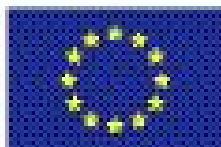


Magyarország célba ér

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



INFORMATIKAI ALAPOK



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

INFORMATIKAI ALAPOK

Szerkesztő:

Dr. Harnos Zsolt
Budapesti Corvinus Egyetem

Dr. Herdon Miklós
Debreceni Egyetem

Szerző:

Dr. Cser László
Budapesti Corvinus Egyetem

Nagyné dr. Polyák Ilona
Debreceni Egyetem

Németh Zoltán
Budapesti Corvinus Egyetem

Lektor:

Dr. Berke József
Pannon Egyetem

Csikós Miklósné
Szent István Egyetem

© DE AMTC AVK 2007

ISBN 978-963-9732-51-3

**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.

1. BEVEZETÉS	7
1.1. KÉPZÉSI KÖVETELMÉNYEK	7
1.2. MI IS AZ A GAZDASÁGINFORMATIKA?	7
1.3. HELYZETKÉP: AZ INFORMÁCIÓTECHNOLÓGIA FORRADALMA	9
1.3.1. Átalakulások az üzletben és menedzsmentben	9
1.3.2. A virtuális vállalatról	10
1.4. AZ INFORMATIKAI TECHNOLÓGIA	11
Irodalomjegyzék	12
Ellenőrző kérdések	12
2. A RENDSZEREK ÉPÍTŐKÖVEI: A PROGRAMOZÁS	13
2.1 A PROGRAMOZÁSI NYELVEK	14
2.1.1. Gépi nyelv és assembly	14
2.1.2. Magas szintű programnyelvek és fordítók	15
2.1.3. Neumann-elvű és nem Neumann-elvű nyelvek	17
2.2 ALAPVETŐ PROGRAMOZÁSI STRUKTÚRÁK	18
2.2.1. Szekvencia	19
2.2.2. Döntés	20
2.2.3. Ciklus	20
2.3 AZ OBJEKTUM-ORIENTÁLT PROGRAMOZÁS ALAPELVEI	21
2.4. AZ OOP-ELVEK NÉHÁNY FONTOS JELLEMZŐJE	22
2.5. STRUKTURÁLT PROGRAMOZÁS ÉS GRAFIKUS OBJEKTUMOK A VISUAL BASIC NYELVBEN	22
2.5.1 A Visual Basic indítása és elemei	23
2.5.2. Beolvasás, kiírás, értékadás	25
2.5.3 Feltételes elágazások	26
2.5.4. Ciklusok	27
2.5.5. Tömbök	28
2.5.6 Grafikus objektumok a VBA-ban	29
2.5.6.1 Tulajdonságok	30
2.6.4.2. Metódusok és események	31
2.6.4.3 Az ActiveX vezérlők	32
Ellenőrző kérdések	34
3. ELEMI ALGORITMUSOK	35
3.1 AZ EUKLIDESZI ALGORITMUS	35
3.2 ÖSSZEGZÉS	36
3.3 TULAJDONSÁG KERESÉSE, KIVÁLASZTÁS, MAXIMUM ELŐÁLLÍTÁSA	37
3.3.1 Negatív szám lineáris keresése	37
3.3.2 Kiválasztás és kiválogatás	39
3.3.3 Maximum kiválasztása	40
3.4 RENDEZÉS	41
3.4.1 Buborék-rendezés	41
3.4.2 Minimum-rendezés	42
3.4.3 További rendezési módszerek	42
Irodalomjegyzék	43
Ellenőrző kérdések	43
4. AZ ADATTÓL AZ ADATBÁZIS KEZELÉSIG	44

4.1. AZ ADATBÁZIS.....	44
4.2. AZ ELEMI ADATSZERKEZETEK	45
4.3. ADATBÁZIS-KEZELŐ RENDSZEREK.....	48
4.3.1. <i>Néhány példa az adatbázis-kezelő rendszerre</i>	49
4.3.2. <i>Adatbázis-kezelő rendszerek felépítése</i>	51
4.4. ADATMODELLEK	53
4.5. RELÁCIÓS ADATMODELL	56
4.6. A RELÁCIÓS ADATBÁZIS KEZELŐ RENDSZEREK A GYAKORLATBAN	59
4.6.1. <i>Adatműveletek, adattábla kezelése</i>	59
4.6.2 <i>Adatbázis tervezése (relációs adatbázis tervezése). Adatbázis létrehozása</i>	61
4.6.3 <i>Lekérdezések. Szűrések a lekérdezésben</i>	64
4.6.4. <i>Úrlapok készítése</i>	68
4.6.5 <i>Jogosultság, biztonság, adatvédelem</i>	69
Irodalomjegyzék.....	72
Ellenőrző kérdések	73
5. SZÁMÍTÓGÉP-HÁLÓZATOK	74
5.1. CÉLJAI, ELEMEL.....	74
5.1.1 <i>Hálózati struktúrák</i>	74
5.1.2. <i>Hálózati hardver</i>	75
5.1.3. <i>Hálózati szoftver</i>	77
5.2 HIVATKOZÁSI MODELLEK	80
5.2.1 <i>Az OSI hivatkozásai modell</i>	80
5.2.2 <i>A TCP/IP hivatkozási modell</i>	81
5.3. A HÁLÓZAT FIZIKAI MEGVALÓSÍTÁSA.....	84
5.3.1. <i>A fizikai réteg</i>	84
5.3.2. <i>Hálózat felépítés (topológiájuk)</i>	87
5.3.3. <i>A hálózatok részelemei</i>	88
5.4 HÁLÓZATOK ÖSSZEKAPCSOLÁSA.....	90
5.5. KLIENS-SZERVER MODELLEK	91
Irodalomjegyzék.....	92
Ellenőrző kérdések	92
6. AZ INTERNET	94
6.1. A HAZAI INTERNET HÁLÓZAT.....	94
6.1.1. <i>Telefonvonalon való kapcsolódás</i>	95
6.1.2. <i>ADSL</i>	96
6.1.3. <i>Wi-Fi</i>	97
6.1.4. <i>Wi-MAX</i>	97
6.1.5. <i>Mobil Internet</i>	98
6.1.6. <i>Kapcsolat két pont között</i>	98
6.2. CÍMZÉSI RENDSZER.....	99
6.3. A DOMÉN NÉV RENDSZER (DNS – DOMAIN NAME SYSTEM)	100
6.4. INTERNET SZOLGÁLTATÁSOK	101
6.4.1. <i>Elektronikus levelezés</i>	101
6.4.2. <i>Állományok átvitele - FTP - File transfer protokoll</i>	102
6.4.3. <i>TELNET</i>	104
6.4.4. <i>A World Wide Web</i>	105
6.4.5. <i>A WWW alkalmazások fejlesztésének eszközei</i>	108
6.4.6. <i>Valós idejű kapcsolattartás</i>	111

