll feladata?
4. Ismertesse az Internet címzési rendszerét! Mik azok az A, B, C osztályú címek?
5. Magyarázza el a domén nevek rendszerét!
6. Mutassa be az elektronikus levelezés legfontosabb jellemzőit! Mi az SMTP? Milyen funkciókat kell megvalósítani egy levelező programnak?
7. Mutassa be az FTP alkalmazás legfontosabb jellemzőit! Mi az anonymous FTP?
8. Mi az a TELNET? Milyen biztonsági problémákat okozhat?
9. Mi a WWW? Mi a HTML, URL?
10. Mi az a HTTP protokoll? Hogyan működik?
11. Mire szolgál az XML és a PHP?
12. Melyek a Java fejlesztőeszközök fontosabb jellemzői?
13. Hogyan kapcsolódhatunk telefonvonalon az Internet hálózatra?
14. Melyek az ADSL fontosabb jellemzői?
15. Jellemezze a Wi-Fi, a WiMax és a GSM rendszerek jellemzőit!

Irodalomjegyzék

- Davies D.W. – Barber D.L. A: *Számítógép-hálózatok*. Műszaki Könyvkiadó, Budapest, 1978.
- Füstös J.: *World Wide Web - Bevezetés a hálózati információszolgáltató rendszer tervezésébe és használatába* Szak Kiadó Kft, Bicske, 1996.
- Hargittai P. – Kaszanyiczki L.: *Internet haladóknak* LSI Oktatóközpont. Budapest
- HegedűsG. –Herdon M. – Kovács Gy. – Némédi J.: *Számítógép-hálózatok* (Számalk, 2002)
- Kónya L.: *Számítógép-hálózatok* (INOK Kiadó Budapest, 2006)
- Magó Zs. – Nagy S.: *Hálózati felhasználói ismeretek* (Számalk, 2003)
- Martin J. - Chapman, K.: *Lokális hálózatok* Novotrade, Budapest, 1992.
- Tanenbaum A. S. : *Számítógép – hálózatok* Novotrade, Budapest, 1992.
- Tanenbaum, A.: *Computer Networks. 3rd Edition*. (Prentice-Hall International, Inc., 1996) *Számítógép-hálózatok. 2. kiadás.* (Panem Kiadó, Budapest, 2004)

10. INFORMÁCIÓS RENDSZEREK

10.1. Rendszer modell

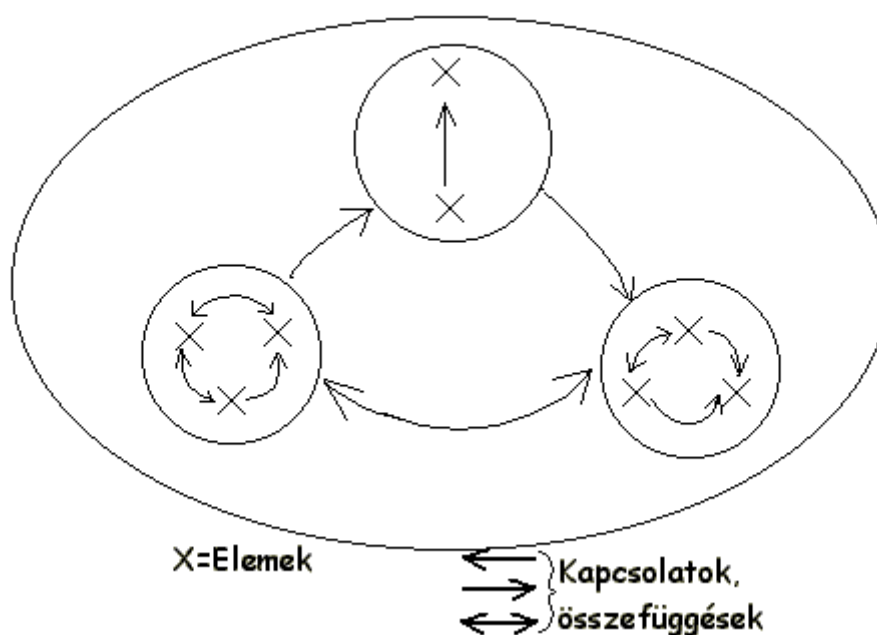
10.1.1. A rendszer fogalma

A gazdasági irányításban és a vállalati vezetésben, s általában a gazdasági életben gyakran találkozunk a kibernetika, rendszerelmélet, rendszerkutatás rendszerszervezés, információ elmélet, információszervezés, információs rendszerek szervezése fogalmakkal, illetve azok gyakorlati alkalmazásával. E fogalmak, illetve az ezzel kapcsolatos tudományos eredmények egyre inkább gyakorlattá válnak.

A gazdasági problémákban való elmaradás, a gazdasági rendszerek vezetése és irányítása, a rendszerek áttekintése mindinkább megkívánja, hogy az agrármérnök is tisztában legyen a kibernetika, a rendszerelmélet és rendszerszervezés, információelmélet és információszervezés legfontosabb fogalmaival, összefüggéseivel, módszertani kérdéseivel és különösen gyakorlati alkalmazási lehetőségeivel, módjával.

A "rendszer" fogalmat a mindennapi élet jelenségeinek igen széles körére használjuk. Beszélünk pl. szabályozó-, jog-, termelési és bérrendszerekről stb., melyek részben fogalmi meghatározások, részben a fizikai valóság bizonyos körülhatárolható területét jelentik. A rendszerfogalom tehát a jelenségek széles körére utal és jelzi a bennük lévő általános, illetve azonos tulajdonságokat, sajátosságokat és törvényszerűségeket, melyek felhasználhatók a gyakorlati életben és a tudományos kutatás területén.

A rendszerelmélet fejlődése során a rendszer-meghatározások sokasága látott napvilágot. Ezek közül a leggyakrabban említett, Bertalanffy által adott meghatározás a következő: A rendszer egymással kölcsönhatásban, kapcsolatban lévő elemek együttese. Ezt szemlélteti a 10.1. ábra.



10.1. ábra: A rendszer

Ludwig von Bertalanffy magyar származású bécsi biológus a rendszereket két nagy csoportra tagolja; nyílt és zárt rendszerre.

- **Nyílt rendszernek** nevezte azokat a rendszereket amelyek áramlási mennyiségekként anyagot, energiát és /vagy/ információkat vesznek le a környező világtól, ezeket belsőleg feldolgozzák és feldolgozott formában visszaadják a világnak.
- **Zárt rendszerek** pedig azok a rendszerek, amelyek az őket környező világgal egyedül és kizárólag energetikai kapcsolatban állnak.

A nyílt rendszerek állapota ezek szerint „**dinamikus vagy folytonos egyensúlyi állapot**”, a zárt rendszerek állapota pedig statikus egyensúly.

A rendszerek felépítésének és működésének egyik fontos jellemzője, illetve meghatározója a **struktúra**. **Struktúrának** nevezzük az **elrendeződés vagy kapcsolódás módját és formáját**.

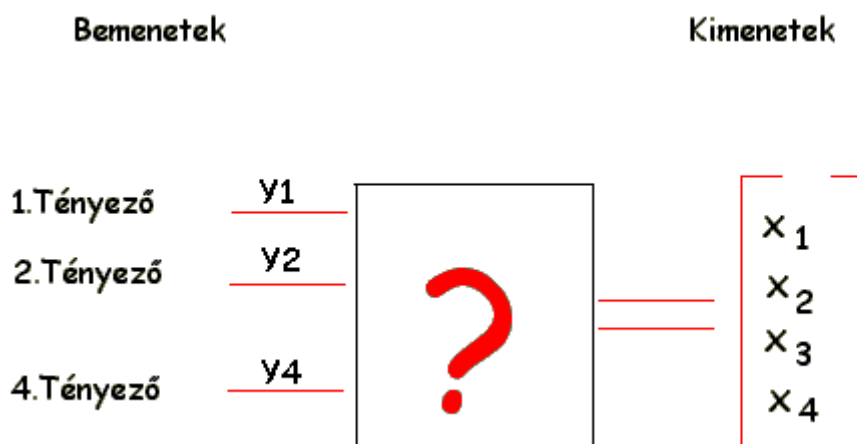
Piaget szerint az ember viselkedésének és gondolkodásának műveleti elemei struktúrák szerint rendeződnek. Azt azonban, hogy mely elemek épülnek e struktúrák által összetettebb viselkedési vagy gondolati építménnyé, mindig az ember aktuális állapota és körülményei határozzák meg. Ezért az a mód, ahogyan az elemek elrendeződnek /azaz a struktúra/ jellemzője az adott rendszernek, behatárolja képességeit és viselkedését

A fentiek alapján megállapítható, hogy:

- a rendszerek tervezése, szervezése során azok viselkedését struktúrájukon keresztül vizsgálhatjuk és
- a rendszer tervezésénél figyelembe kell venni, hogy a struktúrán keresztül befolyásolhatjuk a rendszer viselkedését.

A rendszer struktúráját az elemei közötti, illetve az elemek és az egész közötti viszonyok összessége alkotja. A rendszer eleme a rendszernek az a legkisebb része, amelyet a rendszer működése szempontjából, tovább nem bontunk. A rendszer elemeinek ötféle összekapcsolási módját különböztetjük meg.

A továbbiakban a vizsgálódási területünket leszűkítve a nyílt rendszerekkel foglalkozunk. Az általános rendszerelméleti alapokra támaszkodva megszerkeszthetjük a rendszer elemi logikai sémáját a tér olyan részeként, amelyben valamennyi elemét és környezethez való kapcsolatát sűrítjük (10.2. ábra).



10.2. ábra: A rendszer elemi logikai sémája

Az egyszerű ábrázolhatóság miatt a sémát síkban ábrázoljuk. A környezetből jövő hatásokat BE és a rendszerek a környezetre gyakorolt hatásait KI nyilakkal jelöljük. A hatás a környezet és a rendszer között mindig kölcsönhatás formájában nyilvánul meg. Azt, hogy mi tartozik a környezethez és mi a rendszerhez, meg lehet pontosabban határozni.

A rendszer egészének működéséről tájékoztatást nyújt a bemenetek és a kimenetek állapota, a közöttük lévő összefüggések vizsgálata.

A **bemenet** a rendszerbe kerülő materiális tényezők összessége, s ezekből - a rendszer jellegétől függően - hol az egyik, hol a másik /anyag, energia, információ, munkaerő/ lehet túlsúlyban. A **bevitel helye** a rendszernek lehet egy bizonyos pontja vagy szétszórtan kiterjedhet az egész rendszerre, de meghatározásukat pontosan és objektíven kell elvégezni, mert csak így ítéltük meg a kimenetek várható értékeit.

A **kimenet** a rendszert elhagyó materiális tényezők /anyag, energia, információ, munkaerő/ összessége. A **kimeneti értékek** alapján győződhetünk meg arról, hogy a rendszer miként teljesíti feladatát.

Azt az összefüggést, amely megadja, hogy adott bemenet hatására hogyan változik a **rendszer állapota** /milyen folyamat zajlik le benne/, a **rendszer átmeneti függvényének nevezzük**.

Ha a bemenő adatokat az

$y_i = f/y_{1i}, y_{2i} \dots y_{ni}/$ függvény adja meg és a rendszer kezdeti állapotát az $a_i = /x_{1i}, x_{2i} \dots x_{ni}/$ állapotvektor jellemzi, akkor az állapotváltozást az f folyamatvektor felhasználásával az

$a_{i+1} = a_i + f/y_i/$ összefüggéssel adhatjuk meg. Ebből az átmeneti függvény jelölésére az $f/a_i, y_i/-$ t bevezetve,

$f/a_i, y_i/ = a_{i+1}/$

A kimeneti függvény arra ad feleletet, hogy a rendszer a_i . állapotban milyen hatásokat ad át környezetének.

Ha

$l_i / g_{1i} \dots g_{2i} \dots g_{ni}/$

akkor

$l/i+l/ ::: R /li+l/ ::: R /li + l/Zi/ /e$

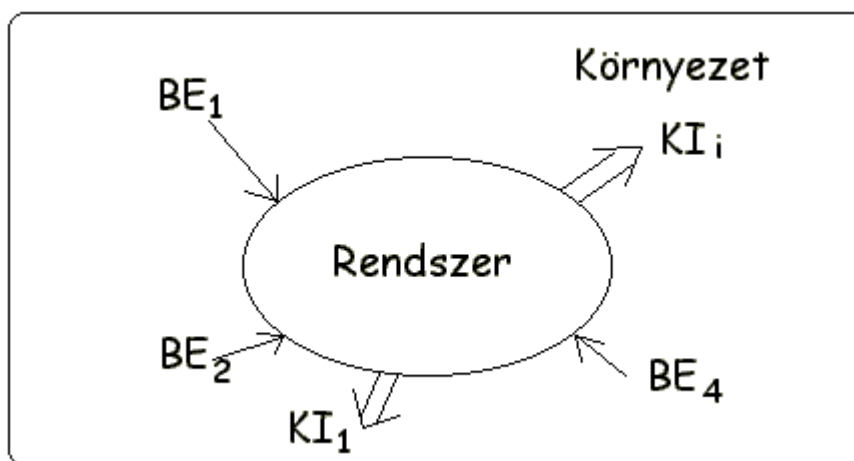
A bevezetett R operátor azt fejezi ki, hogy a rendszer li állapotban milyen li hatásokat ad át környezetének.

A kibernetika új szemléleti módot és új vizsgálati módszereket adott, amelyek nagy hatással volt a különböző tudományágak fejlődésére, azáltal, hogy egy sor új kérdést vetett fel, sok régi problémát pedig új módon fogalmazott meg. Az új módon megfogalmazott problémák adott tudományágak területén újabb összefüggések feltárását, a törvényszerűségek jobb megismerését tették lehetővé.

A számítógépek pedig lehetővé tették a kibernetika segítségével feltárt összefüggések és törvényszerűségek alapján meghatározott célok könnyebb és gyorsabb realizálását.

A rendszerelmélet és a kibernetika, mint két fiatal tudományág vonatkozásában gyakran felmerül tárgyak és vizsgálódási területeik közötti hasonlóság és különbség kérdése. A kibernetikát felhasználja a rendszerelmélet tudománya és a kibernetika is foglalkozik a rendszerelmélettel, rendszerek vizsgálatával. Így a rendszerelmélethez hasonlóan a kibernetikában is sok rendszer definícióval találkozunk.

A kibernetikának a rendszerek vizsgálatához használt alapvető módszere a **fekete doboz /Black Box/** módszer. Ez a módszer különösen fontos bonyolult rendszerek szabályozási (irányítási) feladatok vizsgálatánál. Ennél a módszernél a külső megfigyelő ismeri a bemeneti és a kimeneti jeleket, mennyiségeket, de a belső szerkezet /struktúra/ és működése /a folyamat/ ismeretlen számára. A vizsgált rendszer néha annyira áttekinthetetlen és oly sok rejtett elemet tartalmaz, hogy csak a rendszer részeinek paramétereit tudjuk megfigyelni, észlelni vagy mérni, de maga a teljes rendszerfolyamat és törvényszerűségei rejtettek (10.3. ábra).

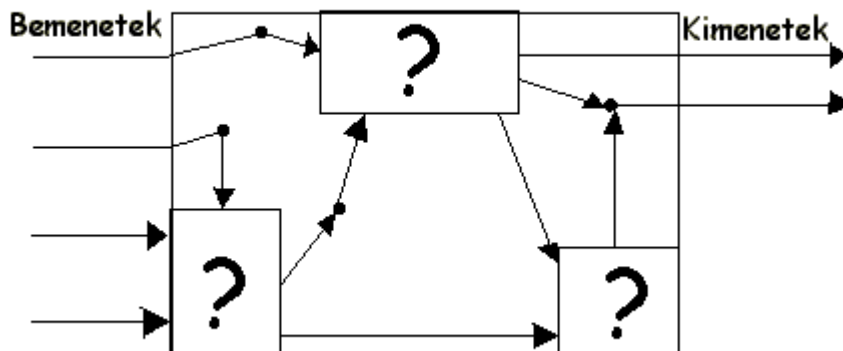


10.3. ábra: A rendszer bemenete, kimenete és a környezet

A fenti módszert a rendszer vizsgálatánál kettős jelleggel használhatjuk.

a./ Megfigyeljük a bemeneteket és kimeneteket, tapasztalatokat gyűjtünk, majd egyrészt az ismert bemenet esetén következtetünk a várható kimenetre, másrészt pedig a keresett /kívánt/ kimenet érdekében következtetünk a szükséges bemenetre. Ez az eljárás **a rendszer valódi természetének felismerése helyett a rendszer viselkedését próbálja utánozni.**

b./ A fekete doboz módszert kísérletezésre használva lépésenként megismerhetjük a rendszer struktúráját és a rendszer folyamatait, így **a fekete doboz lassan szürke dobozzá végül pedig fehér dobozzá alakítható**. A szürke doboz elvi sémáját a 10.4. ábra mutatja be.



10.4. ábra: A szürke doboz elvi sémája

Mint a kibernetika definíciójából is kitűnik, a kibernetika alapvető vizsgálódási területe a rendszer **irányítása**, a **vezérlés** és **szabályozás** módszerével az irányított rendszereket tanulmányozza.

A rendszer **alrendszeréből**, az alrendszerek pedig **elemekből** állnak. Azt, hogy adott esetben mit tekintünk rendszernek és alrendszernek, illetve elemnek, mindig a rendszerek szemlélésének, vizsgálatának célja vagy szintje határozza meg.

Minden irányítási rendszer két alapvető alrendszernek - az irányítás objektumának, illetve irányított alrendszernek és az irányítás szubjektumának, illetve irányító alrendszernek - dialektikus egységét jelenti.

Az **irányítás objektuma** tulajdonképpen a rendszer anyagi része, amelyet egymással kölcsönhatásban lévő, szerkezeti elemek halmaza határoz meg. Ilyen alrendszereket képeznek például egy mezőgazdasági vállalkozáson belül a termelési egységek.

Az **irányítás szubjektuma** az irányított alrendszer irányításában résztvevő elemek együttese. Ezek kapcsolatban vannak egymással, hatnak egymásra. A kibernetikai rendszer elemi sémája a 10.5. ábrán látható.

Definíció: Részrendszer

Részrendszeren önmagában megálló rendszert értünk, egyedül a kívülálló megfigyelő által elfogadott nézőpont különbözik, például egy szervezet információrendszere a szervezeten belül egyszerre részrendszer és önmagában megálló rendszer.

Az informatikában az információs rendszer emberek, gépek, módszerek halmaza, amelyek bizonyos tevékenységek végrehajtására vannak megszervezve.

13.1.1. Rendszerszemléletű megközelítés, rendszertípusok, rendszervizsgálat

A rendszerszemléletű megközelítés

Érdekes erről az általános megközelítésről szót ejteni. Több módszertanról állítják azt készítőik, hogy a rendszerszemléletű megközelítést követik. Az általános rendszerelméletről több alapvető munka is megjelent például [Bertalanffy], [Churchman]). Ezek alapján érdemes egy kicsit az idetartozó fogalmakat absztraktabb módon megtárgyalni.

A rendszerszemléletű megközelítés egységes valaminek (entitásnak, objektumnak stb.) tekinti a rendszert, nem meg feledkezve az alkotó részekről. Ez a szemlélet észleli az alkotórészek aktivitását, tevékenységét, de ugyanakkor figyelmet fordít a rendszer egésze által mutatott aktivitásra.

A zárt rendszerek

Elméletileg a zárt rendszer olyan rendszer, amelyik nem lép kapcsolatba a környezetével, erre példa az olyan kémiai 'reaktor', mint például egy lezárt edény, amelyben valamilyen vegyi reakció folyik. Azonban tökéletesen zárt rendszer gyakorlatilag nem létezik, az elméletileg létező zárt rendszert pedig nem tudjuk sem megfigyelni (hiszen zárt), sem befolyásolni (hiszen zárt). Sokkal hasznosabb, ha zárt rendszernek azt tekintjük, amelyik minimalizálja a környezetével zajló csere bizonytalansági fokát, azaz csak az előre pontosan meghatározott bemeneteket (jól definiált inputok) fogadja el, és csak előre pontosan meghatározott kimeneteket bocsát ki a feldolgozás után. (Nem véletlenül tűnik ismerősnek ez a meghatározás, ez a strukturáltság alapelve.) Ezzel szemben a nyílt rendszereknek meg kell birkózniuk a bizonytalan bemenetekkel, ezért nagyon alkalmazkodó képeseknek (adaptívnek) kell lenniük, ilyenek például az emberek vagy a társadalomban létező szervezetek.

A zárt és nyílt rendszereket egy skála két szélső pontjának tekinthetjük, annak megfelelően, hogy a bemeneteiknél milyen fokú bizonytalansággal kell számolnunk. A tipikus rendszerek, amikkel a gyakorlatban találkozunk, általában egyik szélsőséghez sem tartoznak.

A részrendszerek: rendszerek különböző szinteken

Egy rendszert leírhatunk néhány mondattal, egy diagrammal vagy akár egy több ezer oldalas dokumentummal. Az olyan leírást, amely rengeteg részletet tartalmaz, alacsony absztrakciós szintűnek tekintünk, míg egy magas absztrakciós szintű leírás kevés részletet tartalmaz. Egy rendszert általában szisztematikus módon, részekre bontva, dekomponálva vagy finomítva (ekvivalens kifejezéseket használva) írunk le a folyamatainak magas absztrakciós szintjéből indulva és lejutva egy alacsonyabb absztrakciós szintre.

Ezt a lépéssorozatot rendszerint nagy és bonyolult rendszereknél hajtjuk végre. Ezek a rendszerek olyannyira komplexek, hogy praktikusán nem lehet leírni, sem felfogni vagy megérteni őket egyetlen rendszerként. Ezért bontjuk fel olyan méretű részrendszerekre, amelyeket képesek vagyunk kezelni és megérteni. Elképzelhető, hogy még tovább bontjuk ezeket a részrendszereket is, hogy további részleteket tudjunk leírni. Gyakran valamilyen hierarchikus diagram segítségével ábrázoljuk a köztük fennálló összefüggéseket, egyes részrendszerek több másik (rész)rendszerhez tartozhatnak, így egy hálós formájában és nem faszterkezetben tudjuk leírni a kapcsolódásukat egymáshoz.

A már idézett definíciónk szerint minden részrendszert önmagában rendszernek tekinthetünk, és ezért további részrendszerekre vagy elemekre bonthatjuk.

A rendszermodellek mint absztrakt rendszerek

Absztrakt rendszerek csak fogalmakból állnak - szemben a fizikai rendszerekkel, amelyek valamit csinálnak, valahogy viselkednek -, amelyek a fizikai valóságban nem léteznek, csak a fogalmak, az ideák világában. Az absztrakt rendszerek nem csinálnak semmit - ez a legfőbb megkülönböztető jel a fizikai rendszerekhez képest -, azonban van céljuk, valamilyen fizikai rendszer ábrázolása, leírása.

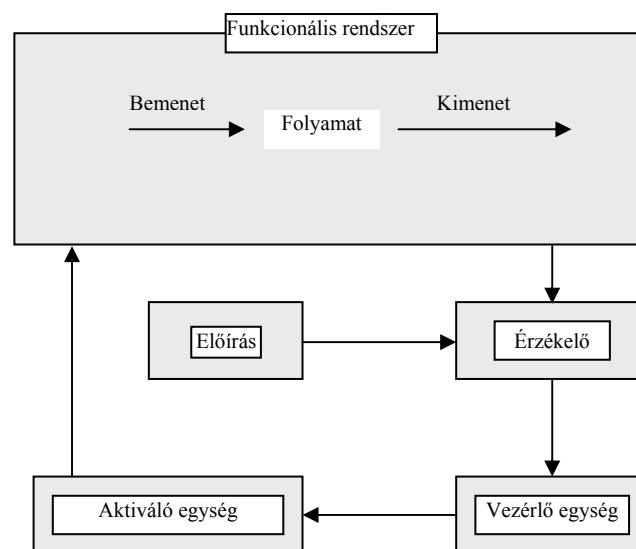
Információrendszer készítésénél általában az első lépés egy absztrakt rendszer létrehozása, amely egy szervezet bizonyos részét modellezi. A következő lépésekben ezt az induló modellt bővítik, és ily módon modellek sorozatát hozzák létre, amelyek egyre pontosabban írják le a kívánt fizikai rendszert. (Más mérnöki tudományokban is hasonló elvek szerint járnak el, például hídépítésnél vagy épületek tervezésénél; az absztrakt rendszer tartalma ábrázolja a kívánt fizikai rendszert.)

A nyílt rendszerek

A nyílt rendszer a rendszerek legáltalánosabb típusa, ezt mutatjuk be a 10.5 ábrán. Vagyis egy nyílt rendszer folyamatokból áll, amelyek a környezetükből valamilyen bemeneteket (input) kapnak, és kimeneteket (output) állítanak elő. Természetesen közben eredmények is képződhetnek a folyamatok egyes szakaszaiban. A rendszer határa választja el a rendszert magát a környezetétől, és tulajdonképpen a folyamatok, a bemenetek és a kimenetek határozzák meg. A bemeneteket és kimeneteket a nyílt rendszer statikus, míg a folyamatokat a rendszer dinamikus elemének tekintjük. Például egy cég iktatási és postázási rendszere nyílt rendszer, és egy információrendszer is az.

A kommunikáló részrendszerek

Egy nagy rendszer leírását egyszerűsíthetjük azzal, hogy részrendszerekre bontjuk, de ezért komoly árat kell fizetnünk, nevezetesen meg kell határoznunk a köztük fennálló kapcsolatokat, felületeket (interfész), amelyeken keresztül a kommunikáció lezajlik. Egy felület alatt azokat a bemeneteket és kimeneteket értjük, amelyeket egynél több rendszer közösen használ. Nyilvánvaló, hogy a bemenetek/kimenetek leírásának az összes, az adott kommunikációban részt vevő rendszerben úgy kell létezniük, hogy a kommunikáció zavartalanul folyjon. Ennek szükséges feltétele a kompatibilitás, és elégséges feltétele az azonosság. Ez különösen fontos számítógéprendszerek, -folyamatok és -programok esetén.



10.5. ábra: A vezérlő rendszer részei és a funkcionális rendszer

A részrendszerek csatolása

Amikor a rendszerek kölcsönhatását részletesebben vizsgáljuk, akkor időnként azt találhatjuk, hogy az egyik rendszer kimenetét egy másik rendszer azonnal felhasználja. Az ilyen eseteket nevezzük szoros csatolásnak. Vannak esetek, amikor ez egyáltalán nem kívánatos, mert például a fogadó rendszer csak bizonyos diszkrét állapotokban vagy bizonyos időpontokban tudja a saját bemenetét (a másik kimenetét) feldolgozni. Egy másik ok lehet a két feldolgozási folyamat sebességének különbsége.

Ekkor egy lehetséges megoldás az, ha a két rendszer szoros kapcsolatát megszüntetjük, a két rendszert szétkapcsoljuk, azaz a két rendszer egymástól függetlenül működik, legalábbis egy ideig. Ezt úgy valósíthatjuk meg, hogy buffert használunk, ahol az első rendszer kimenete várakozik addig, amíg fel nem használják.

Determinisztikus rendszerek

Bizonyos rendszerek viselkedését meg tudjuk jósolni, azaz egy adott bemenetre tudjuk, hogy milyen kimenetet fog a rendszer előállítani. Létezik tehát egy szabály, amely a bemenetet a kimenethez rendeli. Az ilyen rendszerek általában egyszerű folyamatokból állnak, és ahhoz, hogy a rendszer célja megvalósuljon, elegendő a megfelelő bemenetről gondoskodni.

Nem determinisztikus rendszerek

Nem determinisztikus vagy sztochasztikus rendszerek viselkedése előre nem jósolható meg, azaz nem tudjuk azt, hogy hogyan idézhetnénk elő a rendszer céljának megvalósulását, sőt még azt sem tudjuk, hogy egy adott bemenetre milyen kimenetet fog előállítani. Ez a helyzet vagy nagyon bonyolult folyamatoknak a következménye, vagy nagyon kevésbé megértetteknek, vagy ismeretlen, esetleg meghatározhatatlan bemeneteknek vagy ilyen bemenetek kombinációjának. Például ilyen rendszer az időjárás, Magyarország gazdasága stb.

Rendszerek viselkedésének kézben tartása

Ha azt akarjuk garantálni, hogy a rendszerek céljai megvalósuljanak, folyamatosan ellenőriznünk kell a működésüket. A bemenetek lehet, hogy pontatlanok, vagy a kevésbé megértett folyamatok között összeütközések merülnek fel, ezért a rendszerhez hozzá kell kapcsolnunk egy vezérlő rendszert, amelyik ellenőrzi az eredeti rendszert (ezt most funkcionális rendszernek hívhatjuk) vajon helyesen viselkedik, működik-e.

Kimenetellenőrzés

Valamilyen eszközzel észlelik a rendszer kimenetét és összevetik valamilyen előre rögzített előírással. Bármilyen eltérés egy helyreigazítási tevékenységet, korrekciót indít el, amelyet a rendszer bemenetként kap meg, ezzel működésbe hozva a funkcionális rendszert, és az előírásokhoz közelebb álló kimenetet generál. A vezérlő rendszer azonban megváltoztathat folyamatokat, önszervező (adaptív) rendszereknél megváltoztathatja a rendszer célját.

Negatív visszacsatolás azt jelenti, hogyha a rendszer kimenete eltér az előírttól (valamilyen mintavételezés során), akkor a vezérlő rendszer megpróbálja az eltérést csökkenteni, a kimenetet az előírásához közelíteni. A pozitív visszacsatolás ennek az ellenkezőjét jelenti - ha a kimenet, annak legfontosabb jellemzői eltérnek az előírttól, akkor a rendszer megismétli az eljárást, ezzel tovább növelve az előírttól az eltérést.

Bemenetellenőrzés

A vezérlő rendszer gyakran ellenőrzi a rendszer bemenetét ugyanúgy, ahogy a kimenetét. Ezt szokták szűrésnek hívni.

A vezérlés nehézségei

A vezérlő rendszert úgy tekinthetjük, mint olyan rendszert, amely csökkenti a rendszer bizonytalansági fokát. Azonban egy bonyolult rendszer vezérlése gondot okozhat, ugyanis leegyszerűsítve, minden lehetséges rendszerállapothoz tartoznia kell egy vezérlési állapotnak. Ráadásul a rendszerelemektől vezérlési információkat kell kapni és továbbítani nekik, ami lényegesen megnöveli a járulékos információfeldolgozási feladatokat. Egy viszonylag nyitott vagy nem megjósolható bemenetekkel dolgozó rendszer állapotainak meghatározása gyakorlatilag lehetetlen. Ilyen esetekben a gyakori megoldás egy ember-számítógép hibrid rendszer létrehozása, amelyben a számítógéprendszer reagál az előre megállapítható esetekre, míg az ember a nem várt esetekben hoz döntéseket.

Az állapottér

Az állapottér fogalma a rendszerek viselkedésének leírásához kapcsolódó kiegészítő fogalom, amely a folyamatokat és egyéb elemeket (objektumok, entitások stb.) is figyelembe veszi. Ekkor a rendszert úgy fogjuk fel, hogy bármilyen időpillanatban egy adott állapotban van, és ezt az állapotot azok az elemek határozzák meg vagy jellemzik, amelyek ebben az adott időpontban a rendszeren belül léteznek. Ezek az elemek nem csak a fizikai valóságban létezők lehetnek, hanem például egy adott személy kora is lehet egy lényeges tényező/elem/objektum az adott rendszerben.

Ezt a megközelítést csak az ún. diszkrét rendszereknél lehet alkalmazni, azoknál, amelyeknek megkülönböztethetően azonosítható állapotaik vannak. Az információrendszerek diszkrét rendszerek.

Az aktualizálási folyamatok a folyamatok fontos típusát alkotják, ebben a megközelítésben olyan transzformációt jelentenek, amelyek a rendszert az egyik állapotból a másikba viszik.

A rendszerszemléletű megközelítés hátrányai

- o A valós világ leírása különböző nem egyértelmű leírása lehet az eredménye ennek a megközelítésnek: a módszer eltérő alkalmazása különböző eredményekre vezethet.
- o A létrejött modell nem teljes vagy nem pontos, merthogy nincs pontos előírás arra, hogy milyen részletességűnek kell lennie.
- o Nincs kifejezetten javasolt módszer vagy általános egyetértés abban, hogy az esetleg létező módszerek közül melyik a legmegfelelőbb.

- o „Az oroszlán és a ló rendszerszemléletű megközelítésben ugyanaz, mégis egy kicsit másképpen kell velük bánni.”

A rendszerszemléletű megközelítés előnyei

- o A rendszer fogalmának fentebbi meghatározásainak megfelelően közelítjük meg a valós világot vagy annak egy részét.
- o Így ez a megközelítés formális eszközt nyújt a helyzet megértéséhez és leírásához az ösztönösen ismert fogalmak segítségével, mint például a folyamat, bemenet, kimenet.
- o A folyamatok hierarchikus lebontása (dekompozíciója) a bonyolult, összetett folyamatok elemzésének hasznos eszköze.
- o A vezérlő rendszer fogalma illetve ennek fontossága rámutathat arra, hogy egy ilyen hiányzik, nem hatékony, esetleg eredménytelen.
- o A rendszer ilyen jellegű leírása általában elég egyszerű ahhoz, hogy kommunikációs eszközként használjuk a személyek közötti párbeszéd lebonyolításához, természetesen megfelelő magyarázat kíséretében.

10.2. Információs rendszer fogalma

Az információs rendszer definiálására több megfogalmazás született. Burt és Kinnucan a következő megfogalmazást adta:

Az információs rendszer úgy tekinthető, mint információ forrás(ok) bármilyen kombinációban azok bármilyen elérésével és vagy bármilyen visszanyerésével azok manipulálására vagy használatára. Bármilyen információs rendszer célja összekapcsolni a felhasználót egy olyan megfelelő információs forrással, melyre a felhasználónak aktuálisan szüksége van, azzal az elvárással, hogy a felhasználó képes lesz elérni az információt, mely meg fog felelni az igényeinek.

Davis és Olson definíciója szerint egy vezetői információs rendszer

- egy integrált felhasználó - gép rendszer
- információ szolgáltatására
- tevékenységek, vezetés, elemzés, és döntés-előkészítési funkciók támogatására
- a szervezetben.

A rendszer

- számítógép hardvert és szoftvert
- manuális eljárásokat
- modelleket az elemzéshez, tervezéshez, ellenőrzéshez és döntés-előkészítéshez és
- adatbázisokat

használ.

Vezetői információs rendszer. A vezetést támogató számítógép alkalmazások száma növekszik. A mikroszámítógépek forradalma lehetővé teszi, hogy sok vezető asztalára számítógép kerüljön, számos adatbázis elérhetővé válik. Sok magán és közszolgálati szervezet

nem nélkülözheti a számítógépes elemzést a döntéseiben. A hardver és szoftver költség csökken, ugyanakkor az információs rendszerek lehetőségei folyamatosan növekednek.

Azonban, mindezeknek a technológiai fejlődéseknek az ellenére sok vezető egyáltalán nem használ számítógépet, vagy elsősorban egyszerű döntések támogatásához használja. A döntéstámogató rendszereket (Decision Support Systems - DSS), felsővezetői információs rendszereket (Executive Information Systems - EIS) és szakértői rendszereket (Expert Systems -ES) azért tervezték, hogy megváltoztassák ezt a helyzetet.

Ezeket a technológiákat Turban együttesen vezetésként (menedzsmentet) támogató rendszereknek (Management Support Systems - MSS) nevezte.

A vezetésként támogató rendszerek. A Fortune magazinban megjelent 1984-es felmérés szerint az USA-ban 500 vállalat felső vezetőjének 1/3-a használt számítógépet kritikus döntés-előkészítésekhez. Ezeknek körülbelül 1/4-e használt otthon is számítógépet. Ez a szám 1989-re 21 %-ra nőtt. Ez a növekvő szám csak egy jel arra vonatkozólag, hogy valóban információs korszakban vagyunk. Ügyviteli alkalmazottak és műszakiak már több mint 30 éve használnak számítógépeket rutinmunkájuk támogatásához. 1982-vel a helyzet a döntés-előkészítés támogatásában is változást hozott. A mikroszámítógépek lehetősége és a relatívan könnyen használható szoftverek megjelenése kikényszerítette a változást.

A számítógép alkalmazások a tranzakció feldolgozástól és figyelő tevékenységtől a feladat elemzés és feladat megoldási alkalmazások irányába tolódtak el a 80-as években. Szintén egy fejlődési tendencia, hogy a vezetők részére integrált programcsomagot adjanak, amelyek segítik őket a legfontosabb feladatban a döntés-előkészítési munkában.

A számítógépre alapozott technikákat a vezetői döntés-előkészítés hatékonyságának növelése érdekében fejlesztik, különösen a komplex feladatok megoldására.

A döntéstámogató rendszerek készítése több mint 20 évre nyúlik vissza. A szakértői rendszerek kereskedelmi forgalomban való elterjedése az 1980-as évekre tehető, amely a számítógépes döntés-előkészítésben az egyik legfontosabb szerepet játszhatja a jövőben. A felső vezetői információs rendszereket most a felső vezetők munkájának támogatására tervezik. Ezek a technológiák mint független rendszerek, de esetenként azonban előfordul, hogy integrálva jelennek meg.

A vezetésként döntés-előkészítő és a vezetői információs rendszerek. A vezetés (menedzsment) egy folyamat, amelyben bizonyos célokat érünk el *erőforrások* (ember, pénz, energia, anyag, hely, idő) felhasználásával. Ezeket az erőforrásokat *inputoknak* és a *célok elérését* a folyamat *outputjának* tekintjük. A vezetői munka sikerének a mértékét gyakran az output és input közötti aránnyal mérjük. Ez az arány a szervezet termelékenységét fejezi ki.

$$\text{termelékenység} = \frac{\text{output(termékek, szolgáltatások)}}{\text{input (erőforrás ráfordítások)}}$$

A termelékenység egy fő jellemzője bármilyen szervezetnek, mert meghatározza a szervezet és tagjainak jólétét.

A termelékenység szintje vagy a vezetés sikere bizonyos vezetéskénti funkciók végrehajtásától függ, mint pl. tervezés, szervezés, irányítás és ellenőrzés. Hogy ezeket a funkciókat végrehajtsák, a vezetők érdekeltek a döntés-előkészítési tevékenységek állandó és ismétlődő folytatásában.

Az összes vezetői tevékenység a döntés-előkészítés körül forog. A vezető mindenekelőtt egy döntéshozó, mivel minden vezető valójában részt vesz egy szervezet döntés-előkészítési tevékenységének egy bizonyos részében.

Évekkel ezelőtt a vezetők a döntés-előkészítést egyszerűen művészetnek tekintették, így ezek a vezetői stílusok gyakran a kreativitáson, megítélő képességen, intuíción és a gyakorlaton alapultak, nem pedig a szisztematikus tudományos megközelítésű kvantitatív módszereken.

Azonban a környezet, amelyben a vezetésnek dolgoznia kell, változik. Mivel az információ és a mikroelektronika korszakában élünk, ezért ez a technológia életstílusunk meghatározójává válik. Az üzleti élet és környezete sokkal komplexebb ma, mint volt bármikor ezelőtt, ráadásul a fejlődési irány a komplexitás irányába mutat.

A döntés és döntés-előkészítés Turban szerint három ok miatt nehezebb, mint korábban. *Először*, a lehetséges alternatívák száma sokkal nagyobb ma, mint volt bármikor ezelőtt a fejlettebb technológiai és kommunikációs rendszereknek köszönhetően. *Másodszor* a döntések jövőbeni következményeit nehezebb megítélni a megnövekedett bizonytalansági tényezők miatt. *Végül*, a hibaelkövetés költsége nagyon nagy lehet a tevékenységek és az automatizálás komplexitásának és terjedelmének következtében. Ráadásul a komplexitásból fakadó láncreakciók miatt egyetlen hiba elkövetése számos más hibát okozhat a szervezet különböző részeiben.

A vezetőknek ezért sokkal hatékonyabbá, kifinomultabbá kell válni, meg kell tanulniuk, hogyan kell használni az új eszközöket, technikákat, amelyeket az ő területükre fejlesztenek. Ezen technikák közül sokban egy kvantitatív elemzési megközelítést használnak, amelyek egy diszciplínába csoportosítva vezetéstudománynak hívnak (Turban és Meredith).

A vezetési információs rendszer (Management Information System - MIS) egy szabályszerű, számítógépre alapozott rendszer, amelyet a különböző forrásból származó adatok visszanyerésére, csoportosítására és integrálására terveztek, hogy időben nyújtsa a szükséges információt a vezetői döntés-előkészítéshez. A MIS legsikeresebb a rutin, strukturált és megelőző típusú döntésekhez szükséges információnyújtásban. Továbbá sikeresen alkalmaznak nagy mennyiségű, részletes adat tárolására és lekérdezésére a tranzakció feldolgozásokban. A MIS kevésbé hatékony a komplex döntési szituációk támogatásában. Ez a komplex típusú döntések támogatásának hiánya miatt van, valamint azért, mert a MIS-t hagyományos módon nem könnyű fejleszteni, és a vezetőknek nem egyszerű (nem könnyű) használni.

A döntéstámogatás. Simon szerint a döntés-előkészítési folyamatok egy összefüggő sorozatba esnek, amely a nagymértékben strukturált (gyakran programozottnak nevezett) döntésektől a nagymértékben strukturálatlan (nem programozott) döntésekig terjednek. *A strukturált folyamatok* a rutin, ismétlődő feladatok, amelyekre kidolgozott megoldások léteznek. *Nem strukturált* feladatok, amelyekre nincsenek kidolgozott megoldások.

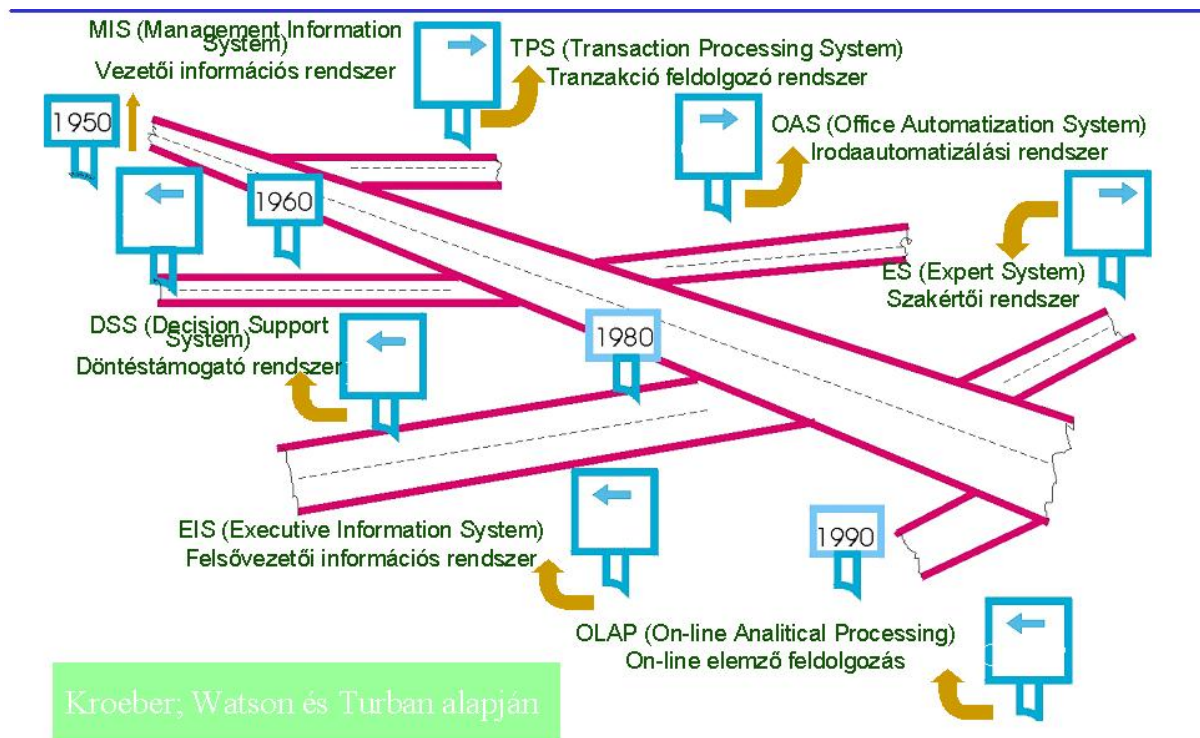
A döntés-előkészítés vizsgálatánál szintén szükséges az emberi döntés-előkészítési folyamat megértése. Ez a folyamat Simon szerint három fázisra osztható :

• a döntéshez szükséges feltételek keresése	Intelligencia,
• a lehetséges tevékenységek felderítése, kidolgozása, elemzése	Tervezés,
• a lehetséges akcióprogram kiválasztása	Választás.

A strukturált feladat az, amelyben ezek a fázisok szabványosítottak (kidolgozott módszerek), a célok tiszták és az input-output egyértelműen specifikált. A nem strukturált

feladatban a három fázis közül egyik sem strukturált. Gorry és Scott-Morton azt a feladatot, ahol a fázisok némelyike strukturált, részben strukturáltnak nevezi.

A számítógépeket már több mint két évtizede használják eszközként a vezetői döntés-előkészítés támogatására. A számítógépesített eszközök vagy döntési segítségek Kroeber és Watson szerint hat kategóriába sorolhatók.



10.6. ábra: Információs rendszerek fejlődési útja

E rendszerek (számítógépre alapozott információs rendszerek) fejlődési útját az 1.sz ábra szemlélteti. A TPS, MIS, DSS, EIS és ES attribútumai néhány dimenzió szerint csoportosíthatók, amelyek közül a legjellemzőbbeket az 10.1. táblázat foglal össze. A különböző kategóriák közötti kapcsolatok a következők:

- Mind az öt kategória az információ technológia egyedi osztályainak tekinthető.
- Ezek kapcsolódnak egymáshoz és mindegyik támogatja a vezetői döntés-előkészítés néhány tevékenységét.
- Az újabb eszközök fejlődése és létrehozása segíti az információ technológia szerepének kiterjesztését a vezetői hatékonyság növelés érdekében.
- A kölcsönös kapcsolatok és koordináció ezek között az eszközök között még fejlődik. Még sok tanulnivaló marad és újabb elméletek szükségesek a további fejlődéshez.

10.1. táblázat Számítógépesített rendszerek attribútumai [Forrás : Turban]

Dimenzió	Tranzakció feldolgozó rendszerek	Vezetői információs rendszerek	Döntés-támogató rendszerek	Szakértői rendszerek	Felső-vezetői információs rendszerek
Alkalmazások	Bérfeldolgozás, készlet, nyilvántartás, termelési és értékesítési információk	Termelés ellenőrzés, értékesítés előrejelzés, megfigyelés	Hosszú távú stratégiai tervezés, komplex integrált feladatterületek	Diagnózisok, stratégiai tervezés, belső ellenőrzés tervezés, stratégiák karbantartása. a. Szűk témakör	Felső vezetői döntés-támogatás, környezeti vizsgálat.
Fókusz	Adat tranzakciók	Információk	Döntések, rugalmasság, felhasználó barátság	Következetesség, szaktudás átadása	Követés, ellenőrzés
Adatbázis	Egyedi minden alkalmazásban, "batch update"	Interaktív elérés programozók számára	Adatbázis kezelő rendszerek, interaktív elérés, tárgyi ismeret	Eljárás és tárgyterület ismeret, tudásbázis (tények és szabályok)	Külső (online) és közös
Döntési képességek	Nincs döntés vagy egyszerű döntési modell	Strukturált rutin problémák hagyományos operációkutatási eszközök használata	Részben strukturált problémák, integrált operációkutatási modellek	A rendszer komplex döntéseket készít, nem strukturált, szabályok használata (heurisztika)	Nincs
Művelet	Numerikus	Numerikus	Numerikus	Szimbolikus	Numerikus (főleg) bizonyos szimbolikus
Információk típusa	Összefoglaló jelentések, részadatok	Tervezett és igényelt jelentések, strukturált folyamat, kivételek szerinti jelentések	Meghatározott döntéseket támogató információk	Javaslat és magyarázat	Állapot elérés, kivételek szerinti jelentések, kulcs jelzések
Legmagasabb szervezeti szint kiszolgálása	Alsó szintű vezetés	Középvezetés	Felső vezetés	Felső vezetés és specialisták	Legfelső vezetés (csak)
Hajtóerő, késztetés	Hasznosság	Hatékonyság	Eredményesség	Eredményesség és elvárás	Időszerűség

10.3. Vezetői tevékenységek és adatszükségletek

Az információ fogalma alatt azokat a híreket, ismereteket értjük, amelyek a valóságra vonatkozó új tényeket és elképzeléseket közvetítenek számunkra. A hírek akkor van információtartalma a hír fogadója számára, ha felfogja, megérti a közleményben foglaltakat. Az érthetőségnek viszont az feltétele, hogy a közlés észlelhető (pl. hallható, papíron látható) legyen, továbbá a címzett számára ismert jelölési (szintaktikai) és értelmezési (szemantikai) szabályok szerint történjen.

Az információs tevékenység szerepe abban áll, hogy a vállalati rendszer különböző szintű vezetőihez, végrehajtó egységeihez olyan információk jussanak el, amelyek azoknál ésszerű döntést, cselekvést, vagyis a rendszer céljával összhangban álló, előrelátható reakciót váltanak ki. Az eredményes kommunikációnak az információáramlás csak szükséges, de nem elégséges feltétele. Hatékony kommunikáció csak akkor jön létre, ha a címzett az információt felveszi, feldolgozza és arra reagál. Különös jelentősége van annak, hogy az információ tartalma milyen hatással van a címzettre, mivel ez összefüggésben áll az információ hasznosságával, vagyis a címzett viselkedésére és célfüggvényére gyakorolt hatással. Az információkat tartalmi jellemzőik alapján szemantikus, pragmatikus és motivációs információknak tekinthetjük.

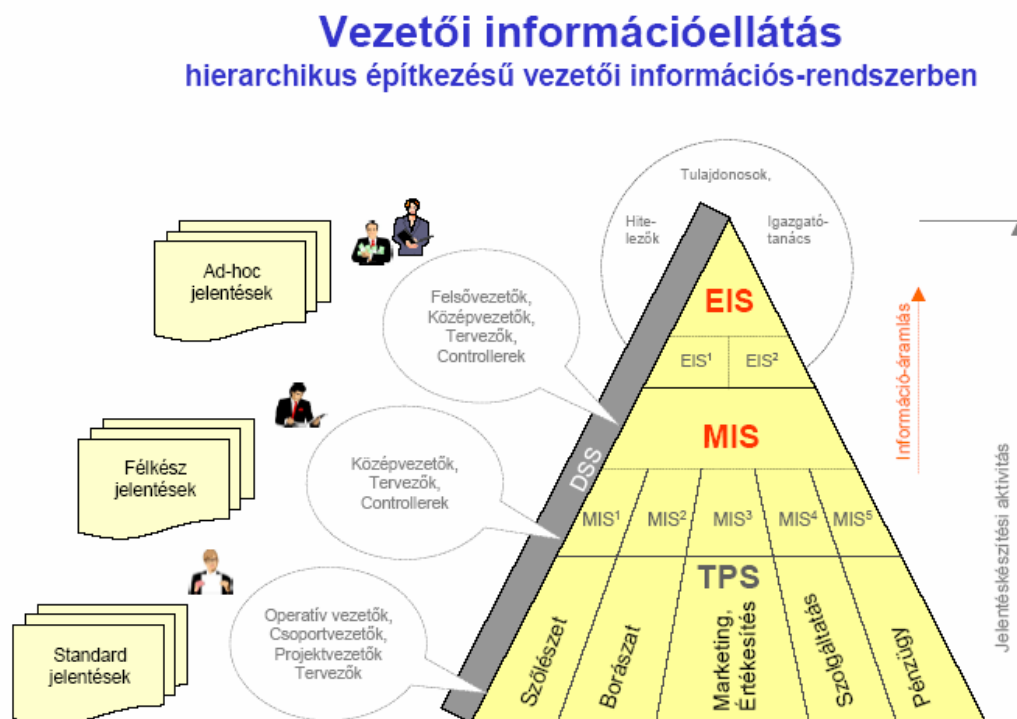
A szemantikus információ útján a fogadó tényekről értesül, ezáltal tájékozottsága nő, választási lehetőségei javulnak. A pragmatikus információk a közlő érdekében álló cselekvés végrehajtását írják elő a címzettnek. Ekkor utasításokról, munkavégzési szabályzatokról, rendelkezésekről, tiltásokról van szó. Ilyen információkat a döntéshozók állítanak elő. A motivációs információk pedig a címzett értékrendszerét, preferenciáit befolyásolják. Ilyenek például az anyagi ösztönzés szabályai, vagy a dolgozóknak a vállalati célokkal való azonosulást kiváltó intézkedések.

A **döntéshozatal** a vezetői funkciók egyik legfontosabb, legnehezebb és legkockázatosabb része. A helytelen döntések helyrehozhatatlan következményekkel járhatnak. A rossz döntéseknek több oka lehet. Egyrészt a döntési folyamatban - pl: a döntéselőkészítő tevékenységekben -, másrészt magában a döntéshozóban rejlik.

A **döntési problémák** meghatározása, elemzése, strukturálása a tervezők fontos feladata. A döntési probléma megoldása, a számszerű eredmények kidolgozása a tervezési rendszer működésének egyik fő funkciója. Az absztrakt döntési probléma az információ felhasználásával, valamint a megoldási szabályok, módszerek alkalmazásával válik a rendszer reál elemévé. A tervezési rendszer elemeit áttekintve megállapíthatjuk, hogy azok tartalmazzanak absztrakt elemeket is: *döntési problémák, megoldási módszerek, szabályok*. A tervezési rendszer olyan sajátos rendszer, amely *anyagi és nem anyagi elemekből épül fel*. A tervezők alapvető feladata, hogy a vállalati gazdálkodás legkülönbözőbb területeit illetően választ adjanak a mit és hogyan kérdésekre. A tervezés tárgyát egy vállalat esetében rendkívül sokféle típusú, nagyságrendű, bonyolultságú döntések halmaza képezi. Valamennyi döntési problémát, mint a tervezés tárgyát meghatározni természetesen nem tudjuk. A vállalat működési modelljére támaszkodva *a tervezés tárgyát képező alapvető döntési problémákat az alábbiak szerint rendszerezjük*: piaci, kereskedelmi, termelési és/vagy szolgáltatási, gazdálkodással, termelési tényezőkkel, és pénzügyekkel kapcsolatos döntési problémák. Hangsúlyozzuk itt a probléma *döntésorientált* megközelítését, *a folyamatszemplétű elemzésre később kerül sor*.

Ahhoz azonban, hogy ki tudjuk elégíteni a vezetői információigényeket, ismernünk kell a vezetői tevékenységek széles körét és azok adatszükségletét. A vezetési funkciókat, a szükséges adatokat, és az adatok forrását tartalmazza a 2. Táblázat. A vezetői

tevékenységeknek egy köre, a vállalati belső és külső változásokkal összhangban állandóan változik, a hangsúlyok eltolódnak.



10.7 ábra: Vezetői információellátás

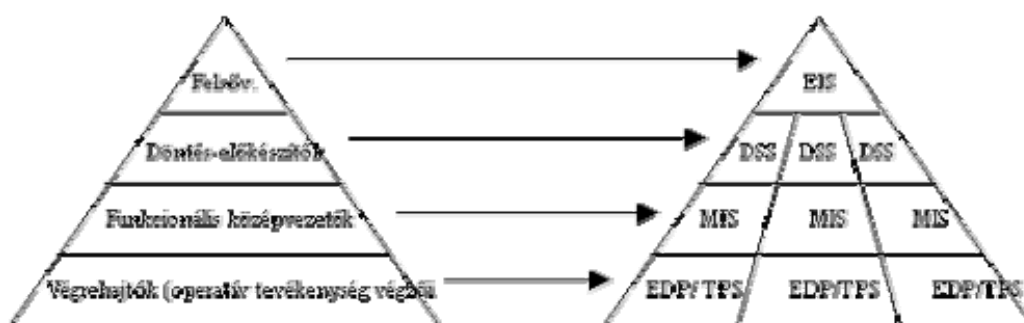
10.4. Vállalati információs rendszerek

A 70-es évekre egyes operatív tevékenységek automatizálását hatékonyan tudták megoldani a vállalatok, az adatok nagy tömegének viszonylag gyors és megbízható rögzítésére, tárolására és kinyerésére alkalmas számítógépes rendszerek felkeltették a vezetők érdeklődését. Érthető módon igényt támasztottak arra, hogy ebből a létező adattömegből viszonyaik, elemzésük révén értelemmel, jelentéssel bíró hasznos adatokat, vagyis számukra hasznos információt nyerjenek ki. A vezetők feladata és felelőssége, hogy döntéseik révén előmozdítsák a szervezet fejlődését, s elfogadható az az érv, hogy ezen döntések minősége javítható, ha megfelelőbb információs háttérre épül. Az automatizálás rendszerei így váltak a menedzseri munkát támogató információrendszerekké.

Az első próbálkozások kudarcaiért az a hibás elképzelés okolható, hogy a szervezet vezető beosztású alkalmazottainak információval való ellátása azt jelenti, hogy a korábbi tranzakciós rendszerek teljes adattömegét hozzáférhetővé tesszük számukra. A vezetői döntéseket azonban mindenkor a döntés meghozatalához szükséges és megfelelő szintű / aggregáltságú információval kell támogatni, vagyis a vezetői információs rendszerek kialakítása együtt járt a vezetői információigény feltérképezésével. Itt kapcsolódik be a controlling. Akár a vezetést támogató eszközrendszerként, akár vezetési funkcióként tekintünk a controllingra, mindenképpen hangsúlyos a vállalatirányítás kérdéseit átfogó információszolgáltató és döntéstámogató jellege, valamint az, hogy figyelembe vegyük a különböző vezetői szintekhez kapcsolódó különböző controlling-feladatok sajátosságait.

A fejlesztések a 80-as években olyan új alkalmazásokat hívtak életre, mint az elsődlegesen középfelvezetői szint információigényét kielégítő, jellemzően egy-egy funkcionális területre kialakított, rögzített formátumú jelentéseket biztosító vezetői információs rendszerek (MIS: Management Information System). Az elemzők és döntés-előkészítők statisztikai, modellező, szimulációs munkáját támogatják az úgynevezett döntéstámogató rendszerek (DSS: Decision Support System), melyek szintén ekkortájt alakultak ki -- egy magasabb funkcionális, megjelenítési és kommunikációs szintet képviselve. A vállalati tranzakciós rendszerekből kinyerhető adatokat azonban egészen felsővezetői szintig igyekeztek hasznosítani, így alakultak ki az operatív rendszerekre épülő felsővezetői információs rendszerek (EIS: Executive Information System), melyek személyre szabottan, jelentős hardver és támogató apparátus igényel nyújtottak főként a múltat vonatkozó, aggregált információkat a vállalatvezetésnek.

A 80-as évek vállalati információrendszereit jól szemlélteti a 10.8. ábra.



10.8 ábra: Tipikus vállalati információrendszer a 80-as évek második felében

A kialakított információrendszereknek több szempontból sem működtek tökéletesen:

Az alsó szint alkalmazásai szigetszerűek voltak, külön fejlesztésű vagy különböző szállítóktól vásárolt szoftvereknek kellett volna kommunikálniuk egymással és együttműködniük, hogy adatokat küldjenek "felfelé". Ezt nem, vagy csak túlzottan nagy ráfordításokkal tudták megoldani a cégek informatikusai. A létrehozott vezetői alkalmazások is szigetszerűvé váltak, nem tudtak összekapcsolódni és adatokat cserélni összvállalati szinten.

Másrészt a tranzakciós rendszerek adatbázisai eredeti feladatuknak megfelelően más elvárásokhoz lettek optimalizálva, ezért még egy-egy tranzakciós rendszer (például: csak az értékesítés, vagy a raktárnyilvántartás) szintjén is lassú és bonyolult volt az összetettebb lekérdezések megválaszolása.

A tranzakciós rendszerek (és a későbbiekben bemutatásra kerülő ERP-rendszerek is) többnyire úgynevezett relációs adatbázisokra épülnek. Az elmúlt mintegy fél évszázadban különböző elképzelések, adatmodellek születtek arra vonatkozóan, hogy miként ragadja meg az informatika a való világ tényeit, változásait. A manapság leggyakrabban használt relációs adatmodell lényege, hogy a tárolandó adatok logikai struktúráját figyelembe véve kétdimenziós táblázatokat (relációkat) alakítanak ki. Például egy tábla mezőiben rögzíthetik a cég termékeinek adatait. Az oszlopok fejlécei a következők lehetnek: cikkszám, terméknév, szín, ár, stb. A sorokban tárolják az adatok összetartozó csoportjait, a rekordokat, az oszlopok pedig a különböző kategóriákat, tulajdonságokat reprezentálják. Egyszerűsített példánk szerint:

			Termék tábla
Cikkszám	Terméknév	Szín	Ár
0011	Asztal	Barna	10000
0012	Asztal	Szürke	11000
0021	Szék	Kék	6000
0031	Szekrény	Barna	18000

Egy másik táblázat tartalmazhatja az értékesítés bevételeit -- régiókra és termékekre bontva.

		Értékesítési tábla
Cikkszám	Bolt	Árbevétel
0011	Budapest/1	150000
0012	Budapest/1	33000
0021	Budapest/1	6000
0031	Vas megye	72000

Látható, hogy az "Azonosító" mint kulcs révén a két táblázat összekapcsolható, belőlük további lekérdezések (nézetek -- view-k), vagy állandó, származtatott táblázatok (úgynevezett pillanatfelvételek -- snapshot-ok) hozhatóak létre.

Természetesen a gyakorlatban csak több, nagyméretű táblázatba fér bele egy vállalatnál a tárolni kívánt adatmennyiség, és igen bonyolult lehet a táblák közötti kapcsolatrendszer. Ennek modellezése komoly feladat, de ez a fajta adattárolás matematikailag és informatikailag is egyaránt biztos alapokon nyugszik. Ugyanakkor a cégek adatállománya több terrabájtnyi lehet, ami további tárolási, frissítési, védelmi és optimalizációs problémákat vet fel.

A controlling szempontjából fontos lekérdezések azonban nehezen futtathatóak ezeken a relációs adatbázisokon, mert gyakran külön programozást igényelnek: a táblázatok több szempont szerinti többszöri végigkeresése hosszú időt vesz igénybe, s sokszor a kért relációk, táblák nem is alakíthatóak ki a hiányzó kapcsolatok, vagyis a hiányzó közös azonosítók miatt (példánkban a cikkszám töltötte be az azonosító szerepét). Másrészt a lekérdezésekhez általában aggregált adatokat is kell számolni, ami további hosszú időt vesz igénybe ilyen óriási adattömegeknél. Végezetül ki kell emelni azt, hogy az így kialakított vezetői információs rendszerekben kevés a vezető mozgási szabadsága és hiányzik belőlük az átfogó, stratégiai szemléletnek megfelelő információkinyerés lehetősége, ezért sem válhattak a menedzserek és a controllerek igazi segédeszközeivé.

Amennyiben a kialakított vezetői információs rendszerekben készültek is beszámolók, a nyújtott információ sokszor mégsem eredményezett helyes döntéseket. Ennek oka lehetett a fentiek mellett az is, hogy csupán a vállalatban belüli, vagy akár csak néhány területről származó információ alapján döntöttek, illetve, hogy nem a megfelelő szinten jelent meg az információ, s született meg a döntés. -- Vagyis: mint minden területen, itt is szükség volt egyfajta tanulási időszakra, hogy pontosabb elvárásokat és jobb válaszokat fogalmazhassanak meg.

Az információrendszerek tehát fejlődésük első évtizedeiben megvalósították a vállalatban belüli adatkezelés automatizálását, valamint több-kevesebb sikerrel kísérletet tettek

a tárolt adatmennyiségből döntéstámogató információk kinyerésére. A 90-es évek elején három komoly kihívással nézett szembe a vállalati informatika:

- egyrészt szükség volt egyössz vállalati integrált IT-környezet kialakítására, mely biztosítja az alkalmazások közötti kommunikációt, adatcserét, és amely nyitott a külső adatok, különböző formátumok bevonására is;
- másrészt hatékonyabbá kellett tenni a vezetők információellátását és döntéshozói igényeiknek megfelelően kialakított vezetői információs rendszerek kifejlesztése lett a cél;
- harmadrészt -- az előzőekkel szoros összefüggésben -- a funkcionális szemléleten túl a vállalat valódi működésének megjelenítése érdekében a szervezeti határokon átnyúló folyamatokkal, az értékteremtés dimenzióinak tudatos kezelésével kezdtek el foglalkozni.
- Az ezekre a kihívásokra adott válaszokat, IT-megoldásokat mutatják be a következő részek.

A vállalatok további területeit kapcsolták be az informatikába, s létrejöttek a manapság is fejlődő és terjedő ERP-rendszerek. Az első alkalmazások MRP II.-k továbbfejlesztéseként és kiterjesztéseként alakultak ki, innen származott a rövidítés is: ERP (Enterprise Resource Planning), vagyis vállalati erőforrás-tervezés. Bár még ma is a régi rövidítéssel (ERP) utalunk ezekre a vállalati rendszerekre, az elmúlt évek innovációi miatt már inkább tekinthetők integrált vállalatirányítási alkalmazásoknak (IEA: Integrated Enterprise Application). Az ERP-rendszerek kialakulásával mind a controlling, mind a menedzsment óriási információs bázishoz jutott, folyamatosan bővülő funkcionalitásukkal ezek a rendszerek jelentik ma a vállalatok informatikai támogatásának alapját.

Az ERP-rendszerek olyan modulokból felépülő alkalmazások, melyek szoftveres megoldást kínálnak a termelés, a logisztika, az értékesítés, az emberi erőforrás gazdálkodás és a pénzügyi elszámolás tranzakcióinak valós idejű, egységes és integrált kezelésére -- a szervezet funkcionális területeit és működési folyamatait lefedő, egységes és integrált vállalatirányítási rendszerek.

Az ERP-knél hagyományosan továbbra is megmaradt tehát a funkcionális szemlélet: az ilyen szoftverek modulokból épülnek fel, melyek megfeleltethetők a vállalat egyes funkcionális területeinek, például: logisztikai, termelésirányítási, értékesítési, számviteli, controlling, eszközgazdálkodási modulokat alakítottak ki.

Az ERP-rendszerek feladata tehát egyrészt a vállalatnál felmerülő tranzakciók kezelése, másrészt (a korai időszaktól kezdve) a vezetők számára információ nyújtása -- a tárolt összes vállalati adattömegből. Az ERP-technológia így ötvözte a TPS/EDP-, illetve a MIS-rendszereket, a középvezetői szint fölött azonban nem igazán alkalmas döntéstámogatásra, mivel a rendszerek óriásira növekedő, relációs adatbázisainál a korábbiakban bemutatottakhoz hasonló problémák merülnek fel. Középvezetői szinten máig működnek az ezekre az adatbázisokra épített, többnyire rögzített beszámolókat előállító alkalmazások, s míg nem jelent meg a vezetői információs rendszerek legújabb generációja, addig a közvetlenül az ERP-ből kinyert lekérdezések felsővezetői szinteken is használatosak voltak.

Nagyjából ekkortájt dőlt el az informatikában a közgazdaságtan egyik hagyományos kérdése: Make or Buy? (Gyártani vagy Megvásárolni?). Az informatikai szállítók standard szoftvereket alkottak, melyek egy általános vállalati modellből levezetve nyújtottak sokoldalú támogatási lehetőségeket az üzleti tevékenység különböző feladatainál. E standard szoftverek személyre -- vállalatra -- szabhatóak, ugyanakkor ez korlátot is jelent, hiszen a használat során

bizonyos mértékben a vállalati működést is kell a választott programhoz "igazítani". Ezt ellensúlyozza az, hogy az ERP-gyártók igyekeznek a szoftver-bevezetések során szerzett iparági tapasztalatokat beépíteni a rendszereikbe.

Az ERP rendszerek évek, s lassan évtizedek óta sikeresek azon célkitűzéseikben, hogy a vállalati tevékenység realizálásához és menedzseléséhez hatékony támogatást nyújtsanak, s ezáltal csökkenthetőek legyenek a szervezet működési költségei, valamint javuljon a belső integráció, az információáramlás és az együttműködés, ennek révén jobb döntések és magasabb színvonalú szervezeti teljesítmény legyen elérhető, ami tükröződik a vevőkör elégedettségében és a vállalat eredményességében.

Az egyre szaporodó vállalati informatikai projektek sikeressége nagyban múlott / múlik az emberi tényező helyes kezelésén: ez nem csak a döntések következetes képviselését, a felhasználók tájékoztatását és oktatását jelenti, hanem az információnak mint hatalmi forrásnak a kezelését is. Az új rendszerek megerősíthetik a régi hatalmi viszonyokat, de sokszor -- például a középvezetői szint helyett az alkalmazottak döntési jogkörének megnövelésével -- át is rajzolhatják azokat.

A 90-es években a vállalatszervezés és az informatika egymással szoros kapcsolatban, egyaránt dinamikusan fejlődött. A vállalatok értékteremtéséről alkotott kép gyökeresen átalakult: a struktúra és az irányítás mindenhatóságába vetett hitet új fogalmak váltották fel: az üzleti és a támogató folyamatok. A folyamatok radikális, vagy éppen fokozatos átszervezésében fontos pozíció jutott az informatikai rendszereknek is: a folyamatok támogatása, egyszerűbbé, hatékonyabbá tétele lett központi feladatuk. Nem csak a gyártási / szolgáltatási technológiára vonatkozott ez, hanem egyszersmind a vállalati számítógépes rendszerek összessége által nyújtott információ újraértékelését is jelentette.

A folyamatszemplélet hangsúlyozza, hogy a folyamatok a vállalati értékteremtés kulcskategóriái, melyek nem csak outputtal, de meghatározható belső vagy külső "vevő"-vel rendelkeznek, átlépik a funkciók, vagy akár az egész szervezet határait. A szerzők a vállalat értékláncán belül legtöbbször megkülönböztetik az alaptevékenységhez kötődő kulcs, vagy operatív folyamatokat, illetve a működtetési, támogató, vagy másként vezetési folyamatokat. A kontrolling és az informatika együttműködésével sok helyütt sikerült eredményesen adaptálni ezt az új szemléletet.

A vállalatok folyamatelvű megragadásához szükség volt a folyamatok műveletekre bontására. A folyamatszintek és dimenziók kialakításakor a kontrollerek vállalati működésre vonatkozó ismeretei nélkülözhetetlenek voltak. Csakhamar megjelentek a folyamatok mentén történő átszervezéseket dokumentáló és támogató folyamatmodellező programok. (például: Micrografx Flowcharter, Visio, Aris, stb.)

A folyamatok kontrollingját szolgálják operatív szinten a folyamatköltség-rendszerek, melyek bekerültek az ERP-k újabb változataiba (például: SAP CO moduljának ABC-része), de külön szoftverek formájában is kaphatóak a piacon (például: H&P Prozessmanager). A fejlődés következő fázisát jelzik a vállalati teljesítmény értékelésébe a folyamatokat is bevonó koncepciók, mint például a Balanced Scorecard, illetve ezek informatikai támogatása (erről részletesebben az Üzleti intelligencia-eszközök kapcsán lesz szó).

A vállalati folyamatok lefutásának megtervezésénél is segítséget jelenthetnek az iparág hasonló cégeinek működési mintái. Ez a fajta benchmark -- üzleti tartalom (business content) beépítése az ERP-rendszerekbe egyszerre hasznos alap, valamint folyamattervezési korlát, amennyiben a megvásárolható vállalati információs rendszerek csak bizonyos határokon belül alakíthatóak a vállalat sajátosságainak megfelelően.

Az átszervezések végrehajtása, illetve az eredmények értékelése szintén a kontrolling feladata volt. A beruházások és a projektek kontrollingja is rendelkezik informatikai támogatással: a személyi teljesítménynövelő szoftverek néhány formája (például: MS Projekt), illetve az ERP-rendszerek önálló moduljai említhetők itt (például SAP IM, illetve PS modulok).

A folyamatszegléslet jegyében új irányzatok indultak el ezekben az években, így az Ellátási lánc menedzsment (SCM: Supply Chain Management), illetve a Ügyfél-kapcsolat kezelés (CRM: Customer Relationship Management). Mindkettő azon a gondolon alapul, hogy a vállalati értéktérítés folyamata nem ér véget a vállalat határainál, valamint, hogy a vállalat csak akkor lehet sikeres, ha megfelelő módon tudja kielégíteni vevőkörenek igényeit. Ezért döntő jelentőségű a vevőkkel, illetve a beszállítókkal kialakított viszony -- értéklánc, illetve még inkább: ellátási lánc szintű optimalizációra van tehát szükség.

A levezethető kontrolling, logisztikai, értékesítési és marketing feladatok a hagyományos tranzakciókezelés mellett magas szintű, on-line, interaktív analitikai képességeket, testre szabható, biztonságos, gyors és a vezetői igényekre optimalizált jelentési formákat, tervezést és tényelemzést egyaránt lehetővé tevő alkalmazásokat igényelnek.

Ezeknek az elvárásoknak a hagyományos logisztikai és értékesítési célszoftverek, a klasszikus ERP-rendszerekbe integrált modulok csak kevésbé tudnak megfelelni. Új típusú alkalmazások hozták a megoldást: ebben az időben kezdtek terjedni az adattárházak, illetve az ezekre épülő, sokoldalú OLAP-rendszerű szoftverek. Ezekkel a technológiákkal újraértelmezték a vezetői döntések támogatását; az egyes szakterületek információigényét kielégítve új informatikai területet nyitottak. Ezeket a szoftvereket alkalmazták az SCM és a CRM támasztotta igények kielégítésére is.

A folyamat-újrászervezések keretében többnyire értékelték a cégek saját informatikai egységüket, rendszerüket is. Az információtechnológia széles körű alkalmazása ugyanis nem mindig eredményezett hatékonyságjavulást, a nagy IT befektetések gyakran nem hozták meg a várt üzleti sikereket. A 90-es években ezért újrafoglalták az üzleti informatikával kapcsolatos alapvetéseket:

- elsődleges cél a vállalat versenyképességének növelése -- ebben kulcsszerepet játszik a megfelelő információ eljuttatása a megfelelő döntéshozókhoz (vezetőkhez, alkalmazottakhoz);
- az informatika az üzleti stratégia (vagy adott esetben újrászervezési koncepció) szerves része, ugyanakkor alárendelt, s nem öncélú eleme, illetőleg az információtechnológiában rejlő lehetőségek feltárása és kiaknázása.

Mindezen változások megerősítették a kontrollingtól és az informatikától elvárt együttműködés szorosságát és fontosságát. A kontrollerek tevékeny részt vállaltak az üzleti folyamatokat támogató, a vezetés információigényét kielégíteni képes vállalati információrendszerek kialakításában, illetve maguk is folyamatosan alkalmazták ezeket tervezési, ellenőrzési és döntés-előkészítői munkájuk során. Az operatív kontrolling mellett ekkortájt kialakuló stratégiai kontrolling is épít az IT-alkalmazásokra, a modern ERP-rendszerek bázisán kialakított Üzleti intelligencia megoldások támogatására.

10.5. Ágazati információs rendszerek

Az EU agrárgazdaságának irányítása hatalmas mennyiségű, pontosan egyeztetett előírásoknak megfelelő információ szabályozott áramlásán alapszik. Az információs csatornák kölcsönösen összekötik a tagországokat a szervezet döntéshozó központjaival. Az ezen információk alapján hozott döntések komoly előnyöket, illetve súlyos hátrányokat jelenthetnek az érintett országoknak, így az adatok hitelességével, megbízhatóságával és összehasonlíthatóságával kapcsolatos követelmények betartása nem csak nagyon szigorú követelmény, hanem egyben elemi érdeke is a tagoknak és a belépni szándékozóknek. Így Magyarország agrárinformatikai rendszerének is meg kell felelnie az EU elvárásainak.

Az EU információs rendszerei szerteágazóak, de lényegében két markáns csoportba sorolhatók. Ezek:

- a primer, vagy elsődleges információs rendszerek,
- valamint a másodlagos, vagy szekunder információs rendszerek.

A primer információs rendszerek az EU nagy adatgyűjtő és feldolgozó struktúrái. Lényegében ezekre épül az Unió agrárinformációs rendszere. Négy meghatározó eleme a következő:

- o **Az agrárstatisztika**, amely az EUROSTAT által koordinálva szerteágazó területeken nyújt statisztikai jellegű információkat az Unión belüli folyamatokról, a főbb tendenciákról. A szerteágazóságra példa, hogy a térinformatika és a távérzékelésen alapuló szántóföldi monitoring is az EUROSTAT Compendiumában szereplő modul.
- o **Az FADN /Farm Accountancy Data Network/** az EU egyik legfontosabb információs rendszere. /Magyarországon a németországi gyakorlatnak megfelelően a Tesztüzemi Rendszer elnevezés a leginkább elterjedt./ Feladata a gazdaságok pénzügyi folyamatainak, jövedelemhelyzetének nyomon követése.
- o **A Piaci Információs Rendszer**, amely szolgálja egyrészt a termelők tájékoztatását a főbb piaci folyamatokról, de fontos feladata a brüsszeli apparátus információs igényeinek a kielégítése is.
- o **Az Integrált Igazgatási és Ellenőrző Rendszert** indokolt kiemelni, amely lényegét tekintve egy „technikai jellegű” információs rendszer, elsősorban az EU adminisztrációjának működését hivatott segíteni. Az adófizetők pénzéből történő kifizetések elszámolását, illetve ellenőrzését végzi.

A másodlagos, vagy szekunder információs rendszerek közös jellemzői, hogy általában nem végeznek nagyobb tömegű közvetlen adatgyűjtést, információikat főleg a primer rendszerek adatbázisaiból nyerik. Céljuk egy-egy „szűkebb” terület speciális információigényének kielégítése. A számlarendszerre épülő feladatok az EU által megadott termékkörre vonatkozó termelési, felhasználási, technológiai adatok konzisztens összefogása révén a termelési érték, hozzáadott érték és jövedelempozíciók rövidtávú előrejelzése, illetve agrárpolitikai intézkedések várható hatásának prognosztizálása.

10.5.1. Az agrárstatisztika

Az Európai Unió - a közösségi költségvetésből legnagyobb arányban részesedő mezőgazdasági támogatásokra való tekintettel - az agrárinformációs rendszerrel szemben megkülönböztetett igényeket és követelményeket támaszt. Az információk döntő többségét az agrárstatisztika szolgáltatja, amely az Európai Unió statisztikai rendszerének egyik legfejlettebb alrendszerét képezi. Az agrárstatisztika alapját az Eurostat jogszabályai, megállapodásai és ajánlásai képezik.

Az EU jogszabályok, megállapodások és ajánlások a közösségi Agrárpolitika (CAP) igényeinek megfelelően folyamatosan változnak. Tekintettel arra, hogy a CAP jelenleg alapvető változáson megy át (termék orientáltból gazdálkodó orientálttá válik) a közeljövőben még több és markánsabb változás várható.

A hivatalos statisztikai szolgálat keretében a mezőgazdasági statisztika fő felelőse a KSH. Az FVM főleg működtetési „operatív” információkat gyűjt. Tevékenységének fő területei az előrejelzések, szakértői becslések, illetve a termelés bevételeire és költségeire vonatkozó adatgyűjtések. Ezen kívül az FVM az erdőgazdálkodás, a halászat és távérzékelés statisztikai témáinak felelőse.

Az agrárstatisztikai információk – amelyeket alapvetően agrárpolitikai, közgazdasági elemzési és általános információs célokra hasznosítanak – az EUROSTAT útján jutnak el a Európai Bizottsághoz. Az EUROSTAT kizárólag az adott ország hivatalos statisztikai szervezetén keresztül történő adatszolgáltatást igényli. Ez a gyakorlatban annyit jelent, hogy a nemzeti statisztikai hivatalok által továbbított adatokat az EUROSTAT mintegy hitelesíti és így adja át az Európai Bizottság számára. Magyarország esetében tehát a Központi Statisztikai Hivatal az ország agrárstatisztikai adatszolgáltatója az EUROSTAT felé. Az EU csak ezeket az adatokat fogadja el hiteles információként.

A közösségi agrárstatisztika legerősebb tartóoszlopa a gazdaságszerkezeti felvételek rendszere, amely gazdag információforrásként szolgál az EU mezőgazdaságának helyzetéről, annak struktúrájáról, a gazdaságok társadalmi-gazdasági jellemzőiről.

1997. óta üzemszerűen működik a műholdas távérzékelésen alapuló szántóföldi növénymonitoring. A térinformatika és a távérzékelés ezzel az agrárirányítás döntéshozó mechanizmusának nélkülözhetetlen részévé vált. A korszerű technika hatékonysága a belvíz helyzet objektív feltérképezésében is szerepet játszik. Az agrárinformatika, ezen belül az agrárstatisztika számára is jelentős fejlődést biztosít az a földügy és térképészet területén megkezdett, a közösségi vívmányok átvételével összefüggő munka, amely a földhivatali információtechnológiára és szolgálatfejlesztésre építve országos szinten kiemelt feladatokat hajt végre. A távérzékelési adatok évente 7-10 alkalommal állnak rendelkezésre az FVM terméshozam-előrejelzési rendszerének megfelelően. A megfigyelés és a mérés folyamatos, kiterjed az ország egész területére. Minden jelentés tartalmaz területre és hozamra vonatkozó előrejelzéseket az őszi búzára, őszi és tavaszi árpára, kukoricára, takarmány-kukoricára, napraforgóra, cukorrépára és lucernára vonatkozóan. Ha szükséges, egyéb adatok is előállíthatók, így például időszakosan vízzel elborított terület, növénybetegség, stb. A távérzékelés technikai hasznosítása a mezőgazdasági statisztika jelenlegi rendszeréhez még nem illeszthető. A távérzékelés 2002 évtől fontos ellenőrző szerepet töltött be a teljes lefedettség megvalósításában (vetésterület növényenként, a termés mennyiség egyeztetésében), továbbá az előrejelzések, prognózisok elkészítésében.

10.5.2. A Piaci Információs Rendszer

A piaci információs rendszerek fő feladata a piaci transzparencia megteremtése, ami fontos előfeltétele egy működő versenynek. Ez akadályozza meg ugyanis, hogy valamely értékesítési rendszer egyik szintjén a szokásos mértéket messze meghaladó profit halmozódjék fel, mégpedig általában más szintek rovására. A mezőgazdasági termelők számára a piaci transzparencia azt jelenti, hogy termékeikért azt a bevételt kapják, amely a piacon reálisan elérhető. A kereskedelemnek és a feldolgozóknak is szükségük van piaci transzparenciára annak érdekében, hogy feladataikat a lehető leghatékonyabban teljesíthessék.

A mezőgazdasági piaci információk különösen a mezőgazdasági termelők piaci esélyeit javítják. A túlkínálat és a hiány-szituáció ismeretében ugyanis információt kapnak arról, mikor és hol lehet adott esetben termékeik eladására a legkedvezőbb alkalom. Másfelől a piaci információknak valamennyi potenciális partner számára hozzáférhetőnek kell lenniük, ugyanis csak ez teszi lehetővé, hogy a piacon az ár alakuljon ki, amely az adott piaci helyzetnek legjobban megfelel.

Az EU piaci és árinformációs struktúrája lényegében az információ felhasználói alapján bontható két csoportra:

- a piaci szereplők információs igényeit kielégítő információs rendszerek, valamint
- a központi irányítás, illetve elsősorban az EU Bizottság VI. Főigazgatósága elvárásainak megfelelni képes piaci /statisztikai/ adatszolgáltató rendszerek.

A brüsszeli adminisztráció a tagországoktól pontosan meghatározott mechanizmus szerint igényel piaci /statisztikai/ információkat. Az adatközlési kötelezettséget termékpályánként eltérően írja elő az Európa Bizottság. Az adatszolgáltatást a VI. Főigazgatóság /DG VI./ felé kell teljesíteni. Az adatszolgáltatásért minden tagországban az agrártárca felel.

Magyarországon a médiumokon keresztül széles kör számára elérhető piaci információs rendszer az Agrárgazdasági Kutató és Informatikai Intézetben /AKII/ működik.

10.5.3. A Tesztüzemi Rendszer

A politikai kezdeményezések és döntések megalapozásához az Európai Unió Bizottságának szüksége van:

- információkra az üzemek jövedelmi szintjéről, valamint
- elemzésekre a politikai döntések lehetséges hatásairól.

A Bizottság ezen funkciók ellátásához hozta létre a Mezőgazdasági Számviteli Információs Hálózatot – angolul: Farm Accountancy Data Network /FADN/ -, amely egy minden EU tagállamra kiterjedő reprezentatív adatgyűjtési rendszer. Ez évente biztosít adatokat a Bizottság számára.

Az FADN felmérése lefed minden az üzemekben folytatott mezőgazdasági tevékenységet. Sőt egyes, nem az ágazat keretei közé tartozó tevékenységről is gyűjt adatokat /pl. falusi turizmus, erdészet/.

A reprezentatív adatgyűjtés megvalósításánál kulcsszerepet tölt be a kiválasztási terv, amelynek elkészítéséhez viszont ismerni kell az „alapsokaságot”, azaz a megfigyelésbe bevont gazdasági egységek összességét. A Bizottság pragmatikus szempontok szerint határozza meg az FADN megfigyelési körét. Így a vizsgálatba csak a fő munkaidőben vezetett üzemek kerülnek be. Főmunkaidős üzemnek az minősül, amely elég nagy ahhoz, hogy a gazdálkodó tevékenységének nagyobb hányadát itt folytassa, s elég jövedelem biztosítható innen a család megélhetéséhez. Az alapsokaságról a nemzeti statisztikai hivatalok gazdaságszerkezeti összeírásai szolgáltatnak információkat.

10.5.4. Az Integrált Igazgatási és Ellenőrző Rendszer

Az Integrált Igazgatási és Ellenőrző Rendszer /Integrated Administrative and Control System/, – a továbbiakban Integrált Rendszer – az EU primer információs rendszerének negyedik eleme. Rendeltetése alapjaiban különbözik a többi primer rendszertől. Míg ugyanis a statisztikai, a piaci információs és a tesztüzemi rendszerek döntően a gazdaságpolitikai döntések megalapozását, utólagos kontrollját szolgálják, addig az Integrált Rendszer elsősorban, sőt kizárólagosan a Közös Agrárpolitika /KAP/ egyik meghatározó elemének, a kompenzációs /vagy direkt/ támogatásoknak az odaítélését, illetve az odaítélés jogosságának ellenőrzését szolgálja. Ezen keresztül egy „technikai jellegű” információs rendszer.

A kompenzációs támogatási konstrukciónak a bevezetése azonban hatalmas többletfeladatot jelentett mind a közösségi intézményekre, mind a nemzeti és regionális szervezetekre, mind az egyes gazdálkodókra vonatkozóan. A támogatások igénybevételenek szabályozása, a támogatási kérelmek benyújtása, ellenőrzése, a kifizetések engedélyezése és lebonyolítása, az esetleges visszaélések felderítése és szankcionálása részletes adatszolgáltatást, bonyolult nyilvántartások egységes rendszerét, a támogatott objektumok /vetésterületek, állatok/ megbízható azonosítását tette szükségessé.

Ezt a célt szolgálta az Integrált Rendszer kialakítása, amely lehetővé tette, hogy a gazdák egy támogatási jogcímre csak évente nyújtsanak be támogatási kérelmet, ami egyúttal a megfigyelés és ellenőrzés alapjául is szolgál. Ez a rendszer szolgálja a gazdálkodók támogatási célú regisztrációját is. Ha bármely az Integrált Rendszer keretei közé sorolt támogatásra jelentkezik a gazdálkodó, ki kell töltenie egy olyan nyomtatványt is, amely gazdasága adatait részletesen tartalmazza.

A rendszer biztosítja, hogy az EU kompenzációs támogatásából részesülő termelőkről a lehető legrészletesebb információk álljanak rendelkezésre. Ezért a termelőnek a kérdőívek kitöltésekor szinte teljesen „átláthatóvá” kell tennie magát.

10.5.5. A Mezőgazdasági Számlák Rendszere

A Mezőgazdasági Számlák Rendszerét /Economic Accounts for Agriculture = EAA/ a Nemzeti Számlák Rendszeréhez kapcsolódva – e rendszer úgynevezett szatelit számláiként – az EU agrárpolitikai információ igényeihez igazodva alkották meg. A rendszer keretében a mezőgazdasági tevékenységről kapni átfogó, és részletes elemzésekre is lehetőséget adó képet. /A mezőgazdasági tevékenység keretébe a nemzetgazdaság valamennyi mezőgazdasági tevékenysége beletartozik, végezzék azt bármely nemzetgazdasági ágban./ A Mezőgazdasági Számlák Rendszerét /továbbiakban MSZR/ nemcsak országos szinten, hanem regionálisan is elő kell állítani, illetve működtetni. Az MSZR a mezőgazdasági termelést, annak felhasználását, a termelés ráfordításait és a jövedelemalakulást összefüggő rendszerben írja le.

10.6. Elektronikus kereskedelem, elektronikus üzletvitel

Az elektronikus kereskedelmet már terminológiai szempontból sem könnyű meghatározni. Bár kétségkívül létezik magyar megfelelője az e-commerce-nek (elektronikus kereskedelem), csakúgy mint az e-business-nek (elektronikus üzletvitel) még a szakirodalomban sem alkalmazzák következetesen ezeket a kifejezéseket. Az e-business kifejezést először az IBM használta, ám milyen könnyű elméletileg elválasztani a két fogalmat, oly nehéz megfelelően alkalmazni egy-egy gyakorlati problémára. A magyar nyelvben az „elektronikus kereskedelem” kifejezés sokkal elterjedtebb az elektronikus üzletvitel meghatározásánál, függetlenül attól, hogy helyesen vagy helytelenül alkalmazzuk-e.

Az elektronikus gazdasághoz tartozó tevékenység, tranzakcióknak a Világhálón való lebonyolítását jelenti. (BÖGEL, 2000)

Az Internet szolgáltatásaira – e-mail, azonnali üzenetküldés, vásárlás, alapozó üzleti kapcsolat a résztvevő felek között. Az e-commerce során tőke, áru, szolgáltatás és/vagy információ cserélhet gazdát két kereskedelmi partner, vagy a kereskedő és a vásárló (végfelhasználó) között. (<http://www.webopedia.com>)

E-business

Meghatározás 1

„Bármilyen internetes kezdeményezés – taktikai vagy stratégiai, ami átalakítja az üzleti kapcsolatokat, legyenek azok akár fogyasztók és vállalatok, vállalatok és vállalatok, fogyasztók és fogyasztók közötti vagy vállalaton belüli relációk” (BÖGEL, 2000)

Meghatározás 2

„Az üzleti irányítás bonyolítása az Interneten keresztül. Ez a tevékenység magába foglalhatja áruk és szolgáltatások adásvételét, technikai vagy információs segítségnyújtást az interneten keresztül”. (<http://www.webopedia.com>)

Meghatározás 3

„Minden üzleti tevékenység, amely részben vagy egészen digitális úton zajlik.” (COLTMAN, 2002)

A jól látható terminológiai különbségek, ellentétek, vagy ha úgy tetszik, a zavar ellenére úgy tűnik, hogy az e-business szélesebb területet ölel fel, mint az e-commerce. Míg az e-commerce konkrétan az üzleti tevékenység magvalósulására, a folyamatra és a részt vevőkre koncentrál, addig az e-business magában foglalja az e-commerce-t, de a tágabb világgazdasági környezettel, a belső vállalati mechanizmusokkal, adott esetben a teljes piaccal és azok hatásával is foglalkozhat. (NEMESLAKI, 2004)

Az elektronikus kereskedelem hatása a gazdasági életben

„Az e-commerce megjelenése szükségessé teszi a jelenlegi üzleti modellek radikális újragondolását. Azok a vállalatok, amelyek kereskedelmi modelljei rugalmasak, vagy könnyen megváltoztathatók, nagyobb eséllyel lépnek be az elektronikus kereskedelembe. Az új

üzleti helyzetek, mint amilyen az e-commerce, újfajta gondolkodást igényelnek.” (BARNES, HUNT, 2001)

Az elektronikus kereskedelem előnyei:

A kereskedelemben régóta ismert és elfogadott 6M szabály szerint tevékenységünk akkor lehet sikeres, ha

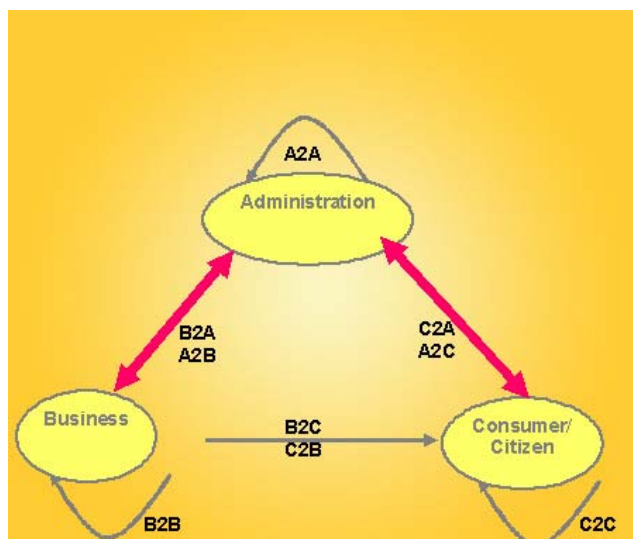
- a megfelelő anyagot, energiát, információt, személyt;
- a megfelelő mennyiségben;
- a megfelelő minőségben;
- a megfelelő időpontban;
- a megfelelő (minimális) költséggel juttatjuk el;
- a megfelelő helyre.

Bár az e- business messze túlmutat a kereskedelmen, jelentősége abban rejlik, hogy új technikai és megközelítései révén hatékonyabbá teszi a folyamatokat, ezáltal tőke és erőforrások takaríthatóak meg. (BOCK, SENNÉ, 1997)

E-business modellek

Az elektronikus kereskedelmet alapvetően kétféle módon tipizálhatjuk: az összekapcsolt felek közötti kapcsolat vagy kapcsolatrendszer jellege, illetve annak specialitásai és remélt előnyei alapján. Jól látható, hogy míg az első egyértelműen elméleti kategória, a második sokkal inkább a gyakorlati megfigyelések alapján született, az organikus fejlődés rendszerbe foglalásának céljából. Ebből adódóan, míg az elektronikus kereskedelemben részt vevő felek minőségének rendszerezése tekintetében nincs jelentősebb véleménykülönbség a szakemberek körében – legfeljebb a rendszer finomítása, elmélyítése területén merülhetnek fel komolyabb viták – addig a kapcsolatrendszerek minőségi tipizálása területén számos eltérő modellt találhatunk. Túlzás lenne azt állítani, hogy ezek a modellek alapjaiban térnek el egymástól, de mindenképpen említést érdemel a szakirodalomban kimutatható sokszínűség.

Az elméleti modellek az üzletben résztvevő szereplők minősége alapján foglalják rendszerbe az elektronikus kereskedelmet. A fontosabb szereplők közötti kapcsolatokat és modell típusokat a *10.9. ábra* mutatja.



10.9. ábra: A modellek kapcsolatai

Administration to Administration (A2A)

Az A2A az elektronikus közigazgatás meghatározása, melynek során az egyes kormányzati intézmények elektronikus úton cserélnék és szolgáltatnak információt egymásnak. Ki kell emelni, hogy az A2A nem csupán ezt a napi eljárási rendnek megfelelő adatáramlást jelenti, amelyet megkönnyít és olcsóbbá tesz az IT, hanem azt a folyamatot is, melynek során az egyes kormányzati adatok a köz- (a magán és vállalati) szféra által felhasználható, lehetőleg minél könnyebben hozzáférhető információvá válnak.

Administration to Business (A2B)

Az (A2B) vagyis az elektronikus közbeszerzés nem csak Magyarországon, de jellemzően az egész világon gyermekcipőben jár. Hasznosságát azonban jól mutatja, hogy a vállalati, elsősorban a bankszektorban már régóta használják sikerrel. Alapja a nyílt, interneten történő tendereztetés, amelyhez elméletileg bárki hozzáférhet – nyilvánvalóan a rendszernek megvannak a maga korlátai, hiszen csak könnyen összehasonlítható célok és konstrukciók esetében alkalmazható.

Business to Administration (B2A)

Elektronikus ügyintézés. Egyes vélemények szerint nem tartozik az e-kereskedelem körébe. (KONDRICZ, 2000)

Business to Business (B2B)

A B2B, az elektronikus vállalkozási kereskedelem az elektronikus kereskedelem legnagyobb szelete. Ez a kijelentés sok tekintetben meglepő lehet, hiszen az IT fejlődésének leglátványosabb pályáját a B2C (Business to Consumer – elektronikus „kiskereskedelem”) területén írta le. Azonban – a számítási módok miatt erősen eltérő, ám ugyanazt a tendenciát erősítő – adatok azt mutatják, hogy a B2B kereskedelem volumene messze meghaladja a B2C-t.

A B2B magába foglal szinte minden területet, amely elektronikus úton végbemenő vállalkozási együttműködésnek tekinthető. Ebből a kijelentésből is jól látszik, hogy az IT mennyire megváltoztatta az „érték” fogalmát. B2B kapcsolatokra példák:

- A termelő-feldolgozó-kereskedő lánc (supply chain)
- Logisztika
- Az elektronikus vásárterek legtöbbje (általában vertikálisan, egy-egy ágazat köré szerveződve)
- Információ vétele és eladása
- K+F együttműködések

Business to Consumer (B2C)

A business to consumer kereskedelem talán az e-business legismertebb formája. Ahogy már korábban jeleztem, az ily módon megvalósuló üzletkötések volumene jelentősen elmarad a B2B üzletkötésektől, ám az üzletkötések száma nagyságrendileg meghaladja azt. Mindez a két kereskedelmi tevékenység jellegéből adódik: a vállalkozási kereskedelem és a kiskereskedelem összehasonlítása nyilvánvalóan inkább csak jelzés értékű eredményekkel szolgálhat, noha az elektronikus kereskedelem legnagyobb mértékben a B2C esetén tágította ki a korábbi határokat. (Bowden szerint csupán akkor lehetne számítani rá, hogy egy újabb fellendülés kezdődjön az B2C kereskedelemben, és legalább megközelítse a B2B fejlődését, ha a hozzá kapcsolódó költségek drasztikusan csökkennének.)

Ez kereskedelmi forma a legsokszínűbb is, hiszen az elektronikus kereskedelem jellegéből adódóan szükségszerűen sokkal inkább képes alkalmazkodni a megújuló igényekhez, mert az egyéni vásárlók a vállalatoknál lényegesen gyakrabban változtatják vásárlási szokásaikat, illetve termékhűségük is alacsonyabb.

Amennyiben az elektronikus kereskedelemnek a legtágabb értelmezését használjuk, úgy minden a kereskedelem ösztönzésére vonatkozó tevékenység is – teljesen vagy legalábbis részben – az e-business kategóriájába tartozik,

- az elektronikus médiában (televízió, rádió), folytatott reklámtevékenység;
- az interneten folytatott nem interaktív reklámtevékenység;
- ugyanezekben a helyeken folytatott ismeretterjesztő tevékenység (dokumentumfilmek, riportok, passzív weblapok).

Fontos rámutatni, hogy a felsorolt tevékenységek mindegyike passzív, vagyis egyutas. A hirdető csak csupán információt juttatnak el a potenciális vásárlóhoz, nincs lehetőség sem a vásárlási szokások feltérképezésére, sem pedig vásárlásra. Az információ minősége természetesen eltérhet, hiszen technikailag jelentős különbség mutatkozik egy termék közvetlen reklámozása és egy cég tevékenységi körének, múltjának és jelenének, céljainak bemutatása között. Azonban ma már minden reklámozható terméket reklámoznak, és egyetlen nagyobb cég (függetlenül attól, részt vesz-e a B2C kereskedelemben) sem engedheti meg magának, hogy ne legyen saját weblapja.

A szűkebb értelemben vett elektronikus vásárlás, amelynek alapja a kereskedő és a vásárló közötti interaktív együttműködés, egyetlen jelentős kategória mentén osztható meg: a termék minősége alapján. Az e-business lényege az a képesség, hogy a vásárló anélkül az

által kiválasztott termékhez juthasson, hogy neki fizikailag el kelljen mennie egy kereskedelmi egységbe.

Lényeges azonban az érem másik oldala is, miszerint a megrendelt, adott esetben elektronikus úton már kifizetett terméket el kell juttatni a vásárlóhoz. E tekintetben alapvetően eltér a hagyományos „anyagi jellemzőkkel is rendelkező” áruk kereskedelme az elektronikus tartalomszolgáltatástól. Hiszen az első esetben az e-buisnessnek ki kell egészülnie egy logisztikai szolgáltatással (amely nyilvánvalóan jelentős többletköltségeket eredményez), míg a második esetben megvalósul a „tökéletes elektronikus kereskedelem”, amely során elméletileg lehetőség van rá, hogy az összes vásárlási folyamat – a megismerés, a kiválasztás, a megrendelés, a fizetés és az átvétel – elektronikus úton menjen végbe.

Anyagi dimenzióval rendelkező tárgyak kereskedelme

- interaktív hirdetés;
- egyedi vásárlás (egyszerű internetes vásárlás, pizza rendelés, vagy TV Shop)
- elektronikus üzlet igénybe vétele (nagy, sokféle árú, széles termékpaletta)

Anyagi dimenzióval nem rendelkező tartalom kereskedelem esetén

- interaktív hirdetés;
- tartalomszolgáltatás megrendelése (elektronikus könyv, zene, szoftver, csengőhang, logó);
- információ/hozzáférés

A B2C kereskedelem utolsó jellemzője, hogy képes átlépni a hagyományos termelő-nagykereskedő-kiskereskedő láncolatot, az eladó a termelő lánc bármely tagja lehet, hogy ha megfelelő (marketing) stratégiával képes vásárlókat vonzani magához.

Consumer to Consumer (C2C)

A C2C kereskedelmet sokan az eBay modellként is emlegetik, mivel a rendszert az említett szolgáltató alakította ki 1995-ben, és azóta is piacvezető. Éves forgalma már 2002-ben elérte a 15 milliárd dollárt. A rendszer lényege, hogy a potenciális eladókat és vevőket egy jól strukturált, könnyen kezelhető rendszer részeivé teszi, megkönnyíti számukra a vételt vagy eladást.

Az elmondottak alapján jelentős párhuzam vonható a B2B és C2C piactér modell között, de ezért szükséges a különbségek jellemzése is. Bár a mindkét struktúra ugyanarra a modellre épül, alapvetően más a kettőben az üzleti szereplők minősége. A B2B esetében cégek, míg a C2C-nél a magánszemélyek lépnek egymással üzleti kapcsolatba. Ezen kívül az elektronikus nagykereskedelmi piacok általában előfizetési rendszerben működnek, vagy tranzakciós díjakat kérnek, tehát szolgáltatásaikat pénzért árusítják. A C2C kereskedelemben ez nem jellemző; a tranzakciókat regisztráció ellenében lehet igénybe venni.

A C2C-ben az elektronikus kereskedelem szinte minden előnye megmutatkozik: minimális anyagi ráfordítással (internet kapcsolat, és számítógép, valamint az erre áldozott idő) üzleti kapcsolatba kerülhetünk a világ bármely más táján élő személlyel. Ugyanakkor a C2C kereskedelem döntően „anyagi” árucikkek vételével-eladásával foglalkozik, ezért itt is beleütközünk a logisztika, – a B2C fejezetnél már említett, a későbbiekben kifejtésre kerülő – problémájába. Ebből adódóan főleg különleges, egyedi, vagy legalábbis érdekesnek

minősíthető árucikkek forgalma jelentős, bár a kínálatban igen nagy számban találhatóak „lejárt” más formában nehezen értékesíthető árucikkek is.

A C2C kereskedelem lényege a kritikus tömeg, vagy kritikus méret elérése. Mind a vevők, mind az eladók szívesebben választanak olyan elektronikus piacot, amelynek magas a látogatottsága, ennek köszönhetően pedig nagy a forgalma is. Ez a természetes tendencia a C2C kereskedelemben a legnagyobbaknak kedvez és koncentrálnak hatása van. Magyarországon a legjelentősebb kezdeményezés, a Vatera forgalma 2003-ben alig érte el a 250 millió forintot.

Ennek ellenére a regionális e-piacok versenyképesek lehetnek a globális nagy elektronikus piacokkal.

Peer to Peer (P2P)

A P2P rendszer – talán nem túlzás kijelenteni – az elektronikus kereskedelem legmodernebb, ugyanakkor kifejezetten speciális formája. Speciális azért, mert a tranzakciók ingyenesek, az adatforgalommal párhuzamosan nem generálódik haszon egyik félnek sem.

A P2P csak az interneten működik, a felhasználók számítógépeinek összekapcsolódása segítségével. A felhasználók összekapcsolódása közvetlen, vagy legalábbis semmilyen külön szolgáltató nem vesz részt benne.

P2P –nek minősülnek:

- a file-cserélő rendszerek;
- a számítógépes kapacitás megosztó hálózatok

Egyes vélemények szerint gazdasági szempontból csupán a B2B, a B2C, és a C2C létező modell, a többi inkább tudományos szempontú megközelítés. (HWA, 2002)

A magyar mezőgazdasági vállalkozások a beszerzésben és az értékesítésben leggyakrabban a B2C modellt alkalmazzák, a B2B csak ritkán jelenik meg a gyakorlatban.

Az élelmiszeriparban azonban a B2B használata a jellemző. Természetesen mindkét esetben döntően a cég profilja határozza meg az alkalmazható modellt.

Az állami szabályozásoknak köszönhetően a B2A alkalmazása is elterjedt, többnyire a nagy vállalatok esetében. Ez elsősorban a VPOP-t, az APEH-et, és a nyugdíjpénztári bizonylatokat érinti, de mint azt az előzőekben jeleztem, a szakirodalom ezt nem tartja valódi elektronikus kereskedelemnek.

Ellenőrző kérdések:

1. Ismertesse az információs rendszer fogalmát!
2. Mire szolgál a vezetői információs rendszer?
3. Írja le az információrendszer típusok közötti kapcsolatokat!
4. Határozza meg a stratégiai információrendszer lényegét!
5. Miként támogathatják az információrendszerek az üzleti stratégiát?
6. Mi a döntés előkészítő rendszerek feladata?
7. Mi az integrált vállalatirányítási rendszer?
8. Hogyan változott az integrált információs rendszerek fejlesztési filozófiája?

9. Mik az ágazati információs rendszerek?
10. Mire szolgál az elektronikus kereskedelem? Melyek az előnyei a hagyományos formákhoz képest?
11. Az elektronikus kereskedelemnél hol jelenik meg az állam, mint szereplő?

Irodalomjegyzék

- Barnes, S., Hunt, B. (2001): E-Commerce&V-Business. Butterworth-Heinemann, Oxford.
- Bock, W. H., Senné, J. N. (1997): Jövedelmező internet. Bagolyvár Kiadó, Budapest
- Bowden, S. és társai (2000): Adoption and implementation of e-business in New Zealand. Preliminary Results. Hamilton
- Bögel, GY. (2000): Verseny az elektronikus üzletben. Műszaki kiadó, Budapest..
- Coltman COLTMAN, T. és társai (2002): E-business: Revolution, Evolution, or Hype? California Management Review. Vol. 44. No. 1.
- Fajszai B., Cser L.:Üzleti tudás az adatok mélyén, IQSYS-BME, 2005.
- Gábor A.: Információmenedzsment, AULA 1997.
- IQSyS Szakmai Nap az adatbányászat alkalmazásáról, 2005.
- Kondricz P., Tímár A. (2000): Az elektronikus kereskedelem jogi kérdései. KJK KERSZÖV. Budapest.
- Laudon, K. C., Traver, C. G. (2002): E- commerce. Addison-Wesley. Boston
- Nemeslaki A. és társai (2004): E-business. Mentor Kiadó. Marosvásárhely

11. TÉRINFORMATIKAI ALAPOK, ALKALMAZÁSOK

A térbeli adatok tárolásának és megjelenítésének régi eszköze a térkép. Sok évszázadnak kellett eltelnie ahhoz, hogy a térképek olyan rendszert, tartalmat és formát kapjanak, mint az napjainkban megszokott. Nagyon sok szakma bemenő adatait megfelelő típusú térképekből nyeri. Ezért a térképek fejlődése minden időben szoros kapcsolatban volt az őket felhasználó szakmák fejlődésével. Napjainkra kialakultak azok a legfontosabb térképtípusok, melyekre támaszkodva fejlődtek és működnek a különböző mérnöki, közlekedési, építészeti, mezőgazdasági tervező szervezetek. A térképek egy csoportja a gazdaságot, a szociális-foglalkoztatási szférát, a népesség- nyilvántartást, az egészségügyet, a környezetvédelmet, a meteorológiát és még számtalan más tematikus ágazatot szolgál.

A térképeket három nagy csoportra oszthatjuk:

- geodéziai térképek
- topográfiai térképek
- tematikus térképek.

A **geodéziai térképek** fő jellemzője, hogy közvetlen mérések alapján készülnek. A mérési eredmények minimális általánosítással és szimbolikával kerülnek ábrázolásra. Méretarányuk 1:500 és 1:5 000 közé esik.

A **topográfiai térkép** a legszélesebben használt térképtípus. Méretaránya 1:10 000 -tól 1:200 000 -ig terjed. A méretarány csökkenésével az általánosítás foka nő. A Föld felszín mesterséges és természetes objektumainak ábrázolása mellett adminisztratív, gazdasági tematikákat is tartalmazhat. Ennek megfelelően ábrázolásmódja gazdag, melyet színek és szimbólumok segítségével valósít meg.

A **tematikus térképek** gyakran kisméretarányúak (1:500 000 -1:2 000 000). Ezeket a térképeket gyakran áttekintő céllal készítik, mert vizuálisan egyszerűen lehetetlen globális kérdéseket nagyméretarányú térképeken ábrázolva szemlélni és értelmezni. A tematikus térképek olyan jelenségeket ábrázolnak, melyek mérésére olyan ritka mérőhálózat áll rendelkezésre, hogy nagyméretarányú térképeken értelmetlen volna e jelenségek ábrázolása. Jellemzője továbbá, hogy egy-egy tematikát viszonylag szűk felhasználói réteg hasznosít, ezért a nagyméretarányú tematikus térképek térképszériaként történő nyomdai előállítás esetleg pénzügyi ellehetetlenüléssel járna.