6.4.6.1. IP telefon	111
6.4.6.2. IPv6	112
Irodalomjegyzék.....	112
Ellenőrző kérdések	112
7. INFORMÁCIÓS RENDSZEREK.....	113
7.1. INFORMÁCIÓS RENDSZER FOGALMA	113
7.2. AZ INFORMÁCIÓRENDSZEREK FŐ TÍPUSAI	113
7.3. VEZETŐI TEVÉKENYSÉGEK ÉS ADATSZÜKSÉGLETEK	117
7.4. VÁLLALATI INFORMÁCIÓS RENDSZEREK.....	120
7.5. INTEGRÁLT VÁLLALATI RENDSZER FELÉPÍTÉSE	125
7.6. ÁGAZATI INFORMÁCIÓS RENDSZEREK	127
Irodalomjegyzék.....	128
Ellenőrző kérdések	128
8. AGRÁRINFORMÁCIÓ-RENDSZEREK	129
8.1. AZ EURÓPAI UNIÓ AGRÁRINFORMÁCIÓ RENDSZEREI.....	129
8.2. AGRÁRSTATISZTIKAI RENDSZER.....	131
8.3. TESZTÜZEMI RENDSZER - A FADN (FARM ACCOUNTANCY DATA NETWORK)	133
8.4. A PIACI INFORMÁCIÓS RENDSZER.....	135
8.5. INTEGRÁLT IGAZGATÁSI ÉS ELLENŐRZŐ RENDSZER	137
8.6. SPECIÁLIS SZAKTERÜLETI INFORMÁCIÓS RENDSZEREK.....	138
Irodalomjegyzék.....	138
Ellenőrző kérdések.....	138
9. TÉRINFORMATIKA.....	140
9.1. FÖLDRAJZI INFORMÁCIÓS RENDSZEREK	140
9.2. TÉRINFORMATIKAI RENDSZEREK TÍPUSAI.....	141
9.3. A HELYMEGHATÁROZÁS, GEOMETRIAI ADATOK.....	142
9.4. GRAFIKUS ADATMODELLEK	143
9.5. RASZTERES ÉS VEKTOROS ADATOK, ADATKEZELÉS, ALAPMŰVELETEK.....	144
9.6. A TÉRINFORMATIKA FONTOSABB ESZKÖZEI, ALKALMAZÁSA	145
9.7. TÉRINFORMATIKAI RENDSZEREK, AZ ADATOK KEZELÉSE	146
9.8. PRECÍZIÓS GAZDÁLKODÁS.....	147
9.8.1. A precíziós gazdálkodás eszközei.....	147
9.8.2. Adatgyűjtés.....	148
9.9. ORSZÁGOS MEZŐGAZDASÁGI TÉRINFORMATIKAI RENDSZEREK	148
Irodalomjegyzék.....	149
Ellenőrző kérdések	150
10. ADATBIZTONSÁG ÉS AZ ELEKTRONIKUS KERESKEDELEM.....	151
10.1. AZ ADATBIZTONSÁG KÉRDÉSEI.....	151
10.1.1. Veszélyforrások és lehetséges károk	151
10.1.2. Védelmi intézkedések.....	152
10.2. E-ALÁÍRÁS.....	153
10.3. AZ ELEKTRONIKUS ALÁÍRÁS FUNKCIÓJA	155
10.3.1. Az elektronikus aláírás módszerei.....	155
10.3.2. Elektronikus aláírással kapcsolatos szolgáltatások.....	157
10.4. ELEKTRONIKUS KERESKEDELEM, ELEKTRONIKUS ÜZLETVITEL	159
10.4.1. E-business.....	159
10.4.2. Az elektronikus kereskedelem hatása a gazdasági életben	159

10.4.3. E-business modellek	160
10.4.3.1. Administration to Administration (A2A)	161
10.4.3.2. Administration to Business (A2B)	161
10.4.3.3. Business to Business (B2B)	161
10.4.3.4. Business to Consumer (B2C)	161
10.4.3.5. Consumer to Consumer (C2C)	163
10.4.3.6. Peer to Peer (P2P)	164
Irodalomjegyzék	164
Ellenőrző kérdések	164
11. E-KORMÁNYZAT, E-SZAKIGAZGATÁS	166
11.1. E-KORMÁNYZAT FOGALMA	166
11.2. AZ ÖNKORMÁNYZATOKKAL SZEMBENI ELVÁRÁSOK	169
11.2.1. Az ügyintézésrel kapcsolatos, főbb elvárások	169
11.2.2. A szervezettséggel, szervezéssel kapcsolatos fontosabb elvárások	169
11.3. ELEKTRONIKUS KÖZSZOLGÁLTATÁSOK, ÜGYINTÉZÉS	169
11.3.1. Az e-közigazgatás programja	171
11.3.2. Szolgáltató közigazgatás	172
11.4. KÖZIGAZGATÁSI ÜGYFÉL-TÁJÉKOZTATÓ RENDSZER	175
11.4.1. Ügymenet modellek	175
11.4.2. Rendszerfejlesztés	176
Irodalomjegyzék	176
Ellenőrző kérdések	176

1. BEVEZETÉS

A Gazdaságinformatika, mint önálló szak oktatása a 2002/2003-as tanévben kezdődött a Budapesti Corvinus (akkor még Közgazdaságtudományi és Államigazgatási) Egyetemen, mint ötéves hagyományos szak a Gazdálkodástudományi Karon. A 2005/2006-os tanévtől azonban, az országban először, az u.n. bolognai elveknek megfelelően elindult a gazdaságinformatikus BSc képzés is. Jelen tankönyv a lineáris kétlépcsős képzési forma keretében azon alapozó tantárgy tanulását hívatott segíteni, amely felvázolja azt a keretet, amely az informatikai szaktárgyak alapjául szolgál a gazdaságinformatikai képzésnél, valamint az informatikai alapokat jelenti a gazdaságtudományi és közgazdasági képzésnél.