E fejezetben a térképen, ezek új korszerű digitális formában tárolt formáin alapuló térinformatikai rendszerek alapjait tekintjük át. A térinformatikai rendszerekben azonban nemcsak az adatok, hanem a végezhető műveletek, illetve a rendszerek segítségével nyerhető információk is alapvető jelentőséggel bírnak. Az adatnyerés, az információ előállítás, valamint az ezekhez szükséges térinformatikai hardver eszközök megismerését követően a mezőgazdaságban egyre inkább tért hódító precíziós gazdálkodás alapelveit és az ágazati szintű néhány alkalmazást ismertetünk röviden

11.1. A térinformatikai rendszerek, digitális térképek

A hagyományos adatbázis koncepcióban szereplő adatok alfanumerikus adatok voltak. Természetesen ezek az adatok jelölhettek földfelszínen elhelyezkedő pontokat, vonalakat vagy területeket (házzszám, kerület stb.). Az adatok formája, szervezettsége, az adatbázis lekérdezési technikája azonban nem tették lehetővé, hogy a térbeliség tényét térbeli feladatok megoldására is felhasználjuk. Gondoljunk csak arra, hogy abból a tényből, hogy egy személyzeti fájlban rögzítjük a dolgozók lakcímét, valamint a munkahely címét, még nem

lehet kitalálni, hogy az egyes dolgozóknak mennyit kell utazniuk a munkába járás során. Az adatbank rendszerek fejlődésével ez a hiányosság egyre több felhasználó számára vált világossá.

A 60-as években fejlesztették ki a globális térbeli információk feldolgozását célzó földrajzi információs rendszer (*Geographical Information System* vagy **GIS**) szoftverjét, majd a 70-es évek elején megfogalmazták a nagyobb felbontású, de szűkebb tematikájú földinformációs rendszer (*Land Information System* vagy **LIS**) koncepcióját is.

A 80-as évek elejére kialakult az úgynevezett több célú kataszter (*multi-purpose cadaster*) koncepció, mely Európa és Észak-Amerika jelentős számú nagyvárosában, mint városi térbeli információs rendszer vált realitássá.

11.1.1. Kapcsolat a digitális helyzeti és alfanumerikus adatok között

A **geokód** valamely terület vagy területfüggő objektum, esetleg objektum csoport azonosítója, mely lehetővé teszi a kapcsolatot a területek vagy objektumok és a hozzájuk kötődő tulajdonság értékek között.

A geokód minden térbeli objektum egy pontjára vonatkozó olyan azonosító, mely tartalmazza az objektum jellegét, valamint a kérdéses pont geodéziai koordinátáit. Ha ezzel az azonosítóval ellátjuk azokat a számítógépes adatbázisokat, melyek az objektumhoz kötődő információkkal rendelkeznek, úgy lehetőségünk van ezeket az adatbázisokat az objektum alapján összekapcsolni, de arra is lehetőségünk van, hogy az objektum helyzetét jellemző koordináták segítségével bizonyos kezdeti térbeli feldolgozásokat is végrehajtsunk.

A geokód két kötelező és egy opcionális mezőből áll. Az első két karakterből álló mező az objektum jellegkódja. A második mező 12 karakterből áll, első hat karaktere a kérdéses pont y, a második hat karaktere a kérdéses pont x koordinátája méterben kifejezve. Mind az első, mind a második mező kötelezőek. A harmadik mező opcionális, négy karakterből áll, és a kérdéses pont magasságát hivatott megadni.

11.1.2. Kataszteri rendszerek

A nemzetközi földmérő szövetség (FIG) 1974-ben Washingtonban megtartott XIV. kongresszusán fogalmazták meg először a "Land Information System" (LIS) fogalmát. Az angol Land szó a kifejezésben a föld értelmet viseli, azaz ezek az újonnan meghatározott információs rendszerek a földhöz kapcsolódnak. A meghatározás szerint ezeknek az információs rendszereknek az alapeleme az egyértelműen meghatározható és lokalizálható parcella vagy földrészlet. Mivel az európai országok többségében a földrészletek és a rajtuk lévő épületek nyilvántartásának hagyományos rendszere fejlett volt, természetesnek tűnt, hogy az új információs rendszer fogalom e meglévő hagyományos ingatlan kataszterek automatizálásával realizálható.

11.1.3. Digitális magasságmodellek

A föld felszínéhez kapcsolódó tulajdonság jellemzők közül nem elhanyagolható szerepet játszanak a földi pontok egységes rendszerben kifejezett magasságai. A magasság növekvő jelentőségére az is utal, hogy a hagyományos földrajzi adattárolók a térképek fejlődésük folyamán egyre szélesebb körben (többféle terméken) és egyre tökéletesebb módszerekkel igyekeztek a magasságokat ábrázolni. A hagyományos térképi magasságábrázolás "végállomása" a szintvonalas ábrázolás volt, melynek lényege, hogy az azonos magasságú tereppontokat a térkép méretarányától és a terep domborzati jellegétől függő magasság lépcsőnként (alapszintköz) folyamatos vonallal összekötik. Az

alapszintvonalakkal nem kifejezhető idomok ábrázolására ún. felező és negyedelő szintvonalakat is alkalmaznak.

A digitális térkép fogalom megjelenésekor a szakemberek egyetértettek abban, hogy a digitális térkép magassági adatait nem célszerű szintvonalak formájában tárolni a számítógépben. A szintvonalak ugyanis nagyon nagy tárolási helyet igényelnek, s ugyanakkor a legtöbb mérnöki feladat számítógépes megoldását kevésbé támogatják.

Az optimális modelltípus kialakításánál nem elhanyagolható körülmény az adatnyerés mikéntje sem. A magassági adatokat földi úton rendszerint tahimetriával ritkábban területszintezéssel, fotogrammetriai kiértékeléssel vagy meglévő térképek szintvonalainak digitalizálásával határozzuk meg. A földi módszerek alkalmazása esetén a szintvonalak minden esetben levezetett termékek, tehát semmi sem indokolja modellként való felhasználásukat.

A digitális magasságmodellek két legáltalánosabb formája a háromszöghálózat és a gyakran törésvonal poligonokkal kiegészített szabályos négyzethálózat.

A háromszögmodell létrehozására leggyakrabban szórt pontos földi (tahimetria), vagy a ritkább, szórt pontos sztereofotogrammetriai kiértékelés kapcsán kis projekt modellekben kerül sor. Ha a háromszögek sarokpontjaira felmérjük a magasságokat és ezekre síklapokat fektetünk, úgy megkapjuk a terep magassági modelljét. Ezen a modellen a különböző tervezési és elemzési feladatok egyszerűen megoldhatók. A módszer hátránya, hogy minden modellpont mindhárom koordinátáját, valamint az összekötési előírást is tárolni kell.

A nagy kiterjedésű (regionális vagy országos) rendszereknél a szabályos négyzethálós tárolást részesítik előnyben. E módszer legfőbb előnye áttekinthetősége és jelentős tárolóhely megtakarítása. Hátránya viszont, hogy a tárolt pontok gyakran nem közvetlen mérés, hanem interpolálás eredményeképpen jönnek létre.

11.1.4. Földrajzi Információs Rendszerek

Az első működőképes GIS szoftvert a 60-as évek végére Kanadában dolgozták ki (Canadian Geographic Information System) azzal a céllal, hogy segítségével optimalizálják a fakitermelés és szállítás tervezését.

A 80-as évek elejére, amint azt a városi rendszerekkel kapcsolatban említettük, az észak-amerikai nagy városok is elkezdtek digitális térképi, illetve azt tovább fejleszteni, digitális térbeli információs rendszereik kialakítását. Az igények oldaláról tehát integrálódott a kis és nagyfelbontású térbeli információs rendszer koncepció. Ezt ismerte fel zseniálisan az *Environmental System Research Institute (ESRI)* nevű szoftverház, mely 1982-ben első verzióban kibocsátott ARC/INFO nevű **GIS** szoftverével az első általános s valóban kereskedelmi szoftvert bocsátotta ki e területen.

A GIS egy megfelelő hardver környezetben működő olyan szoftver együttes, mely eljárásai révén támogatja a területfüggő (térbeli) adatok nyerését, kezelését, manipulálását, analízisét, modellezését és megjelenítését komplex tervezési és működtetési feladatok megoldása érdekében.

A térinformatika nagy jelentőséggel bír a természeti erőforrások kutatásában, állapotának figyelésében; a közigazgatásban; a földhasználati- és tájtervezésben; az ökológiai- és gazdasági összefüggések feltárásában, a döntés-hozásban; ugyanakkor a közlekedési-, szállítási-, honvédelmi-, piackutatási feladatok megoldásában; a szociológiai-, társadalmi összefüggések vizsgálatában; a település-fejlesztésben és a létesítmény-tervezésben. A

Földrajzi Információs Rendszerek az informatika diszciplínán belül az utóbbi évtizedben csaknem önálló tudománnyá nőtte ki magát.

Egy GIS-nek válaszolnia kell a következő kérdésekre:

1. Mi van egy konkrét helyen? leltár készítés és\vagy monitoring.
2. Hol található vagy nem található egy kiválasztott jellemző? leltár készítés és\vagy monitoring.
3. Mi változott egy bizonyos idő óta? leltár készítés és\vagy monitoring.
4. Milyen térbeli típus (forma) létezik? térbeli analízis.
5. Mi lenne, ha ...? Modellelés.

11.1.5. Valódi térbeli (3D-s) információs rendszerek

A térképre alapozott rendszerekre is a "térbeli" jelzőt alkalmaztuk. E rendszerek az euklidesi tér 3 dimenziójából valójában csak két dimenziót alkalmaztak, ily módon talán helyesebb lett volna, ha a nemzetközi gyakorlattal ellentétben térbeli sík rendszereknek nevezzük őket. Kivételt ezek közül csak a digitális magasságmodellek képeztek, melyek elvileg egy felület - a terepfelszín - térbeli ábrázolására szolgálnak. Sajnos a legtöbb korai automatizált térképkészítő és **GIS** szoftver a magasságokat attribútum adatként tárolta és kezelte, és csak a valóban nyílt rendszerek biztosították, hogy ezekhez az adatokhoz olyan felhasználói szoftverek csatlakozzanak, melyek a térbeli műveleteket realizálják.

A földfelszín bármennyire is bonyolult, csak egy felület a föld- és bányászati tudományok jelentős része az adatbázist is beleértve olyan **GIS** szoftvereket igényel, melyek természetes működési területe a földfelszín alatti és fölötti szférákat leíró háromdimenziós valóság.

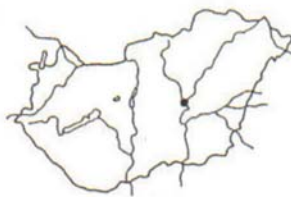
A háromdimenziós alakzatok térbeli kialakítása, leírása, manipulálása és megjelenítése, vagyis a háromdimenziós modellezés először a 70-es évek végén az építészeti és gépészeti tervezéseket segítő számítógépes rendszerekben jelent meg. Ezeknek a rendszereknek lényeges jellemzője, hogy az alakzatok szabályos elemi testekből, illetve felületekből kerülnek kialakításra (néhány szobrászati rendszer kivételével) s hogy a tárolást csak korlátozott mennyiségű elemi test vagy felület, illetve kész modell vonatkozásában kell megoldani.

11.1.6. Térinformatikai rendszerek típusai

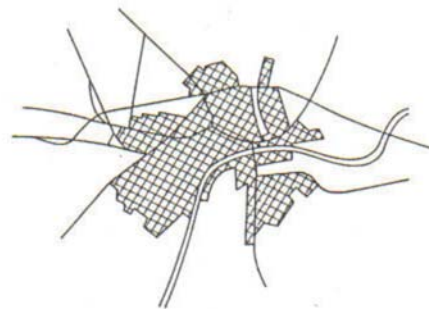
A geoinformációs rendszereket területi kiterjedésük szerint lokális, regionális és globális kategóriákba soroljuk.



Globális



Regionális



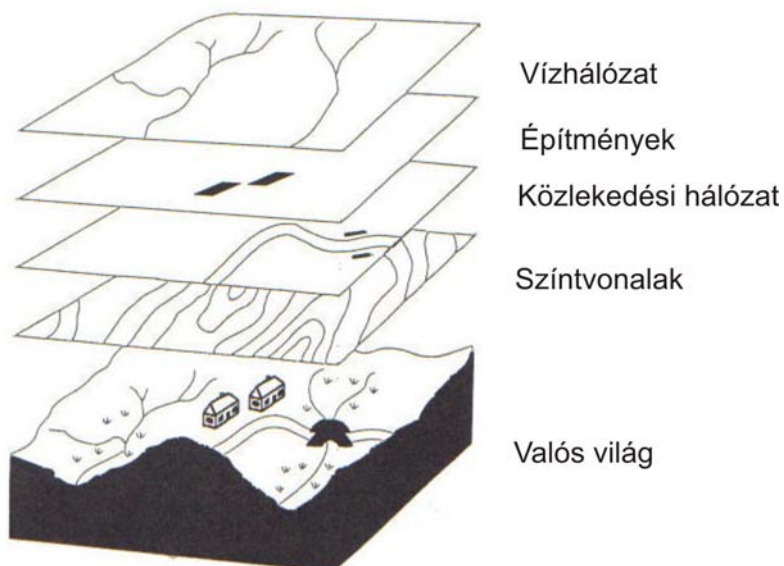
Lokális

Forrás: Detrekői –Szabó (2002)

11.1. ábra Térinformatikai rendszerek területi kiterjedése

A rendszerek felhasználása rendkívül sokrétű, a fontosabb felhasználási területek:

- kataszteri információs rendszer
- közművek információs rendszere
- közgazdasági-, marketing információs rendszerek
- közigazgatási- és önkormányzati információs rendszerek
- topográfiai-kartográfiai információs rendszer
- katonai geoinformációs rendszer
- természettudományi információs rendszerek (*geodézia, geofizika, geológia, talajtan, botanika, ökológia, hidrológia, meteorológia, környezettudomány*)
- társadalomtudományi információs rendszerek (*művészettörténet, régészet, szociológia, politológia, etnográfia*)
- mérnöki információs rendszerek (agrár-, bányá-, erdő-, földmérő-, építő-, építész-, közlekedési mérnöki rendszerek).



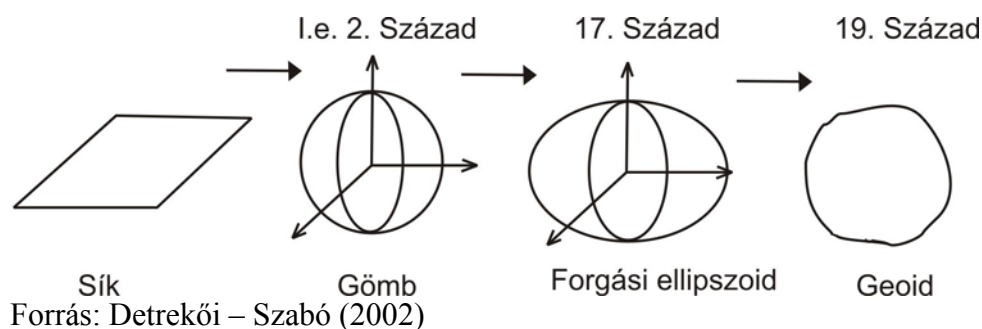
Forrás: Detrekői – Szabó (2002)

11.2. ábra A valós világ fedvényekkel történő ábrázolása. Az ábrán a tematikus dimenziók száma négy

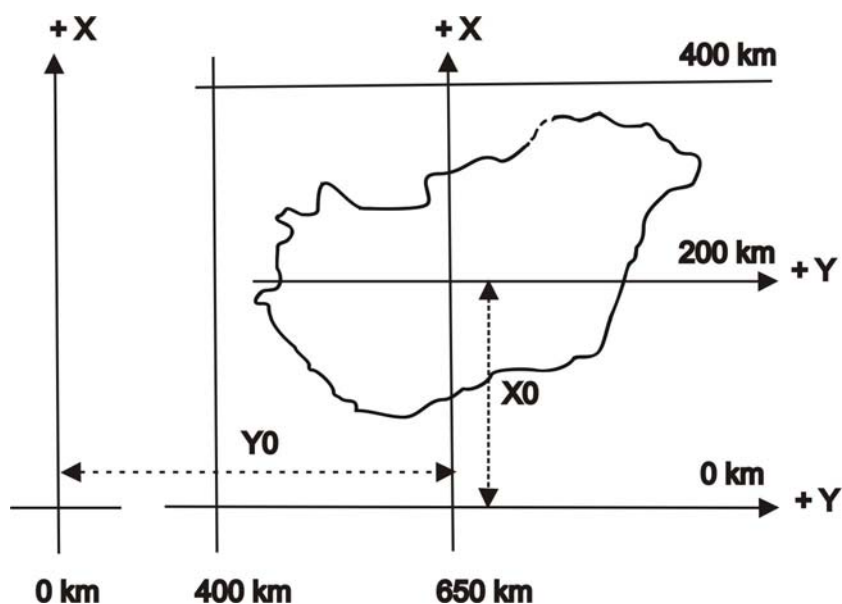
Az objektumok osztályba sorolásához a jobb áttekinthetőség kedvéért gyakran az egyes objektumosztályokat objektumoknak tekintik, ebből komplex objektumosztályokat hoznak létre. Például, ha a településekből hoznak létre objektum osztályt, akkor a komplex objektum osztályt a megyék, az ennél magasabb komplex objektumosztályt a régiók alkotják. Az objektumok osztályainak megválasztásakor gyakran alkalmazzák a fedvényekkel (layers) történő leírást. A fedvények alkalmazását szemlélteti a 11.2. ábra.

11.1.7. A helymeghatározás, geometriai adatok

Az objektumok helyzetét gyakorlati feladatok esetén a fizikai megvalósítását biztosító pontokból kiindulva – mérések segítségével határozzák meg. A térinformációs rendszerek területi kiterjedésétől függően más és más jellegű vonatkozási rendszerek felhasználása indokolt. A vonatkozási rendszerek definiálásához elméleti előfeltétele a Föld alakjának ismerete. A Föld elméleti alakjával kapcsolatos elképzelések az idők folyamán változtak. Hosszú időn keresztül a Föld alakját kizárólag geometriai felületek felhasználásával jellemezték. Az elméleti alak az ókorban kezdetben sík, majd gömb volt. A felvilágosodás korában vezették be a forgási ellipszoidot mint elméleti földalakot. A 19. század elején került bevezetésre a geoid alak.



11.3. ábra Az elméleti földalak fejlődése.



Forrás: Detrekői – Szabó (2002)

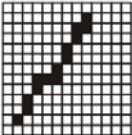
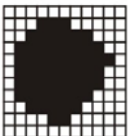
11.4. ábra Egységes Országos Vetületi Rendszer (EOV) koordináta-rendszere

A térinformációs rendszerek jelentős része síkhoz kapcsolódik, s ennek megfelelően síkfelületi koordináta rendszert alkalmaz. A síkhoz kapcsolt vonatkozási rendszerek mind lokális, mind a regionális térinformációs rendszerek alapvető eszközei. Síkfelületi koordinátákhoz jutunk, a Földet síkkal helyettesítjük, ha az ellipszoidról vagy a gömbről síkra vetítünk, vagy ha fotogrammetriai módszerrel gyűjtünk adatokat. A síkfelületi koordináta-rendszerek origójának megválasztása, a tengelyek irányának értelmezése és a tengelyek elnevezése különböző lehet. Az Egységes Országos Vetületi Rendszer (EOV) északkeleti tájékozású rendszer, mely koordináta rendszerének origóját az ország területén kívül vették fel, oly módon, hogy minden koordináta pozitív legyen.

11.2. Raszteres és vektoros adatok, adatkezelés, alpműveletek

A térinformációs rendszerek szoftverei a térbeli adatokkal különböző műveleteket hajtanak végre. A műveletek egy jelentős részénél az alfanumerikus illetve grafikus adatbázisokkal kapcsolatost lekérdezési mechanizmusok egyszerű vagy összetett alkalmazásairól van szó. Más feladatok ugyanakkor geometriai illetve halmazműveletek végrehajtását igénylik. Ezeket az alapfeladatokat műveleti alapeszközöknek hívják.

A térinformatikai rendszerekben nagyon fontos a **raszter - vektor, vektor - raszter** átalakítás. A nyolcvanas évek végéig a témát elsősorban a szkenneléssel és digitális fotogrammetriával kapcsolatban tárgyalták. Napjainkban azonban egyre jelentősebb szerepet játszik a **hibrid adatmodellű GIS** szoftver koncepció, mely korrekt, egyértelmű raszter - vektor, vektor - raszter konverziók nélkül nem képzelhető el. Ennek a koncepciónak az a lényege, hogy a térbeli műveleteket mindig olyan modellben kell végrehajtani, amelyikben egyszerűbb.

Elem	Vektor	Raszter
Pont		
Vonal		
Felület		

11.5. ábra Vektor- és raszteradatok grafikus jellemzése

Ilyenek például a fedvénymetszési műveletek (overlay), amelyek a raszteres adatmodellel igen egyszerűen végezhetők, míg a vektormodellben igen sok számítást és rendezést igényelnek, a távolság és kerületszámítások ugyanakkor, pontosan és egyszerűen csak a vektoros adatmodellben hajthatók végre.

11.2.1. Raszteres adatmodell átalakítása vektoros adatmodellé (raszter-vektor konverzió)

Raszteres adatok digitális kamarákkal történő fényképezés, mesterséges holdakról történő szkenneres felvételek, fényképek és térképek vagy rajzok szkennelése (letapogatása) útján jöhetnek létre. A digitális kamarák, a fényképek szkennelése és a fényképszerű szkennelés tónusos raszteres adatmodellt eredményez, ami számítástechnikailag azt jelenti, hogy minden egyes képelem (pixel) egyedi, adott határok közötti (egy vagy két byte-al

leírható) attribútummal, úgy nevezett szürkességi értékkel rendelkezik. A szkennelt térképek illetve rajzok rendszerint bináris képet szolgáltatnak, azaz a letapogatott rajzi tartalom feketének, a háttér fehérnek tekinthető és egy bit két értékével (0,1) jellemezhető.

Az analóg képek pixelekre bontása (diszkretizálása) alapfeltétele a számítógépes grafika s ezen túl a számítógépes alakfelismerés alkalmazhatóságának.

A szkennelt kép kisebb-nagyobb kiterjedésű objektumokból áll. A vektoros adatmodell ugyanakkor elméleti vonalakból építi fel objektumait. A térképek különösen a kataszteri térképek látszólag szintén csak vonalakból állnak. Ha célunk a raszteres modellből vektoros modellt előállítani, úgy meg kell különböztetnünk a szkennelt kép kisebb nagyobb objektumai közül azokat, melyek valóban területet kívánnak modellezni azoktól, melyek tulajdonképpen vonalak és csak azért tűnnek területnek mivel az analóg térképen nem lehet elméleti vonalat rajzolni. Ilyenek az olyan vonalas létesítmények is, melyeket a térképen szélességi kiterjedéssel is rendelkező egyezményes jelekkel ábrázolunk, információs rendszerünkben azonban geometriai adatként csak tengelyvonalukat kívánjuk tárolni, szélességüket egyéb tulajdonság jellemzőikkel együtt külön adatbázisban szerepeltetjük.

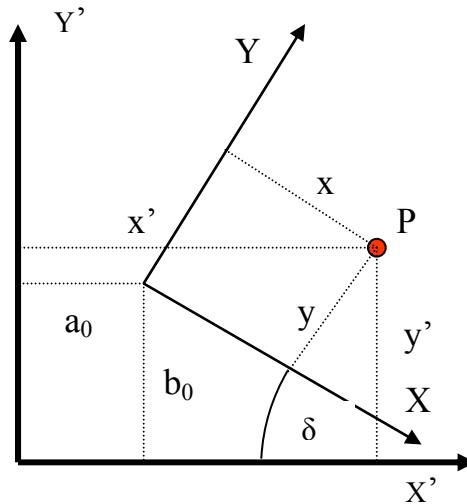
11.2.2. A vektor-raszter konverzió

A téma első megjelenése annak köszönhető, hogy a számítástechnikában a korai vektoros grafikus képernyők helyét elfoglalták a raszteres grafikus eszközök. Az egyik megoldandó probléma tulajdonképpen az volt, hogy a különböző matematikai görbéket milyen sűrű paraméter értékekkel kell kiszámolni ahhoz, hogy a képernyőn a görbét reprezentáló pixelek, függetlenül a görbe szakaszától, egyenlő távolságokra jelenjenek meg, s ennek következtében a görbe rajzának egyenletes vonalvastagsága és tömörsége legyen az egész ábrázolási tartományban. A másik probléma, a kerekítési hibákból adódó fogazottság, különösen a kis felbontású monitorokon rontotta az ábrázolás minőségét.

11.2.3. Síkbeli transzformációk

A térbeli objektumok helyzetét úgy tudjuk a legkényelmesebben figyelembe venni, ha valamilyen, derékszögű vagy görbe vonalú, *alapnak elfogadott*, **koordináta rendszerben** rögzítjük jellemző pontjaik koordinátáit. Az alapként elfogadott koordináta rendszert úgy nevezett referencia rendszerek-nek nevezzük. E rendszereket csak két csoportra osztják abszolút és relatív (a Föld olyan fizikai tulajdonságaival meghatározott mint a tömegközéppont és a forgástengely, illetve önkényesen a Földhöz rögzített), mégis mindkét csoportnak a gyakorlatban sokféle realizálását használják. Ezek a koordináta rendszerek térbeliek. A gyakorlati térképezés igényeiből kiindulva azonban ma még az esetek túlnyomó többségében nem a pontok térbeli koordinátáit használják, hanem azok vetületeit valamely célszerűen felvett síkra vagy síkba fejthető felületre (henger, kúp). Ha a vetítés során a harmadik komponenst a magasságot is meghatározzuk úgy azt a térinformatikai rendszerek szinte kizárólag mint attributív adatot kezelik. Ha ezen kívül még arra is gondolunk, hogy a különböző referencia rendszerekből különböző módszerekkel különböző elhelyezkedésű síkokra vetített térképek elvileg végtelen sok méretarányban készülhetnek, úgy feltehetőleg igazolt az az elképzelésünk, hogy a transzformációknak súlyponti szerepük van a térinformatikai rendszerek alapműveletei között.

A sík transzformáció legegyszerűbb esete, ha pontjainkat olyan koordináta rendszerbe akarjuk átszámítani, mely kezdőpontja nem esik egybe az eredeti koordináta rendszer kezdőpontjával, X tengelye pedig δ szöget zár be az eredeti koordináta rendszer X' tengelyével.



11.6. ábra Hasonlósági transzformáció

Viszonylag kis terület és jó minőségű adatok esetén végezhető transzformáció az elforgatás, eltolás, méretarány-változtatás.

$$x' = a_0 + a_1x - a_2y$$

$$y' = b_0 + a_1x + b_2y$$

ahol x , y és x' , y' a két rendszerben adott koordinátaértékek, a_0 , b_0 a két rendszer közötti eltolás. Az a_1 , a_2 pedig a két rendszer közötti elforgatás és méretarány változást tükröző együttható.

11.2.4. Távfolságfogalmak

Vektoros adatmodell alkalmazása esetén megszokott szemléletünkhöz legközelebb az Euklides-i távfolság fogalom áll, mely két, egy síkban fekvő pont távfolságát a **Pythagoras** tétel segítségével definiálja:

$$d = [(x_2 - x_1)^2 + (y_2 - y_1)^2]^{1/2}$$

azaz a két pont közötti távfolság a két pontot összekötő egyenes hosszával egyenlő.

Számunkra szokatlan, de a térinformatikában gyakran használt távfolság fogalom u.n. **Manhattan** vagy **háztömb távfolság**, mely meghatározás szerint

a

$$d_k = [|x_2 - x_1| + |y_2 - y_1|]$$

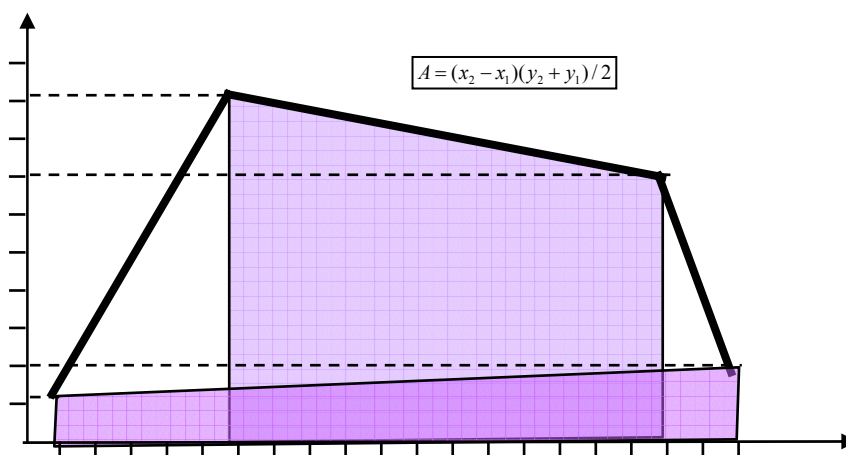
azaz a háztömb távfolság nem más, mint a két pont koordinátakülönbségei abszolút értékeinek összege. E távfolság fogalom keletkezésével (amint elnevezése is mutatja) az amerikai városok derékszögű utcahálózatához kötődik, hisz reálisabban tükrözi az Euklides-i távfolságnál a két különböző utcán található pont közötti valódi távfolságot. Különösen használatos e fogalom a

raszter grafikában, ahol két pixel közötti diszkrét távolság, pixelekben kifejezve, másképpen nem is adható meg.

11.2.5. Hossz-kerület, terület és súlypont számítások

Vektoros adatmodellben a vonalak hosszát a vonalszakaszok hosszainak összegzésével nyerik. A vonalszakaszok hosszát végpontjaik koordinátaiból a *Pythagoras* képlet felhasználásával számítják. Hasonlóképpen határozható meg a poligonok kerülete is annak figyelembevételével, hogy a kezdőpont koordinátái kétszer is szerepelnek a számításokban, mind az első, mind az utolsó egyenes szakasz hosszának meghatározásánál.

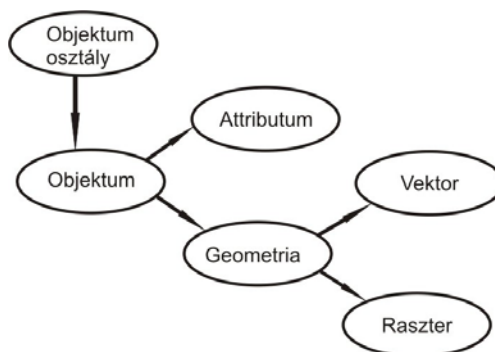
A térinformatikai szoftverek egyik leggyakrabban alkalmazott számítása a **poligonok területének** meghatározása. A számításhoz azt a geodéziában jól ismert algoritmust használják, mely a területet trapézok összegéből számítja.



11.7. ábra Területszámítás

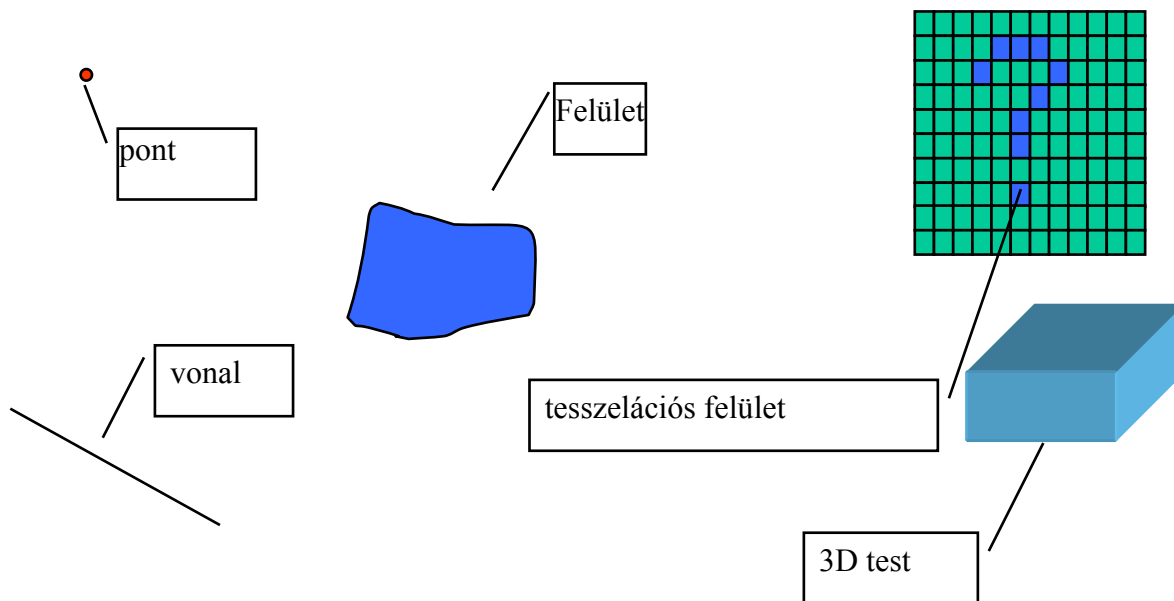
11.3. Grafikus adatmodellek

A térinformatikai rendszerek alkotóelemeinek fontos csoportját alkotják az adatok. Az adatok a valós világ objektumainak jellemzésére szolgálnak. A kiválasztott objektumokat objektumosztályokba sorolják. A valós világot reprezentáló objektumok jellemzésére az adatok két különböző típusa szolgál. Az objektumok helyzetét geometriai adatokkal, az objektumok tulajdonságait pedig szakadatokkal (attributomokkal) jellemzik. A geometriai adatok lehetnek vektoradatok vagy raszteradatok. (ábra).



Forrás: Detrekői – Szabó (2002)

11.8. ábra Az objektumok definiálása



11.9. ábra Az objektumok típusai

Az adatbázis rendszereket az alfanumerikus adatok tárolására és kezelésére dolgozták ki. A fejlődés során a különböző adatmodelleken és szervezési elveken nyugvó adatbáziskezelő rendszerek közül a relációs adatbáziskezelő rendszerek szereztek meg a győzelmet.

A grafikus (helyleíró) adatok olyan speciális tulajdonságokkal rendelkeznek, melyek figyelembe vétele jelentősen csökkenti az igénybe veendő tárhelykapacitást, és egyszerűsíti a végrehajtandó műveleteket. A grafikus adatok ugyanakkor nem igényelnek sok olyan műveletet illetve kapcsolatot, melyek meglehetősen az alfanumerikus adatbáziskezelő rendszerekben meghatározó. Ez az oka annak, hogy a jelenleg működő térinformatikai rendszer (LIS, GIS) szoftverek jelentős része, legalább ideiglenesen, külön adatbázisban tárolja a grafikus adatokat és egy másik, rendszerint relációs adatbázisban az alfanumerikus adatokat. A két adatbázis közötti kommunikációról egy kapcsoló szoftver rendszer gondoskodik.

Mivel napjainkban a hardverköttőségek jelentősége egyre inkább csökken (mind a tárhelykapacitás, mind a műveleti sebességek rohamosan nőnek), újabban olyan szoftvereket is találhatunk már, melyek a két fajta adatállományt közös relációs adatbázisban kezelik. Ezeknél a megoldásoknál speciális szoftvermodul gondoskodik arról, hogy a grafikus adatok a jellegükből következő egyszerűsítési lehetőségek kihasználásával kerüljenek ideiglenes tárolásra és manipulálásra.

A helyleíró adatok hagyományos ábrázolási formája a **vektor modell**. A tárolás alapeleme ebben a megközelítésben a vonaldarab. Ez a modell mindannyiunk számára közelebbi, hiszen geometriai, geodéziai, (manuális) térképezési módszereink mind ezt az alapelvet követik. A terep vektoros ábrázolása azonban csak jelentős egyszerűsítő munka eredményeképpen jöhet létre, hiszen a terep minden pontja önálló, a többi ponttól eltérő sajátosságokkal is rendelkezhet.

Az absztrakció egy alacsonyabb szintjén kézenfekvő, hogy a terep képét kis, homogén területekre osztjuk és az így kapott területeket tároljuk illetve manipuláljuk. A valóság képét területelemekben rögzítő módszereket tesszellációnak nevezzük. Azon az alapon, hogy milyen az elemi területek alakja szabályos és szabálytalan tesszellációról beszélünk. A

szabályos tesszelláció általában hierarchikus struktúrába szervezhető, azaz a kisebb területekből szakadásmentesen hasonló alakú nagyobb területek hozhatók létre.

A tesszellációs modell elterjedése kapcsolatban van azzal a ténnyel is, hogy a korszerű grafikus képernyők mind *raszter grafikával* dolgoznak, azaz függetlenül az adatbázisban alkalmazott grafikus adatmodelltől a megjelenítés mindig raszteres lesz (ezért néznek ki az alacsony felbontású monitorokon a nem függőleges vagy vízszintes vonalak fűrészfogszerűen).

A raszteres modell a tesszellációs modellnek az a legelterjedtebb változata, amelyben a területegység a négyzet.

Másik oka a raszteres modellek terjedésének abban rejlik, hogy a volumenben legjelentősebb adatnyerő eszközök (műholdas érzékelő és továbbító rendszerek, digitalizált, vagy digitális kamarával készült légifényképek, scanneres technikával digitalizált térképek) mind raszteres termékeket szolgáltatnak. Ebből következik, hogy még akkor is, ha egy térinformatikai szoftver vektoros adatmodellt alkalmaz, célszerű ha rendelkezik olyan programmodulokkal, melyek a raszteres bemenő adatok fogadását, illetve a raszteres megjelenítést lehetővé teszik.

A grafikus adatmodell kialakításánál két alapvető szempont játsza a főszerepet: mennyire tömöríthető a modellben az információ, illetve hogyan hat a (tömörített) modell a műveleti sebességre. A két követelmény együttesen sohasem elégíthető ki optimálisan, ezért rendszerint az adott hardver-feltételekhez legjobban illő kompromisszum szolgáltatja a megoldást.

A valódi térbeli háromdimenziós számítógépes testábrázolás először a gépészeti tervező rendszerekben jelent meg. Itt is megtaláljuk a vektoros és tesszellációs módszerek megfelelőit ezek a határleírás módszere illetve a konstruktív test geometria.

11.3.1. Vektoros modellek

A vektoros modellek lényege, hogy az ábrázolandó területet és a rajta lévő objektumokat pontok és a köztük lévő egyenesek együtteseként fogja fel. Ezen az elven az sem változtat, hogy a legtöbb rendszer alkalmas szabályos ívek generálására is a pontok között, mivel az ívek is elképzelhetők differenciálisan kicsiny egyenes darabokból (húrokból) alkotott poligonokként.

A terep absztrakciója következtében a térképen, így a digitális térképen is **pontszerű, vonalas és területi objektumok** találhatók.

A pontszerű objektumokat

a vektoros modell alap meghatározását alkalmazva úgy értelmezzük, mint egy olyan nulla hosszúságú egyenest, melynek kezdő és végpontja azonos.

A vonalszerű objektumok,

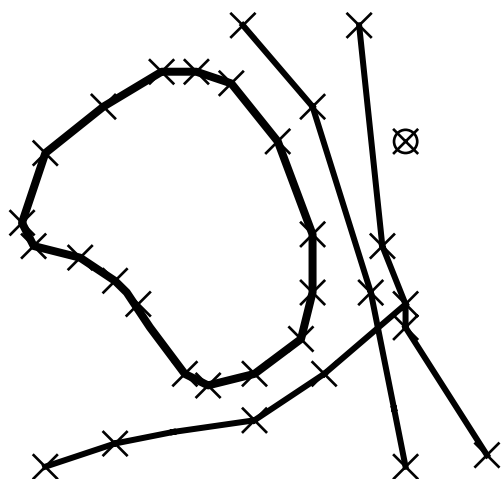
mint egyenesek folytonos halmazai értelmezhetők,

a területi objektumok

explicit értelmezéséhez és különösen a területekkel való folytonos lefedéshez még tovább kell fejleszteni modellünket bizonyos topológiai megfontolásokkal.

Spagetti modell

A módszert a digitális térképezés kezdetén alkalmazták, napjainkban már inkább csak történelmi, módszertani jelentősége van.



Pont-táblázat					
ID	X	Y	Ép. Típus	Tul.	...
1	1223,426	6666,789	templom	Róm.Kat.	

Vonal-tábla						
ID	X1	Y1	X2	Y2	Út. Típus	Száma
1	111,111	1111,222	111,333	1111,444	Főút	70
2	111,333	1111,444	222,333	2222,444	Főút	70
3	333,111	3333,222	333,333	3333,444	Főút	8
4	444,111	4444,222	444,333	4444,444	Telekhatár	2224
5	555,111	5555,222	555,333	5555,444	Telekhatár	2224
6	666,111	6666,222	666,333	6666,444	Főút	8

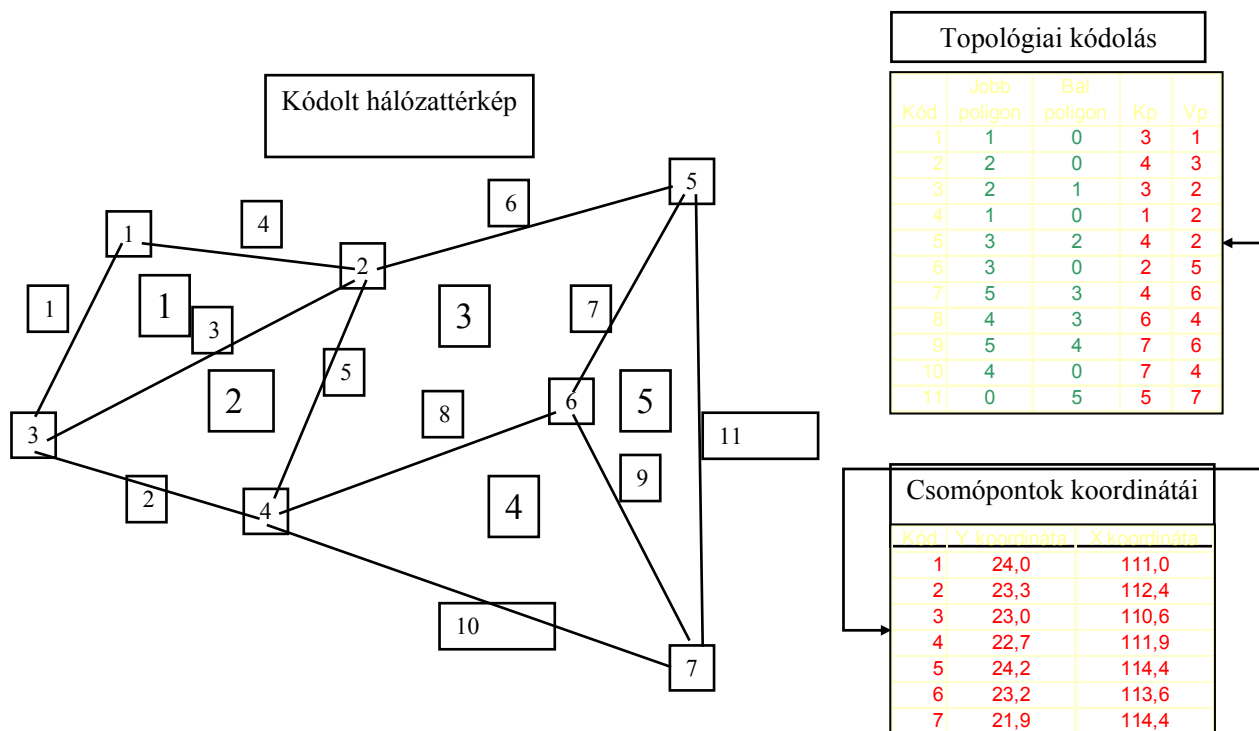
11.10. ábra A spagetti módszer

Lényege az, hogy minden egyes vonalas objektumot egy a vonal töréspontok koordinátáiból álló sorozat (adatmondat vagy sztring) ábrázol a gép memóriájában.

A módszernek több hátránya van, a legnagyobb az, hogy a koordináták keresése szekvenciális s ez lelassítja a műveleteket különösen a szerkesztési-javítási eljárásokat. A másik probléma a területek digitalizálásánál lép fel.

Topológiai modell

A topológia egy olyan matematikai tudomány, mely bizonyos geometriai tulajdonságokból kiindulva, azok általánosítása alapján, algebrai törvényszerűségeket határoz meg. **A geometriai topológia a téralakzatok azon tulajdonságait vizsgálja, melyek nem változnak az idomok szakadásmentes torzítása során.** Ilyenek a *szomszédság, folyamatosság, tartalmazás.*



11.11. ábra Topológiai adatmodell

Míg a spagetti modell csak vonaldarabokkal operál, a topológiai modell a topológiai törvényszerűségek kiaknázását lehetővé tevő adat típusokat is értelmez. Bár a topológiai modell is metrikus térben helyezkedik el, melynek alapja a koordinátás pontok halmaza s a köztük definiált távolság fogalom, a pontok a topológiai struktúra felépítésében játszott szerepük alapján különböző típusokra oszthatók, ezek:

- a. önálló pont;
- b. lánc (ív) részét képező pont;
- c. csomópont.

Az **önálló pont** a valóság modellezése során kapott olyan objektumok leírására szolgál, melyek területi kiterjedése elhanyagolható. Ilyen objektumok lehetnek a kutak, források, geodéziai alappontok, tv-, vagy rádióadó antennák, hidrolóbuszok, kémények, vezetéktartó oszlopok stb.

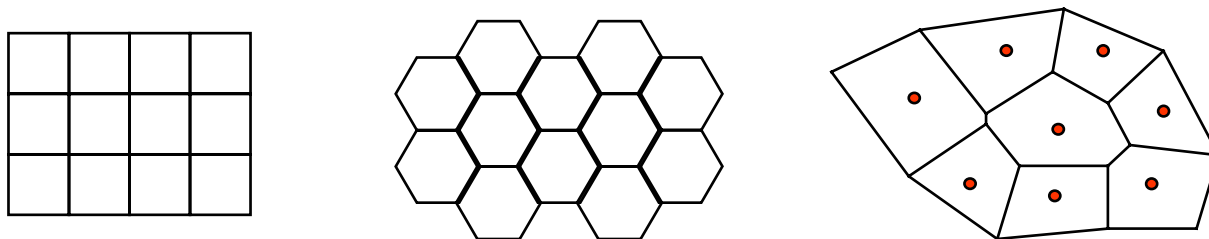
A terepi objektumok egy másik csoportja **vonalakkal** modellezhető, ilyenek a vízfolyások, utak, vasutak, csővezetékek, föld alatti és föld feletti kábelek stb. A vonalas objektumok **töréspontokat** tartalmaznak, melyek valamely (általában lineáris) törvényszerűsége alapján alapuló összekötése szolgáltatja a vonalas objektumot. Míg azonban a spagetti modellben a vonalat alkotó pontok egyenrangúak, addig a **topológiai modell a vonalakban kétféle ponttípust különböztet meg**: a láncolatot (ívet) alkotó **töréspontot** és a **csomópontot**.

A töréspontok valamely egymásutánja alkotja a **láncot**, mely mindig két csomópont között helyezkedik el és az egyik csomópontból a másikra mutat, **azaz a lánc irányított**. A csomópontok a láncok végein helyezkednek el, ez vagy abból adódik, hogy a vonalas objektumnak a csomópontban vége van, vagy abból, hogy a csomópontban az objektum elágazik, vagy más vonalas objektummal metsződik. Következésképpen egy vonalas objektum általában kettőnél több csomópontot is tartalmaz.

A topológiai modell fő előnyei a spagetti modellel szemben elsősorban a tárolás, lekérdezés és karbantartás területén jelentkeznek.

11.3.2. Raszteres-teszellációs modellek

A vektoros ábrázolás a hagyományos ceruza-, vagy tusrajzból ered, hisz a toll is elemi egyenes darabok összegeként húzza meg azokat a vonalakat, melyek a térbeli objektumokat alkotják illetve határolják. A teszellációs modellek lényegében a fényképek és a számítógép monitorok ábrázolásmódját választották mintaként a grafikus adatok modellezéséhez.



11.13. ábra Tesszelációk

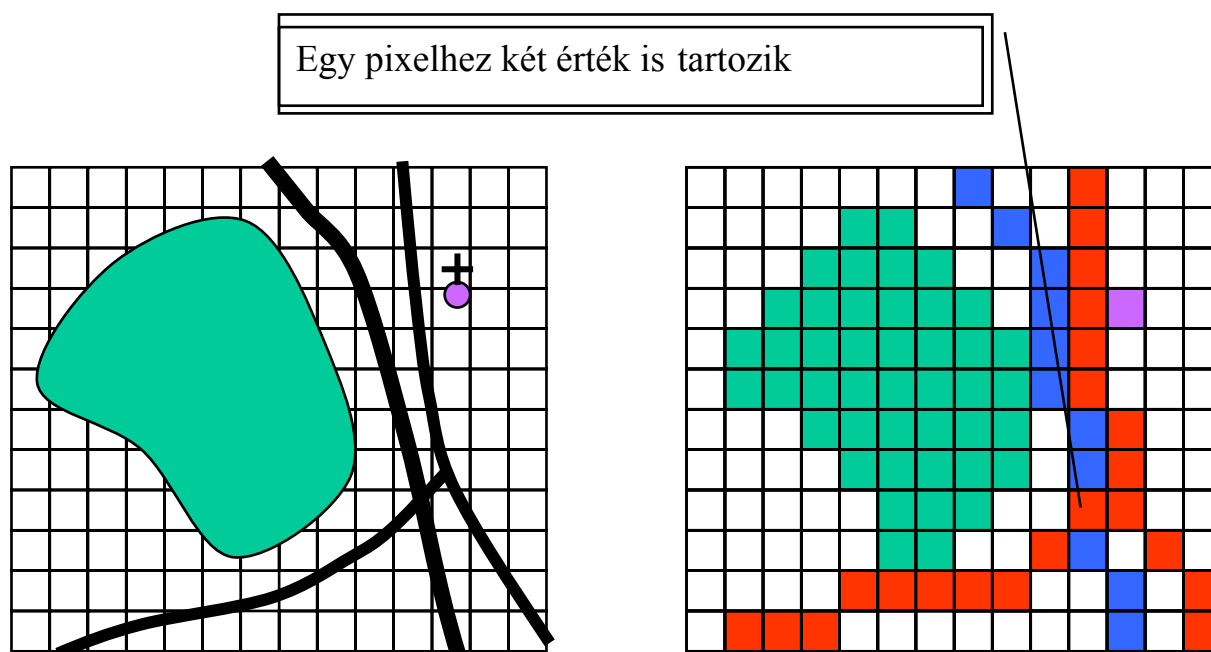
A módszer lényege, hogy a kérdéses területet felosztja elemi területekre oly módon, hogy azok folyamatosak és kihagyásmentesek legyenek. Az objektumok képét a benne foglalt elemi területek befekettítésével (színezésével) oldja meg a modell.

Az elemi felületek a legtöbb alkalmazásban szabályos idomok, négyzetek és háromszögek, de esetenként hatszögek is játszhatják ezt a szerepet. A tesszelláció hierarchikus modellekbe szervezhető.

A legáltalánosabb esetben a képbontási módszer két alapvető tulajdonsággal kell, hogy rendelkezzen:

1. *A partíció (alapelem) tetszés szerint ismételhető kell, hogy legyen azért, hogy segítségével bármilyen méretű képet le lehessen írni;*
2. *A felbontás tetszés szerinti finomítása az alapelem korlátlan tovább bonthatóságát igényli.*

Szabályos tesszellációról akkor beszélünk, ha az elemi (atomi) felületek szabályosak: oldalai és szögeik egyenlőek. Az elemi felületekből (atomokból) nagyobb egységek (molekulák) szervezhetők, ezek azonban nem feltétlenül konformak a felületelemekkel. Amennyiben a konformitás fennáll, a felosztást **hasonlónak** nevezzük.



11.14. ábra Térképábrázolás

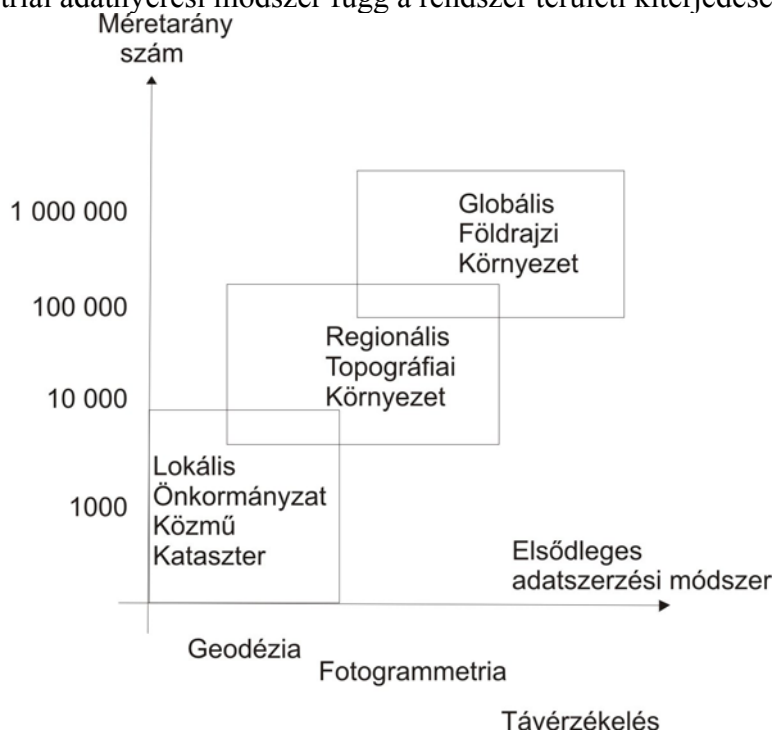
11.4. A térinformatika fontosabb eszközei, alkalmazása

A számítógépek a digitális formában leképezett különböző adatokból információt, tudást állítanak elő. A különböző alkalmazások speciális – adott feladat hatékony megoldását támogató – adatokat, feldolgozó, alkalmazói szoftvereket és megjelenítési módszereket igényelnek. A térinformatikai rendszerek összetett funkciói tovább gazdagítják a hardvereszközök választékát. Az adott munkahely technológiai jellege (adatgyűjtés, adatkezelés, elemzés, megjelenítés) eltérő hardver eszközök, erőforrások használatát igényelheti. (11.15. ábra)

Adatgyűjtés	Adatkezelés	Szerkesztés	Megjelenítés
- Számítógép - Másodlagos adatnyerés eszközei * Digitalizáló * Szkenner - Elsődleges adatnyerés eszközei * Elektronikus tahiméter * GPS vevő * Képrögzítő eszköz, műhold * Digitális fotogrammetriai munkahely * Mobil térképező rendszer	- Számítógép - Mágneslemez tárolók - Mágnesszalagos tárolók - Optikai tárolók	- Számítógép - Nagyfelbontású grafikus képernyő - Egér - Tablet	- Számítógép - Printer - Plotter

11.15. ábra A térinformatikai rendszerek hardver eszközei

A térinformatikai rendszerek fontos feladata az adatgyűjtés. A geometriai és az attribútumadatok különböző módszerekkel nyerhetők. Az adatnyerés módja elsősorban a térinformációs rendszer alkalmazási területétől, felépítési elvétől, a rendelkezésre álló adatforrásoktól és az adatsűrűségtől függ. A térinformatikai rendszerek létrehozásakor a geometriai adatnyerési módszer függ a rendszer területi kiterjedésétől is. (11.16. ábra.)