1.1 Képzési követelmények

A tantárgy az informatika hardver és szoftver eszközeit mutatja be, különös tekintettel azok működési elveire, lehetőségeire és korlátaira. Támaszkodik a hallgatók egyre magasabb szintű informatikai ismereteire, amelyet a középiskolából hoznak, de - figyelembe véve annak inhomogenitását – törekszik az ismeretek rendszerezésére.

1.2 Mi is az a gazdaságinformatika?

„A gazdaságinformatika a közgazdasági és az informatikai tudományok ismereteinek egyfajta kombinációja. A fogalom az üzleti szférában és a gazdálkodási területeken kezelt szociotechnikai rendszereket, az emberek és gépek által fejlesztett, illetve kezelt információ- és kommunikációrendszereket jelenti. A középpontban a gazdasági/üzleti feladatok támogatása áll. A gazdaságinformatikai szakembereknek alapos információtechnológiai (helyzetelemzési, rendszerfejlesztési, programozási, adatbázis-kezelési módszerek és technikák), matematikai és gazdasági ismeretekkel kell rendelkezniük, de tájékozottnak kell lenni jogi, szociológiai vonatkozásban is, és idegen nyelveket is ismerniük kell. A képzésben megszerzett ismereteket integrálni kell tudni, és képesnek kell lenni arra, hogy az üzleti tevékenységet számítógépes alkalmazásokkal tegyék hatékonyabbá.” [1.1].

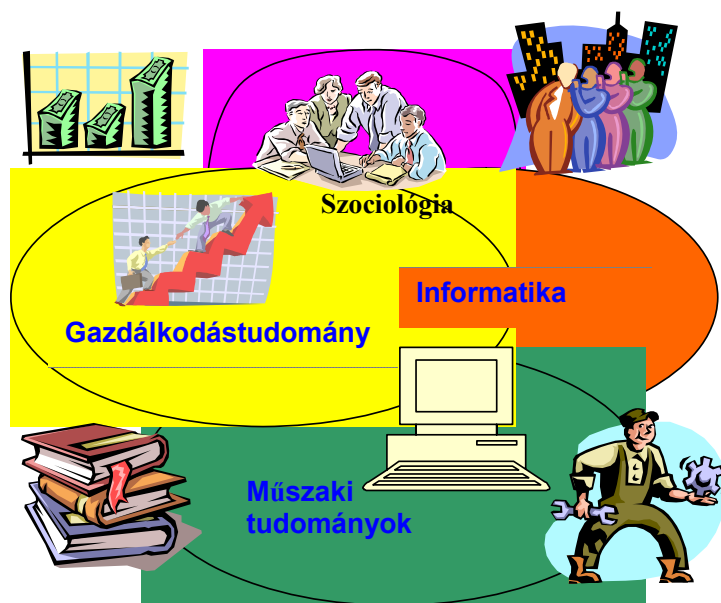
„A gazdaságinformatika az a tudományterület, amely a gazdasági kommunikációs és információrendszerek számítógéppel támogatott feldolgozási módszereivel és technikáival foglalkozik, magában foglalva a közgazdasági, a szociológiai és az informatikai diszciplínákat. A gazdaságinformatika az alábbi részterületek vizsgálatára és elemzésére terjed ki:

- gazdálkodó szervezetek adminisztrációs, kontrolling, diszpozíciós, tervezési, döntési információfeldolgozó rendszerei,
- adatstruktúrák és rendszerfunktionalitás elemzése és modellezése gazdasági alkalmazások szoftverrendszereinek fejlesztése céljából,
- szakértői rendszerek fejlesztése és implementálása a különböző gazdasági ágazatok területén,
- a gazdasági területeken alkalmazandó számítógéprendszerek (hardver, szoftver, gazdasági alkalmazások) kiválasztási kritériumainak meghatározása,
- a korszerű kommunikációs technológiák alkalmazása az üzleti folyamatokban (irodaautomatizálás, elektronikus üzletvitel),
- gazdálkodó rendszerek információrendszereinek fejlesztése és menedzselése,
- az informatikai infrastruktúra hatékony üzemeltetése.” [1.2].

„A gazdaságinformatika a gazdasági rendszereket, a szervezetek gazdálkodását kifejező információrendszerek fejlesztési és üzemeltetési elveivel, módszereivel és technikáival foglalkozik. A gazdasági alkalmazások, illetve alkalmazási rendszerek olyan szoftverrendszerek, amelyek segítik és hatékonyabbá teszik a vállalatok és egyéb gazdálkodó szervezetek gazdasági, üzletviteli feladatainak elvégzését. A GI olyan interdiszciplináris tudományterület, amely a közgazdaságtudomány és az informatika közös ismeretanyagán és kutatási eredményein alapul.” [1.3].

Ennek megfelelően a gazdaságinformatikus szakembernek *képesnek kell lennie*

- az információs társadalom feltétel- és értékrendszerében a valós üzleti folyamatok megértésére,
- az értékteremtő folyamatokat támogató informatikai feladatok menedzselésére,
- az információtechnológia korszerű lehetőségeit kihasználva a szervezetek tudásbázisának és üzleti intelligenciájának a növelésére,
- az infokommunikációs folyamatok és technológiák együttműködésen alapuló modellezésére,
- folyamatok szabályozására és tervezésére,
- a problémák feltárására, a problémátér definiálására, alkalmazások fejlesztésére, működtetésére és a működés elvárt minőségnek megfelelő felügyeletére [1.4.] .