Forrás: Detrekői – Szabó (2002)

11.16. ábra Térinformációs rendszerek területi kiterjedésének és az adatnyerési módszerek kapcsolata.

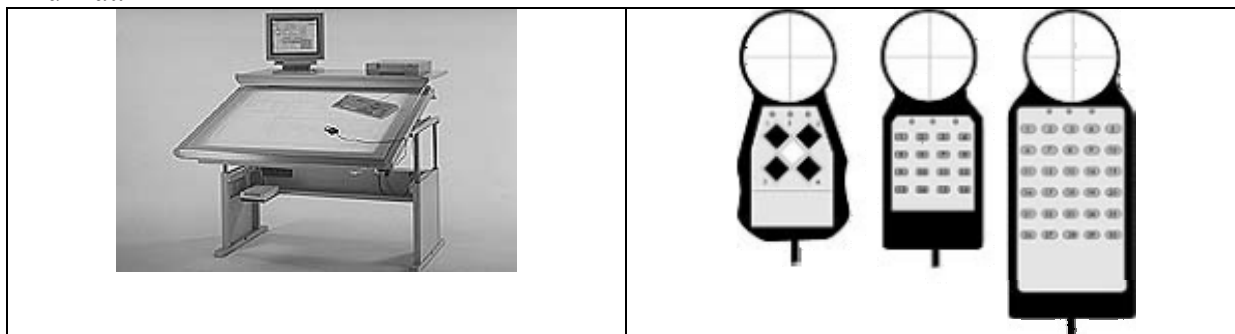
11.4.1. A digitalizálás

A térképek a területfüggő információk felhalmozott együttesei. Ha ezeket az analóg adatbázisokat digitális térinformatikai rendszerünkben fel akarjuk használni, úgy digitalizálnunk kell a térképeket. Amikor azonban kiválogatjuk a digitalizálandó térképeket több dologra figyelemmel kell lennünk.

Az első figyelembe veendő körülmény a **térkép státusa**. A térkép státusa alatt azt értjük, hogy a kérdéses térkép a földmérési-térképészeti hatóság esetleg más állami irányító szervezet által hivatalosan elrendelt térképmű **aktuális** példánya-e vagy sem.

A hivatalosan elrendelt térképmű ugyanis *szigorú szabványok szerint készül*, ezért pontosságára mind a helyzeti, mind a leíró adatok vonatkozásában megbízható mérőszámokkal rendelkezünk. Az aktualitás két szempontból is érdekes: egyrészt mivel a térképek szabványai az idővel változnak az aktuális szabvány csak az aktuális térképre érvényes, másrészt a gyorsan változó valóság, különösen az *iparilag fejlettebb* körzetekben gyorsan elavulttá teszi a térképtartalmat, ezért a már hivatkozott szabványok előírják, hogy mely területeken milyen térképeket milyen gyakorisággal kell felújítani.

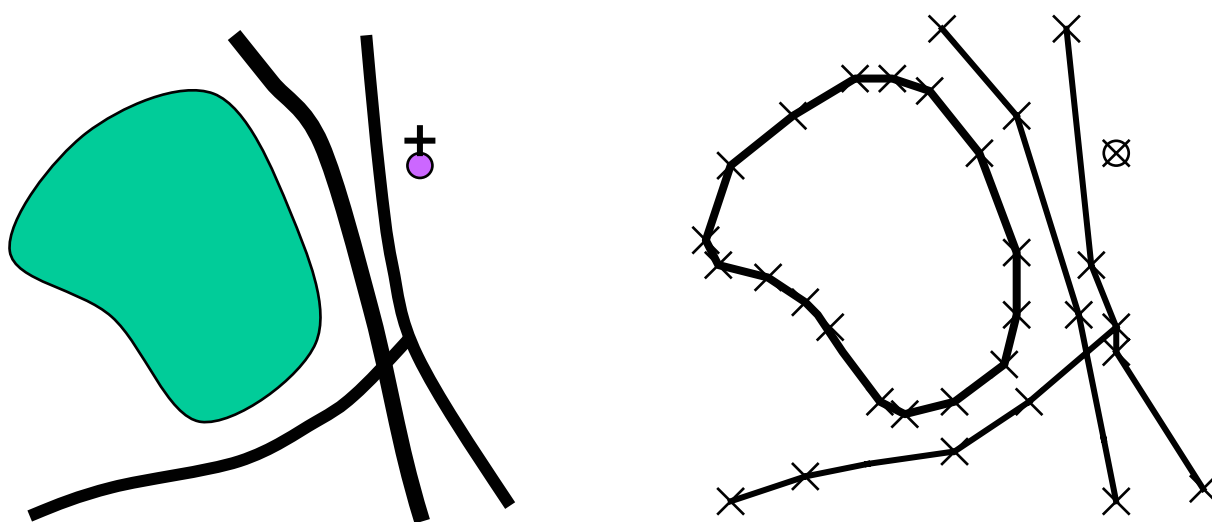
A kézi digitalizálás eszköze a számítógéphez kapcsolt digitalizáló tábla vagy tablet az irányzó (pozicionáló) eszközzel a **kurzorral**. A digitalizálást valamely GIS vagy CAD szoftver modul támogatja és ez jeleníti meg a ledigitalizált elemeket a számítógép képernyőjén is. Általában minden GIS szoftver rendelkezik digitalizáló modullal, de nagy különbségek vannak az egyes szoftverek digitalizálási hatékonyságában. Tömeges digitalizálás esetén igen fontos, hogy a legmegfelelőbb szoftver támogatásával végezzük a munkát.



11.17. ábra Állványra szerelt digitalizáló tábla és különböző számú vezérlógommbal felszerelt kurzorok

Maga a digitalizáló tábla (11.17. ábra) különböző méretű lehet A3-tól A0-ig. A kisebb táblákat tabletnek nevezik. Szerkezetileg a tábla műanyagba ágyazott szabatos, sűrű drótháló, mely különböző elemeiben (rácsszemeiben) attól függően indukálódik feszültség, hogy a kurzor szátkeresztjét koncentrikusan körülvevő elektromágneses tekercs hol helyezkedik el a táblán. A tabletek tömeges perifériaként való megjelenése ahhoz kapcsolódik, hogy a Windows előtti környezetben az egyszerű programvezérlést a tabletekre erősített grafikus menük segítségével oldották meg. Ez az alkalmazás azonban nem igényelt nagy felbontást és pontosságot, ezért a piacon lévő tabletek jelentős része alkalmatlan a szabatos digitalizálásra. A digitalizáló táblák és megfelelő tabletek digitalizálási hibája eszköztől függően 0.1 mm. - 0.02 mm. A hiba nem csak a háló és a tekercs kialakításától, hanem a szátkereszt formájától, elhelyezésétől is nagymértékben függ.

Bármilyen lelkiismeretes is a digitalizálást végző munkaerő, a digitalizálásba óhatatlanul hibák is előfordulhatnak. Ezek ellenőrzése és kiszűrése a digitalizált állományból átlátszó fóliára készülő próbarajz és az eredeti térkép egybevetésével történik.



11.18. ábra Digitalizálás, vektoros állomány létrehozása

11.4.2. Szkennelés

Az első szkennerek nagymértékben térképészeti igényekre jöttek létre, s ez meghatározta azt az alapvető követelményt, hogy alkalmasak legyenek nagy (1 m²-nél nagyobb) térképek digitalizálására, s egyben kiemelten törekedjenek a leképezés geometriai pontosságának biztosítására.

A nagyméretű térképeket dobra erősítették, mely a szkennelés folyamán forgott, a megvilágító berendezéssel kombinált szenzoros optika pedig a dob tengelyével párhuzamos haladó mozgást végzett. A két mozgás kombinációjából kialakuló letapogatási pálya csavarvonal jellegű volt. Amennyiben nem papír, hanem átlátszó fólia szkennelése volt a feladat, úgy a megvilágítást a tejüvegből készült dob végezte.

A következő fejlődési szakaszban (80-as évek közepe) a szabatos szkennelésre a síkágyas plotterek mintájára készült síkágyas szkennereket alkalmazták elsősorban. A kocsira szerelt, színenként alkalmazott, egyedi vagy sordetektorok és megvilágító berendezéseik kétirányú mozgását egy mozgó híd *X* irányú illetve a hídon lévő kocsi *Y* irányú mozgásával érik el a léptetőmotorok.

Napjainkban mind a dob mind a síkágyas megoldásra találunk példákat, az előbbieket inkább a térképek, utóbbiakat inkább a fényképek digitalizálására használják. Mégis a több ezer elemi szenzort egyesítő sorszenzorral ellátott korszerű szkennerek többsége - a nyomtatókhoz hasonló módon - 'beszívja' a letapogatandó térképet vagy műszaki rajzot, s ily módon tetszőleges hosszúságú rajz szkennelésére képes. Ezek a berendezések csak a szélességre adnak típusuktól függően más-más korlátot. A 11.19. ábra két ilyen szkennerek külső megjelenését illusztrálja.



11.19. ábra Korszerű asztali szkennerek és korszerű állványos szkennerek

Nem hagyhatjuk említés nélkül, hogy a napjainkban tömegesen forgalomba kerülő A4-es laptájkennereket, de különösképpen az igen olcsó A4-es kézi szkennereket nem térinformatikai adatnyerésre találták ki, hanem elsősorban szövegek számítógépbe vitelére (ebben a vonatkozásban használhatók, ha hagyományos nyilvántartásokat kívánunk adatbankosítani), valamint különböző rajzi és fényképi input adatok digitalizálására illusztratív jellegű számítógépes alkalmazásokban.

11.4.3. Fej feletti digitalizálás

A fej feletti digitalizálás lehetőségét az a tény teremtetette meg, hogy egyes szoftver készítő is kénytelenek voltak elismerni annak a gondolatnak a helyességét, hogy **a jövő a hibrid - rászteres-vektoros rendszereké.**

Ez az **első stádiumban** azt jelentette, hogy az alapvetően rászteres rendszerek képesek kezelni bizonyos vektoros objektumokat, hasonlóképpen a vektoros rendszerek lehetővé tették, hogy *háttérként* a vektor rajz mögé bevigyék a kérdéses terület ortofotó térképét.

A **következő lépésben** a 'háttér' újabb funkciókhoz jutott, többek közt ahhoz, hogy a képernyőn az egér vezérelte szátkereszttel digitalizálni azaz vektorizálni lehessen a 'háttér' rászteres képét. Természetes, hogy ezt akkor is meg lehet tenni, ha a háttér nem ortofotó, hanem szkennelt térkép.

Felmerül a kérdés, hogy miért előnyösebb a képernyőn digitalizálni mint a térképen. **Az első különbség a megvilágításban és a parallaxisban van.** A papír térképet jól megvilágítani nagyon nehéz, a kurzor vagy az operátor gyakran árnyékot vet a szátkeresztre, és ha a szátkereszt nincs a papír síkjában, ez a helyzet pedig igen gyakran előfordul, akkor parallaxis lép fel azaz más szögből nézve a szátkereszt más térképi pontra mutat. A képernyőn ez a két probléma nem létezik. További előnye a képernyőnek a **zoomolási lehetőség**, mely jelentősen növeli a pontosságot. Végül igen előnyös, hogy a képernyőn a digitalizált és még nem digitalizált vonalak **egymás mellett, de különböző jelöléssel jelentkeznek** (például a már digitalizált vonalak villognak) s így egyszerűen biztosítható, hogy semmi se maradjon ki a digitalizálásból.

A módszer jelentőségét nagymértékben növeli, hogy több olyan országban, melyekben még nem kezdték meg a vektoros digitális alaptérképek forgalmazását, mint például Csehországban, az alaptérképek **rászteres digitális formában** már a felhasználók rendelkezésére állnak.

11.4.4. Fotogrammetriai módszerek

Napjainkra a különböző irányzatok, technikák, műszerek gyakorlatilag egybeolvadtak és átalakultak a **digitális fotogrammetriai munkaállomásnak** nevezett feltupírozott, speciális perifériákkal ellátott, gazdag szoftver választékú, munkaállomás osztályú számítógéppé. A fotogrammetriai feldolgozás tehát átalakult az optikai mechanikai berendezések manuális vezérléséből a feldolgozó programok vezérlésébe. Mivel nem lehet a célunk e rövid összefoglalásban egy vagy több feldolgozóprogram ismertetése, az input és output felvázolása mellett a kettő között elhelyezkedő alap összefüggésekre igyekszünk a figyelmet fordítani.

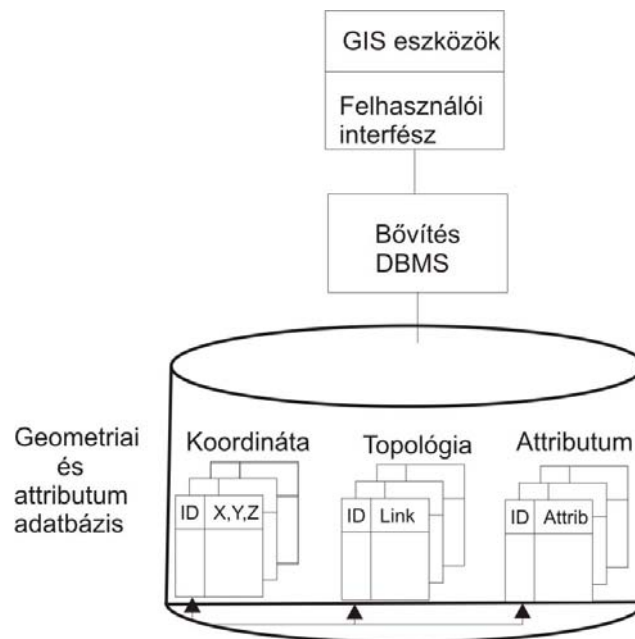
A légi fotogrammetria bemenő adatai az **analóg vagy digitális légifényképek**. Az analóg felvételeket a **légi felvevő kamerák** állítják elő, melyek a felvevő repülőgépekben kerülnek rögzítésre. A kamerák bonyolult automatikus rendszerek, melyek elemei a legmagasabb műszaki színvonalon kerülnek kialakításra. A kamerák feladata, hogy a repülési paraméterekhez (repülési magasság, repülési sebesség) kapcsoltan meghatározott időközökben, a megvilágítási körülményeknek, a sebességnek, a film érzékenységeknek, stb. megfelelő időtartamra megvilágítsák a filmet.

E közben a kamera objektívje nagyfelbontású, nagy fényerejű, lencsehibákra korrigált, kis elrajzolású kell hogy legyen. A fényképezés pillanatában a filmnek szigorúan sík felületűnek kell lennie (ezt vákuumos leszívással érik el), és az előhívás, szárítás folyamatában eredeti alakját nem szabad megváltoztatnia. A fotogrammetriai feldolgozása digitális fotogrammetriai munkaállomásokon megy végbe. Ezek a számítógépek pedig digitális formában várják az inputot. A digitális bemenő adatokat kétféle képen hozhatjuk létre: vagy digitalizáljuk a fényképet vagy analóg helyett digitális kamerát használunk.

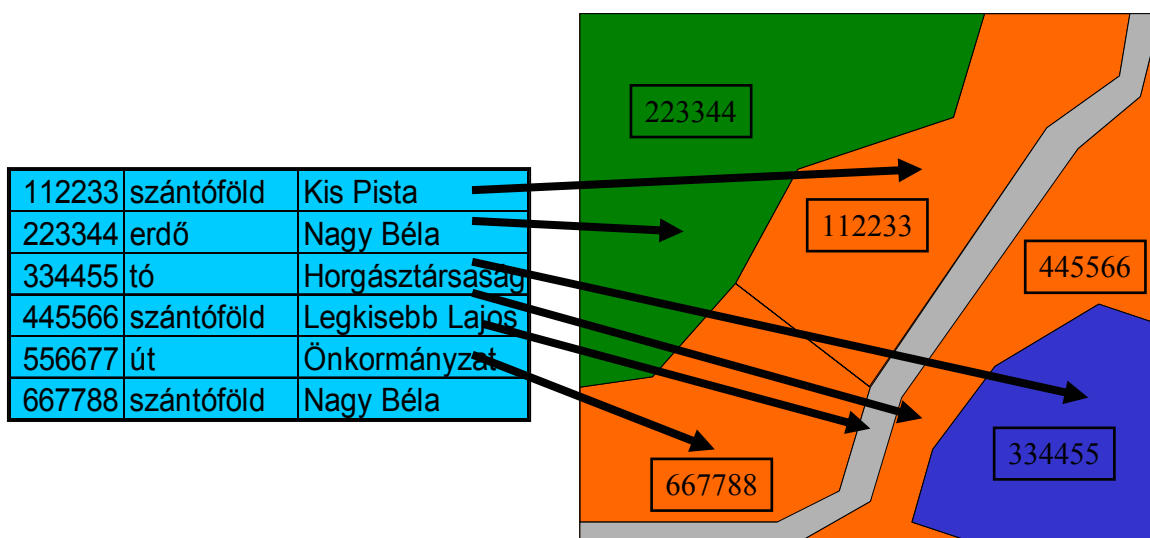
Bár a fénykép digitalizálók nem olcsó berendezések napjainkban mégis inkább ezeket s hozzájuk kapcsolva a hagyományos légi felvételeket részesítik előnyben a szintén nem olcsó, de még nem teljesen kiforrott digitális légi kamerákkal készült digitális fényképekkel szemben.

11.4.5. Térinformatikai rendszerek, az adatok kezelése

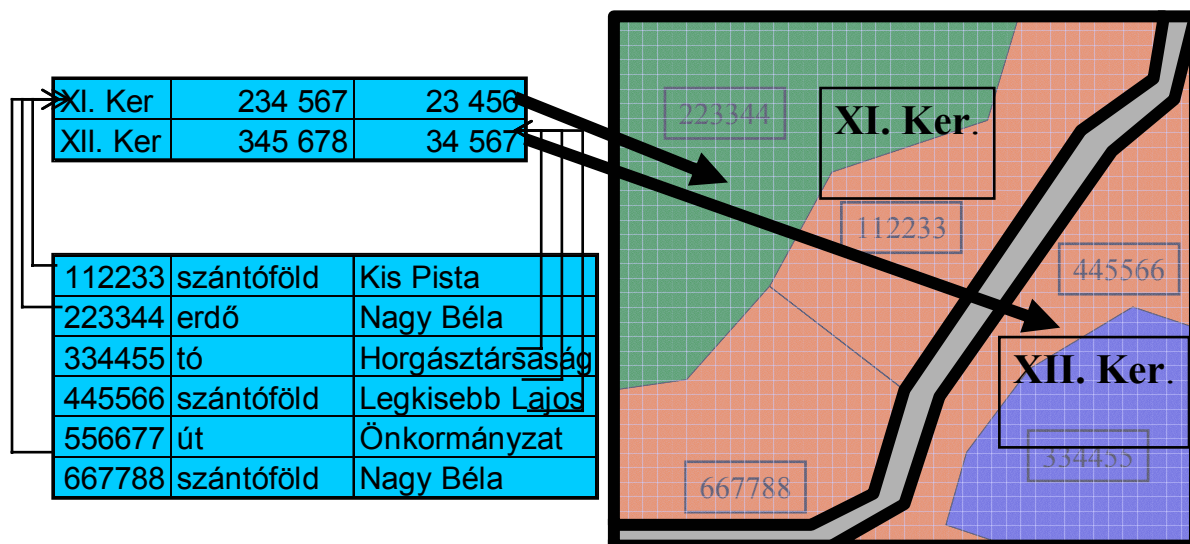
A térinformatikai alkalmazások üzemszerűvé válásával a rendszerekkel kapcsolatos konzisztencia- és adatelérési elvárások fokozódtak. Kézenfekvő lehetőségként kínálkozott a geometriai és szakadatok kezelését is egy adatbázison belül megoldani. A térinformatikai adatok egységes konzisztens kezelését biztosító integrált rendszerek a térinformatikai objektumok tulajdonságait egy adatbázisban, de három eltérő halmazban képezik le. Relációs adatbázis esetén független relációs táblákba kerülnek az objektumok geometriai tulajdonságainak metrikus, koordinátaállományai, külön relációs táblába kerülnek a geometriai elemek kapcsolatait tartalmazó topológiai állományok, és egy harmadik relációs halmazba kerülnek tárolásra az objektumok attribútumai (11.20. ábra)



11.20. ábra Geometria és szakadatok kezelésének integrált rendszere



11.21. ábra Megfeleltetés



11.22. ábra Hierarchikus kapcsolat

11.5. Precíziós gazdálkodás

A térinformatika különböző lehetőségeket ajánl a gazdálkodók számára termelésük növelésére, a költségek csökkentésére, és a földművelés hatékonyságának növelésére. A kézi számítógépek táblákon történő térképészeti alkalmazásától a gazdálkodó irodájában lévő számítógépeken végzett termelési adatok tudományos elemzéséig a térképészet fontos szerepet játszik.

A precíziós termesztés a táblán belüli helyi viszonyokhoz és igényekhez igazodó termesztést jelent, amelynek szerves része a szabatos mérés és ehhez kapcsolódóan a pontosan szabályozott beavatkozás. Ezen termesztési technológia kifejlesztésének és terjedő alkalmazásának alapvető oka: a termesztés gazdaságosságának fokozására irányuló törekvés, valamint a környezetvédelem, a környezetkímélő gazdálkodás szükségessége és az ellenőrzött minőségű termény iránti igény.

A precíziós termesztési rendszer fő összetevői: a szántófieldi termesztési technológiák, a mezőgazdasági gépek és automatikák, a különböző szoftverek és információs rendszerek, melyeket a szántófieldi termesztési műveletek és folyamatok irányítására integráltak.

A precíziós termesztés irányítási rendszere a következő folyamatokra bontható fel: adatgyűjtés, adatfeldolgozás, döntéshozatal és beavatkozás. Ezek a folyamatok számos részfolyamatra oszthatók tovább, melyekben különös szerepet kap a mérés, a vezérlés, a szabályozás és a számítógépes felügyeleti irányítás.

11.5.1. Adatgyűjtés

A gazdálkodók és akik érdekeltek a tábla szintű termelésben elsődleges ismeret igényük aziránt jelentkezik, hogy hogyan tudnak a gazdaság természeti erőforrásaival a legjobban gazdálkodni. Az ilyen szakértői ismeretek, technológiák segítséget nyújtanak a

földműveléssel kapcsolatos fontos döntések előkészítésében és megvalósításában. Azért, hogy meghozzák a helyes döntéseket a gazdálkodóknak a legjobb információkkal kell rendelkezniük a táblákról. A földrajzi adatok, amelyek több éve készültek főleg az irodában áll rendelkezésre. Napjainkban az új információ technológiák lehetővé teszik gazdálkodók számára a térinformatika táblákon történő alkalmazását, kézi számítógépek segítségével a térképek aktualizálását, vizualizációt, adatok lekérdezését.



11.23. ábra Adatgyűjtés mobil eszközökkel

A kézi számítógépes megoldások lehetővé teszik a táblákról készült képek tanulmányozását, a helyszínen, azaz a saját táblájukon történő tartózkodás közben. A szerkesztési lehetőség segítséget nyújt a gazdálkodó számára a növény-növekedést befolyásoló tényezők megadására, mint például a kártevők által okozott kár, a tápanyag elégtelenség és a vízhiány. A táblák ilyen jellegű adatainak gyűjtési megoldásai kármegelőzési lehetőséget biztosítanak számunkra.

A gazdálkodó a kézi számítógépet az Internethez kapcsolhatja olyan további input adatok letöltése céljából, mint például az időjárás előrejelzés. A táblaszintű vagy résztábla szintű adatok real-time (valós idejű) gyűjtése ma már realitás. A gazdálkodók ezen lehetőség alkalmazásával képesek az aktuális helyzetnek megfelelő korrektd helyszíni, adott földrajzi ponthoz kapcsolódó adatok bevitelére majd később az összegyűjtött adatok elemzésére.

11.5.2. Gazdálkodási adatok integrációja

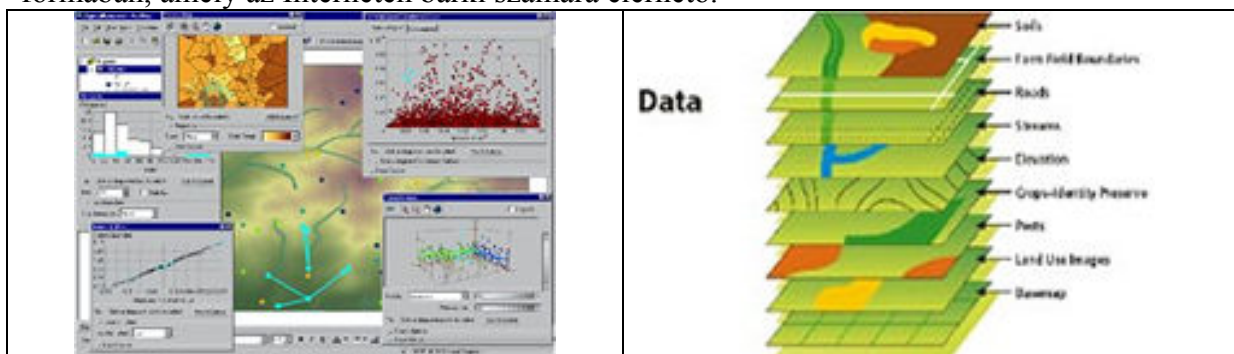
Minden potenciális GIS alkalmazás a mezőgazdaságban különbözik, azonban van néhány alapvető elv, amelyek azonosak maradnak. A mezőgazdaságban dolgozó felhasználók számára szükséges az adatok elérése a döntéseik meghozatalához, ezekben a különböző adattípusokat kezelő különböző eszközök segítik őket, amelyek teljesen integrált szoftver megoldásokkal hatékony segítséget nyújtanak a döntéshozási folyamatban.

A GIS alkalmazások adaptációja a mezőgazdaságban új területnek tekinthető, annak ellenére, hogy légi- és műhold felvételeket már több mint 70 éve használnak. Az ilyen típusú adatok általában raszter típusú adatok, amelyek diszkrét cella vagy pixel adatokból állnak. Az adatrétegek pontokból, vonalakból és poligonokból állnak. A raszter adatok és a vektor adatok kombináltan használhatók egymással, de ezek kezelése és megjelenítése különböző eszközöket igényel.

A légi és a műholdfelvételek képelemzésének kiterjesztése a befektetések jelentős megtérülés növekedéséhez járulhat hozzá a gazdálkodók számára a szükséges eszközök használatával. Az alkalmazói szoftverek képesek a terméshozamok becslésére (előrejelzésére), figyelembe véve a táblákon jelentkező termést befolyásoló tényezőket, mint például a kártevők által okozott kár, vagy a talaj erózió.

A mezőgazdasági adatok gyűjtése és szétosztása a nagy földrajzi távolságok miatt gyakran nehézségekbe ütközik. Az Internet hálózat segítségével azonban ezen adatok könnyen továbbíthatók, de hogyan származik előnye ebből a gazdálkodónak?

A vezető kaphat adatokat a központi irodában, vagy kaphat real-time időjárási adatokat és növelheti ezek megbízhatóságát az Internet segítségével. Potenciálisan probléma származhat a számítástechnikai ismeretek hiányából. Ezért több térinformatikai szoftverrel könnyen használható eljárások valósíthatók meg. Az Internetes térképszerver lehetővé teszi olyan Web oldalak készítését amelyek mezőgazdasági adatokat tartalmaznak térkép formában, amely az Interneten bárki számára elérhető.



11.24. ábra Adatintegráció, adatelemzés

11.5.3. Adatok interpolálása

Nem minden táblaszintű adat hasznos a gazdálkodó számára. A táblákon összegyűjtött adott pontokra vonatkozó adatokat fel kell dolgozni, hogy folytonos adatréteg információkat kapjunk. Ezért sok egyedi pont interpolációjára van szükség, hogy teljes felszín térképet kapjunk, melyet az eredeti pontokból térbeli statisztikai eljárásokkal nyerünk. Ilyen pontok lehetnek növény terméshozamokra vonatkozó adatok, melyeket a kombájnok mérőeszközeiből nyerünk, a táblán manuálisan gyűjtött talajminták adatai vagy vízellátottságra vonatkozó adatok. Ezeknek a pontoknak az interpolálása még hasznosabb információkat nyújtanak a gazdálkodók számára. Ilyen módon készíthetünk hozam térképeket, tápanyag térképeket, szennyező anyagokra vonatkozó térképeket.

A tábla elemzője agronómiai szempontból azonosította a problémás területeket, a térinformatikai rendszer alkalmas az adatok közötti kapcsolatok megmutatására. A hozam monitoring adattérképek ezeken általában alacsonyabb hozamszintet mutatnak. Az alacsony hozamszintek okának azonosításával növelhetjük a terméshozamokat. A térinformatikai rendszer lehetővé teszi a felhasználónak a kapcsolatok feltérképezését, a hozam, a műtrágyázás illetve a kártevők elleni védekezés között.

Egy földrajzi helyhez kapcsolódó mezőgazdasági adatok gyűjtése természetesen csak kezdete a munkaszervezés és termelékenység növekedésére vonatkozó döntéshozatalhoz. A kulcskérdés a gazdaságos hozam növelésének, ha használjuk és feldolgozzuk ezeket az információkat, mellyel növelhetjük a hatékonyságot, hogy az adott táblákra megfelelő növényeket választjuk, vagy az input ráfordításokat helyesen határozzuk meg az adott évszakban. A gazdaság adatainak elemzése segítheti a gazdálkodót ezen döntések meghozatalában, amellyel növelheti a jövedelmezőséget és csökkentheti a környezeti hatásokat. A földterületre vonatkozó adatokat (jellemzőket) úgynevezett rétegekbe szervezhetjük.

Ezek a rétegek tartalmazhatnak hidrológiai jellemzőket, talajjellemzőket, szennyezettségi adatokat, stb. Ha ezeket az adatokat létrehoztuk, lehetőség van hatékony térbeli elemzésekre. A térinformatikai szoftverek számos elemző eszközzel rendelkeznek. Az információs rétegek összehasonlítási lehetősége gyakran nem elegendő, mivel szükség lehet nem kapcsolódó adattípusokra, vagy a gazdálkodó gyakorlati tapasztalatára. Szemléletes példái ennek a vetésforgókkal kapcsolatos döntések.

11.5.4. Adatelemezés

További, bonyolultabb elemzések lehetségesek a geo-statisztikai elemzések segítségével, amely a térinformatikai rendszereknek egy kiterjesztése, ezek segítségével különböző interpolációkat és statisztikai próbákat végezhetünk egy adott területen egy bizonyos jellemzők előfordulására vonatkozóan. Ez segíti a gazdálkodókat, hogy csökkentsék a drága adatgyűjtéseket és helyette megbízható, ténystatisztikai módszerekkel határozzák meg a térképhez rendelt bizonyos jellemzők értékét. Ilyen példa lehet a talajminta vételezés.

11.5.5. Eredmények megjelenítése

A képek általában több információt hordoznak, mint a szöveges leírások, de egy intelligens térkép még többet ér. A mai gazdálkodás még inkább az üzleti szabályok és kormányzati szabályozások függvénye, összehasonlítva a korábbi időszakokkal. A gazdálkodóknak nemcsak a növények biztonságos termesztéséről, de számviteli, agrárszabályozási, támogatási, adózási ismeretekkel is kell rendelkezni, és ami szintén fontos a növények biztonságáról. Ezeknek a tényezőknek a figyelése és feldolgozása sok időt igényel.

11.6. Országos mezőgazdasági térinformatikai rendszerek

A térinformációs rendszerek mindegyikéhez szükség van földfelszíni geodéziai pontokban vagy szatellitpályák által fizikailag megvalósuló koordináta keretrendszerre, vetületi keretre és a helyfüggő információ helyzetének megítélését viszonyítási alapon lehetővé tévő, a legáltalánosabban használt térbeli objektumokat tartalmazó térképekre.

Az információtechnológia világában alapkövetelmény, hogy ezek az alapadatkörök digitálisan álljanak rendelkezésre, hogy tartalmukban és méretarány (felbontás) kiterjedésüket illetően teljes körűek legyenek. További alapkövetelmény, hogy az alapadatkörökre vonatkozó adatminőségi jellemzők digitálisan létezzenek. A térinformatikai társadalom elvárja azt is, hogy ezek az alapadatok és minőségi jellemzőik teljes körűen felhasználhatók legyenek és erről részletes információk adjanak tájékoztatást a metaadatok szintjén interneten, illetve a digitális világban. Lényegesek továbbá a nevezett adatok felhasználására, felhasználhatóságára vonatkozó ún. adat- és árpolitikai szempontok is.

Az állam az ország térképellátását az állami földmérési alap (kataszteri térképek), azok átnézeti térképei és az állami topográfiai térképek (a továbbiakban együtt: állami térképek) készítésével, fenntartásával, korszerűsítésével, tárolásával és szolgáltatásával, illetve az e térképekről való adatszolgáltatással, továbbá légifényképekkel és űrfelvételekkel biztosítja.

Az állami térképeknek alkalmasnak kell lenniük:

- o A hatósági nyilvántartások,
- o Térinformatikai rendszerek,
- o A honvédelmi és rendvédelmi tevékenység,

- o A helyi önkormányzatok feladatai, illetve településfejlesztési és településrendezési, vagyon-nyilvántartási, információs és településirányítási tevékenység,
- o A közlekedési, a hírközlési, a vízgazdálkodási tevékenység,
- o Az infrastruktúra-fejlesztés,
- o Az agrár-és térségfejlesztési tevékenység,
- o A természet-és környezetvédelmi tevékenység,
- o A bányászati szakigazgatás által elrendelt térképészeti tevékenységek, geológiai nyilvántartások,
- o Adózási célú feladatok térbeli térképi referenciájaként.
- o A meglévő térképállományt folyamatosan fel kell újítani, a fel nem újítható térképek helyett újakat kell készíteni.
- o Új állami földmérési alaptérképet számítógépen kezelhető módon, számszerű (numerikus) meghatározással kell készíteni. Ugyanúgy kell eljárni térképfelújítás esetén is.

A következőkben néhány fontos országos térinformációs rendszert ismertetünk röviden.

11.6.1. Mezőgazdasági Parcella Azonosító Rendszer (MePAR)

A Mezőgazdasági Parcella Azonosító Rendszer (MePAR) az agrártámogatások eljárásainak kizárólagos országos földterület-azonosító rendszere. Kizárólagos abban az értelemben, hogy a földterülethez kapcsolódó részben vagy egészben európai uniós támogatások igénylése során csak ennek az azonosítási rendszernek az adatait lehet használni. Az ilyen jellegű támogatások igénylésekor semmiféle más nyilvántartás (pl. az ingatlan-nyilvántartás) adatait a MePAR adataival szemben nem lehet figyelembe venni, legyenek azok a mezőgazdasági táblák elhelyezkedésére, azonosító számára, vagy éppen a tábla területére vonatkozó adatok. A MePAR-t a kérelmezéskor maguk a gazdaságok, a kérelemkezelés és az ellenőrzés során pedig a hivatal használja, tehát a MePAR használatát a gazdálkodóknak is meg kell ismerniük.

Alapvetően azért van szükség erre a rendszerre, mert az Európai Unió jogszabályai az érintett támogatások vonatkozásában kötelező jelleggel előírják. A 2004-es támogatási évtől a MePAR biztosítja a földterülethez kapcsolódó támogatások alapját képező mezőgazdasági táblák helyének egyértelmű azonosítását, valamint adataival segíti területük egyszerű és pontos megadását. Térinformatikai rendszerben és nyomtatott térképeken, térképhelyes légifelvétel háttérrel megjelenítve állnak majd rendelkezésre a kérelmezéshez szükséges és azt segítő adatok.

A mezőgazdasági parcella (más néven mezőgazdasági tábla) egy olyan összefüggő mezőgazdasági földterület, amelyen egyetlen termelő egyetlen növényfajt (vagy növényfajta) termeszt.

gazdákhoz tartozó mezőgazdasági földterületek, a térképhelyes légifelvétel háttérén ugyanis nagyon könnyen felismerhetőek az állandó földfelszíni elemek (utak, épületek, csatornák, erdős részek, stb.).

11.6.2. A TAKARNET rendszer

A rendszer a Földművelésügyi és Vidékfejlesztési Minisztérium Földügyi és Térinformatikai Főosztálya (FVM FTF) az EU PHARE programja segítségével, kisebb mértékben pedig a svájci és a német kormány, valamint a magyar költségvetés támogatásával jelentős beruházásokat hajtott végre az ingatlan-nyilvántartás területén.

E beruházások eredményeképpen az összes tulajdoni lap adat számítógépre került, valamint 2000 júniusa óta a TAKAROS (Térképen Alapuló KAtaszteri Rendszer Országos Számítógépesítése) rendszer üzemel az ország körzeti földhivatalaiban. Ez lehetővé teszi, hogy az adatok karbantartását és az ügyiratkezelést számítógéppel végezzék. Elkészült a földhivatalokat összekötő TakarNet hálózat is, amely az ingatlan adatok távoli, elektronikus elérését biztosítja.

A fejlesztések fontos állomása a TakarNet rendszer kialakítása, amely egyrészt biztosítja a földügyi szakág hivatalainak számítógépes összekapcsolását, másrészt lehetővé teszi a földhivatali adatbázisok távoli adathozzáférést külső felhasználók (regisztrált fizető felhasználók) pl. bankok, közigazgatási intézmények, önkormányzatok, közjegyzők, ügyvédi irodák stb. számára. A TAKARNET koncepciója

- o egységes rendszerbe integrálja a földhivatali intézményeket
- o többszintű biztonsági stratégiát támogasson
- o differenciált szolgáltatások a különböző felhasználóknak
- o egyszerű, olcsó elérést biztosítson
- o kompatibilitás a kormányzati hálózattal
- o INTERNET felől elérhető legyen
- o modulárisan kiépíthető és bővíthető legyen
- o biztosítsa a hálózat központi felügyeletét, elszámolását, kezelését, hozzáférést biztonság és tartalom szempontjából
- o hasonló adattulajdonságok is integrálhatók legyenek

A TAKARNET hálózat a TAKARos NETwork rövidítése. A hálózat kialakításakor az alábbi szempontokat tartották fontosnak:

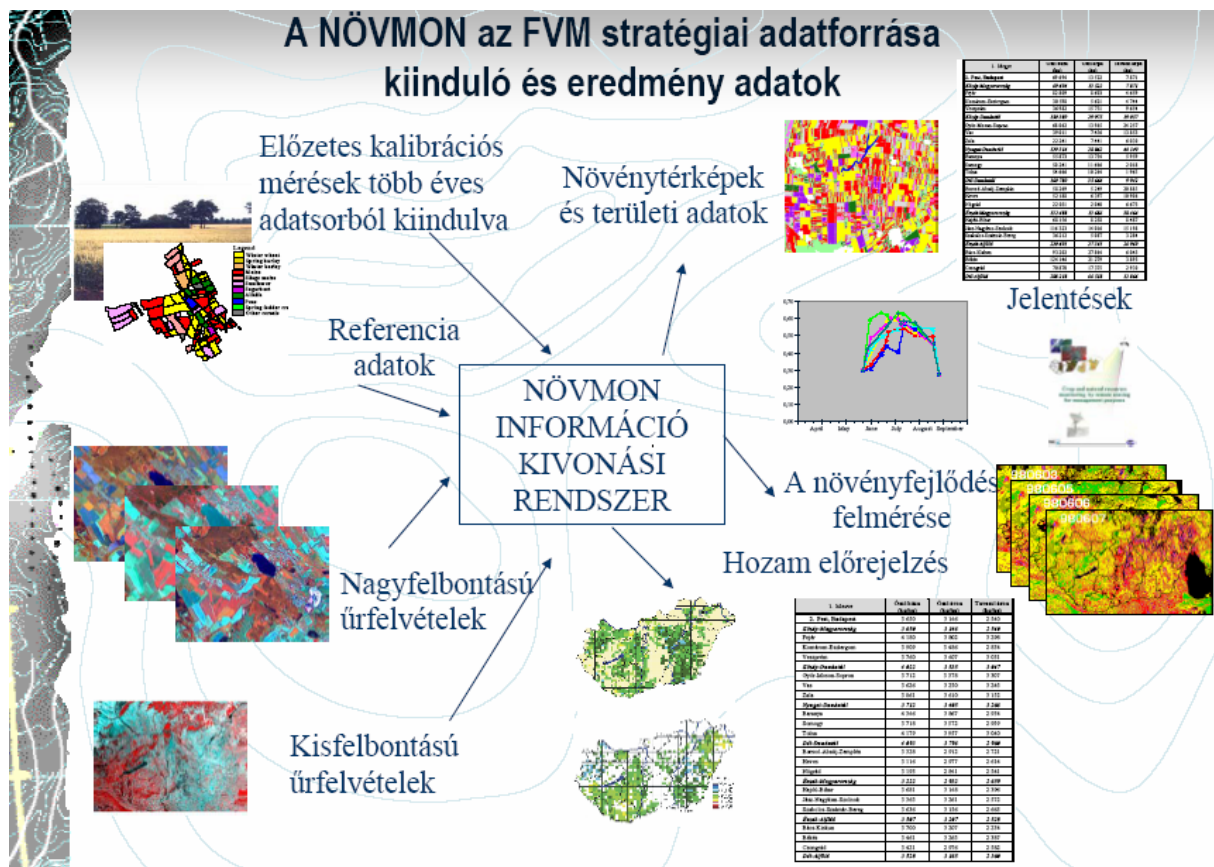
- o a hálózat egységes rendszerbe integrálja a földhivatali szektort, beleértve a FÖMI-t és a minisztérium illetékes osztályait
- o a rendszer többszintű biztonsági stratégiája az egyes felhasználók számára eltérő szolgáltatásokat a szolgáltatásokhoz rendelt biztonsági kritériumok mellett nyújtja
- o olcsó piros/kék számos behívhatósággal az ország egész területéről elérhető legyen modem, és egyszerű Internetes kereső szoftver alkalmazásával az elérési pont(ok) keresztül
- o egyszerűbb adatszolgáltatások, vagy tudakozódás céljára Internet felől is elérhető
- o egységesen felépítése miatt, igény és pénzügyi lehetőségek szerint legyen kiépíthető és bővíthető
- o a hálózatot alkotó berendezések a hálózat egy (bizonyos) pontjáról a kezelhetők, módosíthatók, beállíthatók, központilag felügyelhetők

- o egységes biztonságtechnikai hozzáférési és elszámolási rendszeren alapuljon
- o további adattulajdonosokkal is bővíthető legyen

A TAKARNET egy privát hálózatnak tekinthető, amely a fenti intézményeket kapcsolja össze megyei és területi földhivatalok, FÖMI, a Minisztérium osztályai. Ezen felül, a hálózat - megfelelő védelmi eszközökön keresztül - kívülről behívható akár modemmel is, miközben védett kapcsolóeszközökön keresztül kapcsolódik a kormányzati hálózathoz és az Internethez is.

11.6.3. Országos Távérzékeléses Szántóföldi Növénymonitoring és Termésbecslés

A FÖMI Országos Távérzékeléses Szántóföldi Növénymonitoring és Termésbecslés (NÖVMON) programja 1997-től a 8 legnagyobb területű szántóföldi növényre ad pontos termésbecslést az aratás előtt az FVM-nek. Az évente négy előre rögzített időpontban átadott jelentésekben a számszerű adatokat országos vetésszerkezeti és a terméshozam területi változását mutató térképek egészítik ki. A különböző műholdak által készített felvételek felhasználásával megfigyelt növények pontos országos és megyei terület- és terméshozam adatait, a várható össztermést tartalmazzák e jelentések.



Forrás: FÖMI

11.26. ábra A NÖVMON rendszer összefoglaló ábrája

E termésbecslő módszer kizárólag űrfelvételek kiértékelésén alapul. A módszer fizikai háttere az, hogy a műholdfelvételek a földfelszín és a növénytakaró, a mezőgazdasági táblák és táblarészek objektív, részletes, pontos, torzítatlan (sugárzási) képét rögzítik. A látható fénynél jóval szélesebb sugárzástartományt rögzítő műholdfelvételekből milyen módszerrel

lehet nagymértékben automatikusan és pontos adatokat eredményezve a földfelszíni állapot információt megszerezni!

A NÖVMON technológia a növényzet fejlődését objektíven tükröző, térben és időben reprezentatív igen hatékony mérési eljárás. A felhasznált nagyfelbontású űrfelvételekkel a szántóföldi növények elkülönítése és pontos feltérképezése 0,04-0,1 ha-os földfelszíni részletességgel valósul meg. A növényfejlődés számszerűen is jellemezhető a növénytakaróról visszavert elektromágneses sugárzásokat rögzítő, különböző időpontokban készült műholdfelvételek alapján.

Ellenőrző kérdések:

1. A térképek milyen fő csoportjait különböztetjük meg?
2. Mire szolgál a geokód?
3. Mire használjuk a kataszteri rendszereket?
4. Mi jellemzi a digitális térképeket?
5. Mire kaphatunk választ egy földrajzi információs (GIS) rendszertől?
6. Területi kiterjedés szerint milyen információs rendszereket különböztetünk meg?
7. Mit jelentenek a fedvények?
8. Mi a vetületi rendszer? Mi az EOV?
9. Mi a raszteres adatkezelés lényege?
10. Mi a vektoros adatkezelés lényege?
11. Mi a síkbeli transzformáció?
12. Mi a tesszeláció?
13. Mi a spagetti modell lényege?
14. Melyek a fő jellemzői a topológiai adatmodelleknek?
15. Melyek a térinformatikai rendszerek fontosabb hardver eszközei?
16. A digitalizálás milyen fő módszereit ismeri?
17. Mi a fotogrammetria?
18. Hogyan tároljuk a térinformatikai rendszerek adatait?
19. Mi a precíziós gazdálkodás
20. Térinformatikai rendszerekben milyen módon integrálhatjuk az adatokat?
21. Miért van szükség adatok interpolálására?
22. Mi a MePAR rendszer alkalmazásának a célja?
23. Melyek voltak a TARNET hálózat kialakításának a fontosabb céljai?
24. Hogyan lehet térinformatikai, távérzékelési eszközökkel termésbecslést végezni?
Milyen rendszert fejlesztettek ki erre a célra Magyarországon?

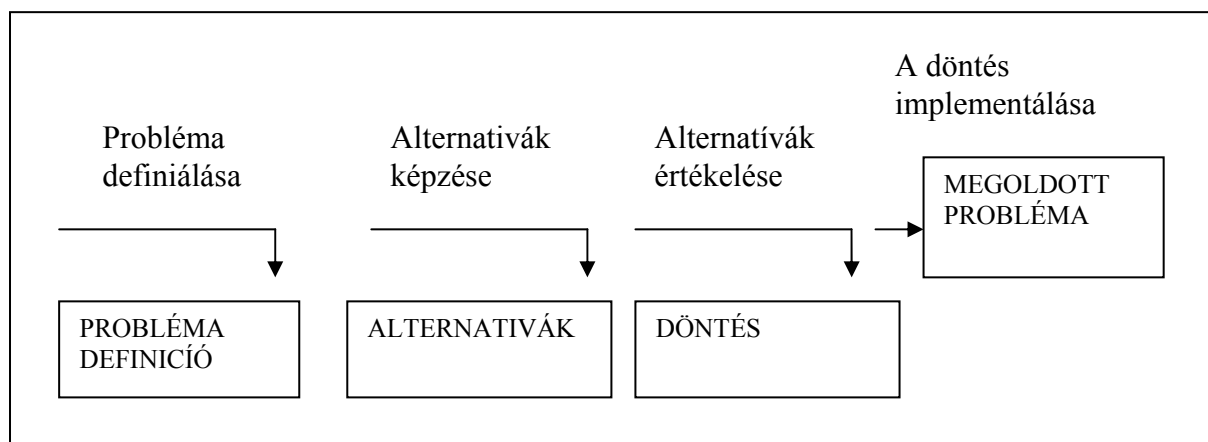
Irodalomjegyzék

- Detrekői Á. – Szabó Gy. (2000): Bevezetés a térinformatikába. Nemzeti Tankönyvkiadó Rt. Budapest. ISBN 963 19 0783 X.
- ESRI: (2006) About GIS for Agriculture. <http://www.esri.com/industries/agriculture/>
- Fölmérési és Távérzékelési Intézet (FÖMI) (2006): <http://www.fomi.hu/>
- Sárközy F. (2006): Térinformatika http://www.agt.bme.hu/tutor_h/terinfor/tbev.htm

12. DÖNTÉSTÁMOGATÓ RENDSZEREK

A döntéstámogató rendszer fogalmát olyan rendszerek leírására használjuk, amelyek támogatják, de nem helyettesítik a vezetőket döntéshozatali tevékenységükben. Ezeket a rendszereket általában részben olyan döntéseknél használják, ahol emberi ítélet is szükséges. A jó döntéstámogató rendszert a nem professzionális felhasználó is kezelni tudja, az adatok elég széles skálájához biztosít hozzáférést, többféle modellezési és elemzési eszközzel rendelkezik. A döntéstámogató rendszerek fejlődése vezetett a (felső)vezetői információs, a csoportos döntéstámogató és a szakértői rendszerekhez.

A döntéstámogató rendszer (DSS) a számítógépre készített szoftver eszközök egy készlete, amelyet egy menedzser használ a probléma megoldására a döntéshozói tevékenysége során. Természetesen a menedzser hozza meg az aktuális döntéseket és oldja meg a speciális problémákat. Ezek a problémák jöhetnek a szervezet ellenőrző mechanizmusán keresztül vagy származhat közvetlenül a menedzsertől ("pl. szükséges egy új hirdetési szlogen").



12.1. ábra A döntéselőkészítési folyamat lépései

A döntések bizonyos lépéseket igényelhetnek a problémamegoldásban, melyek szervezetén belül egy folyamaton keresztül történnek. A döntéselőkészítési folyamatstruktúráknak sok variációja van. Azonban a legtöbb szervezetén belül vannak közös elemei. A folyamatokban az 2.1. ábra mutatja ezeket az elemeket. A folyamatban (William E. Leigh és Michael E. Doherty [37] szerint) alapvetően három együttműködő számítógépes információs rendszer (CIS - Corporate Computer Information Systems) vesz részt. Ezek az adatfeldolgozó rendszer (DP - Data Processing), vezetői információs rendszerek (MIS - Management Information Systems) és döntéstámogató rendszerek (DSS - Decision Support Systems). Az adatfeldolgozó rendszer az operatív munkafolyamatokban, a vezetői információs rendszer az irányításban és ellenőrzésben, a döntéstámogató rendszer pedig a tervezésben, stratégiai tevékenységben kerül alkalmazásra.

12.1. A döntéstámogató rendszerek alapjai

A DSS számos olyan értékes eszközzel segítheti a menedzsereket az alternatívák közötti választásban, mint például

- o hasonló múltbeli döntési feladatok kikeresése adatbázisból;
- o számítási modellek alkalmazása segíthet a lehetséges döntések hatásainak megmutatásában;
- o grafikus alapú megjelenítésnél segíthetnek az aktuális döntés trend hatásainak mérlegelésére, vizuális megjelenítésére;
- o számszerűsített becslések készíthetők.

A döntéstámogató rendszerek legnagyobb része a szervezeti szintű tevékenységet támogatja. Fejlődése a népszerű pénzügyi tervezési rendszerekkel kezdődött, később kiterjedt a piackutatásra, előrejelzésre, vállalati elemzésekre, stb. A személyi számítógépek elterjedése, és különösen a könnyen kezelhető statisztikai programcsomagok és táblázatkezelő programok megjelenése ösztönözte a személyi szintű támogatást biztosító döntéstámogató rendszerek elterjedését is. Az intézményi döntéstámogató rendszereket általában döntéstámogató csoportok fejlesztik ki, valamilyen döntéstámogató (programozási) nyelvben. Régebben ezek nagygépes alkalmazások voltak, ma már azonban egyre inkább mikro gépekre készülnek. Az intézményi döntéstámogató rendszerek mellett gyakran készülnek ún. „ad hoc” rendszerek is, ezek korlátozott alkalmazhatósággal rendelkeznek, és rövid idő alatt elkészíthetők. Az ilyen „ad hoc” rendszerek azért hasznosak, mert egyrészt a vezetők rajtuk keresztül megismerkedhetnek ezzel a technikával, másrészt végül is bizonyos típusú döntéshozatalt támogatnak, továbbá használatuk során kiderül, hogy érdemes-e egy igazi döntéstámogató rendszert készíteni. Ezenkívül támogatást nyújtanak a vezetői döntésekhez, ha nincs idő egy teljes körű döntéstámogató rendszer elkészítésére. Az „ad hoc” döntéstámogató rendszerek jelentés generátorok, rövid elemző programok lehetnek, vagy készülhetnek DSS generátorok segítségével.

A jelentés generátorok a vezetőket speciálisan érdeklő adatok kiválasztására, összegzésére és megjelenítésére vagy listázására szolgálnak. A kiválasztott adatokon, az összegzésen kívül általában csak néhány egyszerű aritmetikai műveletet kell elvégezni. Amennyiben grafikát is használnak, akkor trendek és varianciák is megjeleníthetők. Egyrészt azt lehet mondani, hogy a jelentés generátorok még hosszú ideig a leginkább használt döntéstámogató rendszerek maradnak, másrészt viszont belőlük nőttek ki a vezetői információs rendszerek.

A rövid elemző programok általában kis, 80-100 soros programok, melyeket akár maguk a vezetők is meg tudnak írni, itt az adatokat kézzel is be lehet vinni. Egyszerűségük ellenére igen hatékonyak tudnak lenni.

A DSS generátorok olyan eszközök vagy programozási nyelvek, amelyek segítségével gyorsan lehet döntéstámogató eszközöket készíteni. Ezek a termékek tartalmazzak programozási nyelveket, interfészeket és további olyan szolgáltatásokat, amelyek egy-egy konkrét „ad hoc” döntéstámogató rendszer elkészítésénél segíthetnek.

Míg korábban az egyszemélyi döntéseket támogató rendszerekre koncentráltak, ma egyre inkább a csoportos döntést támogató rendszerek megvalósításával foglalkoznak (GDSS – Group Decision Support Systems). A csoportos döntéshozatal előnye a nagyobb ösztudás, a rendelkezésre álló nagyobb információ, több lehetséges alternatíva figyelembe vétele, a kidolgozott megoldások elfogadásának nagyobb valószínűsége, a folyamatban résztvevők részéről a probléma és a megoldás jobb megértése. Minden résztvevőnek lehetősége kell legyen a többiekétől független munkavégzésre úgy, hogy bármikor közzétehesse, a többiek számára hozzáférhetővé tehesse az eddig elvégzett munkáját. A csoportos döntéstámogató rendszerek részei az adatbázis, a modellbázis, a speciális alkalmazói programok, a jó felhasználói interfészek és végül maga az emberi komponens. További szolgáltatások szükségesek a csoporton belüli kommunikáció támogatására, ilyenek az elektronikus levelező

rendszerek, az egyidejűleg többek által írható (konzultációs) munkaállomások, a videó konferencia rendszerek, a gondolatok és a szavazások grafikus összegző rendszerei.