1.1. ábra: A gazdaságinformatika helye a tudományterületek között

Mint az a 1.1. ábrán is látható, a gazdaságinformatika szorosan kötődik a gazdálkodástudományhoz, de támaszkodik a műszaki tudományokra és az informatikára. Miután az informatikai rendszerek és a gazdálkodástudomány elválaszthatatlan a szociológiai vonatkozásoktól, a gazdaságinformatika társadalmi aspektusai sem elhanyagolhatók.

Jelen tankönyv ebben a szemléletben próbálja meg bemutatni a gazdaságinformatika alapjait azzal a feltételezéssel, hogy az itt leírt ismeretek rendszerét ki fogják egészíteni és el fogják mélyíteni a szaktárgyak tanulása során szerzett részletes ismeretek.

2.1 Helyzetkép: az információtechnológia forradalma

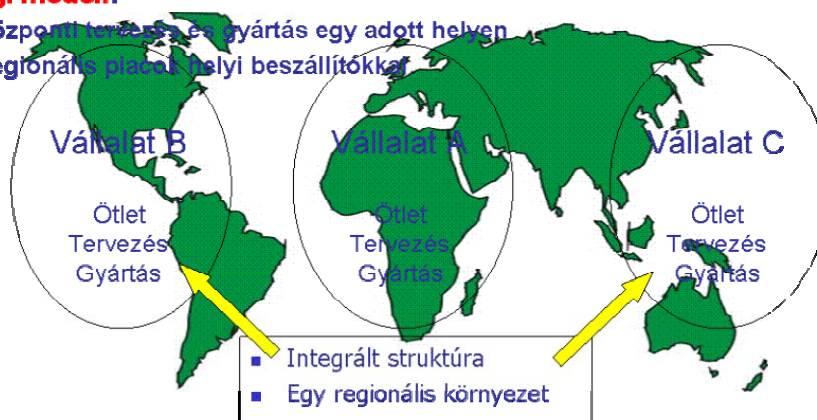
1.1.1. Átalakulások az üzletben és menedzsmentben

Korunk gazdasága óriási átalakuláson megy keresztül. Ennek főbb tényezői az alábbiak:

- Menedzsment és kontroll rendszerének átalakulása,
- Nemzetközi verseny, világpiac fokozott érvényesülése,
- Globálissá váló termelés,
- Globális vásárlási lehetőségek.

A régi modell:

- Központi tervezés és gyártás egy adott helyen
- Regionális piacok helyi beszállítókkal



1.2. ábra: A hagyományos termelési struktúra

Mint az 1.2. ábrán látható, a hagyományos termelési struktúra a regionális környezetre alapozott, termék-orientált vállalatokra épült. Némi egyszerűsítéssel ez úgy írható le, hogy pl. az autógyárban a gyártelepen belül folyt a gyártmányfejlesztés, a gyártásfejlesztés és az alapvető gyártási műveletek (öntés, kovácsolás, hidegalakítás, lemezalakítás, forgácsoló megmunkálás, festés, szerelés stb.), a beszállítók csak a speciális szerelvényeket (pl. autóvillamossági elemeket) biztosították. Kimondható általában, hogy a vállalatok a regionális struktúra mellett elsősorban a termékek szerint szerveződtek. Láthatóvá vált azonban, hogy az egyes vállalatok korántsem rendelkeztek azzal a technológiai háttérrel (többnyire), amely az ilyen szerveződésű termelés gazdaságosan fenntarthatóvá tette volna.

A verseny olyan technológiai fejlesztéseket tett szükségessé, amelyekkel egymagában egy – még oly nagy – vállalat sem tudott lépést tartani, ezért a termelési struktúra elsősorban a technológiák köré szerveződött. Pl. a kovácsoló-gyár nemcsak egy, hanem több (esetleg egymással is versengő) autógyárat szolgált ki egyre magasabb technológiai színvonalú termékekkel. Ez azt jelenti, hogy a végtermék kibocsátó vállalat egyre inkább csak rendszerintegrátorként működik. Az pedig, hogy ezek a beszállítók területileg hol helyezkednek el, a kommunikáció és a szállítás korszerű lehetőségei mellett már egyre kisebb szerepet játszik. Az alrendszereket gyártó kapacitások telepítésénél a munkaerő képzettsége és olcsósága, valamint az infrastruktúra fejlettsége nagyon nagy súllyal esik latba (1.3. ábra).

Ehhez járul még az egyes kiegészítő tevékenységek (pl. szállítás, marketing, szervíz stb.) kiszervezése a vállalat alapvető struktúrájából (l. 1.4. ábra)..

Az új modell:

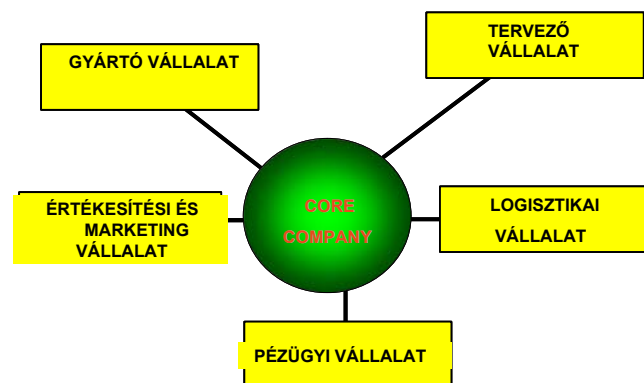
- A világpiacon egyre versenyképesebb
- Sajátos szakértelemmel rendelkező partnerek
- A helyi piac adottságainak és költségvetésének megfelelő gyártás



1.3. ábra: A globalizált termelés alap gondolata

1.1.2. A virtuális vállalatról

Így – lényegében - elérkeztünk a jelenlegi globális termelés alapvető szerkezeti egységéhez, a *virtuális vállalathoz*, amely a termék integrációját és kibocsátását szervezi és valósítja meg, s működését két tényező fogja össze: az *informatika* és a *logisztika*.



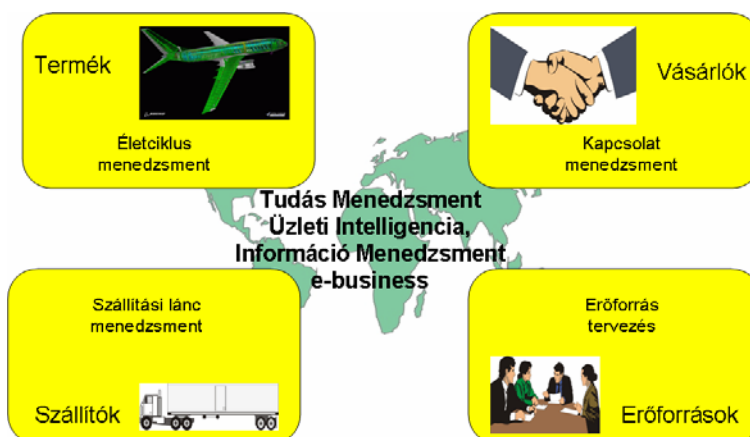
1.4. ábra: A virtuális vállalat gondolata

Az informatika és a vele integrált kommunikációs technika (számítógépi hálózatok) a vállalat belső szerkezetét is átalakították. A vállalati szervezet hagyományos pillérei (a termék, a vásárlók, szállítók és erőforrások) továbbra is megmaradtak (1.5. ábra), de bonyolultsági szintjük miatt új informatikai támogatást kellett bevezetni.

Ez az informatikai rendszer fokozatosan átszötte a teljes működést és a szervezetet integráló alapvető elemévé nötte ki magát (1.6. ábra).



1.5. ábra: A szervezet „tartóoszlopai”



1.6. ábra: A szervezetet integráló informatikai rendszer szegmensei

1.4. Az informatikai technológia

A korszerű informatikai technológia – mint az ismeretes – az alábbi fő témakörökre támaszkodik:

- hardver,
- szoftver,
- adattárolás,
- kommunikáció,
- számítógépi hálózat.

Jelen bevezető tantárgy tematikája úgy épül fel, hogy a később részletesen (önálló tantárgy keretében) elsajátítandó ismeretek összefüggéseit megalapozza. Ezért az elkövetkező fejezetek elsősorban rendszerező jellegűek, s bemutatják azon összefüggéseket, amelyek a szaktárgyakban már evidenciaként jelennek meg. Az elkövetkező fejezetek témakörei az alábbiak:

- A rendszerek építőkövei (a programozás alapjai)
- Az alap algoritmusok

- Az adattól az adatbázisig
- Számítógép hálózatok
- Az internet
- Az információrendszerek a vállalatnál
- Agrárinformációs rendszerek
- Térinformatika
- Adatbiztonság
- e-kormányzat

Irodalomjegyzék

- 1.1. Wirtschaftsinformatik, Prospekt der Johannes Kepler Universität, Linz,
- 1.2. Zilahi-Szabó: Wirtschaftsinformatik, Oldenburg Verlag,
- 1.3. Peter Mertens: Wirtschaftsinformatik, Springer Verlag 2004.
- 1.4. Gazdaságinformatikai BSc Képzés Akkreditációs anyaga, Budapesti Corvinus Egyetem, 2003.

Ellenőrző kérdések

Melyek a gazdaságinformatika főbb részterületei?

Mit értünk az alatt, hogy a gazdaságinformatika interdiszciplináris tudomány?

Jellemezze a virtuális vállalatot!

2. A RENDSZEREK ÉPÍTŐKÖVEI: A 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 gyűjtőkategóriaként, 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.

Neumann János (1903-1957) által az 1948-ban megfogalmazott követelmények a mai napig meghatározzák a számítógépek többségének architektúráját és működését. Az azóta Nemann-elv néven a tudományba bevonult alapvető gondolatok az alábbiak:

- a számítógép legyen teljesen elektronikus,
- a kettes számrendszert használjuk alapul,
- mind az adatok, mind a programutasítások azonos alakban a gép belső memóriájában kerülnek tárolásra,
- a program végrehajtás belső állapotok sorozataként automatikusan történik,
- a számítógépnek legyen logikai műveletek végrehajtására is alkalmas egysége.

(A teljesség kedvéért megemlítjük, hogy Neumann János a számítógépi rendszerek megbízhatóságával kapcsolatban megfogalmazta azt a tételt, ami a jelenlegi repülési és űrkutatási rendszerek informatikájának alapja, amely szerint lehetséges olyan rendszereket létrehozni, amelyek megbízhatósága nagyobb, mint elemeinek megbízhatósága).

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 hardverköltiségeknél. 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 fejlesztése¹.

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 nyelv környezetét felhasználva áttekintjük a legfontosabb gyakorlati programozási ismereteket.

¹ Gondoljunk például a Windows-sorozatra, amelyek kernelje az 1995 és 2002 között kiadott verziókban alig változott.

2.1 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 BASIC2 és Pascal nyelveket.

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.

2.1.1. Gépi nyelv és assembly

A Neumann János által megfogalmazott 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.