A döntéstámogató rendszerek legújabb generációja a mesterséges intelligencia eszközök körébe tartozó szakértői rendszer. Szakértői rendszereknek azokat a tudásalapú rendszereket (KBS – Knowledge Based Systems) szoktuk nevezni, amelyek szakértői ismeretek felhasználásával magas szintű teljesítményt nyújtanak egy problémakör kezelésében. A tudásalapú rendszereknél a problématerületet leíró ismeretek explicit formában, a rendszer többi részétől elkülönítve, az ún. tudásbázisban vannak tárolva. Ennek megfelelően egy szakértői rendszernek alapvetően három komponense van: a felhasználói interfész, a következtető gép és a tudásbázis. A felhasználói interfész definíciója a szokásos, általában valamilyen grafikus felület. A következtető gép a rendszer azon komponense, amely a tudásbázist felhasználva többlépéses logikai következtetéseket képes végrehajtani. Végül a tudásbázis tartalmazza a felhasználási területre vonatkozó tényeket, adatokat és ismert összefüggéseket, következtetéseket – ez utóbbiakat általában „ha, akkor” jellegű szabályok formájában. Egy szakértői rendszer hasonló javaslatokat tud adni, mint az (emberi) szakértő, kérdéseit megmagyarázza (WHY funkció), javaslatait megindokolja (HOW funkció), bizonytalan helyzetben képes meghatározott bizonyosság mellett elfogadható válaszokat adni. A tudásalapú rendszerek tudásbázisában a tárgyköri ismeretek szimbolikusan vannak tárolva, a feladatmegoldás pedig szimbólum-manipulációk révén történik.

A számítógépes információrendszerek egyik nagyon fontos válfaját alkotják a döntéstámogató rendszerek (DTR), angol elnevezést használva Decision Support Systems (DSS). Döntenünk kell az élet minden területén, döntéseink rövidebb vagy hosszabb távon meghatározzák további életünket, tevékenységünket – s ha vezető funkcióban vagyunk – mások életének kereteit is. Döntéseink információkon alapulnak, amelyeket felhasználva választunk a kínálkozó lehetőségek közül.

Nyilvánvaló, hogy a menedzserek döntéshozatala azokon az információkon alapul, amelyeket sikerül nekik összegyűjteniük. Itt egyaránt gondolnunk kell ezeknek az információknak mennyiségére és minőségére. A teljeskörűség és a relevancia egyaránt követelmény volt a hagyományos, nem számítástechnikán alapuló információrendszerekben és napjaink informatikai forradalmában is. A globalizációs és integrációs tendenciák fényében, a világ felgyorsulásának korában persze már elképzelhetetlen a döntéshozatal, a jó döntések meghozatala adekvát információs rendszerek használata nélkül.

Az üzleti szervezetek felépítésének legkülönbözőbb formái is megőrizték a hagyományos hierarchikus modellből a vezetés kitüntetett szerepét, függetlenül a szintek és kapcsolódások számától és kialakításától. Az üzleti szervezetek minden szintjén felmerül az igény a megfelelő információkra, ezek gyűjtésére, tárolására és feldolgozására. (Pontosabban az adatok gyűjtésére, s ezek feldolgozásával az információk kinyerésére.)

12.1.1 A döntéstámogató rendszerek használatának előnyei

Milyen előnyöket várhatunk a döntéstámogató rendszerektől? A legfontosabb tényezők a következők:

- ☐ hatékonyabb döntéshozatal (minőség)
- ☐ költségcsökkentés
- ☐ a döntéshozók közötti jobb kommunikáció
- ☐ a vezetők (döntéshozók) gyorsabb betanulása

Nem szabad persze kritikátlanul elfogadni egy rendszer eredményeit. Magának a döntéstámogató rendszernek a minőségétől is függ az eredmény értéke, felhasználhatósága. S gondoljunk a minden számítástechnikai, informatika rendszer alapösszefüggésére, a GIGO (Garbage In, Garbage Out=szemét be, szemét ki) elvre, mely szerint az eredmény a bevitt adatok minőségétől függ.

Döntéshozatali problémára tekintsünk példaként egy kisüzemet, amelyik ajándéktárgyakat gyárt, mondjuk karácsonyra. Ha túl keveset gyártanak, akkor a készletük hamar elfogy, esetleg már 3-4 héttel az ünnep előtt, s tetemes nyereségtől esnek el, ami a le nem gyártott ajándéktárgyak eladásából származna. Ha viszont túl sokat gyártanak, akkor rosszabbik esetben dobhatják el az egészet, jobbik esetben csak a tárolást kell megfizetni a következő karácsonyig, amikor is újra lehet próbálkozni az eladással. A tárolás azonban lehet nagyon költséges, s nincs igazán garancia arra, hogy a vásárlói szokások nem változnak meg a kárukra egy év alatt.

Milyen információra lenne szüksége a kisüzemnek? Nyilván arra, hogy mennyi ajándéktárgyat lehet eladni az év karácsonyán? Természetesen nem rendelkeznek ezzel az információval. Megbecsülhetik a várhatóan eladható mennyiséget, de ez a becslés igen bizonytalan. Éppen ezért a döntés kockázatos lesz. Hogy mennyire az a körülményektől függ. Bizonyos kockázat még viszonylag stabil piacon is előfordul. Ha a gyártó biztosra akar menni, akkor kevesebbet termel, tehát kevesebb lesz a haszna, de ezzel együtt a kockázata is. Ha nagyratörő tervei vannak, akkor sokat nyerhet, de sokat veszthet is. Egy számítógépes döntéstámogató rendszernek csak akkor fogja hasznát venni, ha az képes ilyen típusú, azaz bizonytalan kimenetelű döntések támogatására.

Nagyléptékű ipari alkalmazásokban, ahol egyáltalán érdemes költséges döntéstámogató rendszereket használni, szinte mindig jelen van a bizonytalanság, a kockázat. Szerencsére az operációkutatás és az informatika rendelkezik olyan módszerekkel, amelyek képesek megbirkózni ilyen problémákkal. Hasonlóan nehéz olyan döntések hozatala, ahol nem rendelkezünk a döntéshozatalhoz szükséges valamennyi információval. Mondani sem kell, hogy az ilyen helyzetek vannak túlnyomó többségben. Azonban az ilyen problémák is kezelhetők a modellezés kifinomult módszereivel.

12.1.2. A döntéshozatal folyamata

A döntéshozatal olyan folyamat, melynek során különböző cselekvési alternatívák közül választunk bizonyos cél vagy célok elérése érdekében. Herbert Simon szerint a vezetői tevékenység szinonimája a döntéshozatalnak, hiszen mind a tervezés, szervezés, ellenőrzés, általában a vezetői feladatok végső soron döntések meghozatalában nyilvánulnak meg. A döntési folyamatot vizsgálva, a következő részfeladatok határozhatók meg:

- o feladat-meghatározás és adatgyűjtés,
- o tervezés,
- o választás,
- o megvalósítás.

Az irodalomban két koncepciót találhatunk arra nézve, hogy mely mozzanatok alkotják a problémamegoldást ezek közül. Az egyik szerint a problémamegoldás a fentiek közül a 4., amelyhez az 1-3. lépésből álló döntési folyamat vezet, a másik szerint a két elnevezés ugyanazt a fogalmat takarja, s e szerint problémamegoldásnak tekinthetjük a fenti lépések sorozatát együttesen.

Feladat-meghatározás és adatgyűjtés fázisa

Habár a problémák jelentkezése általában igen feltűnő szokott lenni, mégis az első lépés logikailag a fennálló helyzet folyamatos figyelése, monitorozása, hogy legyen összehasonlítási alapunk a problémás és problémamentes időszakokról. Ehhez definiálni kell a normális, ill. kívánatos állapotot, az egyéni ill. szervezeti célokat. A probléma észlelése annak tudatosodása, hogy a meglevő és a kívánt állapot között eltérés van, s az ezzel való elégedetlenség ennek megszüntetésére sarkall. A legfontosabb feladat a probléma azonosítása, az eltérések valódi okának, okainak kiderítése. Sajnos nagyon gyakran nehéz megkülönböztetni a problémát annak szimptomáitól.

Például egy vállalkozásban a nyereségesség csökkenése jelzi, hogy valami baj van, de hogy az eladások visszaesése, vagy a ráfordítások növekedése, illetve azok milyen tényezőkre visszavezethető okai, annak kiderítése egyáltalán nem egyszerű.

A probléma azonosítása után látszólag a legegyszerűbb a vele kapcsolatos adatok begyűjtése. Azonban ez egyáltalán nem így van. Nehézséget okozhat a problémával kapcsolatos adatok meghatározása, teljeskörűségének biztosítása, különösen az előrejelzése, egyáltalán a relevancia érvényesülése.

Amennyiben lehetséges, a problémát részproblémákra bontjuk, mivel gyakran a részprobléma önállóan elemezve könnyebben megoldható. A probléma dekompozíciónak ez a módszere azonban csak körültekintéssel használható, szerencsétlen esetekben a rosszul felbontott részproblémák megoldása nem alkalmazható az eredeti feladatra.

Az igazi probléma azonosításakor az sem mellékes, hogy megtudjuk, ki a probléma tulajdonosa, kinek a számára probléma a probléma. Ez leggyakrabban a szervezeti felépítéssel kapcsolatos, a rossz szintre adresszált problémával nem lehet mit kezdeni.

Tervezési fázis

Ennek a fázisnak a feladata megtalálni, kifejleszteni és elemezni a lehetséges cselekvési alternatívákat. Ez a probléma mély megértését, a megoldásváltozatok kialakítását és tesztelését jelenti.

Ehhez el kell vonatkoztatni a probléma felszíni jegyeitől és meg kell találni a mélyebben rejlő összefüggéseket, vagyis absztrakció segítségével szimbolikus formába kell önteni. Ezt hívjuk modellezésnek.

A modellek megválasztása rengeteg, gyakran nem tudatosuló előfeltevést hordoz magában, s ezek gyakran jobban meghatározzák a kapott eredményeket mint a számszerűsítéshez felhasznált adatok. A modellezés hátterében álló koncepciók, iskolák, szaktudományos sőt filozófiai eredmények kikerülhetetlenek az eredmények értékelése, az azokon alapuló döntések meghozatala kapcsán.

Alternatívák generálása

Az alternatívák generálása viszonylag hosszú folyamat, amely keresést és kreativitást igényel (brainstorming). Az egyik alapvető kérdés: mikor kell abbahagyni a keresést?

A DSS-ekben ez általában manuálisan történik, az ES-ekbe automatikusan be van építve egy megfelelő leállási kritérium.

Az alternatívák kimeneteleinek értékelése függ a modelltől és az adatoktól.

Biztos kimenetel a teljes információon alapuló, ezt determinisztikusnak nevezzük. Valószínűségi vagy sztochasztikus a részleges információon alapuló, ekkor általában csak

rövid időtávra mondhatunk valamit, azt is csak fenntartásokkal - ekkor kalkulálni kell a kockázatot is. Bizonytalanságról beszélünk, ha csak minimális információ áll rendelkezésre - a döntéshozó a kimenetek valószínűségeit sem ismeri.

Értékelés

Az eredmények, alternatívák kiértékelése nagyon sok nehézséget rejt magában. Tudni kell, mit mértünk, miben mértük, miért optimalizáltuk és hogyan. Ráadásul a legtöbbször nem lehet egyetlen célváltozó meghatározásra visszavezetni a problémákat. A többszörös célok esetén nem egy optimalizálandó cél van, hanem több, amelyek egymásnak is ellentmondhatnak, s ellent is mondanak.

Ezek kiküszöbölésére számos megoldási módszert dolgoztak ki - hasznosságelmélet, célprogramozás, lineáris programozás (célok mint feltételek), pontozási rendszerek stb., de ezek csak korlátozott esetekben használhatók, bizonyos helyzetekben nincs egyértelmű megoldás.

Ha van is megoldás, a következő problémákat kell megoldani:

- ☐ nehéz explicit célhierarchiát felállítani az adott vállalatnál
- ☐ különböző résztvevőknek mások a célprioritásai
- ☐ a döntéshozó is változtathatja céljait az idővel és a körülmények változásával
- ☐ a szervezet különböző szintjei és részlegei más és más célokkal rendelkeznek
- ☐ a célok maguk is változnak a környezet kihívásaival
- ☐ nehéz pontos kapcsolatot találni az alternatívák és célok tekintetében

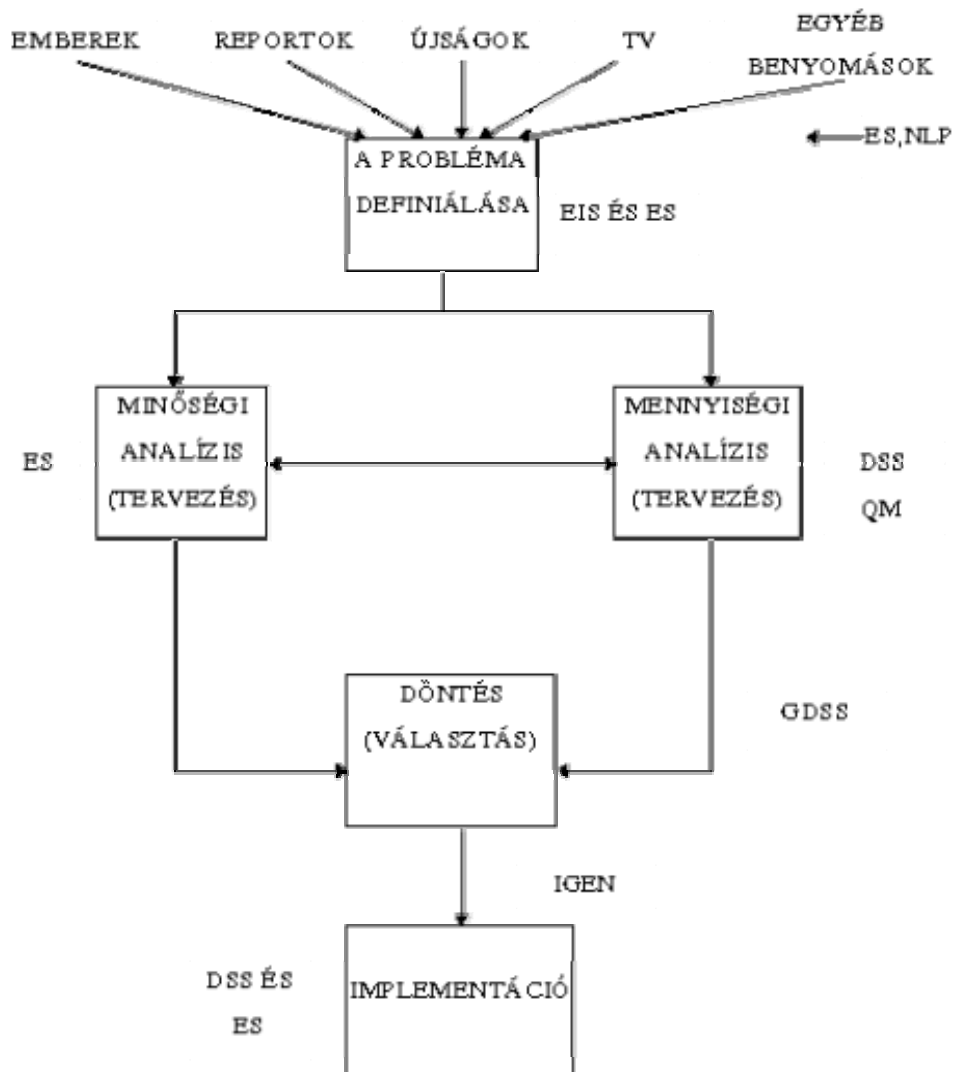
Az eredmények elvileg a modellből és a megoldási eljárásból fakadóan helyesek, mégis ellenőrzésre és legfőképpen érzékenységvizsgálatra van szükség velük kapcsolatban.

Megvalósítás fázisa

Mindig a legnehezebb dolog, mert a kiválasztott megoldást át kell ültetni a gyakorlatba, s még ha nem is itt ugranak ki a modell gyengeségei, akkor is minden változás ellenállást szül, ill. a modellben optimális megoldás megvalósítás után nem biztos, hogy optimális marad.

12.1.3. A döntési folyamat támogatása

A döntési folyamat fenti lépéseit a különböző vezetést támogató rendszerek segíthetik - különböző módokon.



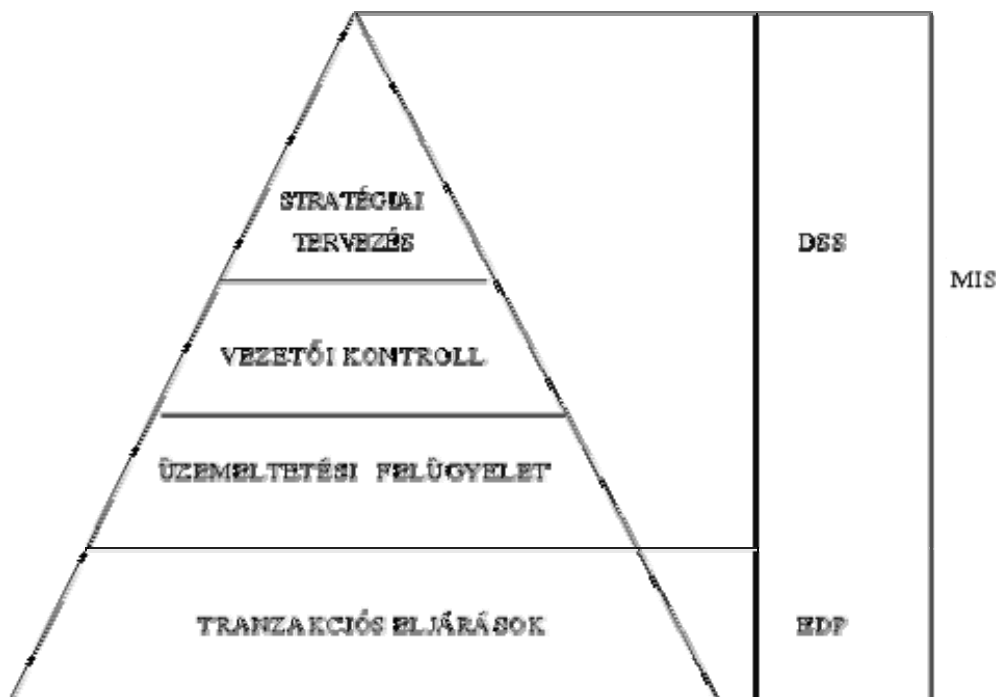
12.2. ábra A döntési folyamat támogatása

Adatgyűjtés fázisa - adatbázis kezelés, riport generálás, adatfeldolgozás - MIS, EDP, EIS, ES, DSS

Tervezés fázisa - alternatíva generálás, előrejelzés készítés - DSS, ES

Választás fázisa - lehetséges megoldások felismerése, érzékenységvizsgálat - DSS (nem dönt, csak ajánl változatokat) ES (dönt és indokol is)

Megvalósítás fázisa - alátámasztja a döntést, segít a kommunikációban, magyarázatban, igazolásban - DSS, ES



12.3. ábra: Döntéstámogató eszközök és vezetési szintek

12.1.4. A döntéshozatal módjai

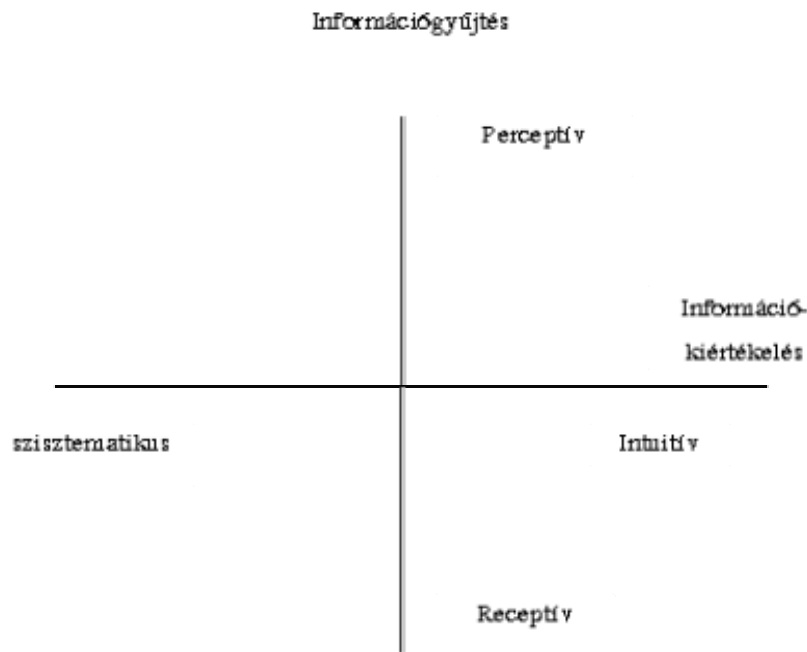
Egyéni döntéshozatal

A döntéshozó intuíciója, preferenciái, szubjektív értékelése és tapasztalatai lényegesek a döntéshozatal tényleges lefolytatása során. Az emberi megismerési és döntési stratégiák az egyes egyéneknél különbözőek. A probléma megismerési (kognitív) folyamat az a folyamat, melynek során az egyének feloldják a világról alkotott belső képük és a valóságról érkező információk különbségeit. Ez sohasem befejezett, egyetlen aktus, hanem az egyén mindennapi tevékenységébe beágyazott, attól elkülöníthetetlen folyamat.

A kognitív folyamat stílusa lehet:

- ☐ perceptív – általánosító, a kapcsolatok, összefüggések feltárását előtérbe helyező, ill.
- ☐ receptív – a részleteket vizsgáló, a specifikumokat hangsúlyozó megközelítés.

Másik oldalról nézve lehet szisztematikus vagy intuitív a munkastílusa valakinek, ahogy az adatokat, információkat feldolgozza.



12.4. ábra: A kognitív folyamat

A különböző stílusok nagymértékben befolyásolhatják az adott személy és döntéseinek kapcsolatát, milyen modelleket választ, azok eredményeit hogyan értékeli, hogyan hajtja végre a döntéseit, stb. Ennek ismerete fontos a döntéstámogató rendszer szempontjából is, egyrészt, hogy milyen humán interface-t, másrészt, hogy milyen belső szerkezetet igényel a MSS-től az adott döntéshozó.

Az emberi probléma megoldás mind heurisztikus, mind transzformációs elemekből áll. A szimbolikus logika e második típus kiragadása és önálló fejlesztése, amelynek számítógépes megvalósításai az ún. produkciós rendszerek. Ezek tanulmányozásával kezdetben nagy sikereket lehet elérni az emberi problémamegoldás vizsgálatában, modellezésében, de egy szint felett már elégtelenek.

A heurisztikus problémamegoldás belső szerkezete nem ismert – ezért is hívjuk ezzel az eufemisztikus kifejezéssel, mindenesetre jellemzők rá a logikai ugrások, az analógiák, a váratlan, ihletett megoldások (és tévedések).

A produkciós rendszerek szimbólummanipulációs technikák, gyakran a formális logika kibővítései, de általánosságban egy zárt világ elemeinek transzformációs szabályrendszereként határozhatók meg. Egy egyszerű logikai transzformáció az a következtetés pl., hogy ha az A személy B anyja és B a C személy anyja, akkor A csak C nagyanyja lehet.

Szervezeti döntéshozatal - csoportos döntések

Sok esetben a döntéshozatal nem egyetlen ember feladata és kiváltsága, hanem több ember kell, hogy meghozza a döntéseket. Tárgyalás, alkudozás, megegyezés szükséges. A szervezeti döntéshozatal tehát általában csoportos döntést igényel. A döntéstámogató

rendszereknek nemcsak egy vállalat, egy cég igényeivel, szükségleteivel kell foglalkozniuk, hanem az egyes (cégen belüli) osztályok, sőt az egyes alkalmazottakéval is.

- ☐ Vizsgálhatjuk a vállalkozás tágabb környezetét is, egész a kormányzati társadalmi szintekig, bár manapság még egyáltalán nem jellemző ezeken a szinteken a döntéstámogató rendszerek használata.

Egy üzleti szervezet célja tevékenységeinek tervezése, működtetése és ellenőrzése a cég meggazdagodása céljából. A tervezés biztosítja, hogy cég mindig a nyereségesség irányába haladjon, illetve, hogy jól felkészülve legyen a nehéz időkre. Az ellenőrzés, a felügyelet azt biztosítja, hogy a mindennapi tevékenységek a cég életében jól menjenek. Ez a két funkció szervesen összekapcsolódik.

A csoportos döntéstámogató rendszerek (GDSS) kifejlesztésekor számos kísérlet történt arra, hogy az individuális döntésekkel foglalkozó elméletet csoportokra próbálják alkalmazni. Csoportokra, melyek egyedeinek egymástól többé-kevésbé eltérő érdekei és preferenciái lehetnek.

A döntéselmélet módszertana azt igényli, hogy a döntéshozó egyrészt felállítson egy preferencia sorrendet, azaz sorba tudja rendezni a felmerülő alternatívákat a saját szempontjai szerint, másrészt ezen alternatívákhoz valamilyen módon valószínűségeket tudjon hozzárendelni.

Csoportokkal foglalkozva nyilvánvaló, hogy sokkal nehezebb közös érdekről és csoportpreferenciáról beszélni. Kézenfekvőnek tűnhet többségi szavazással eldönteni a kérdéseket. Mint az alábbi példa mutatja, ez nem mindig lehetséges.

A szervezetek döntési folyamatának számítógépes támogatására korán megszületett az igény, párhuzamosan a szervezetek információfeldolgozásának automatizálásával. A fejlődés egy evolúciós pályát futott be párhuzamosan a technológiai lehetőségek kiszélesedésével.

A kezdeti MIS-ek a „Mi a helyzet?” kérdésre kívántak válaszolni a rendelkezésre álló adathalmaznak a standard TPS szolgáltatásokat meghaladó feldolgozásával. A következő fokozatot a prediktív MIS -ek jelentették a „Mi történik, ha ?” kérdések kezelésével, végül igazán DSS-nek a „Milyen alternatíva a legjobb?” kérdésére válaszoló rendszert tekintjük. (Ez a változás egyszerű funkcióbővülésként indult, de minőségileg új rendszer alakult ki.)

Ahhoz, hogy egy DSS a célját elérje, szükséges olyan tulajdonságokkal rendelkeznie, amelyek elsősorban a funkcionálisan helyes működést garantálják, másodsorban a kényelmes kezelhetőséget a sokrétű problémaszituációkban, valamint a rugalmasságot és továbbfejleszthetőséget. Ezek pontokba szedve a következők:

- ☐ Dinamikus együttműködés a számítógép és az ember között,
- ☐ A különböző vezetői szintek támogatása,
- ☐ Egyéni és csoportos döntési folyamatok támogatása,
- ☐ Elkülönülő és láncolt döntések kezelése,
- ☐ A döntési folyamat végigkísérése,
- ☐ Különböző döntési stílusok és technikák támogatása,
- ☐ Rugalmasság és adaptivitás,
- ☐ Barátságos felhasználói felület,
- ☐ Hatásosság,
- ☐ Teljeskörű felhasználói kontroll,
- ☐ Fejleszthetőség – belső és külső,

- Végfelhasználói fejleszthetőség.

12.2. Döntéstámogató rendszerek részei

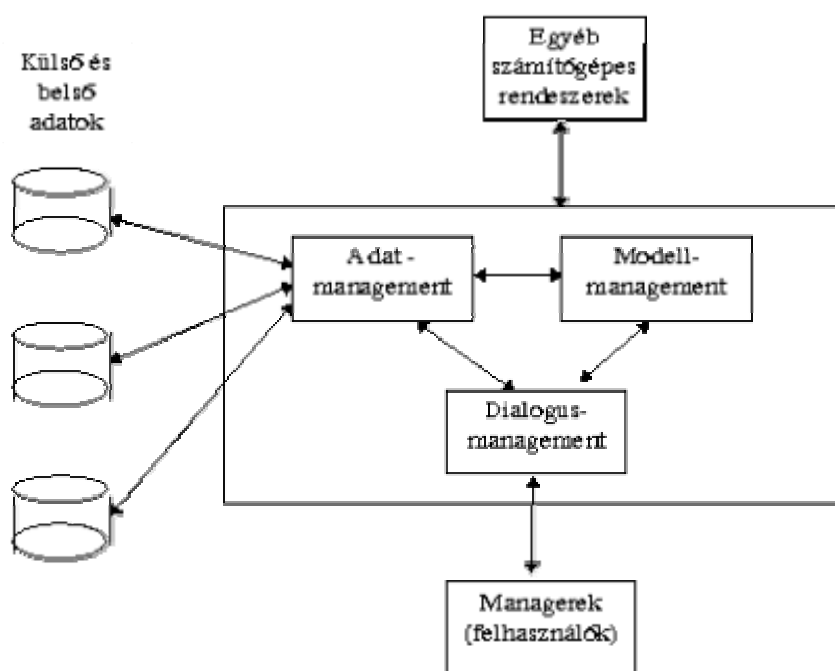
Egy DSS-nek a rendelkezésre álló adatokból az adott szituációt modellezve kell az emberi döntéshozóval együttműködve megoldási alternatívákat előállítani és értékelni. Ennek értelmében egy DSS a következő alrendszerekből áll:

Adatkezelő alrendszer – tartalmazza a döntéshez szükséges adatokat, ill. kapcsolódik a szervezeti TPS-hez vagy külső adatbázisokhoz

Modellkezelő alrendszer – a rendszer elemző képességét biztosítja különböző területek matematikai modelljeinek felhasználásával.

Kommunikációs alrendszer – a felhasználó és a rendszer kapcsolatát biztosítja.

Ezek az alrendszerek mindegyike számos további részekre, funkciókra bontható.



12.5. ábra: A döntéstámogató rendszerek felépítése

12.2.1. Adatkezelő alrendszer

Az adatkezelő alrendszer biztosítja a külső adatbázisokhoz való hozzáférést, a saját adatainak tárolását, feldolgozását, ezeknek egységes és konzisztens adatszótárba rendezését és a lekérdezések elvégzését. Részei így a következők:

- DSS adatbázis
- Adatbáziskezelő rendszer

- Adatszótár
- Lekérdezés

DSS adatbázis

Az adatbázisokban a rendszeresen ismétlődő, jól meghatározott szerkezetű és logikai kapcsolatokkal rendelkező adatokat, adatszerkezeteket tároljuk.

Az idők folyamán több különböző adatbázismodell alakult ki, ezek a következők:

- ☐ rekord-alapú – egyszerű adatállományok, hasonló szerkezetű adatrekordokból,
- ☐ hierarchikus – az adatok logikai hierarchiába vannak szervezve,
- ☐ hálós – ugyanazon az adathalmazon több hierarchia is érvényesül egyidejűleg,
- ☐ relációs – az adatok egyedei tulajdonságainak mátrixába vannak rendezve,
- ☐ objektum-orientált – objektumok tulajdonság rendszerének segítségével tárolja az adatokat.

A különböző adatbázismodellek a valóságos objektumokból különböző szemlélet alapján absztrahálva hozzák létre és tárolják, kezelik az adatrendszereket. Az ábrázolás és kezelés akkor kényelmes, ha az absztrakció szemlélete jól megfelel a valóságos objektumoknak. Pl. hierarchikus viszonyok ábrázolására a hierarchikus adatbázisok hatékonyak. De pl. a többféle hierarchiát is tartalmazó objektumok esetében elégtelen lehet, s ekkor már hálós adatbázisra van szükség. Jelenleg a legelterjedtebb adatbázis modell a relációs, amellyel a legtöbb szituáció jól kezelhető, de úgy tűnik, az objektumorientált modellre szép karrier vár.

A DSS-adatbázisban a szervezet belső tranzakcióinak aggregált adatai, más belső és külső adatforrásokból származó adatok, valamint a felhasználók személyes és egyéni adatai vannak. Ezeknek a kialakítása különböző módon történhet, a fő kérdés a centralizáltság illetve elosztottság foka, az információkhoz való hozzáférés köre, szintjei és jogosultsági rendszere, valamint a külső adatbázisokból való adatkinyerés (extrakció) megszervezése.

Adatbáziskezelő rendszer

Az adatbáziskezelő programrendszerek a különböző adatmodellekre optimalizálva készülnek, skálájuk az olcsó íróasztali gépeken futó verzióktól kezdve az országos statisztikai adatgyűjtéseket kezelő nagygépes (mainframe) rendszerekig terjed.

Funkciói a következők:

- ☐ A DSS-adatbázis elérése, adatok kinyerése,
- ☐ Gyors adatfelújítások,
- ☐ Különböző forrásokból származó adatok együttes kezelése
- ☐ Lekérdezések, riportok generálása,
- ☐ Az adatok biztonságának garantálása,
- ☐ Személyes és szervezeti adatok alternatíváinak kezelése,
- ☐ Az adathasználat nyilvántartása.

A végfelhasználók számára az adatbáziskezelő műveletek nagy része rejtett, s annak is kell lennie. Nincs arra szükség, esetleg csak a rendszergazdák számára, hogy

beleavatkozzanak az adatok tárolásának, felújításának technika részleteibe. Az adatbáziskezelő által kikeresett adatok a végfelhasználó által használt jelentéskészítők, ad-hoc lekérdezések vagy modellek inputját alkotják.

Adatszótár

A belső és külső adatok egységes kezeléséhez szükséges kialakítani egy olyan adatszótárt, amely maga is mint (speciális) adatbázis tárolódik, amellyel a tárolt és felhasznált adatok forrása, állapota, kapcsolatai leírhatók és felhasználhatók.

Lekérdezés

Alapvető funkció, amelyen keresztül lehet az adatokat elérni, a lekérdező nyelvek szorosan kapcsolódnak az adatbázis kezelőkhöz és adatbázis modellekhez, valamint a DSS kommunikációs alrendszeréhez.

12.2.2. Modellkezelő alrendszer

Formailag hasonló részekre bontható, mint az adatbáziskezelő alrendszer:

- ☐ Modellbázis
- ☐ Modellbáziskezelő rendszer
- ☐ Modell szótár
- ☐ Modell végrehajtás

Modellbázis

A modellbázis tartalmazza azokat különböző (statisztikai, pénzügyi, stb.) modelleket, amelyekkel az analitikus feladatait elláthatja egy DSS. Ezeket kell tudni alkalmazni, módosítani, kombinálni a feladatok függvényében.

A modelleket a felhasználási szint és funkció szerint csoportosíthatjuk:

- ☐ Stratégiai,
- ☐ Taktikai,
- ☐ Működtetési,

valamint

- ☐ Modell blokk, ill.
- ☐ Szubrutin

típusúakra.

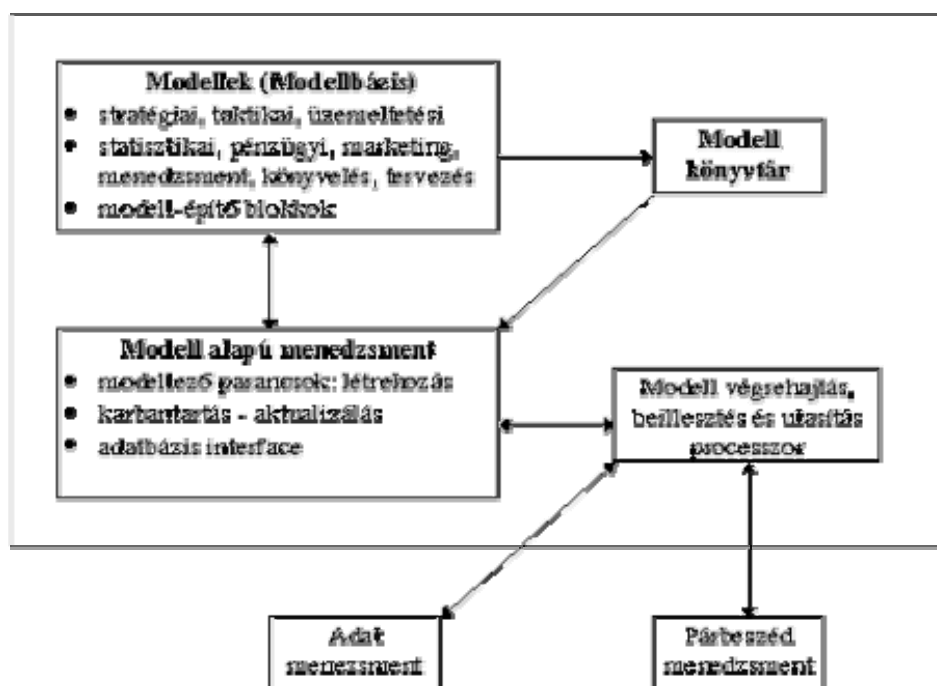
A stratégiai modellek a felső vezetők stratégiai tervezését szolgálják, inkább leíró, mint optimalizációs szerkezetűek. Hasonló részfeladatokat látnak el, mint az EIS (Felsővezetői információs rendszerek) modelljei.

A taktikai modellek a középvezetők igényeit elégítik ki, az erőforrások allokálásának és a kontrollnak a támogatásával. Általában a szervezetek funkcionális részeihez kapcsolódnak.

A működtetési modellek a mindennapi tevékenységek megszervezését segítik a szervezetben, a napi munkafolyamatok ütemezését, a források biztosítását, stb.

Mind a három modell szinthez, a modellek átalakítására, módosítására adnak lehetőséget a modell blokkok és szubrutinok. Ilyenek pl. a regressziós analízis, klaszteranalízis, stb. blokkok, vagy a véletlenszám generátor, jelenérték számító szubrutin stb. Önállóan is felhasználhatók bizonyos adatelemzési célokra, de alkalmasak nagyobb modellek felépítésére is.

A modellek osztályozhatók felhasználási terület (pénzügy, munkaügy, stb.), valamint elméleti alapok szerint is (statisztikai, optimalizálási, stb.) Ezek részletezése több száz típusra vezetne, ezzel most nem foglalkozunk.



12.6. ábra: A modellbáziskezelő rendszer részei

Modellbáziskezelő rendszer

A modellbáziskezelő rendszernek kell létrehoznia a modelleket a meglevő blokkokból és szubrutinokból, bővíteni a blokkok készletét, a modelleket és a hozzájuk szükséges adatokat kezelni, a részmodelleket összekapcsolni.

Modell szótár

Funkciója hasonló az adatszótáréhoz, katalogizálja a modelleket, tartalmazza a modelldefiníciókat, tájékoztatja a felhasználókat a modellek lehetőségeiről és hozzáférhetőségéről. Arra persze csak gyakorlat vezetheti rá a felhasználót, hogy milyen esetekben mely modellek a jók. A szakértő rendszereknek itt tág tér nyílhat ennek a választásnak a segítésére.

Modell végrehajtás

A modellek aktuális futását vezérli, valamint a különböző modellek összekapcsolását, az inputok és outputok átirányítását.

12.2.3. Kommunikációs alrendszer

Ez a komponens biztosítja a felhasználó és a rendszer kapcsolatát, vezérli az interakciókat, felelős a használat kényelmességéért és hatékonyságáért. Ennek minősége meghatározója lehet egy DSS elterjedésének vagy háttérbe szorulásának, függetlenül a többi komponens minőségétől.

A kommunikációs folyamat három részre bontható, az utasításnyelvre, a válasznyelvre és az ezek használatát leíró szabályrendszerre.

Az utasításnyelv elemei azok az eszközök, amivel utasítani lehet a rendszert, azaz a billentyűzet gombjai, az egér, érintő képernyő, botkormány, optikai leolvasó sőt hangérzékelő és értelmező.

A válasznyelvet a rendszer használja a felhasználó informálására az eredményekről, részeredményekről, amellyel utasítást, választást kér. Ebbe beletartoznak a különböző output perifériák, mint a képernyő, nyomtató, hangszóró, ill. ezek felhasználásának módja. Mikor milyen grafikai vagy hangeffektusokat használjon a rendszer, s hogyan függ össze ezekkel a döntéshozó pszichológiai beállítottsága?

E nyelveknek a szabályrendszere ismert kell legyen a felhasználó számára, papíron vagy online help formájában mindig elérhetően.

A felhasználó

Egy DSS-t a felhasználó különböző módokon alkalmazhat a döntési folyamat támogatására, illetve különböző státuszú felhasználók léphetnek a rendszerrel kapcsolatba.

Alter szerint a tipikus felhasználási módok a következők:

- ☐ Előfizetői mód – a döntéshozó rendszeresen elkészített jelentéseket vár el a rendszertől, s nem is interaktívan.
- ☐ Hivatalnoki mód – a döntéshozó változó tartalmú jelentéseket készített a rendszerrel, de azt offline módon készíti elő.
- ☐ Terminál mód – a döntéshozó interaktívan használja a rendszert kérdés-felelet szekvenciákon keresztül.
- ☐ Közvetítéses mód – a döntéshozó másokon keresztül, stábja erre kijelölt munkatársai közvetítésével használja a rendszert, nem ismerve a megoldás és a rendszer részleteit. Ennek a módszernek az elterjedtsége a vezetők számítástechnikai képzettségének növekedésével, ill. a DSS rendszerek használatának egyszerűsödésével a jövőben csökkenni fog.

A közvetítőket három csoportba lehet sorolni, lehetnek:

- ☐ DSS asszisztensek – a vezető stábjának a döntéstámogató rendszerekkel való kapcsolattartásra specializálódott tagja,
- ☐ Specialisták – egy-egy üzleti szakterület szakértője, aki a DSS rendszerek segítségével készíti elő szakvéleményét,
- ☐ Szakértők – egy-egy modellezési módszer szakértője, az ő segítségével lehet a speciálisabb modelleket kidolgozni, ill. megoldani.

DSS hardver és szoftver

A DSS-ek a legtöbb hardver elemen futhatnak, a különbségek a feladat nagyságából, bonyolultságából, időigényéből fakadhatnak.

A sok felhasználót kiszolgáló, nagy erőforrásokat mozgósító rendszereket inkább nagygépekre ajánlatos telepíteni, míg az egyfelhasználós, kis adatbázist használó társaikat lehet asztali számítógépekre. A lokális hálózatok alkalmasak egy határig a nagygépek kiváltására, de mindig a konkrét helyzet analízisa alapján kell a megfelelő kompromisszumot kialakítani.

Steven Alter nyomán megkülönböztetjük a döntéstámogató rendszerek 7 szintjét.

- ☐ Javasló rendszerek (suggestion systems)
- ☐ Optimalizáló rendszerek
- ☐ Reprezentációs modellek
- ☐ Könyvelési modellek
- ☐ Elemző információs rendszerek (analysis information systems)
- ☐ Adatelemző rendszerek
- ☐ Adatkezelő rendszerek (file drawer systems)

12.3. Döntéstámogató szoftverrendszerek és alkalmazások

12.3.1. OPTRANS OBJECT

Az OPTRANS OBJECT egy francia DSS fejlesztő környezet, amely a döntési folyamat valamennyi fázisában segíti a fejlesztőt. Az alábbiakban a legfontosabb jellegzetességeit emeljük ki ennek a szoftvernek.

i) A generátor szerkezete

A generátor legfontosabb részrendszerei (erőforrásai):

- ☐ az adatbáziskezelő rendszer
- ☐ a riportgenerátor
- ☐ a modellező nyelv
- ☐ a fájlkezelő rendszer
- ☐ a statisztikai algoritmusok eszköztára és
- ☐ a felhasználói interfész.

Minden részrendszerhez egy ablak tartozik. Egy DTR alkalmazáshoz egy alkalmazási fájl, a változónevek egy halmaza, fejléccímkék és paraméter nevek társulnak. A rendszernek van egy lekérdező processzora és egy megoldó (solver) könyvtára. A felhasználó egy parancsablakon, menükön, ablakokhoz tartozó lokális menügombokon, párbeszéd ablakokon, editoron keresztül kommunikál a rendszerrel.

12.3.2. Visual IFPS/Plus

Az IFPS az Interactive Financial Planning System rövidítése, vagyis a név egy interaktív pénzügyi tervező rendszert takar. Olyan problémaorientált negyedik generációs nyelvről van szó, amely bármely tervezési, illetve elemzési feladatra használható, feltéve

hogy „spreadsheet” típusúak a felhasznált információk. Számos beépített üzleti függvényt tartalmaz, így pl. nettó jelenérték számítás, értékcsökkenés meghatározása, loan amortization, belső megtérülési ráta, stb. A plusz szó a beépített adatbáziskezelési lehetőségekre utal. Az IFPS egy „spreadsheet” formájú matematikai modell létrehozására alkalmas eszköz.

Megadjuk a probléma (az üzleti rendszer) matematikai leírását (a feltételek és a változók közötti relációk matematikai leírását). Az IFPS modellben a felhasználó matematikai egyenletei az üzleti tevékenységek közötti kapcsolatok logikáját írják le, ez adja a modell logikáját. Az IFPS modell nagyon hasonló egy programhoz, amelyet BASIC-ben, FORTRAN-ban, C-ben, vagy egyéb nyelven írtak. Ennek ellenére az IFPS felhasználóinak nem kell programozóknak lenniük, sokkal inkább érteniük kell az üzleti tervezéshez, illetve elemzéshez. Következésképpen az IFPS-ben a felhasználó logikája adja a modellt, nem egy számítógépes program.

Az IFPS hasznos olyan döntési eljárásokban is, amikor nem precízen definiált problémáról van szó, amely több változót tartalmaz, komplex relációkkal és bizonytalanságokkal. A modell olyan további információkat is ad az elemzőnek, mint a különböző döntések lehetséges hatásainak vizsgálata, a különböző alternatívák közötti választások elősegítése stb. Az IFPS modell könnyen létrehozható és egyszerűen értelmezhető. A kezdeti modell a későbbiekben tetszés szerint bővíthető, a környezet változásainak megfelelően a modellt könnyű módosítani.

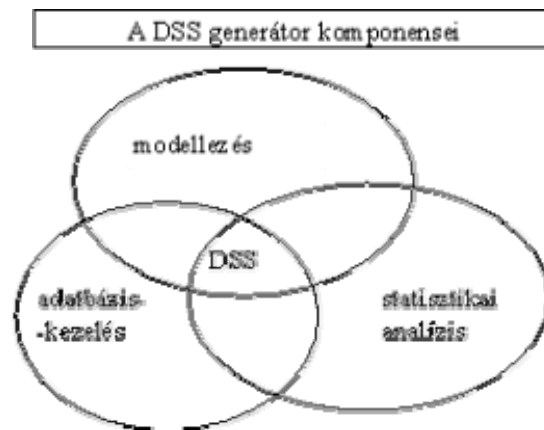
Automatikus spreadsheet funkciók

Az IFPS megengedi a táblázat formájában adott pénzügyi beszámolókat, megoldások létrehozását. A sorok illetve az oszlopok elnevezése tetszőleges lehet, általában az oszlopok jelölik az egyes időszakokat, illetve tervezési szinteket, a sorok pedig a változókat, vagy tervezési szinteket (nincs előre definiált sor, illetve oszlop rendszer). A táblázat használható a kezdeti IFPS modell létrehozására.

Döntéstámogató rendszerek és az IFPS

Az IFPS döntéstámogató rendszerek létrehozására alkalmas számítógépes eszköz. Az eddigiekkel összhangban azt mondhatjuk, hogy a döntéstámogató rendszerek interaktív, rugalmas, adaptív számítógépes információs rendszerek, amelyek döntési szabályokat, modelleket, adatbázisokat és a döntéshozó saját elméleteit használják fel. Így a számítógépes rendszer segíti a managereket a döntéshozásban. Természetesen a döntéseket nem az IFPS modell hozza, hanem a managerek.

Ahogy az alábbi ábra is mutatja a számítógépes DSS három legfontosabb funkciója: az adatbáziskezelés, statisztikai analízis és modellezés. Ezek a komponensek külön-külön nem nevezhetők DSS-nek. Olyan számítógépes eszközök, amelyek segítenek az adott problémára vonatkozó DSS-t létrehozni. Az IFPS nem azonos egy DSS-sel, olyan DSS generátor, illetve eszköz, amely az adott speciális szituációban segíti a DSS kifejlesztését.

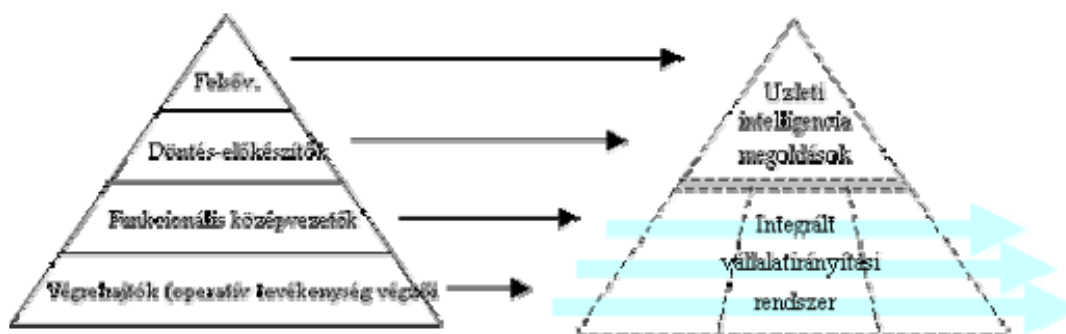


12.7. ábra: Egy DSS generátor részei

12.4. Üzleti intelligencia, mint vezetői támogatás

A 80-as évek jellemző vállalati informatikai rendszerei a 90-es években fokozatosan átalakultak. A 12.8.sz. ábra három hangsúlyos változást mutat.

- A végrehajtás, és a közvetlen fölött elhelyezkedő vezetői szint támogatását integrált vállalatirányítási rendszerek biztosítják. A korábbi EDP/TPS, illetve MIS-rendszerek feladatait az ERP-rendszerek moduljai látják el, melyek bár őrzik a funkcionális tagolást, lehetőséget nyújtanak a horizontális -- folyamatelvű -- szemlélet érvényesítésére is.
- Az alsó szint egyetlen, egységes vállalati adatbázist képes teremteni, azonban a felsővezetői információigény kielégítésére, azösszvállalati controlling feladataihoz speciális adatkezelési, lekérdezési technika szükséges. Ezt biztosítják az Üzleti intelligencia megoldások, melyek egy közvetítőrétteg segítségével épülnek az alsó szintű vállalati információrendszerekre. A felső szint lényegében tehát a korábbi DSS- és EIS-rendszerek modernizált és egységes utóda.
- A szervezet belső és külső határai elmosódnak: a vállalat kapcsolatrendszere és adatforrásainak köre kibővül, a piaci változások a legtöbb ágazatban felgyorsulnak. Ennek a kihívásnak kell megfelelniük mind az ERP-, mind pedig a felsővezetési döntéstámogatást ellátó információs rendszereknek.



12.8. ábra: Üzleti intelligencia és az integrált vállalatirányítási rendszer

A gazdálkodó szervezetek számára rendelkezésre álló, egyre növekvő mennyiségű információ feldolgozása, hatékony felhasználása elengedhetetlen a versenyképesség

megőrzéséhez. A korábban alkalmazott informatikai módszerek, technikák nem képesek megbirkózni a rendelkezésre álló nagymennyiségű adattömeggel.

A feladat megoldását az üzleti intelligencia (Business Intelligence) eszközei nyújtják, ami tulajdonképpen nem jelent mást mint az adatok elérését, elemzését és megosztását az adott szervezetenél. Az alkalmazás során a rendelkezésre álló adatok rendszerint egy központi adattárba kerülnek, amely egyrészt biztosítja az adatok integritását, másrészt alapjául szolgálhat a további elemzéseknek. Ez utóbbi az adatbányászat eszköztárával, multidimenzionális adatbázison, vagy az adattárházra épített, a belső felépítést elrejtő riportfelületen keresztül valósul meg.

Az adatok jobb döntések lehetővé tételét célzó konszolidálásával és újraszervezésével jelentős előnyökre lehet szert tenni: ezen előnyök felfedezését és hasznosítását hívjuk „Üzleti intelligenciának”. Az Üzleti Intelligencia azonban több, mint adatok és technológiák kombinációja: az információ tudássá transzformálásáról szól, a megfelelő adat eléréséről, a benne rejlő lehetőségek felfedezéséről és értékeinek megosztásáról.

Az Üzleti intelligencia megoldások (BI: Business Intelligence) körébe olyan alkalmazások és technológiák tartoznak, melyek célja, hogy a szükséges adatokhoz való hozzáférés biztosításával, ezen adatok megfelelő tárolásával, valamint sokoldalú elemzési lehetőségekkel támogassák a vállalati döntéshozatalt. Az Üzleti intelligencia megoldások magukban foglalják tehát az adattárolási, a valós idejű lekérdezési, analitikai, előrejelzési és adatbányászati eljárások modern formáit.

12.4.1. On-line elemző feldolgozás

Napjainkban a vezetői információs rendszereket, a vezetői döntéstámogatást egy több szintű adatkezelési és analitikai megoldás formájában valósítják meg:

- Az alapot tehát a ***belső, illetve külső adatforrások*** jelentik -- az adattárházak ugyanis nem csak az ERP-rendszerekből vesznek át adatokat, hanem más forrásokból is. Ezáltal szélesebb alapokat adnak a vezetői döntések meghozatalához, másrészt feladatként jelenik meg a különböző helyekről különböző struktúrákban és formátumokban érkező adatok bevonása és kezelése.
- Az adatforrás rétegre épül egy ***közvetítő réteg***, mely biztosítja a kiválasztott adatok beolvasását, konvertálását, szükség esetén tisztítását (újracsoportosítását -- az adattárház tagolásának megfelelően, standardizálását -- konzisztenssé tételét, tartalmi ellenőrzését és illesztését), azért, hogy az adattárház teljes és minőségi adatállománnyal rendelkezzen.
- A következő réteg már maga az ***adattárház***, mely magában foglalja a ***bevont adatmennyiséget***, valamint az ***adatszótár***at: az adatok tulajdonságait, illetve az adatok közötti kapcsolatokat leíró metaadatok rendszerét.
- Mivel a szervezet különböző területein sokfajta, eltérő információigény jelentkezik, ezen igények hatékony kielégítésére a központi adattárház bázisán az egyes területekre optimalizált ***adatpiacokat*** építenek ki. Adatpiacok alakíthatóak ki a controlling szabályozóköreinek támogatására, valamint a vállalati értékteremtés folyamatának állomásainál és a funkcionális egységek szintjén.
- A legfelső szintet az ***OLAP-rendszerű alkalmazások*** jelentik: magas szintű analitikai képességeiket az informatikai szállítók fejlesztései révén ma már számos vállalati területen igénybe vehetik a felhasználók. Ezek az ***Elemzési rendszerek*** alapvetően egyrészt a controlling szabályozóköreinek támogatására, másrészt a vállalati értéklánc,

illetve a funkcionális egységek elemzési feladatainál alkalmazzák. Logikus, hogy az adatpiacok és az OLAP vállalati megjelenése azonos elvet követ -- az egyes szervezeti területek információigényének kielégítését és a speciális területi feladatok hatékony ellátását ez a két szint együttesen biztosítja. Az OLAP-szoftverek valamilyen **adatkezelő nyelven** (legtöbbször SQL) kommunikálnak az adatkezelés alsóbb rétegeivel, legtöbbször osztott rendszerként működnek és lehetővé teszik több felhasználó, felhasználói csoport egyidejű munkavégzését.