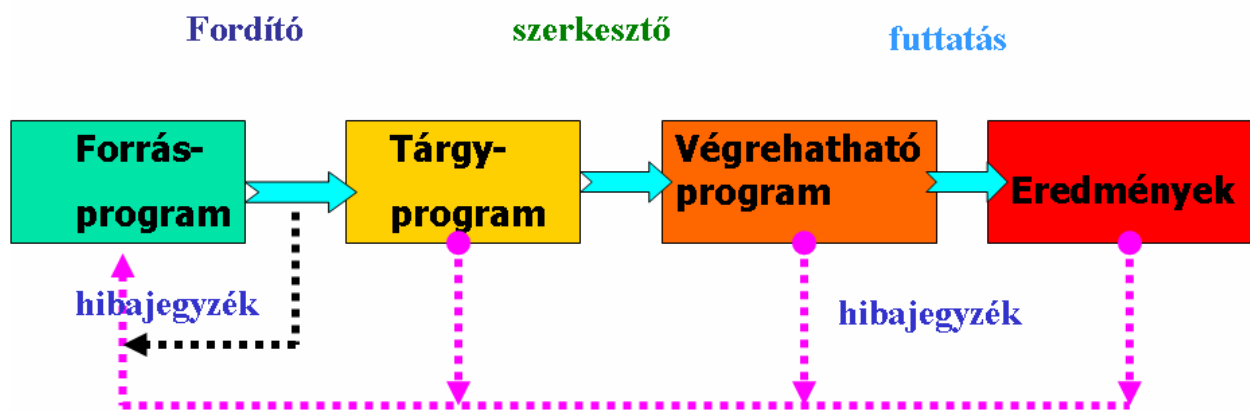
Bár a gépi kód képes kihasználni a számítógép minden adottságát, megkívánja a számítógép teljes ismeretét, a fizikai felépítését is beleértve – nem meglepő, hogy gyakorlatilag az ötvenes évek óta nem vagy nagyon ritkán használják. Az egyes utasítások könnyebb megfogalmazására, jobb olvashatóságára és 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 szerezhetők, 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.

2.1.2. 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, szabadon 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

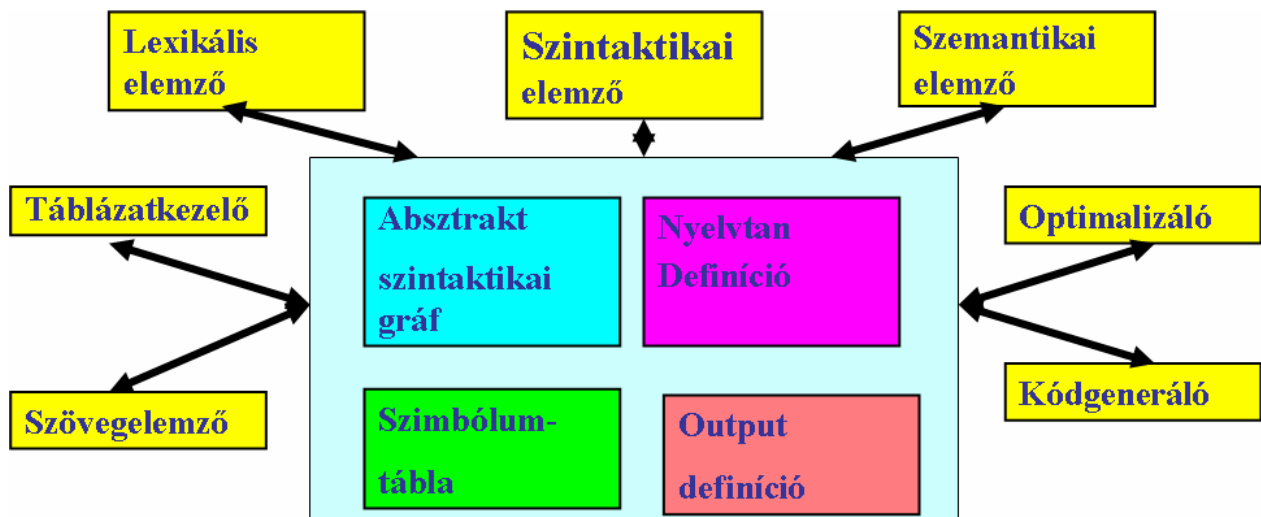
Az **értelmezőprogram** (interpreter) soronként halad a forráskódban, és az egyes sorok feldolgozása után azonnal végrehajtja az utasítást, tehát a programunk gépi kódú megfelelője soronként áll elő. Emiatt a szintaktikai hibákról itt nem jegyzék készül a fordítás után, mint a compilernél, hanem azok 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) is³.

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.

³ 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.



2.2 ábra: Egy komplex fordítórendszer

2.1.3. Neumann-elvű és nem Neumann-elvű nyelvek

A magas szintű nyelvek mindegyik változatában valamilyen fokú absztrakció érvényesül – a gépi kód „nyers” lépéseihez 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 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 változatai⁴, 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.

⁴ 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.

- 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.

2.2 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 lehetségessé 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 változathoz 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. Dijkstra 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.

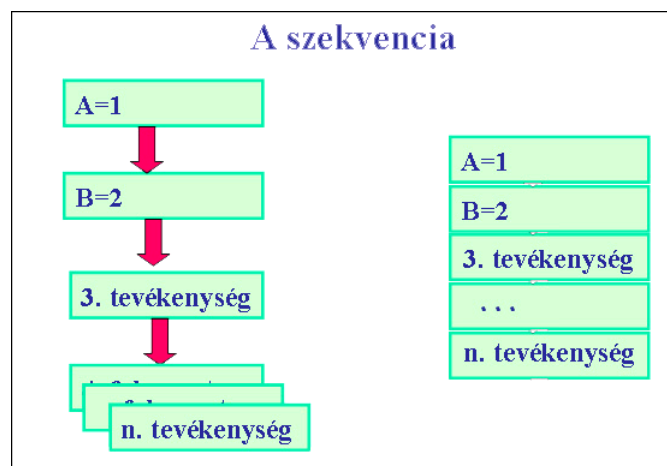
5 Edsger Wybe Dijkstra (1930 -2002) holland tudós

- **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, gyakorlati példákkal a 2.4 fejezet szolgál majd. Egyúttal bemutatjuk az algoritmusok vizuális megadási formáinak (folyamatábra, struktogram) az adott alapstruktúrákhoz rendelt szimbólumait⁶.

2.2.1. 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.



2.3 ábra: A szekvencia

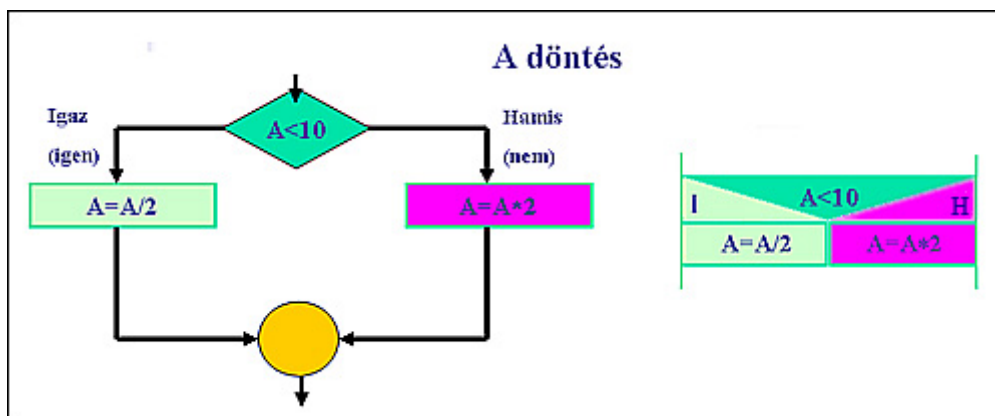
Az algoritmusok grafikus illusztrálására alkalmazott két fő módszer látható a 2.3 ábrán, a baloldali folyamatábrának, a jobboldali struktogramnak hívjuk. A folyamatábra az egyes alapstruktúrákhoz különféle alakzatokat rendel, az egyszerű utasításokat általában téglalap vagy paralelogramma szimbolizálja. 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 értékkadásra, 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.

⁶ Itt jegyezzük meg, hogy a folyamatábrák elemeire nincs konkrét szabály, többé-kevésbé általánosak azonban az alábbi megfeleltetések: bekérés/kiírás/értékkadás: téglalap vagy paralelogramma, feltételes elágazás: elforgatott négyzet – utóbbit a ciklusok feltételének jelzésére is használják. Az egyes elemeket folytonos, egyenes vonallal vagy nyilakkal kötjük össze.

2.2.2 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.



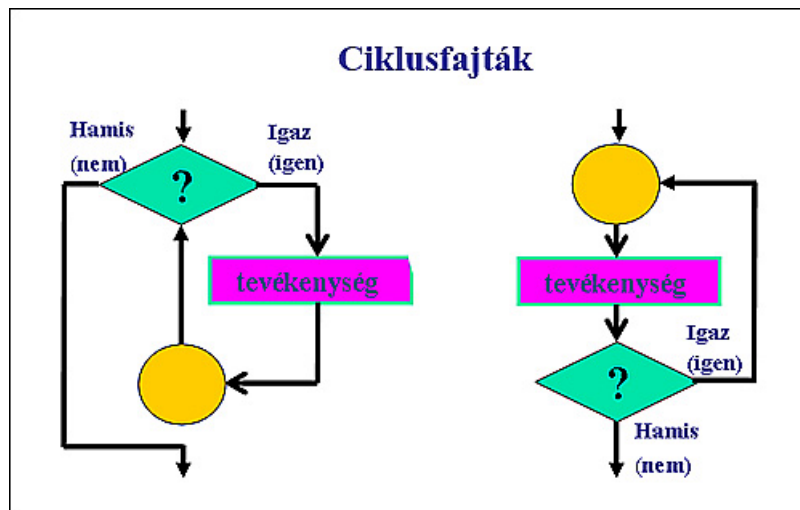
2.4 ábra: Feltételes elágazás

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 ágon 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 2.4 fejezetben látunk rá példát.

2.2.3. 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.



2.5. 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 háttesztelő ciklus utasításai – ha nem használunk ugrásokat – legalább egyszer biztosan végre lesznek hajtva, míg az előtesztelőnél 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 előtesztelő 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 ismételjük meg a ciklusmag utasításait.

2.3 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.

2.4. 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 zárás elve* (egységbe zárás 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*⁷. Amikor a hasonlóság azt jelenti, hogy különböző 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.

A 2.1.3 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#⁸ 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 következő, Visual Basicről szóló fejezetben, a grafikus objektumok kapcsán.

2.5. Strukturált programozás és grafikus objektumok a Visual Basic nyelvben

A Visual Basic (VB) fejlesztőrendszer őse az egyszerűségéről méltán híres BASIC nyelv. A Microsoft által fejlesztett utód (a Visual Studio fejlesztői termécsomag egyik elemeként) professzionális, önálló programozási nyelv, de jelentőségét nagyrészt az Office programcsomag tagjaihoz illesztett változatai adják. A nyelv előnye, hogy Microsoft termék, és mivel az Office szövegszerkesztője és számológépe, azaz a Word és az Excel széles körben elterjedt irodai segédeszközök, ezek makrónyelveként a Visual Basic is könnyen hozzáférhető a felhasználók széles köre számára. Ebben a fejezetben célunk az alapvető programozási struktúrák és egyes grafikus objektumok szemléltetése, ezért az Office-vonatkozásokkal csak röviden foglalkozunk, a fejlesztői környezet elindításának módjára

⁷ Több alaposztály tulajdonságait is egyesítheti egy származtatott osztály, ekkor *többszörös öröklés*ről beszélünk. Megjegyezzük, hogy a származtatott osztályokból lehetséges újabbakat 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.

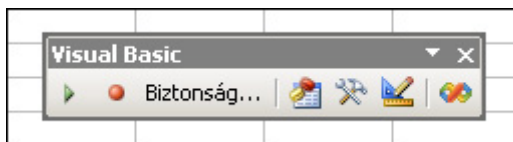
⁸ c# kimondva: c-sharp

szorítkozva⁹. Ugyanezen okokból nem tárgyaljuk a nyelv önálló, Visual Studio-beli formáját sem.