- Az **adatkezelés alkalmazásoldali menedzselése**, illetve az egyes rétegek, szoftverek, hálózati **egységek közötti kommunikáció** biztosítása szintén fontosak, az előzőektől különválasztandó funkciók.

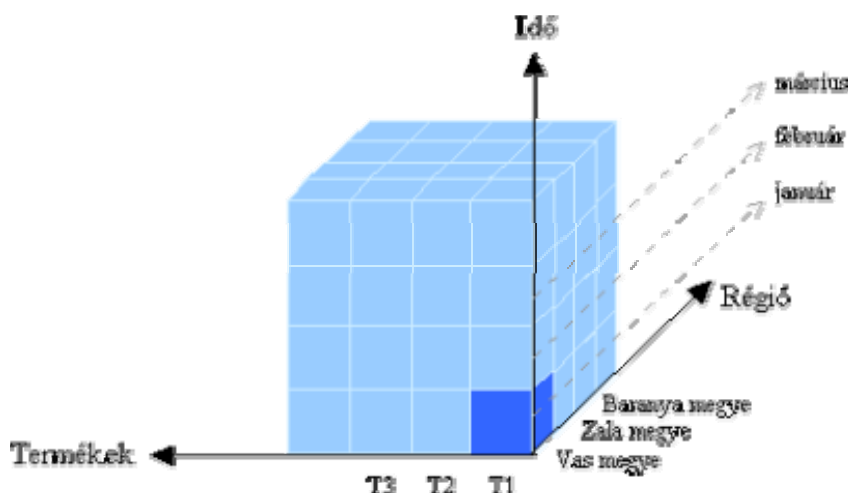
A következőkben részletesebben foglalkozunk az adattárházakkal, bemutatjuk az adatpiacok megjelenését kiváltó gyakorlati okokat, illetve a független adatpiacoktól az adattárház + integrált adatpiacok architektúráig vezető utat; valamint az OLAP-rendszerű alkalmazások két fő területét.

A vezetői információs rendszerekkel kapcsolatban korábban felvázolt kérdések és problémák (lekérdezések sebessége, rugalmassága, stb.) kezelését az adattárházak **négy vezérelv** mentén valósítják meg. Ezek a vezérelvek úgy hoznak megoldást a korábbi MIS-, DSS-, EIS-rendszerek problémáira, hogy alapvetően egy másfajta szemlélettel fordulnak az adattárolás és adatkezelés felé.

12.4.2. Több dimenziós adatbázis

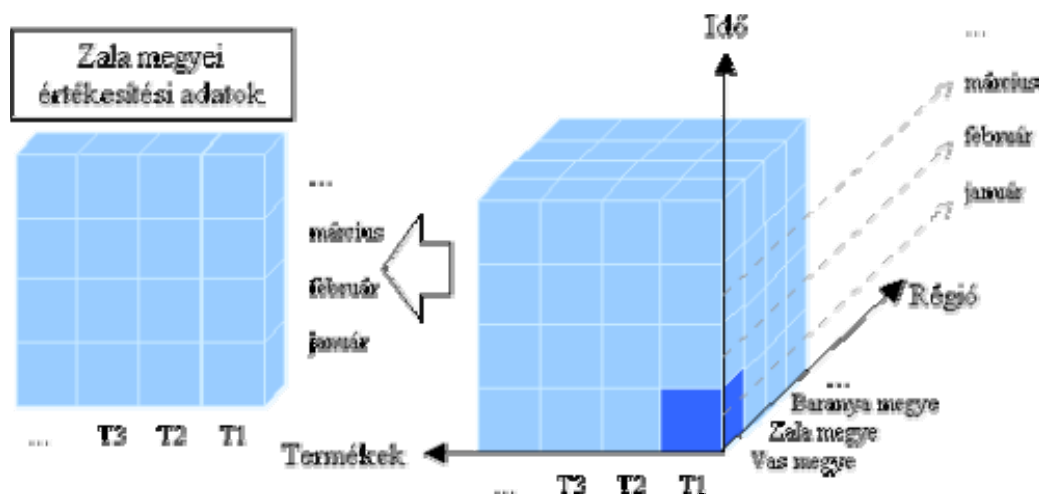
A fenti vezérelvek gyakorlati megvalósulását a hagyományos relációs adatmodell kevésbé tudja támogatni, az adattárházak alkalmazásával előtérbe került a **multidimenzionális adatmodell**, illetve az ennek megfelelő adatkezelés. A multidimenzionális adatmodell elmélete már 1972-ben megszületett a Massachusetts-i Technológiai Intézetben, de igazából csak a 90-es években, az adattárházak és az OLAP-rendszerű alkalmazások sikerével vált széles körben ismertté. Míg a relációs adatmodellt egymással kapcsolatban lévő táblákkal illusztrálhatjuk, addig a multidimenzionális adatmodell egy háromdimenziós **kockával** szemléltethető.

A kocka egyes élei különböző **dimenziókat** jelentenek, például: termékeket (T1, T2, T3, stb.), régiókat (Vas megye, Zala megye, Baranya megye, stb.), illetve időszakokat (január, február, március, stb.). Attól függően, hogy milyen témában, milyen üzleti tevékenységhez építjük a kockát, más és más dimenziókat fogunk használni. (A fenti példa dimenziói tartozhatnak egy, a vállalat értékesítését elemző kockához.) A nagy kocka a dimenziók tagolásával kisebb kockákra, **cellákra** bontható. Példánknál maradván az egyes cellák adott termék, adott régióban, adott időpontban elért eladási adatát mutatják. A sötétebb színnel jelzett cella eszerint azt jelzi, hogy T1 termékből, Vas megyében mennyit értékesítettek januárban.



12.9. ábra: Adatkocka egyszerűsített modellje

Természetesen a valós életben példánknál jóval több dimenzió kapcsolódik egy-egy témához, ezért az adattárházakban többdimenziós kockákat építenek és használnak. A lekérdezések során azonban néhány meghatározott dimenzióérték mellett hív le a vezető vagy a controller kétdimenziós táblákat, vagyis **szeleteljük a kockát**: ez az úgynevezett "slice and dice" funkció. Például a Zala megyei termékmenedzser leginkább az érdekl, hogy havonta mennyit értékesítettek az egyes termékekből, ezért a termékmenedzser számára a termékek-idő szelet releváns.



12.10. ábra: Adott nézet / szelet lekérdezése a kockából

Az adattárházak adatkockáin további alpműveletek végezhetőek. A legtöbb dimenzió esetében igaz, hogy több különböző részletezettség mellett is értelmezhető. A legegyszerűbb példa erre az idő, melynél számolhatunk napi / heti / havi / évi bontással. A felsővezetésnek készített jelentések havi bontásban mutatják például az eladások alakulását, azonban lehetőségük van arra, hogy megtekintsék a heti bontásokat is. A vállalat adatraktárában tárolt adatokat ugyanakkor alsóbb szintű vezetők is használhatják, akik inkább heti vagy napi adatokat akarnak látni. Ez a dimenziók mentén történő **felösszegzés**, illetve alábontás, **lefűrés**: a drill up / drill down funkció lehetővé teszi, hogy ugyanaz a téma (ugyanazon dimenziókkal rendelkező kocka) több aggregáltsági szinten is megjelenjen.

Az összvállalati tervezés, elemzések, szimulációk elvégzésére alkalmas OLAP-rendszerű alkalmazások felhasználják és igénylik az adattárházak nyújtotta adatkezelést.

A multidimenzionális szemlélet megfelel a felsővezetői információigényeknek; az adattárházak konszolidált, integrált és valamilyen szinten aggregált adatokat tartalmaznak, olyan módon, hogy a közöttük lévő bonyolult kapcsolatok mellett is sokféle -- és gyorsan elvégezhető -- lekérdezést tegyenek lehetővé. Ezt elősegítendő a fizikai adattárolás szintjén is új, modern megoldásokat alkalmaznak. A konkrét adatok mellett azonban tárolni kell az adattulajdonságokat, illetve a kapcsolatokat leíró metaadatokat is, valamint a számítások meggyorsítására sokszor több aggregációs szint adatait már **előre elkészítve** tartalmazza az adattárház.

Az újabb és újabb fejlesztések fokozatosan tágítják az adattárházak **méretkorlátait**. A korai időszakban ugyanis a multidimenzionális szemléletet még nem tudták összvállalati szinten alkalmazni. Ezért gyakran azt a megoldást választották, hogy az ERP-rendszerek relációs adatbázisa fölé egy második, a felsővezetői információigényeknek megfelelő aggregációs szintű és szervezettségű adatokat tartalmazó, de szintén relációs modellen alapuló adatbázist alakítottak ki. (Egy köztes alkalmazás, az úgynevezett OLAP-motor szükséges ilyenkor, hogy az OLAP-szoftverek által igényelt adatkezelési minőséget -- táblák helyett témaorientáció és multidimenzionális szemlélet, adatok közötti kapcsolatok hatékony kezelése, műveletek, stb. -- erre a relációs adatbázisra alapozva is biztosítani lehessen.)

A különválasztott **vezetői adatbázis** esetében az elgondolás már jó volt tehát, a módszerek azonban -- mint minden kezdetnél -- még kiforratlanok. A vállalatoknál gyakran évekig dolgoztak azon, hogy kialakítsák ezeknek a korai adattárházaknak a struktúráját, megoldják az adatok bevonását, majd elfogadhatóvá tegyék a lekérdezések futási idejét.

Új megközelítés jelent meg ekkortájt: az egész vállalat adattömegére nehézkesen felépíthető összvállalati adattárházak helyett az egyes vállalati területeken adatpiacok kialakításába kezdtek. Az informatikai cégek azt már meg tudták valósítani, hogy egy-egy funkcionális terület adatmennyiségére alakítsanak ki döntéstámogató multidimenzionális alkalmazásokat; ráadásul ezek a projektek a vállalat részéről is kevesebb ráfordítást igényeltek, s biztosabb eredményeket hoztak.

Az adatpiacok sikere azonban hamarosan új problémát vetett fel. Amikor ugyanis a vállalat egységei sorra építettek saját adatpiacokat, kiderült, hogy a különálló fejlesztések csak az adott funkcionális terület vagy elszámolási egység szintjén működőképesek. Ezek a **független adatpiacok** nem egymással harmonizáltan gyűjtötték az adatokat a belső vagy külső forrásrendszerekből, más struktúrákban, más logikai elvek mentén tárolták azokat, és eltérő elvárásokat fogalmaztak meg. Emiatt az adatpiacok közötti kommunikáció megteremtése csak igen nagy költségek és hosszadalmas informatikai újraszervezések révén volt megvalósítható.

Szemléletesen: ha egy vállalat regionális központjai az évek alatt külön-külön alakítottak ki adatpiacokat és más-más terméklistákat használtak ezekben, akkor adataik nem összevethetőek. Hasonlóan: ha az értékesítés adatpiacában más a terméklista, mint amit a számvitel, a controlling használ, akkor összvállalati szinten már nem tudjuk ezeket összekapcsolni (és nem tudjuk például a különböző fedezetű termékek vevőinek megoszlásáról informálni a felsővezetést) -- illetve csak azután, hogy harmonizáltuk a listákat. (Az együttműködés támogatásában fontos szerepet játszik a honosított IT-rendszer, illetve többnyelvű környezetben az, hogy a felhasználók által használt különböző nyelveken ugyanaz a tartalom és ugyanaz a struktúra jelenjen meg.)

Az adattárházak és független adatpiacok korai megvalósítási tapasztalatai alapján, illetve az informatikai megoldások fejlődésével mára már hatékony bevezetési módszereket és IT-megoldásokat fejlesztettek ki.

- Azért, hogy összvállalati szinten is biztosítva legyen az adatok és műveletek konzisztenciája, először kijelölik és rögzítik a legfontosabb dimenziókat, azok tagolását, valamint meghatározzák a kiépítendő, szervezeti egység szintű adatpiacok kapcsolódási pontjait -- viszonyrendszerüket.
- Erre az alap-keretrendszerre támaszkodva egymás után (vagy elegendő erőforrás esetén egymással párhuzamosan haladva) kialakítják az egységek adatpiacait. Az így létrehozott adatpiacok egymással bizonyosan tudnak kommunikálni, de ezen túlmenően az adott funkcionális terület, folyamatrész speciális igényeinek is megfelelnek. Ekkor tehát **konform adatokkal és dimenziókkal rendelkező adatpiacok integrált rendszeréről** beszélünk.

12.4.3. Programcsomagok

SAS

A SAS olyan teljeskörű megoldásokat, szoftvereket kínál, amelyek megfelelnek az intézményekkel szemben támasztott üzleti kihívásoknak, illetve megoldást adnak az iparág specifikus problémákra. Lekérdező, jelentés- és kimutatás-készítő és OLAP eszközök.

A lekérdezések, kimutatás készítések tipikusan grafikus végfelhasználói felület használatával valósulnak meg, de biztosított a programsorból való lekérdezés, elemzés lehetősége is. A SAS rendszer támogatja az SQL-t. A lekérdezések, jelentéskészítések többféleképpen valósíthatók meg. Egyik lehetőség, hogy az adattárházban előkészített, rendszerezett adatokon egy elemző eszközt használva (SAS/Enterprise Reporter, SAS/Enterprise Guide) a felhasználók saját maguk állítják össze jelentéseiket, kimutatásaikat. A másik lehetőség, hogy a SAS fejlesztőeszközeit használva egyedi, testre szabott jelentéskészítési környezet kerül kialakításra. Az OLAP eszközök segítségével az elemzők grafikus felület segítségével, dinamikusan (a dimenziók és a részletezettségi fok előre definiálása nélkül) állíthatják össze lekérdezésüket és végezhetnek elemzéseket akár vegyes, relációs és multidimenzionális formában tárolt adatokon is. Az adatok megjelenítése történhet táblázatos, grafikus, térképes formában. A dimenziókat (pl. idő, tér, értékesítési csatorna) és a dimenzió belüli hierarchiát (pl. év / negyedév / hónap / hét hierarchia az idő dimenzió belül) elemzés közben on-line változtathatja az elemző, így lehetővé válik a döntéshozatal támogatása a kellő időben rendelkezésre álló, könnyen értelmezhető információk szolgáltatása által.

A SAS/EIS szoftver modul objektum orientált, menüvezérelt, programozást nem igénylő rendszer OLAP alkalmazások fejlesztéséhez, futtatásához és karbantartásához. A kész objektumokból való építkezésnek köszönhetően egyszerűen és gyorsan alakíthatók ki komplex rendszerek. A jelentéskészítő objektumokon kívül a SAS/EIS több mint 30 előre elkészített objektumot tartalmaz. A SAS/EIS szoftver a SAS rendszer többi részének komplett **front-end** eszköze. Fejlesztéskor a SAS/EIS-t gyakran a későbbiekben ismertetésre kerülő SAS/AF fejlesztőeszközzel kombinálva célszerű használni.

A SAS/Enterprise Guide ötvözi a SAS szoftver világszinten elismert statisztikai és adatmanipulációs képességeit egy modern, Windows-os grafikus felhasználói felülettel. Segítségével a felhasználók könnyen és gyorsan végezhetnek statisztikai elemzéseket, adatösszesítéseket, leválogatásokat, az eredményt pedig látványos, jól áttekinthető formában publikálhatják. Használata nem igényel programozói tapasztalatot. A szoftver lehetővé teszi, hogy bármilyen - a SAS rendszer által támogatott - adatot elérjünk, lefutassunk SAS procedúrákat, alkalmazásokat a SAS szerveren, az eredményeket pedig változatos jelentés,

kimutatás, grafikon formátumban jelenítsük meg. Az eredmény automatikusan HTML-ben generálódik, weben publikálható, átemelhető MS Office alkalmazásokba, e-mailen terjeszthető, nyomtatható. Az Enterprise Guide segítségével az elemzéseket megelőző adattisztítás és adatmanipuláció hatékonyan végezhető el. Az alkalmazás - amit a felhasználó többnyire egér kattintásokkal állít össze - elmenthető és egy későbbi időpontban ugyanaz a feladatsor végrehajtható. Az adatmanipulációs lehetőségek mellett egy nagyon komplex statisztikai eszköztár áll rendelkezésre. Az alapstatisztikák mellett klaszter elemzés, regresszió és számos egyéb elemző eszköz is elérhető az Enterprise Guide-ban.

A SAS/Enterprise Reporter egy felhasználóbarát eszköz, ami intelligens szerver hozzáférést kombinál egy PC alapú jelentéskészítő eszközzel. Microsoft Office jellegű megjelenésének köszönhetően használata gyorsan elsajátítható. Tartalmaz egy "Report Gallery" komponenst, ebben előre megírt jelentés minták (template-ek) találhatók illetve ide menthetünk el általunk készített vagy módosított mintákat. Az Enterprise Reporter integrálódik a SAS rendszer többi részével, ennek köszönhetően a felhasználók bármilyen jelentést megvalósíthatnak a scoring adatbázis adatain anélkül, hogy ismernék az adatok fizikai helyét, eredetét. Az üzleti felhasználók számára az adatok fizikai helye, eredete transzparens.

Másik alapvető funkció az OLAP rendszerek esetében a lefűrés. Ez azt jelenti, hogy a dimenziók hierarchikus felépítésűek, és egy-egy kattintással lehet felfelé összegezni, és lefelé szétbontani az adatokat (pl. negyedévről havi bontásra, vagy termékcsoportról termék bontásra).

Microsoft

A Microsoft® SQL Server™ OLAP Eszközök egy kimerítően teljes OLAP implementáció, ami a Microsoft SQL Server 2000 része. Az OLAP Eszközöknek része egy middle-tier szerver, ami a felhasználók számára lehetővé teszi, hogy modern analízist hajtsanak végre adatok nagy volumenén kivételes teljesítménnyel. Egy másik része az OLAP Eszközöknek a kliens cache és a számítási motor, amit PivotTable® -nek hívnak, ami tovább segíti a teljesítmény növelését és csökkenti a hálózati forgalmat. A PivotTable segítségével a felhasználók analíziseket hajthatnak végre, miközben nem csatlakoznak a céges hálózathoz.

Az OLAP egy kulcsfontosságú komponens az adattárházak feldolgozásakor, és az OLAP Eszközök megadják azt a szükséges funkcionalitást, ami applikációk tág körében felhasználható a cég kimutatásoktól a döntéstámogatásig. Az OLAP funkcionalitás SQL Szerverbe helyezésével a multidimenzionális analízis megengedhető lett és nagyobb közönséghez jutnak el az OLAP előnyei. Ezen közönség nem csak a kisebb vállalatokat foglalja magában, hanem csoportokat és egyéni felhasználókat is a nagyobb vállalatoknál, amelyek eddig kimaradtak az OLAP iparból az ár vagy a jelenleg kapható eszközök komplexitása miatt.

Microsoft Office integráció

Az Office 2000-s verziójától kezdve, az OLAP Eszközökkel kompatibilisen sok és gazdag OLAP böngészési lehetőséggel bővült az Excel. Az OLE DB for OLAP interfészen alapulva ezek az új lehetőségek élő kapcsolatot kínálnak a szerverekkel, de ugyanúgy választhatjuk a kapcsolatmentes illetve Web-alapú hozzáférést. Először az Excel 2000-ben megjelent új PivotTable kínált dinamikus vizsgálati lehetőségeket, kapcsolatot egy Excel számológéptábla és az OLAP szolgáltató között. Abban az esetben, ha a Microsoft OLAP

Eszközöket használjuk, további lehetőségeink is vannak, mint például lokális logikai kockaszeleteket hozhatunk létre az Excel segítségével.

Az új Office Web Komponensek segítségével lehetőségünk nyílik, hogy egyszerű OLAP böngészési illetve diagram készítési lehetőségeket nyújtsunk egy ActiveX kontroll segítségével, amit bármilyen Weblaphoz hozzáadhatunk, akár Sharepoint Portal Server DashBoard-hoz is. Mivel ez a technológia is az OLE DB for OLAPon alapul, bármilyen ezzel kompatibilis OLAP szolgáltató használható.

Ellenőrző kérdések:

1. Melyek a döntés-előkészítési folyamat lépései?
2. Milyen eszközökkel segíthet egy döntéstámogató rendszer az alternatívák közötti választásban?
3. Milyen előnyöket várhatunk a döntéstámogató rendszerektől?
4. Melyek a fontosabb jellemzői az egyéni döntéshozatalnak?
5. Melyek a csoportos döntéstámogató rendszerek fontosabb tulajdonságai?
6. Milyen főbb részekből épülnek fel a döntéstámogató rendszerek?
7. Melyek a modellkezelő rendszer fő részei?
8. Melyek a döntéstámogató rendszerek főbb felhasználási módjai?
9. Mi az üzleti intelligencia?
10. Mire szolgál az OLAP technológia?
11. Milyen döntéstámogató programrendszereket ismer? Jellemeze őket!

Irodalomjegyzék

Dobay Péter - Vállalati információ-menedzsment, Nemzeti Tankönyvkiadó, 1999

Gábor András (szerk.) Információ-menedzsment, Aula Kiadó, 1998

Hetyei József (szerk.) Vezetői döntéstámogató és elektronikus üzleti megoldások

Magyarországon, 2001, Computer Books, ISBN: 963 618 246 9

Szentpéteri Szabolcsné: Gazdasági döntések bizonytalanság esetén (Közgazdasági és Jogi Könyvkiadó Budapest 1980

Turban, E.; Meredith, J. (1991) Fundamentals of Management Science. 5th ed. Homewood, IL: Richad D. Irwin

13. E-KORMÁNYZAT, E-SZAKIGAZGATÁS

Az Európai Bizottság 1999. decemberében hozta nyilvánosságra programtervezetét, amely az e-Europe (elektronikus Európa) nevet viselte. Ezzel új politikai programot adott az Új Gazdaság (New Economy) korszakára. A program része az on-line kormányzás, az elektronikus közigazgatás.

Az e-Europe program tíz legfontosabb fejezete:

- A fiatalok beléptetése a digitális korszakba
- Olcsó Internet-hozzáférés
- Az elektronikus kereskedelem terjedésének gyorsítása
- Gyors Internet a kutatók és a diákok számára
- Intelligens kártyák a biztonságos elektronikus hozzáféréshez
- Kockázati tőke (kis és középvállalatok) számára
- Elektronikus részvételi lehetőség a fogyatékos, csökkent munkaképességű és hátrányos helyzetű személyek számára
- On-line egészségügyi szolgáltatások
- Intelligens közlekedés, szállítás
- On-line kormányzás

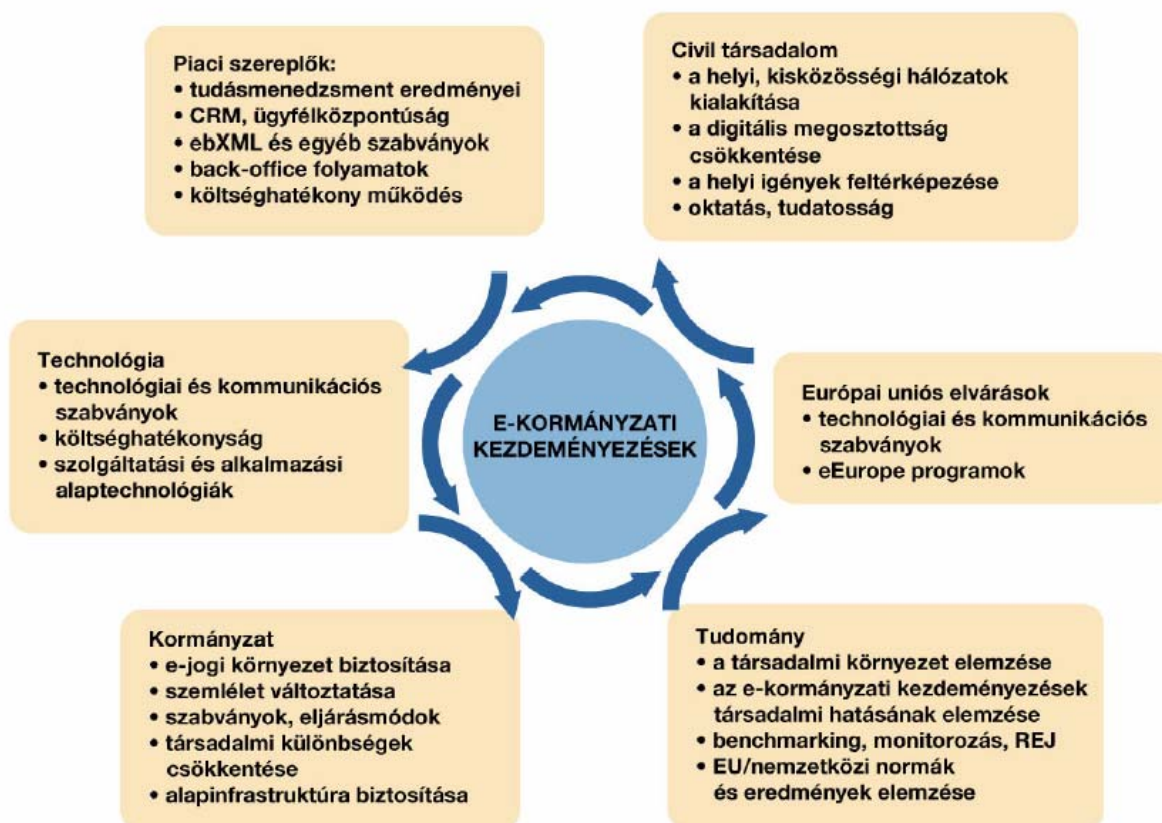
Az e-Europe központi üzenete maga a cím: Információs társadalmat mindenkinek! A második fő üzenet, hogy a digitális korszak technikailag lehetséges alkalmazásaival törekedjünk az életminőség javítására, kezdve az elektronikus kereskedelemtől az on-line egészségügyig. A program utolsó pontja pedig a demokráciamodel megújítását követeli az on-line kormányzásra való áttéréssel. A harmadik átfogó üzenet az, hogy az elektronikus Európa (a digitális Európa, az európai információs társadalom) megerősödése érdekében a tudásgazdaság, az információs gazdaság fontosabb alkalmazásait (e-kereskedelem, intelligens kártya, stb.) gyorsan és minél szélesebb körben terjesszük el.

A fejezetben az e-kormányzat fogalmának megismerését követően az állam és szakigazgatási szervezetek néhány alapvető elektronikus szolgáltatási feladataival és lehetőségeivel ismerkedünk meg.

a. E-kormányzat fogalma

Az elektronikus kormányzat (e-kormányzat) kifejezés mára mind a köznyelvben, mind az államigazgatásban egy univerzális jövőképjavító tényezővé nőtte ki magát. Hatóköre túlterjed a szorosan vett államigazgatáson, központi kormányzati igazgatáson, átfogja a teljes közigazgatást és azon kívül a közösségi szolgáltatásokat is.

A nemzetközi és a hazai tapasztalatok alapján jól látható, hogy az e-kormányzati szolgáltatások legnagyobb igénybevevői, a lakosság és a vállalkozók mellé hogyan zárkoznak fel mindinkább a civil szervezetek. Az igénybevett szolgáltatások egyelőre a kormányzati portálok és egyéb webhelyek szolgáltatásszerkezetét tükrözik vissza: legtöbbször a közigazgatással kapcsolatos információkat keresnek az interneten, de már jól érzékelhető az igény a nyomtatványok letöltése és online visszaküldése, valamint a tranzakciók lebonyolításának lehetősége iránt is. A vállalkozások elsősorban az adózási és társadalombiztosítással kapcsolatos ügyek intézésénél, valamint adatszolgáltatásra vennék igénybe az internetet, azaz sokkal inkább a fejlettebb kétoldalú és tranzakciós lehetőségeket igényelnék, míg a hazai civil szervezetek körében inkább az információs szolgáltatás és egyszer kommunikáció, információ-kérés iránt mutatkozik érdeklődés. Ezekre az elvárásokra épülnek az e-kormányzás működését is meghatározó alapelvek, követelmények:



Forrás: E-Kormányzat Stratégia és Programterv 2005. MeH Elektronikus Kormányzat Központ

13.1. ábra Az e-kormányzati fejlesztésekre ható erőtér

- o **nyitottság:** a közigazgatás legyen kész fogadni és feldolgozni a nyilvánosság, a társadalom és gazdaság szereplőinek elvárásait és javaslatait;
- o **részvétel:** az állampolgárokat, vállalkozásokat, közösségeket érintő kérdések megvitatásába, a döntések előkészítésébe minél szélesebb körben be kell vonni az érintetteket. A központi és a helyi kormánzatnak olyan politikai környezetet kell biztosítania, amely részvételre, bekapcsolódásra ösztönzi a polgárokat, közösségeket. A közigazgatásnak folyamatosan jeleznie kell ezirányú elkötelezettségét;
- o **számokérhetőség:** a döntéshozatali folyamatoknak átláthatóbbakká kell válniuk, lehetőséget kell biztosítani a bekapcsolódásra; a nyilvánosság számára egyértelművé kell tenni, hogy ki miért felel;
- o **visszacsatolás:** az érintettek számára lehetőséget kell biztosítani a vélemények, javaslatok, észrevételek visszacsatolására a döntéshozók, politikusok, köztisztviselők felé;
- o **hatékonyság:** legyen hatékony a döntéshozatali mechanizmus, az elfogadott határozatok, jogszabályok végrehajtása;
- o **elérhetőség:** az e-kormányzati kezdeményezésekkel párhuzamosan folyamatosan biztosítani és bővíteni kell a hagyományos kommunikációs és

szolgáltatási csatornákat azok számára, akik nem kívánnak (vagy nincs lehetőségük) élni az új IKT eszközök adta lehetőségekkel.

A POLGÁROK SZÁMÁRA NYÚJTOTT SZOLGÁLTATÁSOK	AZ ÜZLETI SZFÉRÁNAK NYÚJTOTT SZOLGÁLTATÁSOK
Jövedelemadó – adóbevallás megtétele, értesítés a kivetett adóról Álláskeresés – munkaügyi hivatalok Társadalombiztosítási kifizetések (legalább három az alábbi négy lehetőség közül) – munkanélküliek járadékai – gyermekek után járó pótlékok – gyógyászati költségek – tanulói ösztöndíjak Személyi dokumentumok – útlevél, gépjárművezetői jogosítvány Gépkocsik nyilvántartásba vétele – új, használt, importált autók Építési engedély kérelem A rendőrségnek tett bejelentések – pl. tolvajlás esetében Közkönyvtárak – katalógusokhoz, keresési lehetőségekhez való hozzáférés Születési és házassági bizonyítványok – kérelmezésük és kiadásuk Felsőbb oktatásba történő jelentkezés – ide értve az egyetemeket is Költözés bejelentése – lakcímváltozás Egészségüggyel összefüggő szolgáltatások – pl. interaktív tanácskérési lehetőség a különböző kórházi szolgáltatások elérhetőségéről, kórházi bejelentkezések	Munkavállalók részére biztosított hozzájárulások Társasági adó Általános forgalmi adó Új társaság bejegyzése Adatközlés a statisztikai hivatalnak Vámnyilatkozat Környezetvédelemmel összefüggő engedélyek Közbeszerzés Az egyes szolgáltatások benchmarkja négy minőségi mutató alapján történik, a szolgáltatás jellege szerint: – információs, tájékoztató; – egyirányú interakciót biztosító; – kétirányú interakciót biztosító; – a teljes online tranzakciót (ügyintézési folyamatot) biztosító (ide értve a fizetést és a kézbesítést is) lehetővé tevő

13.2. ábra On-line szolgáltatási elvárások

Az e-kormányzati kezdeményezések egy olyan bonyolult, összefüggő társadalmi erőterben értelmezhetőek, amelyben fő tényezők a piaci szféra, a civil társadalom, a közigazgatás, valamint a tudomány. Nem hagyhatók figyelmen kívül az információs és kommunikációs technológiai eszközök fejlődéséből adódó új lehetőségek, valamint az (Magyarország esetében döntően az Európai Unió felől érkező) elektronikus kormányzattal kapcsolatos előírások, elvárások, szabványok, kötelezettségek. Így érthetően kulcskérdéssé válik, hogy az egyes társadalmi alrendszerek (piac, civil társadalom, közigazgatás), valamint a technológiai fejlődés és a nemzetközi kihívások, elvárások mivel tudnak hozzájárulni az e-kormányzat fejlődéséhez.

Az Európai Unió ajánlása a 20 leginkább keresett szolgáltatás on-line elérhetővé tételéről az állampolgári, illetve a vállalkozói elvárások és igények összegzését adja, egyúttal egyfajta előírásként, megteremtendő normaként működik.

b. Az önkormányzatokkal szembeni elvárások

Az önkormányzatok tevékenységével kapcsolatban az állampolgárok és a gazdasági élet szereplői esetében is igény a gyors, hatékony, átlátható ügyintézés, a hatékony településfejlesztés és –gazdálkodás, a munkahelyek lehetőség szerinti megőrzése, új munkahelyek teremtésének támogatása, és így tovább. A közigazgatási, ügyintézési folyamatok szervezettségének növelése, az önkormányzati szervek belső működési hatékonyságának fokozásával, korszerű információs rendszerek alkalmazásával tervezhetővé, átláthatóvá, követhetővé válik a település, az önkormányzati szervek gazdálkodása, alaposabb, sokoldalúbb lehet a képviselő-testület döntéseinek előkészítése, megalapozottabbakká válhatnak a döntések. Jelentősen javulhat a különböző szervek közötti adat- és információcseré, valamint az információ minősége.

Az ügyintézéssel kapcsolatos, főbb elvárások

- Önkormányzat-lakosság, hivatal-lakosság közti interakció biztosítása
- Az ügyfélfogadás térbeli és időbeli korlátainak kitolása, ill. feloldása
- Gyorsabb, egységes, diszkriminációmentes ügyintézés
- Azonos ügy - azonos ügyintézés

A szervezettséggel, szervezéssel kapcsolatos fontosabb elvárások

- Átlátható folyamatok, jogkövető megoldások
- Felelőségek egyértelmű definiálása, nyomon követése konkrét esetekben is
- Elemzésekhez alapadatok generálása, statisztikai adatok előállítása, jelentések készítése (jelenleg közel 40-féle különböző statisztika létezik; pl. minden tárca, ágazat külön-külön kér adatokat; nincs koordináció a különféle statisztikai adatszolgáltatási igények vonatkozásában)
- Az önkormányzati szervek munkatársai informatikai felkészültségének elmélyítése
- Önkormányzatok közötti információ-csere előmozdítása
- Fokozatosan kiépíthető e-önkormányzati informatikai modell kialakítása
- Eljárási viták, felelősségi kérdések eldöntéséhez tényadatok szolgáltatása
- Szükségtelen adatszolgáltatások, párhuzamos munkafolyamatok elkerülése
- Párhuzamos irattárak kiküszöbölése
- Adatvédelmi szabályzatok kidolgozása, betartásuk biztosítása
- A megfelelő archiválás biztosítása, törvényes adatőrzési időszakok betartatása
- Államigazgatási adatvagyon bővítése
- Egységes közigazgatási fogalomtár definiálása, kialakítása és felhasználása
- Minőségbiztosítási módszerek bevezetése, egységesítése az önkormányzati szerveknél

c. Elektronikus közszolgáltatások, ügyintézés

A közszolgáltatások végzésére vonatkozó Európai Unió ajánlás, a "Common List of Basic Public Services" a tagállamok számára elvárásokat határoz meg az állampolgároknak, ill. az üzleti élet szereplőinek elektronikusan nyújtandó közszolgáltatások körére, és azok interneten keresztül történő igénybevételének szintjeire vonatkozóan. Az ajánlás négy elektronikus szolgáltatási fejlettségi szintet különböztet meg:

1. szint:	információ	On-line információk nyújtása a közigazgatási szolgáltatásokról („ügyleírások”)
2. szint:	egyirányú interaktivitás	Az ügyleírások által nyújtott információkon túl az ügyintézéshez szükséges űrlapok, nyomtatványok is letölthetők, kinyomtathatók
3. szint:	kétirányú interaktivitás	Az ügyintézéshez szükséges űrlapok, nyomtatványok on-line kitölthetők, elektronikusan (elektronikus aláírás segítségével) hitelesíthetők, s ugyancsak on-line továbbíthatók
4. szint:	teljes körű elektronikus ügyintézés	A teljes ügyintézés, ügymenet elektronikus űrlap, nyomtatvány kitöltése, hitelesítés, továbbítás, döntés, kézbesítés, illeték lerovása (természetesen elektronikus aláírás felhasználásával)

i. E-ügyintézés: az önkormányzati ügyintézési szolgáltatások elektronizálása

Az e-ügyintézés megvalósítja az EU eEurope programjának „Common List of Basic Public Services” ajánlásában megfogalmazott mind a négy szolgáltatási szintet, s biztosítja az ügyfelek részére is a függőben lévő ügyeik elektronikus követését. (Minden szint megvalósítása önállóan is működőképes rendszert eredményez, s minden szint bevezetése egy-egy újabb komoly lépés a szolgáltató önkormányzat, ügyfél-barát közigazgatás megvalósítása irányában). A szolgáltatások harmadik és negyedik szintjéhez szükséges a minősített fokozatú elektronikus aláírás használata (amelyre államigazgatási eljárásokban, önkormányzati ügyekben, közjegyzői ellenjegyzéshez, ügyvédi aláíráshoz, sőt nagyobb értékű tranzakció esetén elektronikus beszerzéshez is szükség lehet). Az elektronikus aláírást „A 2001. évi tv. Az elektronikus aláírásról” c. törvény rögzíti. A Hírközlési Főfelügyelet már két minősített biztonságú aláírás-hitelesítő szervezetet is regisztrált, az önkormányzati, közigazgatási alkalmazáshoz azonban a jelenlegi szabályozás korszerűsítése is szükséges (lásd például az önkormányzati ügyiratkezelést szabályozó 38/1998. (IX.4) sz. BM rendeletet a jelen dokumentációnak az önkormányzatok működése jogi környezetét bemutató pontjában). A harmadik és negyedik szint használata szintén feltételezi a széles funkcionalitást megvalósító, egymással és a front-office rendszerrel is integrált back-office alrendszerek meglétét; e nélkül ezen szintek működése nem lehetséges.

Az e-ügyintézés mint szolgáltatás értékelési szempontjai az EU-ban: a különböző ügytípusok milyen arányban szerepelnek a négy szint valamelyikében; milyen arányban érhetőek el a különböző szintű on-line szolgáltatások; az on-line szolgáltatásokat milyen arányban használják az ügyfelek. Nem elegendő tehát az önkormányzati oldalt fejleszteni, a településeken is ki kell építeni ezekhez az információkhoz való közösségi hozzáférés lehetőségeit (teleházak, könyvtárak, egyéb közösségi hozzáférési lehetőségek). A kistépülések rendelkeznek fajlagosan a legkevesebb hozzáférési lehetőséggel, így e településeken feltehetően alacsonyabb az „elektronikus írástudással” rendelkezők száma, de kevesebb a helyi önkormányzatok által nyújtott, azaz a helyben elérhető közigazgatási szolgáltatások száma is. Ahhoz, hogy a kisebb és nagyobb településen lakók egyenlő szolgáltatásokban részesülhessenek, szükséges, hogy a kisebb településeken is legyen megfelelő számú közösségi hozzáférési pont, hogy a legalább a helyben nem elérhető szolgáltatásokat döntően elektronikusan intézhessék. Növelheti a szolgáltatás komplexitását az ügyfelek szempontjából, ha további, jelentősebb ügyfélforgalommal rendelkező szervezetek is csatlakoznak a rendszerhez (APEH, földhivatalok, közüzemi szolgáltatók stb.).

ii. Az e-önkormányzat jövőkép funkcionalitása, alrendszerei, moduljai

Az önkormányzatok ügyfelei, valamint partnerei és munkatársai részére az elektronikus szolgáltatásokat a front-office modul-csoportok valósítják meg, úgymint e-ügyintézés, e-ügyfélkezelés, a közérdekű és közhasznú információszolgáltatás, ügyfélforum, -levelezés, településmarketing, elektronikus közbeszerzés, belső „ügyfelek” kezelése. Az önkormányzati szerveken belüli és az e szervek közötti kommunikációt biztosítja, s „önkiszolgáló” alkalmazásokat kínál a munkatársaknak (pl. személyi adatok megváltozásának bejelentéséhez). Az e-önkormányzás alrendszer a képviselő-testület és a bizottságok munkáját támogatja hatékonyan, illetve infrastruktúrát kínál a helyi demokrácia kiszélesítéséhez. Az elektronikus szolgáltatások háttérét biztosító belső folyamatok, tevékenységek támogatását a back-office modul-csoportok nyújtják, mint például a közigazgatási alkalmazások, az adminisztratív (költségvetési) modulok, továbbá az irodaautomatizálási és kommunikációs, vezetői információs és döntéstámogatási, illetve tudásmenedzsment rendszerek. Az önkormányzati információs rendszer (jövőkép) funkcionalitását a megoldási térkép (13.3. sz. ábra) foglalja össze.

E-ügyintézés (elektronikus ügyintézési szolgáltatások)	On-line információk a közigazgatási szolgáltatásokról (ügyleírások)	Nyomatványok, űrlapok letöltése (az ügyintézéshez)	Nyomatványok, űrlapok on-line kitöltése, hitelesítése, továbbítása	Teljes elektronizált ügyintézés (döntés, kézbesítés, illeték stb.)	Függőben lévő ügyek elektronikus követése				
E-ügyfélkezelés	Call Center		„Önkiszolgáló” alkalmazások interneten keresztül (partnerek részére)		CRM				
Közérdekű, közhasznú információk; település-marketing	Közérdekű, közhasznú információk megjelenítése	Ügyfélforum, ügyféllevelezés		Településmarketing	Elektronikus közbeszerzés				
E-önkormányzás	A képviselőtestület, és a bizottságok munkája támogatása			Helyi elektronikus népszavazás, választás					
Belső „ügyfelek” kezelése	On-line információk (intraneten, ill. extraneten keresztül)			Önkiszolgáló alkalmazások (intraneten, ill. extraneten keresztül)					
Közigazgatási alkalmazások	Adók és egyéb bevételek	Vagyongazdálkodás	Településfejlesztés, beruházások	Településüzemeltetés	Közterület-felügyelet	Ipari, kereskedelmi igazgatási feladatok	Művelődési, oktatási, sport feladatok	Szociális- és egészségügyek	Egyéb önkormányzati feladatok
	Okmányirodai feladatok (üzemeltetés)	Gyámhivatali feladatok		Anyakönyvi hivatali feladatok	Építéshatósági feladatok	Egyéb közigazgatási feladatok		Közigazgatási térinformatika	
Adminisztratív (költségvetési) alkalmazások	Számvitel, pénzügy, kontrolling	Humánpolitika, humán erőforrás-gazdálkodás,		Anyag-, tárgyeszköz-gazdálkodás,	Projekt és programok (beruházások, fejlesztések)	Elektronikus irat-, ügyirat-, dokumentum-, munkafolyamat – kezelés		Jogszabályi információk szolgáltatása (jogtárak)	
Irodaautomatizálás, kommunikáció	Szövegszerkesztés, dokumentációkészítés,		Táblázatos és grafikus kimutatás-készítés,		Elektronikus levelezés (belső, külső),		Prezentáció-készítés		Feladatkezelés, -ütemezés stb.
Vezetői információk és döntés-támogatás. Tudásmenedzsment	Stratégiai tervezés és kontroll		Önkormányzati intelligencia (döntéstámogatás, adattárház, adatbányászat)	Dokumentációmenedzsment	Csoportmunkaszoftverek (groupware),		Internet, intranet, extranet (mint információforrás)	Mesterséges intelligencia	

Forrás: IHM Elektronikus Önkormányzat Stratégia 2003

13.3. ábra Az önkormányzati információs rendszer funkciói

iii. Az e-közigazgatás programja

Az e-közigazgatás a globálisan jelentős szerephez jutó lokalizáció információs kori alapintézménye. Az e-közigazgatás a legkisebb településen is egy olyan nyilvános társadalmi játszma és vállalkozás, amelyben az emberek és érdekeik kifejezésére szervezett csoportjaik elektronikusan együttműködnek az általuk választott önkormányzati képviselőkkel és a helyi közigazgatással, s az együttműködésből született közös döntéseket végrehajtják.

Az eEurope 2002-2005 három vezető prioritása közül az egyik éppen az e-kormányzás, e-közigazgatás. Európában és Magyarországon is a jelen talán legjelentősebb állami s társadalmi feladata az e-közigazgatás bevezetése és elterjesztése. A kormányzati, regionális és helyi megvalósítás egyik legfontosabb, ám csak első feltétele az, hogy mindenki

számára biztosított legyen az intelligens (információs kori) eszközök használata és rendelkezzenek azok használatához szükséges tudással.

Magyarországon ma majdnem minden régiónak és megyének, valamint egyre több kistérségnek, városnak, falunak elfogadták az intelligens térségi-települési stratégiáját, valamint operatív programját, amelyekben szinte kivétel nélkül kiemelt feladat az e-közigazgatás bevezetése és elterjesztése. Ezzel a területfejlesztési tervezés szintjén jórészt lezajlott a közigazgatási paradigmaváltás előkészítése.

Az e-közigazgatás: digitális közigazgatás. De nem(csak) a régi közigazgatás elektronizálása, hanem egy újfajta közigazgatás is. Először lesz valóban szolgáltató közigazgatás, mert az információs kor lehetővé teszi a magasabb szintű és mindenki által elérhető közigazgatási szolgáltatásokat. Az e-közigazgatás négy folyamat és azok minőségi integrálása: a közigazgatási szintek és intézmények belső modernizálása, a helyi társadalmak létezésének és működésének megreformálása, a kormányzati-önkormányzati szintek közötti digitális együttműködés és az e-demokrácia, az e-közigazgatáshoz szükséges minden tudás elérhetővé és használhatóvá tétele.

A közigazgatás tehát a „forró valóság”, a társadalom problémáit nem kezelheti tűzoltásként. Az e-közigazgatás így a forró valóság tényleges és folyamatos hűtése, amely az e-demokrácia megalapozásával és kiépítésével valósítható meg. A közigazgatási rendszereinknek ezért meg kell változniuk, hogy lehetővé váljon a helyi szinten erősödő igény megmutatása – konkrétan az a törekvés, hogy az embereknek nagyobb beleszólásuk legyen a (köz)ügyekbe. Az új közigazgatásnak emiatt az eddiginél jobban segítenie kell a helyi akaratok és szerveződések kialakulását. Az e-közigazgatás tehát nem pusztán az e-ügyintézés bevezetése vagy az elektronikus aláírás alkalmazása, hanem a digitális demokrácia és a szolgáltató e-közigazgatás általánossá tétele.

Az információs és kommunikációtechnológia alkalmazásával a közigazgatás kerüljön közelebb az emberekhez, a társadalom szinte minden tagjához. Az adminisztratív információval kapcsolatos tárgyalási módszereket, valamint az emberek és a közigazgatás közötti párbeszéd módját az új információs- és kommunikációtechnológiák kínálta eszközökkel váltsák fel. Ezáltal a közigazgatás hatékonysága javítható, az állampolgár, az üzleti-, valamint a civil szféra kapcsolata nyitottabbá és együttműködőbbé válik.

A helyi közigazgatási funkciók mindegyike igényli a kommunikációt, az adatszolgáltatást. A funkciókhoz kapcsolódóan a közigazgatás egyben jelentős tartalomszolgáltató is. Az irattárakban levő adatok fokozatosan, módszeresen kerüljenek elektronikus feldolgozásra. Tegyük lehetővé az emberek, a társadalom egyéb szereplői és a közigazgatás közötti elektronikus párbeszédet. Elkerülhető lesz, hogy ugyanazt az adatot ismételtlen megkérdezzék, vagy olyan adatot kérjenek, amit már a korábban megszerzett adatokból a közigazgatás rendelkezésére áll. Ez az önkormányzatok ügyintézésének hatékonyságához vezet és az állampolgárok az üzletei- és civil szféra részére jobb minőségű szolgáltatást nyújtanak.

A szolgáltató közigazgatás feltétele, hogy a szolgáltatásokat az emberek elérjék, s a szolgáltatásokat elektronikusan közvetlenül igénybe vehessék, az ügyeiket sokkal gyorsabban és hatékonyabban elintézhessék. A modernizációs célok felvázolása előtt azonban szükséges áttekinteni a települési önkormányzatok tevékenységét, a feladatkörét érintő legfontosabb szakmai irányokat, egyáltalán a hálózati állam és önkormányzás előtt álló stratégiai tennivalókat.

A nagyobb arányú átalakításhoz szemlélet- és módszerbeli változás, változtatás szükséges, avagy a közigazgatás teljes modernizációjára van szükség. Nem egyetlen masszív

átstruktúrázódásra vagy egyetlen felülről kezdeményezett forradalmi változásra kell gondolnunk, hanem tudatos, decentralizált kísérletek ezreire, amelyek lehetővé teszik számunkra, hogy a politikai döntéshozatal új modelljeit helyi és regionális szinten kipróbáljuk, elindítva ezzel az új típusú demokrácia társadalmi tanulásának folyamatát.

A huszadik században az állam és a társadalom nagyon messze került egymástól. Az emberek sokszor érezhették úgy, hogy az állam nem értük van, sőt a diktatúrák idején ellenük lép fel. A modern demokrácia intézményeinek célja nem lehet más, minthogy az államot és közigazgatását végképpen a társadalom szolgálatába állítsa. Az emberek sokasága csak akkor érzi magát majd állambarátnak, ha - különösen az önkormányzás és a helyi közigazgatás - egyértelműen polgárbarát lesz. Ennek a sokszor meghirdetett fordulatnak a rendszerszerű és egyedül hatékony szisztémája az e-kormányzás és az e-közigazgatás lehet.

A globális információs társadalom kialakulásával párhuzamosan zajló helyi intelligens fejlesztő programok fókuszában a változásokkal együtt járó kockázatok és mellékhatások mérséklése, valamint az új korszak előnyeinek maximalizálása álljon.

A szolgáltató közigazgatás megteremtésének kulcsa a megfelelő integrált informatikai háttér. Nemcsak a meglévő közigazgatást kell gépesíteni és a közigazgatási honlapokat portálokká alakítani, tartalommal feltölteni. Az ügyintézési folyamatokat, a kapcsolatrendszereket kell újragondolni, végrehajtva a szükséges racionalizálásokat, kötelezővé téve az intelligens technológiák alkalmazását.

A közigazgatási információs rendszereket és az adatvagyonot egységesíteni kell, és mindenki számára egyformán garantálni és elérhetővé tenni. A lakosság, a piaci- és civil szervezetek, valamint a közigazgatás közötti kommunikációt tegyük hatékonyvá. Biztosítani kell a nemzetközi információs hálózatokkal való együttműködést, a megfelelő hardver- és szoftverplatformot, amely képes megfelelni az internetes kihívásoknak.

A közigazgatási szolgáltatásokat portál alapú informatikai megoldásokkal kell biztosítani, amely például

- egységesen kezeli az adatvagyonot és a közigazgatási rendszereket,
- a hozzáférés egyszerű és egységes,
- alkalmazásával megvalósul a „mindent egy helyen intézni” elv,
- egyszerű és hatékony információáramlást biztosít,
- interaktív,
- biztosítja a tapasztalatcserét,
- bárki részére, bárhol, bármilyen eszközzel lehetővé teszi a hozzáférést,
- egyszerűsíti a munkafolyamatok menedzselését,
- átlátható feladat- és hatáskörök kialakítását teszi lehetővé.

Ezért e-közigazgatási tájékoztató gyűjtőportálokat alakítanak ki az önkormányzatok különböző szintjein, amelyek egyrészt az adott településről vagy térségről teljes körű információt szolgáltatnak, másrészt az adatok a települési portálok mögött egy egységes szerkezetű adatbázisba kerülnek („gyűjtődnek”) és több lépcsőben elérhetők, elemezhetők. Az e-közigazgatás gyűjtőportál rendszer a régiókban előforduló különböző szintű és nagyságú önkormányzatok számára megoldást képes nyújtani az információs társadalomba való bekapcsolódás elősegítéséhez, mely „egy valós igényeken alapuló, interaktív, integrációs lánc” kialakítását célozza meg a közigazgatás minden területén.

Kiemelt feladat az állampolgárok, a képviselők, a köztisztviselők felkészítése az információs korra, speciális kezdeményezések segítségével az új térségi identitás és életminőség javításának elősegítése. Ennek érdekében minden térségben olyan térségi

informatikai (vezetékes és mobil) hálózat kiépítése szükséges, amely a háztartásokat hozzáférési-kapcsolódási ponttal látja el a telematikus szolgáltatásokhoz.

A vállalkozásokat és professzionális felhasználókat ösztönözni kell az új technológiák hasznosítására a vezetési és a termelési rendszerek hatékonyságának növelése érdekében, valamint az üzleti tevékenység elektronikus szervezésére. Hálózati kapcsolatrendszer épüljön ki a vállalkozások és a K+F tevékenységet folytató intézmények között annak érdekében, hogy a kis- és középvállalkozások elszigeteltsége oldódjon, aktív szerepet játszanak az új technológiákon alapuló fejlesztésekben, ezáltal növelve piaci versenyképességüket.

A közösség fejlesztésének, a társadalmi csoportok támogatásának, a csapatmunka gyakorlatának új technikáját oktató kurzusokat, tanácsadásokat terjesszük el, hogy az emberek elsajátíthassák mindazt a tudást, amely a kezdeményezések elindításához és ahhoz szükséges, hogy ők maguk önszervező csoportokat működtessenek. Szubvenciót és kölcsönt kínálunk a közösségi önszervező, e-közigazgatást segítő kezdeményezéseknek, kezdve a megvalósíthatósági tanulmányok megfogalmazásától annak garantálásáig, hogy a települések és térségek igényeit kielégítő önkormányzati-közigazgatási fejlesztések további átfogó támogatásban részesüljenek.

d. E-aláírás

Az információs társadalom kialakulásához vezető úton mérföldkőnek tekinthető az elektronikus adattovábbítás. Az elektronikus formák térhódításának az előfeltétele a digitális úton történő nyilatkozattétel jogi szabályozása, az elektronikus nyilatkozatokhoz fűződő joghatások állami elismerése. A papír, mint hagyományos adathordozó esetén a nyilatkozatokat az emberi kézírással, aláírással hitelesítik. A számítástechnikai eszközök és az Internet világában szükség van a kézíráshoz fűződő hitelesítési funkció áttünetésére, az aláírás digitalizálására. Erre vonatkozólag az Országgyűlés törvényt is alkotott, amely az elektronikus aláírás jogi szabályozására vonatkozik (2001. évi XXXV. törvény).

Az elektronikus aláírás fogalmát a vonatkozó törvény a következőképpen határozza meg: "az elektronikus dokumentumhoz azonosítás céljából végérvényesen hozzárendelt vagy azzal logikailag összekapcsolt elektronikus adat, illetőleg dokumentum". Egy olyan műszaki, technikai megoldás, amely az egyik, már meglévő elektronikus adathoz egy másik elektronikus adatot kapcsol. Az elektronikus aláírás fogalmán, az elnevezéssel ellentétben, nem egyszerűen az emberi kézírás digitalizált formáját, hanem egy számítógépes adatot, adathalmazt kell érteni. Elektronikus aláírásnak tekinthető például az is, ha valaki az általa írt elektronikus levél végére a saját nevét egyszerűen odaírja (gépele), illetve, ha a saját kézzel írt aláírását elektronikus formában a levélhez csatolja. Ezek a technikai megoldások azonban nem akadályozzák meg az aláírással való visszaéléseket, nem tekinthetők biztonságos eljárásoknak.

Az elektronikus aláírással szembeni alapvető követelmény, hogy hitelesen azonosítsa a dokumentum aláíróját. A használt technikai megoldásoknak biztosítaniuk kell, hogy az aláírás tényét, annak megtörténtét utólag senki ne kérdőjelezhesse meg, ne vonhassa kétségbe, továbbá azt is, hogy az adott aláírás egyértelműen az aláíró személyéhez kapcsolódjon. Elektronikus aláírásként olyan technikai megoldást kell alkalmazni, amely képes megakadályozni a dokumentum tartalmának utólagos megváltozását. A fenti követelményeknek megfelelő elektronikus aláírás digitális jelek sorozatának, egy speciális számsorozatnak fogható fel.

Mielőtt az elektronikus aláírás funkcióit és módszereit ismertetnénk, bemutatjuk az ügyviteli folyamatokban alkalmazott aláírás típusokat.

- láttamozó: az irat kézjeggyel való ellátása. Sem egyetértési, sem pedig döntési hatáskörre nem jogosít, csupán egy olyan jelzés, amely tudtul adja, hogy az ügyintéző elolvasta és tudomásul vette annak tartalmát.
- jóváhagyó: az aláíró egyetért az irat tartalmával, de döntésre nem, csupán véleményezésre jogosít.
- véglegesítő: az aláíró hitelesíti az iratot, amivel jelzi döntése eredményét, hogy egyetért az irat tartalmával. Amennyiben ez az aláírás rá kerül a dokumentumra, a továbbiakban az már nem módosítható.

Ezek után arról is szólnunk kell, milyen fajtái vannak az elektronikus aláírásnak:

- **A fokozott biztonságú elektronikus aláírás** a törvényi megfogalmazás értelmében alkalmas az aláíró azonosítására, és egyedülállóan hozzá köthető. Olyan eszközzel hozták létre, mely kizárólag az aláíró befolyása alatt áll. A fokozott biztonságú elektronikus aláírás úgy kapcsolódik a dokumentum tartalmához, hogy azon minden, az aláírás elhelyezését követően tett módosítás érzékelhetővé, észlehetővé válik. A nyilvános kulcsú eljárással létrehozott aláírás megfelel a fokozott biztonságú elektronikus aláírással szemben támasztott törvényi követelményeknek. A technikai fejlettség jelenlegi szintjén a két kategória egymást lefedi. A törvény indokolása rámutat azonban arra, hogy a jogi fogalomnak technológiától független meghatározást kell tartalmaznia.
- **Minősített elektronikus aláírás** olyan nyilvános kulcsú eljárással készült fokozott biztonságú elektronikus aláírás, amelyet biztonságos aláírás-létrehozó eszközzel hoztak létre, és amelynek a hitelességét minősített tanúsítványt igazolja. A biztonságos aláírás-létrehozó eszközökkel és a tanúsítványt kibocsátó hitelesítési szolgáltatóval szembeni követelményeket külön jogszabály határozza meg. Az elektronikus okirat minősített aláírással történő ellátása a bizonyító erő szempontjából jut jogi jelentőséghez.

i. Az elektronikus aláírás funkciója

A papíron készített iratok hitelességét, az információk valódiságát általában aláírással vagy pecséttel, az esetek többségében a kettő együttes alkalmazásával igazoljuk. Már a kezdetektől fogva jogszabályok írták, illetve írják elő, hogy milyen esetben tekinthető egy aláírás hitelesnek. A legfontosabb szempontok, (pl. magánokiratok esetén) hogy az illető személy saját maga és önként írja alá az iratot (pl. egy adásvételi szerződésnél praktikus dolog két tanúval aláíratni a dokumentumot), de itt sem maradhat el az eladó és a vevő sajátkezü aláírása. Ezek az aláírások az illető személy illetve személyek biztonságát szolgálják. Abban az esetben, ha nem magánokiratokról van szó, hanem közokiratokról, még szigorúbb előírások vonatkoznak az aláírásra (pl. cégbejegyzésnél a cégbíróságon, a cég vezetőjének aláírási címpéldányt kell készíteni közjegyző előtt, amelyet saját kézjeggyel lát el). Ez az illető cég biztonságát szolgálja.

Mindezek mellett a technika fejlődése szükségessé tette, hogy ne csak papír alapú iratokban, hanem elektronikus iratokban is gondolkodjunk, ami egy újabb problémát vet fel, mégpedig azt, hogyan lehet egy elektronikus iratot elektronikus továbbítása esetén úgy aláírni az illetékes személynek, hogy az hiteles legyen. A probléma megoldására született meg az elektronikus aláírás funkciója.

Az elektronikus aláírás funkciói között három nagyon sarkalatos pontot kell megemlíteni:

- titkosság: csak a címzett legyen képes elolvasni az elektronikus aláírással ellátott iratot
- hitelesség: a címzett egyértelműen azonosítani tudja az aláíró
- sértetlenség: az aláírással ellátott irat tartalma változatlan maradjon

ii. Az elektronikus aláírás módszerei

Az irodai rendszereken belül kialakított elektronikus aláírás lehetővé teszi, hogy az ügyek előrehaladását nyomonkövessék, bár itt meg kell említenünk, hogy zárt irodai rendszerekben az elektronikus aláírás helyett elegendő a felhasználók azonosítása felhasználói névvel és jelszóval, mert utána már a rendszer naplózza az egyes személyek tevékenységét. Az iratokra pedig a kézjegyet helyett a felhasználói adatok kerülnek rá. Természetesen, amikor ezek az iratok kikerülnek a zárt irodai rendszerből, azaz az elektronikus folyamatból (pl. postázzák az ügyfél számára), akkor a hagyományos ügymenetnek megfelelően folytatják útjukat. Nézzük meg ezek után, hogy az elektronikus aláírásnak milyen módszerei léteznek, és melyek ezek:

kulcsokat alkalmazó aláírás

Az általánosan megfogalmazott elvárások, követelmények szempontjából a jelenleg ismert és használt technikai megoldások közül az úgynevezett nyilvános kulcsú eljárásokkal létrehozott elektronikus aláírás tekinthető világszerte elfogadottnak. A nyilvános kulcsú eljárás során két kulcsot, egy nyilvános kulcsot (kriptográfiai nyilvános kulcs) és egy titkos kulcsot (kriptográfiai magánkulcs) kell használni. Az aláírókulcs segítségével elhelyezett elektronikus aláírás bonyolult matematikai és kriptográfiai megoldások, műveletek összessége. Mindkét kulcs digitális jelek sorozatának fogható fel, amelyeket sajátos programokkal kell kezelni. A titkos kulccsal az aláíró képes az elektronikus iraton egy kizárólag rá jellemző aláírást létrehozni, illetve adatokat titkosítani. A titkos kulcshoz tartozó nyilvános kulcs segítségével a címzett pedig ellenőrizheti az elhelyezett elektronikus aláírást, sőt képes a titkos kulcs tulajdonosa számára adatokat titkosítani. A titkosításhoz és annak feloldásához eltérő - aszimmetrikus - algoritmusokat használnak, így lehetetlen a titkosított üzenetet ugyanazzal a kulccsal megfejteni és fordítva.

A nyilvános kulcsú elektronikus aláírás alapvető tulajdonságai:

- az adott elektronikus aláírás kizárólag egy aláíró személyéhez kapcsolható,
- az egyedileg azonosítja az aláíró, így a címzett ellenőrizheti a feladó személyazonosságát,
- az aláírás ténye kétséget kizáróan bizonyítható, azaz az üzenet küldője utólag nem hivatkozhat arra, hogy azt nem is írta alá,
- egyértelműen kimutatja, ha az adott dokumentum az aláírást követően megváltozott,
- bizonyos feltételek mellett az aláírás időpontja is hitelesen rögzíthető.

A nyilvános kulcs alapján, abból gyakorlatilag lehetetlen a titkos kulcsot megfejteni, így nincs lehetőség az elektronikus aláírás hamisítására sem. A nyilvános kulcs birtokában kétséget kizáróan megállapítható, hogy a vizsgált aláírás a hozzá tartozó titkos kulcs segítségével készült-e vagy sem.

Az elektronikus aláíráshoz használatos kulcsok az aláíró személyétől fizikailag elkülönülten, fájlok, floppyk, esetleg chipkártyák formájában jelenik meg. A nyilvános kulcs bárki által megismerhető, ezáltal határozható meg a kulcs tulajdonosának a személyazonossága. A titkos kulcsot - akár csak a különféle PIN-kódokat - természetesen titokban kell tartani.

A címzett a dokumentum alapján egy szoftver segítségével újra elkészíti annak a digitális lenyomatát. Ezt követően a digitális aláírást a nyilvános kulcs segítségével dekódolja. A dekódolás folytán megkapja az aláíró által készített digitális lenyomatot. Amennyiben a két digitális lenyomat azonos, egymással megegyező, megállapítható, hogy a dokumentum az aláírása óta nem változott, illetve, hogy a címzett által használt nyilvános kulcshoz tartozó titkos kulccsal készült az elektronikus aláírás.

kártyát igénylő aláírás (chipkártya + PIN-kód)

Optikai, mágneses vagy chipkártyával azonosítják az aláírót. A biztonság növelése érdekében csakúgy, mint a hagyományos bankkártyák esetén PIN-kódot alkalmaznak. Meg kell említenünk, hogy nincs igazán kialakult szabvány e kártyákra, ezenkívül mindenkit el kell látni kártyával (ennek költsége van), és leolvasásához külön kártyaolvasó szükséges.

digitalizált aláírás (biometria + kriptográfia)

Ez a technológia a biometria és a kriptográfia összekapcsolásán alapszik. A digitális táblán rögzített aláírás jellegzetességeit vizsgálja, ez a biometria része, és az illető személyhez kapcsolva tárolja. Ezt követően kriptográfiai módszerekkel ezeket a jellegzetességeket hozzárendeli az irathoz.

Legnagyobb előnye, hogy jogszerű (már akkor is az volt, amikor még az elektronikus aláírás többi módszere nem volt az), mivel a kézzel történő aláíráshoz köti a hitelességet. Az aláírás ellenőrzése a biometria jellegzetességeket hasonlítja össze az előzőleg rögzített aláírásmintákkal és a hasonlóságot százalékos értéként adja meg. Hátrányaként talán azt említhetjük meg, hogy az aláírás rögzítéséhez digitalizáló tábla szükséges.

iii. Elektronikus aláírással kapcsolatos szolgáltatások

Az elektronikus aláírással kapcsolatos szolgáltatásokat külön-külön vagy azok közül többet együttesen is lehet nyújtani. A hitelesítési szolgáltató a hitelesítési szolgáltatás részeként köteles valamennyi alábbiakban ismertetett szolgáltatást ellátni. Valamennyi szolgáltatás lehet fokozott biztonságú vagy minősített szolgáltatás.

iv. Aláírás-létrehozó adat elhelyezése

Az aláírás-létrehozó eszköz egy olyan hardver vagy szoftver, amelynek segítségével, az aláírás-létrehozó adat (jellemzően kriptográfiai magánkulcs) felhasználásával az elektronikus aláírást létrehozzák. Az aláírás-létrehozó eszközön az aláírás-létrehozó adat elhelyezése gyakorlatilag az elektronikus aláírás lehetőségének a biztosítását, a hardver vagy a szoftver és a titkos kulcs rendelkezésre bocsátását jelenti. Az eszköz vagy program és a titkos kulcs birtokában lesz képes a felhasználó az adott dokumentumot elektronikusan aláírni. A titkos kulcshoz tartozó kriptográfiai nyilvános kulcsot, mint aláírás-ellenőrző adatot ugyancsak a szolgáltató állapítja meg. A magánkulcsot minden esetben nyilvánosságra kell hozni, illetve kérelemre a címzett rendelkezésére kell bocsátani, ennek hiányában nem lehet ugyanis a titkos kulccsal kódolt üzenetet elolvasni.

v. Időbélyegzés

Az időbélyegzés során a szolgáltató az elektronikus dokumentumhoz időbélyegzőt kapcsol. Az időbélyegző tanúsítja, hogy az adott időpontban mi volt a lebélyegzett dokumentum tartalma. Az időbélyegzés tulajdonképpen az időpont bizonyítására is alkalmas speciális elektronikus aláírásnak tekintendő, amely ennek következtében lehet egyszerű vagy minősített időbélyegzés is.

vi. Hitelesítési szolgáltatás – tanúsítvány

A hitelesítési szolgáltatás keretében a hitelesítési szolgáltató

- azonosítja a hitelesítési szolgáltatást igénylő (avagy aláíró) személy adatait,
- tanúsítványt bocsát ki,
- nyilvántartásokat vezet,
- fogadja a tanúsítványokkal kapcsolatos változások adatait,
- nyilvánosságra hozza a tanúsítványhoz tartozó szabályzatokat,
- az aláírás-ellenőrző adatokat (kriptográfiai nyilvános kulcsokat),
- a tanúsítvány aktuális állapotára (különösen esetleges visszavonására) vonatkozó információkat.

A hitelesítési szolgáltatás is lehet minősített vagy nem minősített elektronikus aláírás-hitelesítési szolgáltatás. A minősített hitelesítési szolgáltató jogosult nem minősített tanúsítványt is kibocsátani.

A hitelesítés-szolgáltató a tanúsítvány kibocsátását megelőzően azonosítja az igénylő (későbbi aláíró) személyét, majd a saját elektronikus aláírásával aláírt tanúsítvánnyal hitelesíti az igénylő elektronikus aláírását. A tanúsítvány tartalmazza az aláíró titkos kulcsához tartozó nyilvános kulcsot, az aláíró azonosító adatait, a két kulcs összetartozását és érvényességét. A tanúsítvány alapján győződhet meg a címzett arról, hogy az elektronikus aláírás magától az aláírótól származik.

A tanúsítványt a feladó hozzácsolja az általa aláírt - saját titkos kulcsával titkosított - dokumentumhoz. Az aláíró aláírása a saját nyilvános kulcsával, a tanúsítványon szereplő aláírás pedig a hitelesítés-szolgáltató nyilvános kulcsával ellenőrizhető.

A hitelesítési szolgáltatási tevékenység lehet fokozott biztonságú vagy minősített szolgáltatás, ennek megfelelően a szolgáltató által kibocsátott tanúsítvány is lehet nem minősített, avagy minősített tanúsítvány.

Minősített elektronikus aláíráshoz azonban minősített tanúsítványra van szükség. A minősített tanúsítvány kibocsátására kizárólag a minősített szolgáltatók jogosultak.

A minősített tanúsítvány tartalmazza:

- annak megjelölését, hogy a tanúsítvány minősített tanúsítvány,
- a hitelesítés-szolgáltató és székhelyének (ország-) azonosítóját,
- az aláíró nevét vagy egy becenevét, ennek jelzésével,
- az aláírónak külön jogszabályban, a szolgáltatási szabályzatban, illetőleg az általános szerződési feltételekben meghatározott speciális jellemzőit, a tanúsítvány szándékolt felhasználásától függően,
- azt az aláírás-ellenőrző adatot (nyilvános kulcsot), amely az aláíró által birtokolt aláírást készítő adatnak (magánkulcs) felel meg,
- a tanúsítvány érvényességi idejének kezdetét és végét,
- a tanúsítvány azonosító kódját,
- az adott minősített tanúsítványt kibocsátó hitelesítés-szolgáltató fokozott biztonságú elektronikus aláírását,
- a tanúsítvány használhatósági körére vonatkozó esetleges korlátozásokat,
- a tanúsítvány felhasználásának korlátait,
- más személy (szervezet) képviselőjére jogosító elektronikus aláírás tanúsítványa esetén a tanúsítvány ezen minőségét és a képviselt személy (szervezet) adatait.

A minősített hitelesítés-szolgáltatónak lehetősége van alacsonyabb biztonsági fokú nem minősített tanúsítvány kibocsátására, továbbá arra is, hogy különböző tanúsítványtípusokat állítson ki. Meghatározhatja a tanúsítvány felhasználásának tárgybeli, földrajzi vagy egyéb korlátait, illetve az egy alkalommal vállalható kötelezettség legmagasabb értékét.

A tanúsítvány kibocsátható olyan céllal is, hogy az aláíró más személy (szervezet) képviseletében történő aláírásra jogosítsa fel, de ebben az esetben a tanúsítvány kibocsátását megelőzően a képviseleti jogosultságot igazolni kell. A képviseleti jogosultság meglétét a hitelesítés-szolgáltató köteles ellenőrizni. A tanúsítvány kibocsátásról a képviselt személyt (szervezetet) haladéktalanul tájékoztatni kell.

Ennek a folyamatnak részeként komoly szerepet szánunk az egyetemeknek és a főiskoláknak, mint regionális információs központoknak. Ennek több előnye ígérkezik. Nem kell új szervezetet létrehozni, ezért költségtakarékosabb. Könnyebben alakulhat ki bizalmi viszony az adatot gyűjtő és az adatot szolgáltató között. A gyűjtött és elsődlegesen feldolgozott adatbázis segítené a gyakorlati oktatást, az egyetemi kutatást, különösképpen pedig a felsőoktatásra is épülő szaktanácsadást.

e. Közigazgatási ügyfél-tájékoztató rendszer

A társadalmi rendszerváltás és az információs társadalom kihívásai egyaránt a közigazgatási tevékenység új alapra helyezését indokolják. Az Európai Unió elvárásainak és törekvéseknek megfelelő ügymenet jellegű rendszerfejlesztés adhat választ ezekre a kihívásokra.

A rendszer a teljes magyar joganyag felülvizsgálatával kialakított hatásköri rendszerre épül, ez feladatkörökbe szervezve kétszintű kulcsszó-rendszerrel tartalmazza az önkormányzatok és dekoncentrált szervezetek hatásköreit havi követéssel. Megjelöli az I. II. fokon, illetve felülvizsgálati ágban eljáró szervezeteket, szétválasztja az államigazgatási (hatósági) feladatköröket az önkormányzatiaktól, lehetőséget biztosít átruházott hatáskörök esetén az eredeti és aktuális jogosított megkülönböztetésére. Tartalmazza a hatáskör szövegét és az ezt előíró vagy lehetővé tevő jogszabályt. A rendszerből elkülönülnek nagyságrend szerint az önkormányzatok, tehát főváros, megye, megyei jogú város, község. A hatáskörök szervezeti egységekhez, bizottságokhoz kapcsolhatók, támogatva ezzel a szervezeti Működési Szabályzat naprakész vezetését. Ugyancsak lehetőség van a hatáskörök személyhez kötésére, amely biztosítja, a munkaköri leírások, ügyfél-tájékoztatás hatékony megoldását. A kötelező hatáskörök mellett önként vállalt feladatok kezelésére is alkalmas.

Ügymenet modellek

A feladat- és hatáskörök kiegészülnek az eljárások indításához szükséges dokumentumok (formanyomtatványok, tájékoztatók) részletes leírásával. A rendszer tartalmazza a jogi háttér ismertetését, kiemelve az Államigazgatási Eljárás közös elemeit. Első fázisban ezek az információk az állampolgárok korszerű tájékoztatását teszik lehetővé papíron, ügyfélszolgálaton vagy interneten keresztül.

Tudásbázis jelleggel magába foglalja a közigazgatási eljárások teljes ügyintézési folyamatának leírását. Ez alapját képezi az EU-ban is követendőnek tartott Workflow alapú fejlesztésnek. A kistélepülések számára, ahol nehezen biztosítható minden területen magasan képzett szakmai háttér, lehetőséget biztosít a jogszerű, gyors, pontos, olcsó, polgárbarát ügyintézésre. A nagytelepüléseken a munkaszervezést teszi könnyebbé, segítve a nagyszámú ügyintézési folyamatot. A tájékoztató funkcionálisan túl a hivatal zárt rendszerében a tájékoztató formanyomtatványokat beviteli képernyőként kezelve alapjául szolgál egy tudás alapú rendszer kialakításának.

A rendszer tartalmazza továbbá a folyamatok lezárásául szolgáló határozat-ügyiratminták részletes leírását, valamint a döntést befolyásoló Alkotmánybírósági, Legfelsőbb Bírósági határozatokat, ugyancsak tartalmaz információkat az elérhető szakirodalomról.

A rendszerfejlesztés célja első fázisban korszerű tájékoztatás és szakszerű ügyintézés segítése. Második fázisban a hivatalban működő Workflow és tudásbázis alapú tudásbázisú ügyintézői rendszer kifejlesztése. Harmadik fázisban az elektronikus aláírás EU szabványjogi megvalósítását követően a nap 24 órájában működő elektronikus ügyintézés megvalósítása, ahol az eljárás indítása az ügyfelek (állampolgárok, szervezetek) otthonukból, telephelyeikről kezdeményezhetik elektronikusan ügyeik intézését. Természetesen ehhez szükséges az Államigazgatási Eljárás elektronikus hitelesítése is.

f. Agrár szakigazgatási intézmények

Az agrárszakigazgatási intézmények köréhez tartoznak: az állategészségügyi- és élelmiszerellenőrzési intézmények, a növényegészségügyi és talajvédelmi szolgálatok, a megyei (fővárosi) földművelésügyi hivatalok, az erdészeti szakigazgatás, a Nemzeti Földalapkezelő Szervezet és a Magyar Élelmiszer-biztonsági Hivatal.

A növényegészségügyi szakigazgatás 20, az állategészségügyi szakigazgatás 25, a földművelésügyi szakigazgatás 19 részben önálló intézményből áll.

Az állategészségügyi szakigazgatási intézményhálózathoz összesen 25, részben önállóan gazdálkodó intézmény tartozik, ezen belül 19 megyei és a Budapesti-Fővárosi Állategészségügyi és Élelmiszer-ellenőrző Állomás működik. Ezek az állomások hatósági állategészségügyi és szolgáltatási tevékenységet végeznek. A hálózatban további két diagnosztikai szolgáltatási tevékenységet végző Állategészségügyi Intézet, az országos hatáskörű Élelmiszervizsgáló Intézet, továbbá az Oltóanyag- Gyógyszer- és Takarmány-ellenőrző Intézet tartozik.

A fővárosi és a megyei állomások hatósági és szolgáltató, az állategészségügyi intézetek kizárólag hatósági szolgáltató tevékenységet látnak el. Ennek keretén belül kiemelt jelentőségű a veszélyes állatbetegségek leküzdésével kapcsolatos feladat, az élelmiszer- és takarmányellenőrzési hatósági munka, az országhatáron folyó állategészségügyi ellenőrző és vizsgáló tevékenység, az állategészségügyi intézetek diagnosztizáló és betegséget megállapító tevékenysége. 2004. évben megalakultak az EU-konform állategészségügyi határállomások, amelyeknek fontos feladatai közé tartozik az állategészségügy, az állatvédelem és a takarmányellenőrzés. Az intézmények elvégzik az uniós és hazai rendeletekben előírt, az egyes állatbetegségek felderítésére irányuló monitoring vizsgálatokat, a betegségmentesítési programokat. Az állatvédelmi előírások betartásának ellenőrzéséről, a hatósági ellenőrzések számáról, a talált hiányosságok százalékaról, a megtett intézkedések számáról a szakmai főosztálynak az EU illetékes bizottságát rendszeresen tájékoztatnia kell.

A növény-egészségügyi szakigazgatási intézmények hálózatát 19 részben önállóan gazdálkodó megyei növény- és talajvédelmi szolgálat, illetve az országos hatáskörű középírányító szerv, a Növény- és Talajvédelmi Központi Szolgálat alkotja. A növény- és talajvédelmi szolgálat a növényi károsítók fellépésének megelőzését, szükség esetén a károsítók elpusztítását, a növény-védőszerek, terméshozó, talajjavító anyagok ellenőrzését, a növény-egészségügyi határállomások működtetését végzik. További kiemelt feladataik:

- növényvédő szerek és termésnövelő anyagok forgalomba hozatalának és felhasználásának engedélyezése, az engedélyezés minőségbiztosítási rendszerének fejlesztése ill. monitoring vizsgálata;
- termelők nyilvántartása, növényegészségügyi útlevel, termőhelyi ellenőrzés;
- területi növényegészségügyi diagnosztikai laboratóriumok fejlesztése;
- központi Károsító Diagnosztikai Laboratórium létrehozása;
- szennyvizek és szennyvíziszapok mezőgazdasági felhasználásának és kezelésének szabályozása;
- vizek mezőgazdasági eredetű nitrát szennyezéssel szembeni védelme;

A földművelésügyi szakigazgatási intézmények feladatait 19 megyei (fővárosi) földművelésügyi hivatal látja el részben önállóan gazdálkodó intézményként. A földművelésügyi intézményhálózat szolgáltatási és hatósági, államigazgatási tevékenységet egyaránt végez. Kiemelt feladataik:

- a nemzeti agrárpolitika képviselte és érvényre juttatása, valamint a különböző szintű területekre vonatkozó stratégiák, koncepciók, programok és tervek kidolgozása és egyeztetése;
- nemzeti támogatási rendszer működtetésével összefüggő feladatok ellátása;
- elemi károkkal és azok biztosításával kapcsolatos állami támogatások működtetése;
- hegyközségek törvényességi felügyelete, halászati, vadászati igazgatási feladatok;
- mezőgazdasági vízgazdálkodással összefüggő feladatok.

Az erdészeti szakigazgatási intézmények körét az Állami Erdészeti Szolgálat (AESZ) és a hozzá kapcsolódó részjogkörű költségvetési egység (igazgatóság) tartozik. Az AESZ az Erdőtörvényben és a külön jogszabályokban meghatározott erdészeti igazgatással összefüggő irányítói, szervezési és hatósági alapfeladatokat, valamint szakágazati statisztikai-, erdészeti ismeretterjesztési feladatokat látja el. Működteti továbbá az Országos Erdőállomány-adattárat és az Erdővédelmi Mérő- és Megfigyelő Rendszert.

A földügyi szakigazgatási tevékenységet a földhivatalok (20 megyei intézmény és a 116 területi szervként működő körzeti földhivatal), valamint a Földmérési és Távérzékelési Intézet (FÖMI) látja el. Az intézmények feladata az ingatlan-nyilvántartás vezetése, a földtulajdoni és földhasználói előírások érvényesülésének ellenőrzése, továbbá ellátják a polgári célú állami földmérési, térképészeti és távérzékelési feladatokat, és közreműködnek a Nemzeti Kataszteri Program végrehajtásában. A Földmérési és Távérzékelési Intézet hatósági jogkörében végzi az államhatárral kapcsolatos tárgyevre meghatározott földmérési és nyilvántartási feladatokat. Teljesíti a központi adat- és térképtárral összefüggő, a Földrajzinév-tár vezetésére vonatkozó-, továbbá az alappont-hálózatokat és az állami topográfiai térképeket érintő, jogszabályban előírt feladatokat. Feladata a TAKARNET program továbbfejlesztése, továbbá a MePAR rendszer működtetése.

A Mezőgazdasági minősítés és törzstenyészet-fenntartás intézményei közé tartozik az Országos Mezőgazdasági Minősítő Intézet, az Országos Borminősítő Intézet, az Agrobotanikai Intézet és az Állami Ménesgazdaság. Az intézmények feladata a termelésben alkalmazható állat- és növényfajták elismerése és minősítése, a fajtafenntartásról, a géntartalékok képzéséről és fenntartásáról való gondoskodás, a vetőmagvak és szaporítóanyagok ellenőrzése, az állatfajok és fajták tenyésztésének irányítása, tenyésztésszervezés, a borgazdasági termékek előállításával és forgalomba hozatalával kapcsolatos feladatok. A minősítő intézetek is szerepet kaptak az Integrált Igazgatási és Ellenőrzési Rendszer működtetésében.

A Mezőgazdasági és Vidékfejlesztési Hivatal 81/2003. (VI. 7.) Korm. rendelet, alapján az Agrárintervenciós Központ és SAPARD Hivatal jogutódjaként jött létre 2003.

július 1-jén. Az MVH ellátja az egyes közösségi termékpiaca rendtartások szabályozási eszközeinek működtetésével kapcsolatos feladatokat. Az MVH az Európai Mezőgazdasági Orientációs és Garancia Alap (a továbbiakban: EMOGA) Garancia Részlege tekintetében ellátja:

- az Európai Unió (a továbbiakban: EU) belpiaci támogatásainak kezelésével;
- az EU külpiaci támogatásainak kezelésével;
- az intervenciós rendszer működtetésével;
- a közvetlen kifizetések kezelésével és ehhez kapcsolódóan az Integrált Igazgatási és Ellenőrzési Rendszer (a továbbiakban: IIER) fejlesztésével és működtetésével, a vidékfejlesztési (kísérő) intézkedések bonyolításával – a vidékfejlesztési intézkedések bonyolításával összefüggő - a Halászati Orientációs Pénzügyi Eszközből (a továbbiakban: HOPE) folyósított támogatások végrehajtásával összefüggő feladatokat.

g. e-Agrárium, e-szakigazgatás

Az Unió legrészletesebben szabályozott területe a mezőgazdasági termelés, valamint az agrár- és vidékfejlesztéshez kapcsolódó támogatások igénybevételének lehetősége. A vidékfejlesztési támogatások elérésére és hatékony felhasználására projektek bonyolítására képes, megbízható belső nyilvántartást vezető, stratégiával és üzleti tervvel rendelkező gazdaságok aspirálhatnak sikerrel. Meghatározó pozícióba kerülnek a gazdaságszervezést végző nem kormányzati szervezetek, kamarák, amelyek a gazdasági érdekek mentén létrejövő bizalommal rendelkeznek.

Az agrárvállalkozók és agrárvállalkozások versenyképes piaci szereplésük biztosítása érdekében, az általános vállalkozási ismeretek és információk mellett több piac, piacsabályozási rendszer, minőségi előírás napi ismeretével, valamint a versenyképes termékek előállításához szükséges korszerű technológiai ismeretekkel kell rendelkezniük – többek között – a következő területeken:

- o hazai (regionális) és EU agrár-, vidékfejlesztési- és környezetvédelmi szabályozás
- o EU és fennmaradó nemzeti agrár-, vidékfejlesztési támogatási rendszer
- o WTO agrárszabályozás
- o hazai (helyi, regionális) EU és világ agrárpiaci információk (trendek, árak, hírek, minőségi elvárások)
- o input termékek piaca (trendek, árak, hírek)
- o szolgáltatási, finanszírozási és föld piaci (trendek, árak, hírek)
- o termeléshez szükséges környezeti információk (időjárás, növényvédelem, állat- és növényegészségügy)
- o technológiai ismeretek (alkalmazott kutatási eredmények)

Az e-agrárium jövőképe kiinduló pontja, és a versenyképes mezőgazdasági termelés egyik alapfeltétele a fenti információkhoz való minél szélesebb körű elektronikus hozzáférés biztosítása, az agribusiness szereplői, de kiemelten a mezőgazdasági termelők részére. Az agribusiness szereplőinek kiszolgáló állami- és köztestületek, civil szervezetek és vállalkozások által üzemeltetett integrált információs rendszereken túl szükséges:

- o az e-közösségek kialakulását lehetővé tevő kommunikációs, e-commerce és e-közigazgatási szolgáltatások elindulása és térnyerése
- o Az informatikai technológián alapuló információ források és szolgáltatások elérését lehetővé tevő egyéni és közösségi Internet elérések penetrációjának növelése

- o a gazdaságok belső nyilvántartását, ügyvitelét, menedzsment döntéseit és technológiai tevékenységét támogató szoftverek elterjedése
- o az informatikai eszközök, szolgáltatások és lehetőségek igénybevételéhez és hasznosításához kompetens humánerőforrás.

A hazai mezőgazdasági termelőket több, a versenyképességüket hátrányosan érintő gazdasági és közigazgatási hatás érinti. Az informatikai fejlesztések az információk biztosításával, az elektronikus szolgáltatások, ügyintézés és az e-business fejlesztésével hozzájárulhat az agribusiness versenyképességének fokozásához.

A közigazgatási informatikai fejlesztéseken túl ki kell építeni az e rendszerek adataira épülő közhasznú információkat szolgáltató rendszereket. A rendszerek kiépítésénél figyelembe kell venni, hogy a hazai kormányzati és köztisztviselői agrárirányításnak az Unión belüli politizálásához az EU elvárásain túl is szüksége van gazdasági információkra, valamint, hogy a hazai termelők is birtokolhassák az Unió versenyhátrányainak információit. A közigazgatási informatikai fejlesztéseknél, a mezőgazdasági termelők számára jelentős előnyt jelent, ha az agrár portálrendszerek tartalmán belül a gazdasági döntéseiket érintő, más ágazatok szabályozási és közhasznú információi is megjelennek. Az Unió várható reformjai alapján a kapcsolódó ágazatok között a vidékfejlesztés, agrárkörnyezet gazdálkodás és a környezetvédelem fontossága kiemelkedő.

Az élelmiszer piacok globális fejlődési trendjeinek megfelelően, az egészséges táplálkozás, egészséges élelmiszerek biztosítása érdekében fejleszteni kell a technológiai színvonal növeléséhez szükséges tudományos és szakmai adatbázisokat, elérhetőségüket. Biztosítani kell, hogy a technológiai információkhoz a termelők és a szaktanácsadók is a napi munkájuk kapcsán hozzáférjenek. A magas minőségű termékeket előállító termelés technológiájának betartásához elengedhetetlen a környezeti információkhoz való hozzáférés. A környezeti információk közül az agrármeteorológia és az állat-egészségügyi és növényvédelmi előrejelzések a legfontosabbak.

A korszerű agrárinformációs rendszer kiépítésének alapvetően három fő célja lehet:

- o Naprakész információkkal segítse a vállalkozások működését és fejlődését.
- o Szolgáltassa ki az agrárkormányzat információszükségletét, a döntéshozatali mechanizmus aktuális információk iránti igényét.
- o Készüljön föl és a csatlakozást követően tegyen eleget az EU nyilvántartási és információszolgáltatási követelményeinek.

A piacgazdaság működésének egyik legfontosabb feltétele a magán- és társas vállalkozások sikere. Az agrárinformációs rendszer egyik célja ezért az, hogy segítse a vállalkozások eredményes működését és fejlődését a működéshez és a döntéshozatalhoz szükséges folyamatosan frissített, naprakész, aktuális információk biztosításával. Ezek alatt elsősorban a gazdasági szabályozás, támogatások, kormányzati-, műszaki és kereskedelmi-, szaktanácsadói-, a gazdálkodás eredményességére vonatkozó összehasonlító információk, valamint a gazdálkodók szakmai felkészültségének fejlesztésére vonatkozó információk értendők.

Az EU tagságtól függetlenül egyre sürgetőbb az agrárkormányzat döntéseit támogató új információs rendszer kialakítása szükségessége, mivel az elmúlt időszakban felmerült új követelményeknek az információszolgáltatás jelenleg nem tud minden elemében megfelelni. Kormányzati célú információs rendszereinket szakmai, tartalmi, adatgyűjtési kör és fegyelem, valamint a nyújtott szolgáltatások és az azokhoz való hozzáférhetőség tekintetében is alakítani szükséges.

A magyar agrárgazdaságnak az Európai Unióhoz történő integrálódást követően fontos az európai követelményekkel összhangban álló nyilvántartási rendszer és információs hálózat üzemeltetése. Az EU agrárgazdaságának irányítása hatalmas mennyiségű pontosan egyeztetett előírásoknak megfelelő információ szabályozott áramlásán alapszik. Az információs csatornák kölcsönösen összekötik a tagországokat a szervezet döntéshozó központjaival. Az ezen információk alapján hozott döntések komoly előnyöket, illetve súlyos hátrányokat jelenthetnek az érintett országoknak, így az adatok hitelességével, megbízhatóságával és összehasonlíthatóságával kapcsolatos követelmények betartása nem csak nagyon szigorú követelmény, hanem egyben elemi érdeke is a tagoknak.

Az ágazatirányítás fontosabb feladatai és informatikai eszközei röviden a következők. A feladatok egyaránt jelentenek horizontális (pl. állat- és növényegészségügy, élelmiszerjog), strukturális (pl. kedvezőtlen adottságú területek, beruházások támogatása) és piacsabályozási (termékpálya-szabályozások, intervenció, export-visszatérítés) feladatokat. Ezekben belül mindenhol megjelenik az intézményfejlesztés és a gazdaságfejlesztés információs igénye. Az előbbi jelenti a már meglévő, működő szervezeteink fejlesztését, új feladatokra történő felkészítést, de sok esetben szükséges új hivatalok kialakítása is. **Az intézményfejlesztés többek között a felszerelésben, a technika bővítésében és megújításában is jelentkezik. Szinte mindenhol jelentkezik az adott feladat ellátásához szükséges informatikai támogatás kialakításának, illetve fejlesztésének szükségessége.** Így például:

- o **Az állategészségügyi** ellenőrzési rendszer átvétele szükségessé teszi az állategészségügyi informatikai rendszer további fejlesztését az EU különböző informatikai és adattovábbító rendszereihez (ANIMO, ADNS, SHIFT) való csatlakozás elősegítése érdekében. Ugyancsak feladat a szarvasmarha egyedi jelölési és nyilvántartási rendszer kiteljesítése és kiterjesztése más gazdasági haszonállatokra, járványtani adatbázis és monitoring rendszer kiépítése. Az állatgyógyászati készítmények ellenőrzése, törzskönyvezése, regisztrációja megköveteli egy gyógyszerügyi adatbázis felépítését, karbantartását is, melyen keresztül megvalósul a csatlakozás az EU gyógyszerügyi telematikai hálózatához.
- o **A növényegészségügyi** szolgálat fejlesztésének feltétele az EU belső piacán alkalmazott hatályos vizsgálati módszerek, szabványok és ellenőrzési rendszer, illetve az előírt nyilvántartási, információs és jelentési rendszer bevezetése. Határkirendeltségek fejlesztése, az országos növényvédelmi intézményrendszert és a növényegészségügyi határkirendeltségeket integráló (*LAN-WAN*) hálózat kiépítése, korszerűsítése, a növényegészségügyi és talajvédelmi szakigazgatás informatikai hálózatának fejlesztése, a hatósági feladatellátás és tervezés statisztikai rendszereinek felállítása magas fokú adatbázis védelmi feltételek alkalmazásával (tűzfal-szoftverek).
- o **Az élelmiszer-ellenőrzésre és minőségbiztosításra** szolgáló különleges EU-konform laboratóriumok és adatbankok kialakításához, működtetéséhez és összekapcsolásához szükséges infrastrukturális (informatikai) fejlesztéseket kell végrehajtani. A zöldség-gyümölcs minőségellenőrzési rendszer továbbfejlesztése érdekében létre kell hozni egy információs adatbank-rendszert.
- o Kiemelt fontosságú terület a **statisztikai és piaci információs rendszer**, amely a mezőgazdasági termelés és értékesítés üzemgazdasági és makrogazdasági adatainak olyan nyilvántartását jelenti, amely a kellő részletességű adatok szolgáltatására képes úgy az ágazatirányítás, mint az EK Bizottság és az Eurostat részére, s emellett biztosítja a termelők és értékesítők megfelelő informálását is. A korszerű, az EU-val is harmonizáló agrárinformációs rendszer és intézményi feltételeinek kialakítása szempontjából kiemelt jelentőségű terület.

- o 1998-ban elkezdődött az Integrált Igazgatási és Ellenőrzési Rendszer (*IIER/IACS*) bevezetéséből a **földhivatali hálózatra** háruló új szolgáltatások kifejlesztése (TAKAROS, TAKARNET, META)
- o Az **Agrár-környezetvédelmi Hálózat** már meglévő intézményekre (pl. Talajvédelmi Állomások hálózata) épül, azonban a Hálózat kialakításához jelentős műszaki-, technikai és személyi fejlesztésekre van szükség. Ez elsősorban digitális, térképi adatbázisrendszer, szoftver- és hardverhálózat, térinformatikai eszközök, stb. kialakítását teszi szükségessé. Már meglévő tevékenységek (pl. távérzékelés) további koordinálása és fejlesztése is szükséges.
- o **Országos szőlőkataszter létrehozása**, mely az EU borpiaci szabályozásának végrehajthatóságához alapvető fontosságú. A kataszteri nyilvántartás létrehozásához országos szőlészeti statisztikai felvételezést tervezünk.

Ellenőrző kérdések:

1. Melyek az önkormányzatokkal szembeni elvárások, amelyek fokozottan igénylik az elektronikus ügyintézés megvalósítását?
2. Milyen alrendszerek, funkciók alakíthatók ki egy e-önkormányzati rendszerben?
3. Mit jelent az elektronikus aláírás, miért van rá szükség?
4. Mit jelent a hitelesítési szolgáltatás?
5. Milyen agrár szakigazgatási intézményeket ismer?
6. Milyen e-agrárszakigazgatási rendszereket ismer?

Irodalomjegyzék

IHM: (2003). Információs Társadalom Részstratégia. E-önkormányzat Stratégia
 Kapronczai I. Az agrárinformációs rendszer elemei az EU harmonizáció tükrében.
 MEH: (2001) A mezőgazdaság infokommunikációs eszközökkel történő támogatása
 Várallyai L. e-Közigazgatás

14. ÜZEMELTETÉSI KÉRDÉSEK

14.1 Segédprogramok, kommunikációs és egyéb alkalmazások

Hagyományosan az operációs rendszerekkel szemben a felhasználónak „mindössze” annyi elvárása volt, hogy biztosítsa számára a számítógép kezeléséhez szükséges felületet. Manapság azonban az operációs rendszerek számos olyan szolgáltatást is tartalmaznak, amely nem elsődlegesen a rendszer működését, sokkal inkább a felhasználó kényelmét vagy megelégedését hivatottak szolgálni. Léteznek továbbá olyan felhasználói alkalmazások is, amelyek az operációs rendszer szolgáltatásait kiegészítik, bővítik, esetleg javítják.

A **segédprogramok** (utility) olyan programok, amelyek működésükben az operációs rendszer szolgáltatásaihoz állnak közel, annak hiányzó műveletét valósítják meg (esetleg egy meglevő műveletet oldanak meg szebben-jobban-hatékonyabban, mint az operációs rendszer saját eszközei).

Az operációs rendszerek által nyújtott kiegészítő szolgáltatások megértéséhez nézzük meg a Microsoft Windows XP-ben alkalmazott megoldását: a következőkben néhány általános célú rendszerkomponenssel (vagy ha így jobban tetszik – hiszen tulajdonképpen így kell(ene) neveznünk ezeket az alkalmazásokat – segédprogrammal) ismerkedünk meg.

Alapértelmezett telepítés mellett a START menüben a Kellékek csoport tartalmazza a Windows XP ilyen elemeit.

Szövegszerkesztést támogató alkalmazásból alapértelmezés szerint kettő is van a Kellékek között: a Jegyzetömb formázatlan szöveg kezelését lehetővé tevő szerkesztő program (az angol nyelvben a különböző típusú, szöveget feldolgozó alkalmazásokra – annak funkcionalitása szerint – különböző (kifejező) elnevezések, ezt a csoportot „text editor”-nak hívják), míg a WordPad egy alapszintű formázási műveleteket kezelni képes szövegszerkesztő („word processor”). Ennek megfelelően a Jegyzetömb elsődlegesen olyan jellegű szöveges tartalmak létrehozásának, szerkesztésének vagy megjelenítésének az eszköze, amelyek formai jellemzőket nem tartalmaznak – ilyenek pl. a különböző programkódok. A WordPad segítségével pedig egyszerű formázott szöveget (pl. egy hivatalos iratot, levelet) készíthetünk el. A Számológép a nevéhez mérten sok meglepetéssel nem szolgál, de azt mindenképpen érdemes róla tudni, hogy két működési módja van: normál nézetben hagyományos négy alapműveletes számoló eszköz, azonban tudományos nézetben trigonometriai, statisztikai (és az informatikában jelentős számrendszereket is támogató) műveletek elvégzésére is alkalmas – a két mód között a Nézet menü segítségével lehet váltani.

Szintén a kellékek között érhető el a karakteres felület megjelenítésére szolgáló Parancssor, valamint a Windows Intéző is.

A parancsokról

A Windows erősen kötődik és épít a grafikus felületre, de ez nem jelenti azt, hogy ne rendelkezne karakteres felületen is elérhető utasítás-rendszerrel. Jogosan merül fel a kérdés, mely szerint a grafikus felület eszközrendszere és az operációs rendszer szolgáltatásainak bősége mellett mi szükség lehet utasításokkal beavatkozni a rendszer működésébe? A válasz lehet(ne) alapvetően szemléletet tükröző is: a karakteres felületen való műveletvégzés gyorsabb, mint a grafikus környezet használata – de véleményem szerint az utasítások ismerete elsődlegesen olyan esetekben hasznos, amikor az adott tevékenységnek nincs (vagy nem ismert, esetleg körülményesen elérhető) megfelelő (grafikus felületről is elérhető) alternatívája.

A karakteres felület többféle módon is elérhető: a Kellékek csoportból vagy elindítható a Futtatás menüponthoz begépett cmd vagy command paranccsal is. A következőkben a legáltalánosabban használható parancsok rövid áttekintését adjuk:

1. lemezkezelés parancsai
 - a. **FORMAT** – megformázza a kijelölt háttértárat: új gyökérkönyvtárat (és fájlrendszert) hoz létre (az esetleges előző fájlrendszer és az abban létrehozott állományok természetesen törlésre kerülnek), illetve ellenőrzi a megadott háttértár szektorait. Nem használható rendszert tartalmazó köteten. A helyreállítási konzolból indítva megadható a létrehozandó lemez fájlrendszere is.
 - b. **CONVERT** – FAT vagy FAT32 fájlrendszerű kötetet NTFS kötétté alakít.
 - c. **CHKDSK** – lemezellenőrzést hajt végre a megadott köteten: megvizsgálja a szektorokat, megjelöli a hibás szektorokat (és megpróbálja helyreállítani a bennük található adatokat – sérült lemez esetén célravezetőbb lehet a RECOVER parancs használata).
 - d. **LABEL** – létrehozza, megváltoztatja vagy törli egy meghajtó logikai azonosítóját (kötetcímkéjét). A kötetcímké aktuális értékének lekérdezésére használható a VOL parancs is.
2. könyvtárakra (mappa) vonatkozó parancsok
 - a. **MKDIR (MD)** – létrehozza a megadott könyvtárat (könyvtár-szerkezet megadása esetén a tartalmazó könyvtárak is létrejönnek!).
 - b. **CHDIR (CD)** – megjeleníti vagy kiválasztja (módosítja) az aktuális könyvtárat (könyvtár-váltás).
 - c. **RMDIR (RD)** – törli a megadott könyvtárat. Az aktuális könyvtár vagy annak „szülő”-könyvtára, továbbá nem üres könyvtár nem törölhető!
 - d. **TREE** – grafikus formában megjeleníti a megadott könyvtárból kiinduló könyvtárszerkezetet.
 - e. **SUBST** – virtuális (logikai) meghajtót hoz létre a megadott elérési út, mint gyökérkönyvtárból.
3. állományokra vonatkozó parancsok
 - a. **COPY** – állományokat másol. A parancs általános alakja: COPY forrás cél, ahol általában a „forrás” és a „cél” is elérési út, de „forrás”-ként a billentyűzetet megadva – logikai eszközazonosítója „CON” – szöveges állomány létrehozására is alkalmas, pl: COPY CON level.txt. Alkalmas állományok összefűzésére is.
 - b. **MOVE** – állományokat helyez át.
 - c. **RENAME (REN)** – állományok azonosítóját (név, típus) változtatja meg.
 - d. **ATTRIB** – beállítja, módosítja, megjeleníti vagy törli az állományok karakteres felületen elérhető jellemzőit (attribútumait). (A CACLS parancs hasonló műveleteket végez el az állományok jogosultsági jellemzőivel kapcsolatban.)
 - e. **TYPE** – egy (szöveges) állomány tartalmát megjeleníti.
 - f. **DEL (ERASE)** – állományokat töröl. Figyelem: a karakteres felület törlési művelete nem logikai – a törölt fájlok ténylegesen és véglegesen eltávolításra kerülnek (és nem a Lomtárba...)!
4. könyvtárakra és állományokra egyaránt alkalmazható parancsok
 - a. **DIR** – állománytulajdonságok megjelenítése: név, típus, időbélyegek, stb.
 - b. **COMPACT** – (csak NTFS partíción) a röptömörítés szolgáltatás használatának beállítása, lekérdezése.

- c. XCOPY – teljes könyvtárszerkezet (állományok és könyvtárak is!) másolása.
- 5. egyéb parancsok: környezet lekérdezésének, beállításának parancsai:
 - a. DATE, TIME – rendszer dátum és idő
 - b. VER – az operációs rendszer verziószáma
 - c. PROMPT – a készenléti jel alakja (alapértelmezés szerint az aktuális könyvtár teljes elérési útja és egy „>” jel)
 - d. PATH – keresési útvonalban szereplő könyvtárak (az ezekben a könyvtárakban található végrehajtható kódot tartalmazó állományok a könyvtárszerkezet bármely pontjáról elindíthatóak)
 - e. GETMAC, IPCONFIG, NETSTAT – hálózati környezet alapvető paramétereinek lekérdezésére szolgáló parancsok.
 - f. HELP – a karakteres felület parancsainak rövid leírása (bármely parancs neve után a „/?” kapcsoló az adott parancsra vonatkozó részletes leírást jelenít meg)
 - g. EXIT – a cmd parancsértelmező befejezése, visszatérés a grafikus felülethez.

Ahogy az a fenti (közel sem teljes) felsorolásból is látható, a Windows karakteres felülete ugyanolyan hatékony (sőt, bizonyos esetekben – pl. hálózati paraméterek konfigurálása esetén – még jobban használható) eszköz a felhasználó (vagy a rendszergazda) munkájában, mint a grafikus környezetben.

Kommunikáció, multimédia

A Windows XP nagy hangsúlyt fektet felhasználói igények minél teljesebb kiszolgálására, ennek következménye a hálózati műveleteket és a multimédia alkalmazását (mint a felhasználói szempontból két legfontosabb informatikai alkalmazási terület) támogató segédprogramok minden korábbinál bővebb választéka. A sok vitát kiváltó Internet Explorer továbbra is az operációs rendszer alapértelmezett böngészőprogramja maradt (bár immár opcionálisan eltávolítható), az Outlook Express (a Microsoft Office programcsomag részét képező Outlook nevű komplex csoportmunka-támogató program „leegyszerűsített” változata) az elektronikus levelezést, a Messenger a közvetlen (on-line) kommunikációt („csevegés”), a NetMeeting a hálózaton keresztüli adat- és információ-megosztást teszi lehetővé (ez utóbbi alkalmazás egyben a multimédia támogatására is példa, hiszen az állományokon túl audio és videó tartalmat is képes a hálózat felhasználói között továbbítani, pl. videó-konferencia formájában), végül pedig a „Távoli asztal” szolgáltatás segítségével terminál-szolgáltatás használatára (azaz fizikailag egy másik számítógépről használni egy távoli számítógép erőforrásait) nyújt lehetőséget az operációs rendszer.

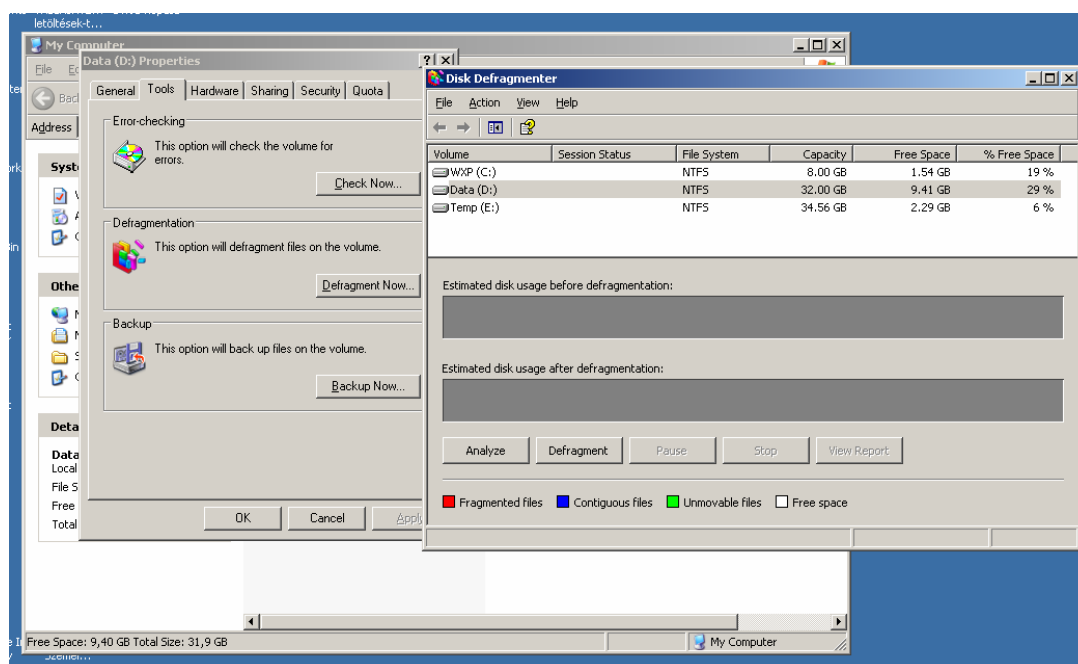
Ami a multimédia támogatását illeti, a Paint nevű grafikus szerkesztő (rajz-) program segítségével rastergrafikus képek készíthetők, szerkeszthetők (a PrtScr billentyűvel lementett képernyőképek pl. kiválóan alakíthatók illetve menthetők a használatával – jelen jegyzet ábráinak egy része is ilyen módon készült), a Médialejátszó egyaránt alkalmas audio- és videófájlok lejátszására, a Hangrögzítő illetve a Movie Maker pedig az ilyen formátumú állományok létrehozására, szerkesztésére.

Felügyeleti eszközök

A segédprogramok között külön csoportot képeznek – nem feltétlen a megvalósító alkalmazások megjelenése vagy elhelyezkedése szempontjából, sokkal inkább a nyújtott szolgáltatások tekintetében – azok az alkalmazások, amelyek az operációs rendszer működésének megváltoztatására szolgálnak, ezek a felügyeleti eszközök. (Ezeket az alkalmazásokat, szolgáltatásokat általában alapértelmezett felhasználói jogosultság mellett

nem vagy csak korlátozott funkcionalitással használhatjuk – a teljes körű felügyelethez kitüntetett felhasználói jogosultságok (a Windows XP fogalmai szerint „kiemelt felhasználó” vagy „rendszergazda”) szükségesek.) Ezek közül – felhasználói szempontból – legfontosabbak a háttértár állapotával és a felhasználói tevékenységhez szükséges programok kezelésével kapcsolatos alkalmazások.

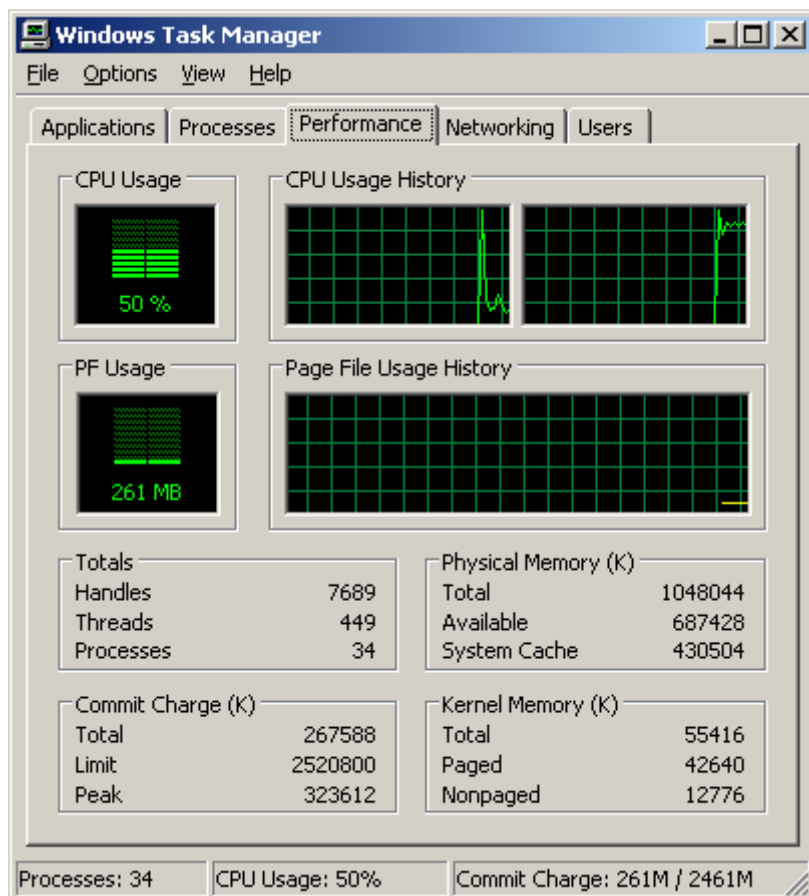
Az előbbire példa a Hibaellenőrző és a Töredezettség-mentesítő (mindkét alkalmazás az ellenőrizni kívánt háttértár Tulajdonságok lapjának Eszközök fülén érhető el). A Hibaellenőrző a fájlrendszer hibáinak felderítésében és kijavításában játszik szerepet, a Töredezettség-mentesítő pedig a tárolás-szervezés logikájából következő elhelyezkedési problémák optimalizálására és ilyen módon a lemez teljesítményének növelésére (gyorsabb lesz a fájlok elérése egy nem töredezett tárolási rendszerben) szolgál (14-1 ábra).



0-1. ábra: Lemezellenőrzés eszközei

A programok kezelésével kapcsolatban alapvetően két eszközt kell kiemelnünk: az egyik a rendszer aktuális állapotát megjelenítő (és szükség szerint befolyásolni képes) alkalmazás a Feladatkezelő, a másik a programok elérhetőségével (is) kapcsolatos műveleteket tartalmazó Vezérlőpult. A Feladatkezelő (15-2. ábra) elindítható a Tálca gyorsmenüjéből vagy a CTRL-ALT-DEL (hagyományosan a számítógép újraindítására szolgáló) billentyű-kombináció segítségével. Használatával áttekinthető a rendszerben futó alkalmazások és az alkalmazások által használt folyamatok állapota, erőforrás-használata, valamint a rendszer általános teljesítményére vonatkozó mérőszámok és a hálózati felhasználásra vonatkozó adatok. Igazi jelentőségét azonban az áttekintésen túl a beavatkozás lehetősége adja: segítségével elindíthatunk és leállíthatunk alkalmazásokat (pl. a nem válaszoló (lefagyott) vagy a nem kívánt (vírus) programokat).

A Vezérlőpult (START menü önálló programcsoportja) a felhasználói beavatkozást támogató eszközök alapvető gyűjteménye, melyek közül a felhasználói felület szempontjából a „Megjelenítés”, a kezelés szempontjából az „Egér” illetve a „Billentyűzet”, a működés szempontjából a „Rendszer”, a programok szempontjából pedig a „Programok telepítése és eltávolítása” a leglényegesebb.



14-2. ábra: Feladat-kezelő

14.1.1 Tömörítés

A segédprogramok különböző típusai közül a felhasználói gyakorlatban talán legtöbbször használtak (és ilyen értelemben az egyik legfontosabbak) a tömörítő programok.

A tömörítés elve egyszerű: az adatot eredeti méreténél kisebb méretűre kell átalakítani. Az átalakítás célja általában a *tárolási helyigény* csökkentése (egy kisebb állomány kevesebb helyet foglal), de manapság ugyanilyen fontos a *továbbítás időigényének* a csökkentése (pl. hálózaton keresztül).

A tömörítés folyamatában az *eredeti adatot forrásnak*, a *tömörítés eredményeként előálló újabb változatot pedig archívumnak* nevezzük. A tömörítés általában (de nem szükségszerűen) kétirányú folyamat: a forrás → archívum irányt *tömörítésnek* (becsomagolásnak) nevezzük, az archívum → forrás művelet neve *kibontás* (kicsomagolás, kitömörítés). A tömörítési eljárásokat alapvetően ezen átalakítási módszerek szerint kategorizálhatjuk:

- a **veszteséges tömörítési eljárások** során a forrásból bizonyos részek kimaradnak az archívum elkészítése során. Ebből következik, hogy ezek a tömörítési eljárások általában nem megfordíthatóak, azaz nem tartozik hozzájuk kibontási művelet. Felmerül a kérdés, hogy mi értelme van tömöríteni, ha a tömörítés adatvesztéssel jár? Természetesen csak akkor alkalmazható ez a fajta tömörítési eljárás, ha az adatvesztés mértéke megengedhető, tipikusan ilyenek a multimédia jellegű adatok tömörítési eljárásai. (Az elv az, hogy az emberi érzékszervek „becsaphatók”: ha a forrásból elhagyjuk azokat a részeket, amelyeket úgysem tudnánk az érzékszerveinkkel felfogni – pl. hangok esetében az emberi fül által érzékelhető 20 Hz – 20 KHz frekvencia-tartományon kívül esőket –, akkor ez a veszteség nem

számottevő.) A veszteséges tömörítési eljárások során készülő archívumok (általában nem is szokás őket archívumnak nevezni, inkább egy újabb adattípusról beszélhetünk a tömörítés eredményeként) közvetlenül felhasználhatóak: megnézhető, meghallgatható, szerkeszthető, stb. Ismertebb veszteséges tömörítési eljárások pl. az állóképek esetén alkalmazott jpeg eljárás, az audió-adatok tömörítésére használt MP3 kódolás, vagy a videók mpeg, illetve DIVX kódolása.

- ezzel szemben a **veszteség-mentes tömörítési eljárások** alkalmazásakor az archívumból (a kibontási folyamat során) a forrás tartalma maradéktalanul helyreállítható. Természetes, hogy ezeket az eljárásokat elsősorban a tényleges adat-tartalmú állományok (pl. dokumentumok, táblázatok, stb.) tömörítésére használhatjuk. A veszteség-mentes tömörítési eljárások eredményeként létrejövő archívumban megtalálható ugyan a forrás teljes (eredeti) tartalma, de közvetlen módon nem dolgozható fel, csak a visszaállítás után. (Ez alatt azt kell érteni, hogy ha pl. egy szöveges dokumentumot betömörítünk, akkor az archívumot hiába szeretnénk egy szövegszerkesztővel szerkeszteni.) Nagyon sok ilyen eljárás (ismétlődés-alapú: RLE, statisztikai alapú: Huffman, lexikai: LZW, stb.) létezik és ennek megfelelően nagyon sok (és sokféle) tömörítőprogrammal is találkozhatunk (különböző ZIP variánsok, RAR, LHA, ICE, stb.).

(Emlékeztetőül: egyes operációs rendszerek (a különböző Linux rendszerek eredendően, de manapság már a Windows XP is) natív módon (az operációs rendszer saját szolgáltatásaként, nem segédprogram formájában) is támogatnak különböző tömörítési eljárásokat.)

A használni kívánt tömörítőprogram kiválasztása a felhasználó szubjektív megítélésén alapul, általánosan igaz az, hogy nincs olyan eljárás (és nincs olyan tömörítő program sem), amely minden szempontból jobb lenne a többinél. Ilyen szempont lehet valamely hatékonysági mérőszám (mennyire (a forrás méretének %-ban) képes tömöríteni a program, milyen gyors, stb), de akár az elterjedtség (pl. a ZIP alkalmazások népszerűségének egyik oka az, hogy az Interneten kvázi-szabvánnyá vált), a kezelhetőség vagy a nyújtott többletszolgáltatások (titkosítás, testre szabhatóság, stb.) is. A hatékonyság vizsgálatánál ne feledkezzünk meg arról, hogy a tömörítés két alapvető jellemzője (a méret és az időigény) általában egymásnak ellen-hat: egyrészt a tömörítés egy kódolási eljárás, és mint ilyen, időigényes, másrészt a tömörítés célja a méretcsökkentés. Könnyű belátni, hogy minél kisebb méretet szeretnénk elérni, (általában) annál több műveletre van szükség a kódolás során – következésképpen annál tovább tart a be- és a kitömörítés. (A két szempont közti preferencia általában minden egyes tömörítési művelet során külön beállítható.) A kezelhetőség szempontjából a grafikus felületű – vagy akár az operációs rendszer saját szolgáltatásaiba integrálódni képes – programok valószínűleg egyszerűbben kezelhetők egy átlagos ismeretekkel rendelkező felhasználó számára, ugyanakkor egy parancsmódú változat számos olyan többletműveletet tartalmazhat (pl. kapcsolók használatával), amely a grafikus felületen keresztül nem érhető el (jó példa erre az arj nevű program).

14.1.2 Vírusok

A segédprogramok másik alapvető csoportjába (sajnos) azok az alkalmazások tartoznak, amelyek nélkül szintén nem lehet meg egyetlen számítógépet aktívan használó felhasználó sem: a vírusok elleni védekezés eszközei.

De mik azok a vírusok? Számos definíció közül a legegyszerűbb (és talán legszemléletesebb) az a megközelítés, amely szerint a vírus egy olyan **program**, amely felhasználói beavatkozás nélkül képes **saját kódjának többszörözésére** (reprodukciós képesség) és **célja a felhasználói tevékenység megzavarása** (ártó szándék). (Az „ártó

szándék” nem minden esetben egyenlő a károkozással, adott esetben a bosszantó jelleg vagy az adatgyűjtés is lehet ártalmas...)

A számítógép-vírusok megjelenése egybeesik a számítógép megjelenésével, hiszen már a hetvenes években, a mai értelemben vett számítógép kifejlesztésének idején írtak programokat, amelyek tartalmaztak olyan utasításokat, amik szándékosan rongáltak, illetve hibákat okoztak. Az ilyen programok kifejlesztésével az volt a készítőik célja, hogy teszteljék a számítógép terhelhetőségét.

Minden vírus általában három alapvető részből áll:

- a *reprodukciós rész* a vírus legfontosabb része, mert ez az, ami terjed, azaz megkeres olyan programokat vagy alkalmazásokat, amelyekbe be tudja írni magát;
- az *aktiválási feltétel* az a feltétel, aminek a teljesülése esetén a vírus kifejti „hatását”. Ez bármi lehet: csak annyi, hogy a kód betöltődjön a memóriába; vagy egy dátum eljövetele; egy rendszerállapot kialakulása (pl. a lemez foglaltsága nagyobb, mint 50%), stb.
- a *kódrész* pedig az az utasítás-sorozat, ami a tényleges károkozást megvalósítja.

A számítógépes vírusokat többféle szempont szerint szokták csoportosítani. Az egyik ilyen szempont lehet, hogy milyen a vírusok terjedési típusa, vagyis az, hogy a rendszerünk melyik elemét szemelik ki szaporodásuk közegének. A másik szempont a vírusok a tulajdonságainak milyensége, vagyis az, hogy hogyan képesek magukat elrejteni, megnehezíteni azonosításukat.

A terjedés módja szerint léteznek **boot szektor vírusok** (a háttértárak rendszerindításért felelős szektoraiban található meg), **fájl-vírusok** (amelyek neve egy kicsit megtévesztő: csak az önálló végrehajtásra alkalmas kódot tartalmazó – parancs- illetve programfájlok, kódkönyvtárak – állományokban előforduló vírusok tartoznak ide) és a manapság legelterjedtebb ún. **makróvírusok** (amelyek többé-kevésbé bármilyen tartalmú állományhoz képesek hozzákapcsolódni, feltéve, hogy az adott típusú állomány feldolgozása során elképzelhető valamilyen parancsértelmezési tevékenység – legtöbb esetben az operációs rendszerek vagy az adott típusú állományt feldolgozó alkalmazások automatikus szolgáltatásai révén).

(Manapság – elsősorban szintén az elterjedtségüknek köszönhetően – külön kategóriába szokták sorolni azokat a vírusokat, amelyek működési mechanizmusa a makróvírusokéval azonos, csak jellemzően az Internet aktív tartalommegjelenítő eszközei révén hatnak, ezek az ún. **script típusú vírusok**.)

(Létezik továbbá egy másik „vírus”-kategória is, amely egyértelműen az elektronikus levelezési rendszerekhez köthető, ami valójában nem vírus, de károkozása kétségtelen, az ún. **hoax** (ál-levél, esetleg lánc-levél). Az ilyen levelek egy kitalált indokkal arra ösztönzik a felhasználót, hogy a levelet több példányban továbbítsa – ezáltal valósul meg a reprodukció, a károkozó hatása pedig a rengeteg feleslegesen elküldött levél rendszerterheléséből adódik.)

A vírusok csoportosításának másik lehetősége a működés módja szerinti csoportosítás: a **memóriarezidens vírus** aktiválása után a számítógép kikapcsolásáig a memóriában marad (és fertőz), az **alkalmazás-vírus** a hatás kifejtése után befejezi a működését (mint bármely más felhasználói program), az **időzített bombák** esetében mindaddig csak az aktiválási feltételt vizsgáló komponens aktív, amíg a feltétel nem teljesül, a **lopakodó vírusok** az operációs rendszer alapszolgáltatásainak meghamisításával próbálnak elrejtőzni a felhasználó és a víruskereső programok elől, a **polimorf vírusok** pedig képesek saját kódjuk

megváltoztatására (mivel a víruskeresők minta alapján ismerik fel a vírusokat, így megtalálásuk lényegesen nehezebb).

A vírusok elleni védekezés eszköze sokrétű lehet – és kell is, hogy legyen! Elsősorban mindenképpen a felhasználói felelősséget kell megemlíteni: megtett-e a felhasználó mindent a fertőzés megakadályozására? A „social engineering”-nek (leginkább „felkészült felhasználók”-nak lehetne fordítani) a lényege az, hogy a felhasználó rendelkezzen azokkal az ismeretekkel, amelyek révén csökkenthető a vírusok rendszerbe való bejutásának vagy aktiválódásának a veszélye: ismeretlen forrásból származó állományokkal szembeni óvatosság, ellenőrizhető szerzőtől származó alkalmazások, etikus magatartás (hoaxot nem továbbítunk...).

Természetesen minden jóindulat és felkészültség ellenére is előfordulhat vírusfertőzés, ilyenkor a megfelelő vírusvédelmi rendszerek nyújthatnak segítséget. (Elvileg az operációs rendszertől is elvárható (lenne) valamiféle védelmi mechanizmus, de ez (pl. a Microsoft rendszerei esetében) nem valósul meg maradéktalanul.) A vírusvédelmi rendszerek esetében az alapszabály a „bármilyen jobb, mint a semmi”: számos kereskedelmi forgalomban kapható termék mellett legalább ugyanennyi ingyenesen használható rendszer is rendelkezésre áll. A rendszerek használatakor pedig a kulcszó a rendszeresség: rendszeresen ellenőrizni kell a számítógépet és rendszeresen aktualizálni (frissíteni) kell a víruskereső program adatbázisát!

14.2 Adatvédelem, adatbiztonság

Az információs társadalom megjelenésével mind az egyén, mind a szervezetek szempontjából új típusú értékek keletkeztek: az adatok. A személyes adatok, a hivatali titkok, szervezetek működését befolyásoló információk ugyanolyan – gazdasági értékkel is mérhető – erőforrásokká váltak, mint a tárgyi anyagi javak. Ennek a folyamatnak természetes velejárója az információ megszerzésére vagy megrongálására irányuló (szándékos vagy véletlen) kísérletek megjelenése is, amelyek kezelésére minden szervezetnek fel kell készülnie. A következőkben megpróbáljuk bemutatni azokat a tényezőket, amelyek befolyásolják a szervezet információ-kezeléssel kapcsolatos feladatait: az adatok jellegéből adódó speciális fenyegetettségek és az ezek ellen való védekezés módszereit, az adatkezeléssel kapcsolatos szabályozási elveket és módszereket, a felhasználói eszközöket. A témakör kiterjedtségére és összetettségére tekintettel meg sem kíséreljük (a valószínűleg nem is létező) „egyetlen és üdvöztető” megoldás bemutatását, helyette példákon keresztül megpróbáljuk bemutatni mindazokat a területeket, amelyek megfontolásra érdemesek.

Manapság az elektronikus kommunikáció minden eddiginél nagyobb mértékű lett. Ezen nem csak az Interneten történő levelezést és adatátvitelt kell érteni, hanem a telefonvonalakon bonyolított egyéb kommunikációt is: fax üzenetek, telebank szolgáltatások igénybevétele vagy egy egyszerűnek tűnő pizza- vagy mozijegy rendelés. Ha egy-egy ilyen kapcsolat alkalmával valaki a személyes adatait is használja (cím, email cím, bankkártya szám, stb.), tovább fokozódik a veszély. A telefonvonalak és központok **digitalizálódásával** egyre könnyebb egy kapcsolatot (annak tényék és tartalmát) rögzíteni és tárolni - esetleg később feldolgozni.

Korábban elképzelhetetlen feladatok és célok ma már megvalósíthatók. Ezzel egy időben a tárolt információk felértékelődtek és gyakori célpontjai lettek az ipari kémkedésnek és a „csakazértis” stílusú hacker valamint a jóval veszélyesebb cracker támadásoknak, különböző direktmarketinges megoldásoknak.

Gyakran felmerülő kérdés, hogy ki a hacker és ki a cracker. Egyértelmű válasz aligha adható, de egy lehetséges kategorizálás a következő: a hacker nagy tudással rendelkező, jól felkészült informatikai szakember, akit elsősorban az ismeretszerzés motivál (a professzionális

hackerek ebből élnek: vagy biztonságtechnikai tanácsadók, vagy adatrablók...), ezzel szemben a cracker tevékenységének alapvető szándéka a rombolás, a károkozás (nem ritkán még az általa használt eszközök működésével sincs tisztában: script kiddie.)

A külső tényezők mellett azonban nem szabad figyelmen kívül hagyni azokat a veszélyeket sem, amelyek nem az adatokban tárolt információra nézve fenyegetőek, hanem magukra az adatokra. Az informatikában az adatok (a tárolás jellegéből adódóan) számos olyan veszélynek vannak kitéve, amelyek nem a feldolgozáshoz, hanem a tároláshoz kötődnek: ha nem tételezünk fel rossz szándékot, akkor is beszélnünk kell a (véletlen, figyelmetlenségből adódó) törlésről, az adatok „elvesztéséről” („nem találom...”), rosszabb esetben a számítógép meghibásodásából, rongálódásából vagy eltulajdonításából következő adatvesztésről.

Az adatvédelem és az adatbiztonság fogalma a magyar nyelvben hasonló (és gyakran keveredő) jelentéstartalommal használt kifejezés. Anélkül, hogy kategorikusan meghatároznánk, hogy melyik fogalom mi mindent jelöl(het), jelen jegyzet olvasása során fogadjuk el alapvetésnek, hogy **adatvédelem** alatt az információs és kommunikációs rendszerek adataihoz való *illetéktelen hozzáféréssel kapcsolatos problémák kezelésével kapcsolatos tevékenységeket* értjük, míg **adatbiztonság** alatt azokkal a módszerekkel foglalkozunk, amelyek *az adatok megsemmisülését hivatottak megakadályozni* (vagy az ily módon jelentkező problémákat orvosolni).

Az adatokat veszélyeztető tényezők:

- fizikai hatások (meghibásodásból, rongálódásból, eltulajdonításból adódó veszteségek): az ilyen típusú fenyegetettségek pusztán informatikai eszközökkel nem kezelhetők (csupán a teljesség miatt szerepelnek a felsorolásban), de ezeknek az ilyen módon keletkező károk mértékének csökkentésére is alkalmasak a „hagyományosan” informatikainak tekintett fenyegetettségek elkerülésére alkalmas módszerek;
- adatvesztés: ide tartoznak a szándékos vagy véletlen törlések miatt elvesztett adatokkal kapcsolatos problémák csakúgy, mint az informatikai rendszerek nem megfelelő működéséből („lefagyás”) adódó veszteségek;
- illetéktelen hozzáférés: az adat vagy információ elérhetősége egy szándékolatlan külső („harmadik fél”) részére;
- hamisítás: lényegében az előző csoportba is sorolható tevékenység, az információ tartalmának (jelentésének) megváltoztatására irányuló kísérleteket soroljuk ide.

Látható, hogy számos tényezőt kell figyelembe venni, amikor az adatvédelemmel kapcsolatos tevékenységeket vizsgáljuk. Az egyes fenyegetettségekhez természetesen alkalmas **védelmi eszközök** is kapcsolódnak, a legfontosabbak ezek közül (az előző felsorolással összhangban, de hangsúlyozottan nem kizárólagosan az egyes fenyegetettségi területekhez kötötten!):

- fizikai korlátozások (elzárás, biztonsági szolgálat, őrzés)
- biztonsági másolatok, archiválás, elosztott tárolás
- azonosítás, jogosultság-kezelés, titkosítás
- módosulás/módosítás elleni védelem

A **sikeres megvalósítás** kritikus tényezői általában a következők:

- A biztonságpolitika, a célok és a tevékenységek az üzleti célokon alapuljanak,

- A megközelítés összhangja a szervezeti kultúrával
- A vezetés elkötelezettsége és támogatása
- A védelmi követelmények, a kockázatelemzés és menedzselés jó megértése,
- Minden vezető és alkalmazott bevonása
- Az útmutatók eljuttatása minden alkalmazotthoz és partnerhez, alvállalkozóhoz
- Megfelelő képzés és oktatás
- Átfogó és kiegyensúlyozott mérési rendszer a végrehajtás kiértékelésére és a fejlesztési javaslatok jelzésére

A fentiek figyelembevételével a biztonsági rendszer megvalósítása öt területen történhet:

- *fizikai védelem*: a gépekhez, adathordozókhoz fizikailag csak az arra jogosult személyek férhetnek hozzá.
- *a felhasználó személyazonosságának és hitelességének megállapítása* („authentication”): az informatikai rendszer felhasználói csak azonosítás után férhetnek hozzá az adatokhoz.
- *jogokkal történő felruházás* („authorization”): a felhasználók számára a tevékenységük elvégzéséhez minimálisan szükséges felhatalmazások biztosítása.
- *rejtjelezés, titkosítás* („encryption”): az illetéktelen hozzáférés megakadályozása a kritikus állományok és adatbázisok kódolt tárolásával vagy a kommunikáció titkosításával.
- *számlázás, ellenőrzés* („accounting/auditing”): a rendszer eseményeinek folyamatos figyelése, a felhasználók tevékenységének naplózása, a naplók elemzése.

14.2.1 Az adatvédelmi törvény

Az állam működtetéséhez, a társadalmi és gazdasági tevékenységek tervezéséhez és szervezéséhez szükséges információ mennyisége folyamatosan növekszik. A nagy tömegű információt viszont mind kevésbé lehet a hagyományos módszerekkel kezelni, így a modern társadalom jelentős részben a számítógépekben tárolt és feldolgozott adatokon alapszik. Az adatkezelés során az adatbiztonsági intézkedések jelentősége egyre nagyobb az automatizált adatfeldolgozás terjedésével. Rendkívüli a jelentősége annak, hogy az adatokhoz az érdekeltek hozzáférjenek. Ezt a hozzáférést azonban megfelelően szabályozni, korlátozni szükséges, hogy az érintett jogai vagy érdekei ne csorbuljanak e megismerés által. Az adatkezelések törvényessége és tisztessége érdekében elengedhetetlen, hogy az informatikai rendszerek üzemeltetői és használói a vonatkozó jogszabályokat megismerjék és munkájuk során figyelembe vegyék.

A legfontosabb vonatkozó törvényi rendelkezések és kapcsolódó szabályok a következők: A Magyar Köztársaság Alkotmánya, és az ún. „Adatvédelmi Törvény” (1992. évi LXIII. törvény A személyes adatok védelméről és a közérdekű adatok nyilvánosságáról), amelyben az adatvédelem alapelve került megfogalmazásra: **Adat csak akkor kezelhető, ha az érintett hozzájárul vagy törvény, jogszabály elrendeli. Kétség esetén azt kell vélelmezni, hogy az érintett a hozzájárulását nem adta meg.** Személyes adatot kezelni csak meghatározott célból, jog gyakorlása és kötelezettség teljesítése érdekében lehet. (Csak érdekességképpen jegyezzük meg, hogy ez a törvény rendelkezik az adatvédelmi biztos feladatáról és jogosítványairól, mint ahogy meghatározza a különböző szintű védettséget

élvező adatok („titok”: államtitok, szolgálati titok, üzleti titok, banktitok) körét, tartalmát és kezelésükkel kapcsolatos előírásokat is.) Végezetül pedig meg kell említeni, hogy a Büntető Törvénykönyv is külön fejezetben rendelkezik az ún. „Számítógépes csalás” mint bűnesetről.

Az adatvédelmi törvény (már csak megalkotásának idejéből következően sem) kezeli az elektronikus dokumentumok és az elektronikus kommunikáció specialitásait. (Sokak szerint ez egyébként olyan területe a jogtudománynak, amelyben általános érvényű szabályozás nem is képzelhető el...) A jelen társadalmi és gazdasági változásai azonban megkövetelnek valamiféle szabályozást ezen a területen is. Az elektronikus adatkezelésnek (és ily módon az elektronikus kereskedelmi, szolgáltatási és közigazgatási tevékenységek elterjedésének) alapfeltétele a kommunikációba résztvevőinek bizalma. Ennek megteremtését hivatott elősegíteni a digitális (vagy elektronikus) aláírásra vonatkozó törvény.

A törvénytervezet kidolgozásának alapját az Európai Közösség vonatkozó Irányelveiben megfogalmazott elvek képezték, figyelembe véve a magyar jogrendszerből és viszonyokból adódó sajátosságokat. Az Európai Parlament és Tanács elektronikus aláírásra vonatkozó irányelvét 2000. január 19-én tették közzé az Európai Közösségek Hivatalos Lapjában. Eszerint a szabályozásnak technológia-függetlennek kell lennie, ezért többszintű szabályozás kialakítása szükséges. A legfelső, törvényi szint az alapelveket rögzíti. A technológia-függő részek szabályozása alacsonyabb jogforrásban történhet, amelyek nemzetközi szabványokon alapuló technikai megoldásokat tartalmaznak.

Jelenleg (világviszonylatban is) az ún. nyilvános kulcsú eljárásokra alapozott elektronikus aláírás terjedt el. A magyar szabályozás is ezen eljárások feltételeinek kialakítására törekszik, nem zárva ki azonban más megoldások létjogosultságát sem.

A törvény egyik legfontosabb szabálya, hogy elismeri az elektronikus aláírás és irat joghatályát, gazdasági életben, közigazgatásban való alkalmazhatóságukat az Európai Közösség elektronikus aláírásra vonatkozó Irányelvének szellemében. Jogi eljárásokban bizonyítékként az elektronikus aláírás és irat elfogadhatósága nem tagadható meg azon az alapon, hogy kizárólag elektronikus formában van jelen.

14.2.2 Informatikai vonatkozások

A társadalmi és gazdasági környezet vizsgálata után térjünk vissza az adatvédelem információ-technológiai jellegzetességeihez. Adatvédelemről beszélve feltételezhetjük, hogy a rendelkezésre álló információhoz a jogosultakon kívül más is megpróbál hozzáférni. A fizikai védelmi megoldásokat kicsit figyelmen kívül hagyva a problémát általánosíthatjuk egy olyan kommunikációs folyamatra, amelyben két érintett (jogosult) fél és egy illetéktelen (behatoló) vesz rész, és amelynek a célja az információ eljuttatása az egyik érintett féltől a másikig. (A fogalmakkal, az adatvédelem kommunikációra történő kivetítésével csak a probléma jobb általánosíthatóságát igyekszünk megteremteni – ugyanezek a feladatok jelentkeznek akkor is, amikor nincs szó a hagyományos értelemben vett kommunikációról (azaz az adatok nem hagyják el a számítógépet), de más az adatokat létrehozó és más a feldolgozó személy.)

A folyamat megértéséhez a következő fogalmak jelentését kell tisztáznunk: azt az üzenetet, szöveget, adatot, amit továbbítani szeretnénk (és nincs szükség semmi extra műveletre annak értelmezéséhez), **nyílt szövegnek** nevezzük; az a művelet, amely a nyílt szöveget elrejt, a **titkosítás**; a létrejövő értelmezhetetlen adathalmaz a **titkosított szöveg**; a titkosított szöveg nyílt szöveggé való jogosult visszaalakítását **megoldásnak**, a jogosulatlan megoldást **megfejtésnek** (feltörésnek) nevezzük; és mindehhez kell a **kulcs** (key).

A titkosító módszereknél alapvető elvárás, hogy egy adott információból úgy készítsen másikat, hogy ez utóbbiból csak egy kiegészítő információ ismeretében lehessen megismerni

az eredeti információt. Ezt a kiegészítő adatot nevezzük kulcsnak, ami egy lehetőleg hosszú, véletlenszerű jelsorozat. Egy speciális kulcs a **digitális aláírás** (esetleg időbélyeggel kiegészítve): a hagyományos aláírás informatikai megfelelője. A digitális aláírás egy olyan üzenet, ami (általában) tartalmazza az „aláírt” adat jellemzőit, hitelesítő ellenőrző összegét, az aláírás idejét, esetleg helyét és az aláíró nevét. Így nemcsak az aláíró azonosítja, hanem hitelesítő eljárásokkal kiegészítve az adatokat is védi hamisítás és ismétlés ellen.

A (személy)azonosság ellenőrzésére alapvetően három lehetőség kínálkozik:

- **ismeret-alapú:** Ez a legegyszerűbb, legolcsóbb és leggyengébb módja az azonosításnak: az ellenőrzés egy „kérdés-válasz” típusú folyamatként írható le. Tipikus példája a jelszó: egy olyan karaktersorozat, amit meg kell jegyezni. (A statisztikák szerint a számítógépekbe való illetéktelen behatolások 80%-a vezethető vissza a jelszavak megfelelő védelmének hiányosságaira (kitalálható tartalom, hozzáférhető helyen való tárolás, stb.).
- **birtoklás-alapú:** Egy sokkal erősebb azonosítási módszer, lényege, hogy az azonosítást egy olyan tárgyhöz köti, amelynek birtoklása igazolható. Klasszikus példája a különböző hardver- és szoftverkulcsok alkalmazása (ld. a regisztrációs kódok), de ilyenek a különböző (programozható) azonosító kártyák (bankkártya, a mobiltelefonok SIM kártyája, stb.). (A gyakorlatban előszeretettel alkalmazzák az első módszerrel együtt, annak kiegészítésére.)
- **tulajdonság-alapú:** Ez a fajta módszer valamilyen egyedi (jellemzően biológiai) sajátosság felismerésén alapszik (ún. biometrikus rendszerek: ujjlenyomat-azonosítás, írisz-diagnosztika, hangminta-elemzés, stb.). A legerősebb és legbiztonságosabb azonosítást garantáló módszer, de komoly (sok esetben még nem eléggé megbízható) technológiai háttérrel igényel az alkalmazása.

14.2.3 Adatvédelmi módszerek a gyakorlatban

Az Internet utóbbi időre jellemző elterjedésével az informatikai fenyegetettségek száma és csoportja is jelentősen kibővült. A következőkben a legjellemzőbb fenyegető tényezőket mutatjuk be:

- **Vírusok**
- **Kéretlen levelek** (levélszemét, SPAM): Az elektronikus levelező rendszereken keresztül továbbítódó olyan üzenetek, amelyeket a címzett akart megkapni. Veszélye (az idegesítő jellegén túl) nagy mennyiségében rejlik: pl. a „lánc-levelek: küldd tovább 10-50-100 embernek” mind a hálózatot, mind a címzett postafiókját leterhelhetik. Hasonló elven (bár más mechanizmussal, a böngészés folyamatához kapcsolódva) működnek a **kéretlen felugró ablakok** (pop-ups) is.
- **Hálózati betörési eszköz** („kiskapu”): Olyan program, amely a (megfelelő paraméterek ismeretében) távoli hozzáférési lehetőséget biztosít az adott számítógéphez.
- **Hamisítás** (keret-csere): Böngészés közben sokszor a szerverek kisebb részekben, ún. *keretekben* (frame) jelenítik meg az információt. Egy több keretre osztott lap minden része általában ugyanarról a szerverről származik, de az is lehet, hogy különböző keretek tartalma különböző szerverekről. Alkalmas módon meghamisítva a keretet, pl. azt hiheti a gyanútlan felhasználó, hogy egy web-es áruház lapján titkosítottan küldi a bankkártya számát, valójában pedig illetéktelenek kezébe juttatja.

- **Adatgyűjtés:** Az Internet tartalomszolgáltatói a felhasználói elégedettség növelése érdekében számos olyan szolgáltatást építenek be rendszereikbe, amelyek révén az egyes Internetet használó felhasználók megkülönböztethetők. Az eredetileg segítő szándékú elképzelés könnyen felhasználható ártó szándékkal is, ha az ilyen módon összegyűjtött információval visszaélnék (pl. felhasználói szokások elemzése, nyomkövetés, stb.)
- **Bénító támadások:** A rosszindulatú tevékenység célja ma egyre többször nem éppen az, hogy a támadott gépről illetéktelenül adatokhoz, szolgáltatásokhoz férjenek hozzá, hanem csak annyi, hogy mások számára ezt lehetetlenné tegyék. Ezek a DoS (Denial of Service) támadások.

A védekezés alapelve: **sose higgyük, hogy tökéletes biztonságban vagyunk!** Nem elég időnként ellenőrizni a rendszereket. Állandóan figyelni kell, és készen kell állni a beavatkozásra. Használjunk akár kereskedelmi forgalomban kapható, akár ingyen rendelkezésre álló (pl. Internetről letölthető, magazinok mellékleteként kapható) eszközöket, programokat (nem feltétlenül biztonságosabb valami, mert drágább, vagy egy csak mert egy nagyvállalat terméke) – de legfőképpen a józan ész és az odafigyelés segíthet. Ez utóbbi a már emlegetett „**social engineering**”: képezzük magunkat (illetve adott esetben beosztottjainkat, munkatársainkat is), hogy legyenek tisztában a fenyegetettségekkel és az ellenük való védekezés módszerével. (A „küld tovább 100 ismerősödnek” című levelekben foglaltaktól még soha egyetlen rákos kiskutya sem gyógyult meg...)

Természetesen az ismeretek mellett alkalmazásokra is szükség van. Egy (Internet kapcsolattal rendelkező) számítógépen általában javasolható a következő adatbiztonságot és/vagy adatvédelmet támogató eszközöknek a feltelepítése:

- **vírusvédelmi rendszer:** valamilyen vírusirtó program. Számos gyártó számtalan terméke rendelkezésre áll, a korszerűbbek a vírusok mellett a hálózati fenyegetések egészével vagy részével szemben is rendelkeznek valamilyen védelmi mechanizmussal.
- **levélszemét-szűrő (spam filter):** egyes megvalósításokban a vírusirtó program, másokban a levelező program kiegészítése, amely bizonyos (általában nyilvántartásokon, szövelemzéseken alapuló) módszerek alkalmazásával megpróbálja megkülönböztetni az érvényes és a kóros leveleket.
- **tűzfal (firewall):** a hálózati forgalmat felügyelő program. Megfelelő beállításokkal biztosítható, hogy csak a felhasználó által kívánt műveletek működhessenek.
- **archiválási rendszer (backup, synchron):** az adatvesztés megelőzésének legegyszerűbb módja a kényes adatok többszörös tárolása (duplikáció). A korszerű mentőrendszerek általában képesek automatikus mentések elvégzésére, illetve a mentési (vagy helyreállítási) művelet során a bevonni kívánt adatok körének meghatározására.
- **egyebek:** ilyenek pl. azok a programok, amelyek megkísérlik kivédeni a kórosan megjelenő ablakok felbukkanását (pop-up blockers), vagy (igény szerint) eltüntetik az Internet-használatból adódó információ-tároló bejegyzéseket (trace cleaners), stb.

Ellenőrző kérdések:

1. Melyek a segédprogramok legfontosabb típusai?
2. Soroljon fel és mutasson be az Ön által megismert segédprogramok közül 2-3-at (az alkalmazási terület megjelölésével)!
3. Milyen tömörítési eljárásokat ismer?
4. Magyarázza el a tömörítő programok működésének elveit, módszereit!
5. Mit nevezünk vírusnak?
6. Melyek az alapvető vírus-kategóriák?
7. Hogyan védekezhünk a vírusfertőzés ellen?
8. Mit jelent a „social engineering”?
9. Melyek az informatikában a legfontosabb (adatokat érintő) fenyegető tényezők?
10. Milyen területeket célszerű szabályozni egy „informatikai biztonsági terv”-ben?
11. Melyek a személyazonosság-ellenőrzés különböző szintű módszerei?
12. Miben áll az adatvédelmi törvény jelentősége? Véleménye szerint mennyire illeszkedik/illeszthető az informatika specifikumaihoz?
13. Milyen adatvédelmi megoldásokat ismer?

Irodalomjegyzék

- [1] Bártfai: *Hogyan használjam? (BBS-Info, 2004)*
[2] Györfi – Györi – Vajda: *Információ- és kódelmélet (Typotext, 2003)*
[3] Tóth: *PC vírusok (LSI, 1999)*

Ellenőrző kérdések

1. AZ INFORMATIKA ALAPJAI

1. Határozza meg az adat és az információ fogalmát!
2. Mivel foglalkozik az információ-technológia (mint tudomány-terület)?
3. Melyek a számítástechnika szempontjából jelentős algoritmusok legfontosabb jellemzői?
4. Magyarázza meg a hardver és a szoftver fogalmát!
5. Ismertesse a Neumann-elveket!
6. Miben áll a „tárolt program elvének” jelentősége?
7. Melyek a számítógép (elvi) funkcionális egységei, mi a feladatuk?
8. Hogyan történhet a numerikus típusú adatok tárolása egy számítógépes rendszerben?
9. Milyen előnyös és hátrányos tulajdonságokkal rendelkezik a fixpontos, illetve a lebegő pontos ábrázolás?
10. Hogyan történik a karakterek tárolása a számítógépekben?

2. SZÁMÍTÓGÉP-ARCHITEKTÚRÁK

1. Mit jelent a negyedik generációs számítógép fogalma?
2. Milyen egységekből épül fel a modern mikroprocesszor?
3. Vázolja fel a processzor működését!
4. Ismertesse a számítógépekben alkalmazott tárhierarchia szintjeit!
5. Jellemezze a processzor és a többi hardverelem kapcsolatrendszerét!
6. Mit jelent az alaplapi chipkészlet fogalma?
7. Milyen háttértároló-kategóriákat ismer?
8. Mit jelent a képernyő frissítési frekvenciája?
9. Sorolja fel a nyomtatók főbb típusait!
10. Foglalja össze a CISC- és RISC-technika különbségeit!

3. OPERÁCIÓS RENDSZEREK

1. Definiálja az operációs rendszer fogalmát!
2. Sorolja fel az operációs rendszerek alapvető feladatait!
3. Hogyan csoportosíthatók az operációs rendszerek? Mondjon példákat!
4. Mit jelent az operációs rendszerek „több-feladatos” és „több-felhasználós” jellege?
5. Hogyan történik az utasítások kiadása karakteres illetve grafikus felületen?
6. Mit nevezünk állománynak, illetve könyvtárnak?
7. Melyek a legfontosabb állomány-tulajdonságok?
8. Melyek a Windows operációs rendszerek grafikus felületének általános jellemzői?
9. Mutassa be a Windows operációs rendszerek állomány-kezeléssel kapcsolatos szolgáltatásait!
10. Mit értünk LINUX disztribúció alatt? Mondjon példákat!
11. Magyarázza meg a következő fogalmakat a LINUX operációs rendszerben: process, konzol, démon!
12. Melyek a LINUX operációs rendszer legfontosabb állomány-kezelő parancsai? Mutassa be egy-két parancs működését is!
13. Melyek a helyi hálózatok alapvető modelljei? Hasonlítsa össze az egyes modelleket (előnyök, hátrányok)!
14. Mit jelent a „dedikált szerver” kifejezés?
15. Melyek a legfontosabb feladatai egy hálózati operációs rendszernek?

4. IRODAAUTOMATIZÁLÁS, FELHASZNÁLÓI ALKALMAZÁSOK

4.1. Szövegfeldolgozás

1. Ismertesse az automatikus helyesírás ellenőrzés használatát a szövegszerkesztő programban!
2. Ismertesse a nyelv megadását a szövegszerkesztő programban!
3. Ismertesse a tabulátorok használatát a szövegszerkesztő programban!
4. Ismertesse az élőfej, élőláb használatát a szövegszerkesztő programban!
5. Ismertesse a körlevélkészítést a szövegszerkesztő programban!
6. Ismertesse a tartalomjegyzék készítését a szövegszerkesztő programban!
7. Ismertesse a tárgymutató használatát a szövegszerkesztő programban!

4.2. Táblázatkezelő rendszerek

1. Ismertesse az Excel diagramkészítés funkcióit!
2. Hogyan történik az Excel adattáblák rendezése?
3. Hogyan történik az Excel adattáblákban a szűrés?
4. Adjon meg néhány Excel függvényt! Ezek mire használhatók?

4.4. Számítógépes grafika

1. Jellemezze a számítógépes grafikai programok csoportjait!
2. Mit jelent a képfelbontás, színmélység, rácsfelbontás, monitor felbontás? Milyen mértékegységekkel jellemezhetők az előzőek?
3. Nevezzen meg néhány vektorgrafikus fájlformátumot!
4. Nevezzen meg néhány pixelgrafikus fájlformátumot!
5. Adja meg a színérzet jellemzőit!
6. Mit jelent a számítógépes grafikában az RGB, jellemezze ezt?
7. Mit jelent a számítógépes grafikában az CMYK, jellemezze ezt?
8. Mit tud a képtömörítésről?
9. Jellemezze a legelterjedtebb grafikus program típusokat!
10. Milyen különleges számítógépes grafikai eljárások vannak? Jellemezze ezeket!

4.5. Prezentáció készítés

1. A diabemutató tervezésének fontosabb szempontjait adja meg!
2. Milyen dianézetek vannak? Jellemezze ezeket!
3. Milyen módokon hozható létre diabemutató?
4. Milyen egyszerű rajzelemeket és beépített rajzobjektumokat lehet használni a PowerPoint-ban?
5. Hogyan lehet a dián a színeket megváltoztatni?
6. Mire szolgál a minta-dia?
7. Mire használható a diarendezés?

4.6. Irodaautomatizálás

1. Sorolja fel a hagyományos irodai funkciók problémáit!
2. Mit jelent az elektronikus iroda?
3. Mit jelent az automatizált iroda?
4. Sorolja fel a hagyományos iroda funkcióit!
5. Sorolja fel az automatizált iroda funkcióit!
6. Mit jelent workflow és mit valósít meg a workflow management?
7. Jellemezze röviden az irodai rendszerek fejlődésének szakaszait!
8. Jellemezze röviden az integrált irodai rendszerek (IIR) csoportjait!

9. Sorolja fel az automatizált irodai rendszerekkel szembeni elvárásokat!
10. Ismertesse röviden a jellegzetes irodai rendszereket és a csoport-munkát támogató szoftvereket!
11. Jellemezzen röviden néhány integrált irodai rendszert!
12. Jellemezze a Notes IIR-t!
13. Mit jelent a workflow rendszer és adjon meg néhány ilyen rendszert!
14. Milyen emberi tényezőket kell figyelembe venni irodaautomatizáláskor?

6. ADATBÁZIS KEZELÉS

1. Mit értünk az adat és az adatbázis kifejezéseken?
2. Milyen adatmodellek ismertek?
3. Jellemezze a hierarchikus adatmodellt!
4. Jellemezze a hálós adatmodellt (kétszintű hálós adatmodell, Codasyl-hálós)!
5. Jellemezze az objektum-orientált adatmodellt!
6. A relációs adatmodell fejlődése.
7. Jellemezze a relációs adatmodellt!
8. Mit értünk táblán, mezőn, rekordon, nézettáblán, kulcson, idegen kulcson?
9. Milyen kapcsolattípusok vannak a relációs adatmodellben? Jellemezze ezeket!
10. Hogyan ábrázolhatók a kapcsolattípusok a relációs adatmodellben?
11. Ismertesse a relációs adatbázis kezelő szoftverek fejlődését!
12. Milyen elvárások vannak az adatbázis kezelő rendszerrel szemben?
13. Az adatbázis kezelő rendszerek felépítése. Az egyes elemek feladatait ismertesse!
14. Mit értünk metaadatok alatt?
15. Mit értünk tranzakción és tranzakció helyes lefutásán?
16. Milyen „technikák” biztosítják a tranzakció helyes lefutását?
17. Mit értünk redundancián, inkonzisztencián, normalizáláson, normálformán?
18. Mit jelent a funkcionális függés?
19. Mit jelent 0NF, 1NF, 2NF, 3NF, BCNF, 4NF, 5NF?
20. Mit jelent a részleges függés?
21. Mit értünk tranzitív függésen?
22. A relációs algebrai műveleteket sorolja fel! Mit jelentenek az egyes műveletek?

7. RELÁCIÓS ADATBÁZIS KEZELŐ RENDSZEREK A GYAKORLATBAN

1. Milyen objektum típusok érhetők el az Access Adatbázis ablakában?
2. Mit tartalmaz és milyen szerkezetű a tábla?
3. Mire használhatók a lekérdezések és az űrlapok?
4. Milyen eszközökkel védhetők az Access-ben az adatbázisok?
5. Milyen engedélyek adhatók a felhasználóknak és a csoportoknak az Access-ben?
6. Milyen adattípusok adhatók meg az Access-ben?
7. Milyen mezőtípusok vannak az Access-ben és mik a jellemzőik?
8. Milyen mező tulajdonságok adhatók meg a táblákban?
9. Milyen típusú lekérdezések készíthetők?
10. Mi a lényege a keresztábrás lekérdezésnek?
11. Hogyan kezdődik a lekérdezés létrehozása?
12. Milyen grafikus eszköz segíti a lekérdezés tervezői munkát?
13. Milyen szakaszok különíthetők el az űrlapon?
14. Mire alkalmazható az Access jelentéskészítője?
15. Hányféleképpen készíthetők az Access-ben jelentések?
16. A Jelentés Varázsló alkalmazásával hogyan lehet elkészíteni egy jelentést?

17. Milyen szerkezetűek az Access jelentések?
18. Milyen résznyelvek alkotják az SQL-t?
19. Milyen adattípusok vannak az SQL-ben?
20. Milyen SQL utasítással lehet táblát létrehozni, módosítani, törölni? Adjon meg példákat!
21. Milyen SQL utasítással lehet a táblát lekérdezni? Adjon meg egy példát lekérdezésre!
22. Milyen SQL utasítással lehet indexet létrehozni, törölni? Adjon meg példákat!
23. Milyen SQL utasítással lehet táblába új sort felvenni, sort módosítani, sort törölni? Adjon meg példákat!
24. Milyen SQL utasítással lehet adatbázist megnyitni, létrehozni, lezárni, törölni? Adjon meg példákat az említett műveletekre!
25. Milyen összesítő függvényeket lehet alkalmazni az SQL SELECT utasításban?
26. Hogyan végezhetők el csoportosítások az SQL SELECT utasításban?
27. Hogyan végezhető el az eredmény táblák rendezése az SQL-ben?
28. Hogyan lehet létrehozni nézettáblát SQL-ben?
29. Milyen SQL utasítással lehet jogosultságokat adományozni, illetve visszavonni?

8. SZÁMÍTÓGÉP-HÁLÓZATOK

1. Melyek a számítógépes hálózatok legfontosabb jellemzői?
2. Sorolja fel miért előnyös a számítógépeket hálózatba kapcsolni!
3. Mi az a hoszt?
4. Magyarázza meg, hogy mit takar az IMP fogalma!
5. Határozza meg a protokoll fogalmát!
6. Mik azok a hálózati rétegek?
7. Mi a hálózati architektúra?
8. Milyen rétegekből épül fel az OSI modell?
9. Mi a fizikai réteg feladata?
10. Mi az adatkapcsolati réteg szerepe, és milyen alrétegei vannak?
11. Mit biztosít a hálózati réteg?
12. Mi a különbség az üzenet és a csomagkapcsolás között?
13. Ismertesse a csavart érpáras átviteli közegeket!
14. Mire használhatók a koaxiális kábelek?
15. Ismertesse az optikai kábeles átviteli közegeket!
16. Milyen elven működik a vezeték nélküli adatátvitel?

9. A TCP/IP PROTOKOLL ÉS AZ INTERNET

1. Milyen rétegei vannak az Internet hálózatnak, és ez hogyan viszonyul az OSI modellhez? Mi az egyes rétegek feladata?
2. Mi a TCP protokoll feladata?
3. Mi az IP protokoll feladata?
4. Ismertesse az Internet címzési rendszerét! Mik azok az A, B, C osztályú címek?
5. Magyarázza el a domén nevek rendszerét!
6. Mutassa be az elektronikus levelezés legfontosabb jellemzőit! Mi az SMTP? Milyen funkciókat kell megvalósítani egy levelező programnak?
7. Mutassa be az FTP alkalmazás legfontosabb jellemzőit! Mi az anonymous FTP?
8. Mi az a TELNET? Milyen biztonsági problémákat okozhat?
9. Mi a WWW? Mi a HTML, URL?
10. Mi az a HTTP protokoll? Hogyan működik?
11. Mire szolgál az XML és a PHP?

12. Melyek a Java fejlesztőeszközök fontosabb jellemzői?
13. Hogyan kapcsolódhatunk telefonvonalon az Internet hálózatra?
14. Melyek az ADSL fontosabb jellemzői?
15. Jellemezze a WiFi, a WiMax és a GSM rendszerek jellemzőit!

10. INFORMÁCIÓS RENDSZEREK

1. Ismertesse az információs rendszer fogalmát!
2. Mire szolgál a vezetői információs rendszer?
3. Az információrendszer típusok közötti kapcsolatok leírása.
4. A stratégiai információrendszerek meghatározása.
5. Miként támogathatják az információrendszerek az üzleti stratégiát?
6. Mi a döntés előkészítő rendszerek feladata?
7. Mi az az integrált vállalatirányítási rendszer?
8. Hogyan változott az integrált információs rendszerek fejlesztési filozófiája?
9. Mik az ágazati információs rendszerek?
10. Mire szolgál az elektronikus kereskedelem? Melyek az előnyei a hagyományos formákhoz képest?
11. Melyek az elektronikus aláírás módszerei?
12. Mi a hitelesítés?
13. Mire szolgál az időbélyegző?
14. Az elektronikus kereskedelemnél hol jelenik meg az állam, mint szereplő?

11. TÉRINFORMATIKAI ALAPOK, ALKALMAZÁSOK

1. A térképek milyen fő csoportjait különböztetjük meg?
2. Mire szolgál a geokód?
3. Mire használjuk a kataszteri rendszereket?
4. Mi jellemzi a digitális térképeket?
5. Mire kaphatunk választ egy földrajzi információs (GIS) rendszertől?
6. Területi kiterjedés szerint milyen információs rendszereket különböztetünk meg?
7. Mit jelentenek a fedvények?
8. Mi a vetületi rendszer? Mi az EOVS?
9. Mi a raszteres adatkezelés lényege?
10. Mi a vektoros adatkezelés lényege?
11. Mi a síkbeli transzformáció?
12. Mi a tesszeláció?
13. Mi a spagetti modell lényege?
14. Melyek a fő jellemzői a topológiai adatmodelleknek?
15. Melyek a térinformatikai rendszerek fontosabb hardver eszközei?
16. A digitalizálás milyen fő módszereit ismeri?
17. Mi a fotogrammetria?
18. Hogyan tároljuk a térinformatikai rendszerek adatait?
19. Mi a precíziós gazdálkodás?
20. Térinformatikai rendszerekben milyen módon integrálhatjuk az adatokat?
21. Miért van szükség adatok interpolálására?
22. Mi a MePAR rendszer alkalmazásának a célja?
23. Melyek voltak a TARNET hálózat kialakításának a fontosabb céljai?
24. Hogyan lehet térinformatikai, távérzékelési eszközökkel termésbecslést végezni? Milyen rendszert fejlesztettek ki erre a célra Magyarországon?

12. DÖNTÉSTÁMOGATÓ RENDSZEREK

1. Melyek a döntés-előkészítési folyamat lépései?
2. Milyen eszközökkel segíthet egy döntéstámogató rendszer az alternatívák közötti választásban?
3. Milyen előnyöket várhatunk a döntéstámogató rendszerektől?
4. Melyek a fontosabb jellemzői az egyéni döntéshozatalnak?
5. Melyek a csoportos döntéstámogató rendszerek fontosabb tulajdonságai?
6. Milyen főbb részekből épülnek fel a döntéstámogató rendszerek?
7. Melyek a modellkezelő rendszer fő részei?
8. Melyek a döntéstámogató rendszerek főbb felhasználási módjai?
9. Mi az üzleti intelligencia?
10. Mire szolgál az OLAP technológia?
11. Milyen döntéstámogató programrendszereket ismert? Jellemeze őket!

13. E-KORMÁNYZAT, E-SZAKIGAZGATÁS

1. Melyek az Önkormányzatokkal szembeni elvárások, amelyek fokozottan igénylik az elektronikus ügyintézés megvalósítását?
2. Milyen alrendszerek, funkciók alakíthatók ki egy e-Önkormányzati rendszerben?
3. Mit jelent az elektronikus aláírás, miért van rá szükség?
4. Mit jelent a hitelesítési szolgáltatás?
5. Milyen agrár szakigazgatási intézményeket ismer?
6. Milyen e-agrárszakigazgatási rendszereket ismer?

14. ÜZEMELTETÉSI KÉRDÉSEK

1. Melyek a segédprogramok legfontosabb típusai?
2. Soroljon fel és mutasson be az Ön által megismert segédprogramok közül 2-3-at (az alkalmazási terület megjelölésével)!
3. Milyen tömörítési eljárásokat ismer?
4. Magyarázza el a tömörítő programok működésének elveit, módszereit!
5. Mit nevezünk vírusnak?
6. Melyek az alapvető vírus-kategóriák?
7. Hogyan védekezhetünk a vírusfertőzés ellen?
8. Mit jelent a „social engineering”?
9. Melyek az informatikában a legfontosabb (adatokat érintő) fenyegető tényezők?
10. Milyen területeket célszerű szabályozni egy „informatikai biztonsági terv”-ben? Melyek a személyazonosság-ellenőrzés különböző szintű módszerei?
11. Miben áll az adatvédelmi törvény jelentősége? Véleménye szerint mennyire illeszkedik/illeszthető az informatika specifikumaihoz?
12. Milyen adatvédelmi megoldásokat ismer?