A Word dokumentummal vagy Excel-munkafüzettel párhuzamosan futó *Visual Basic for Applications* (VBA) programozási környezet a VB nyelv szabályait alkalmazza, és a VB vezérlői mellett az Office objektumait (bekezdések, munkalapok, cellák, diagramok stb.) is kezeli. A VB-alkalmazások eseményvezéreltek: a dokumentumokhoz írt programrészek (makrók) sorrendje nem rögzített, egymástól függetlenek is lehetnek, általában egy adott Office-beli esemény bekövetkeztével indul a végrehajtásuk.

2.5.1 A Visual Basic indítása és elemei

A Word, Excel, Powerpoint kortárs (1997 utáni) változataiban az egyik eszköztár a Visual Basic nevet viseli, többek között a makrók rögzítésére, lejátszására és a Visual Basic Editor indítására szolgáló ikonokat tartalmaz. A makrók fogalmával már az assembly kapcsán találkoztunk, itt VB-nyelven rögzített műveletsorozatokat jelentenek, segítségükkel programozással vagy anélkül automatizálhatunk olyan lépéseket, amelyeket újra meg újra végig akarunk hajtani például egy táblázat rendezése, formázása vagy akár értékadások, számítások során.



2.6 ábra: Visual Basic eszköztár

Az eszköztár negyedik elemére kattintva indul a Visual Basic szerkesztőablak, amelyre gyakran VBA-IDE¹⁰ rövidítéssel hivatkozunk. A 2.7 ábrán látható IDE egy Excellel fut párhuzamosan, és az itt írt programok Excel objektum-hierarchiájának megfelelően szervezhetők modulokba – ez alapján beszélhetünk objektum-típusú (munkalapi modulok, űrlap-modulok – utóbbira példa a képen látható *UserForm1*, egy párbeszédablak tervező nézetben) és standard (azaz általános, valamennyi munkalapról hozzáférhető kódot tartalmazó) programmodulokról. E modularizáció a programjaink fizikai strukturálásának eszköze, míg a logikai strukturálásról a VB-nyelvben *eljárások* és *függvények* gondoskodnak. Az eljárások és függvények tehát egy program részekre bontásának eszközei, de maguk is különálló programoknak tekinthetők. Saját azonosítóval rendelkeznek, deklarációjuk általános formában:

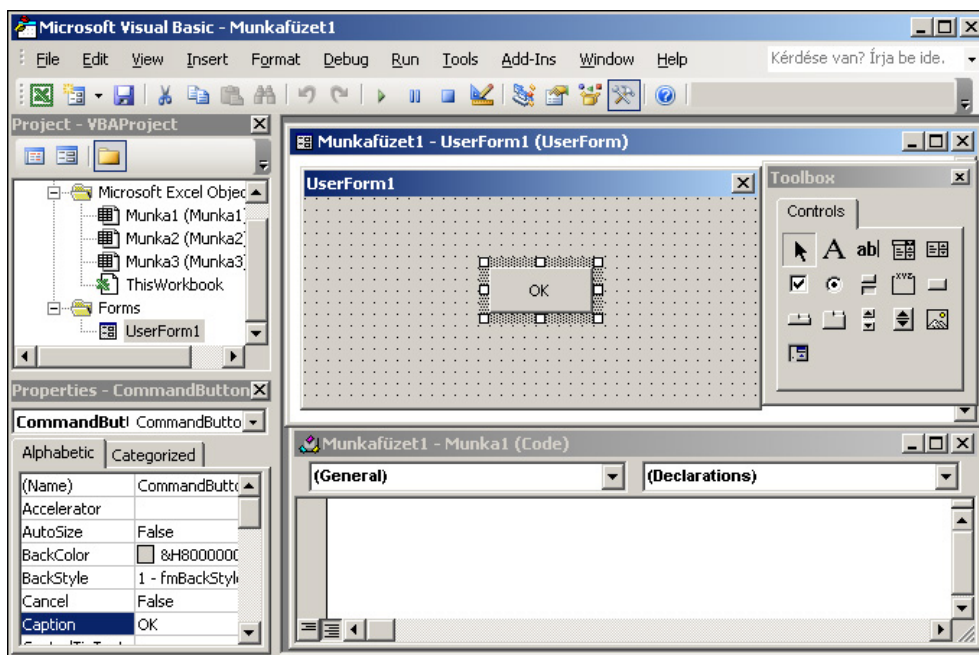
Sub {eljárás neve} ({paraméterek})

Function {függvény neve} ({paraméterek}) *As* {adattípus}

Az eljárások és függvények lezárására az *End Sub*, illetve az *End Function* utasítások szolgálnak

⁹ Megjegyezzük, hogy a linux-platformra készült openoffice.org irodai szoftvercsomag is ismeri a visual basic nyelvet, képes az office-dokumentumokhoz írt vb-programok fogadására. Bizonyos funkciók viszont csak akkor működnek, ha windows-környezetben fut az openoffice.

¹⁰ integrated development environment: integrált fejlesztőkörnyezet. A programkód írása mellett grafikus elemek tervezésére is alkalmas.



2.7 ábra: VBA IDE

Információ ideiglenes tárolására a programozásban változókat használunk. A változókat tehát legkönnyebben tárolórekeszekként képzelhetjük el: valamilyen számértéket, szöveget tárolnak. Olyan azonosítók, amelyekhez a VB ezt a tárolt információt rendeli. A programkódba történő bevezetésüket *deklarációnak* nevezzük, e deklaráció *helye* határozza meg a változó élettartamát, a *módja* pedig azt az adattípust, amelyet tárol. A legfontosabb adattípusok a tárolt információ jellege alapján:

- Numerikus típusok
 - *Integer*: egész számokat tárol a $[-32768 ; 32767]$ tartományon
 - *Long (Long integer)*: egész számokat tárol a $[-2^{31} ; 2^{31}-1]$ tartományon
 - *Single*: valós számokat tárol, a szám abszolút értéke $1,401298 \cdot 10^{-45}$ és $3,402823 \cdot 10^{38}$ között lehet (egyszeres pontosságú lebegőpontos érték)
 - *Double*: valós számokat tárol, a szám abszolút értéke $4,94065645841247 \cdot 10^{-324}$ és $1,79769313486231 \cdot 10^{308}$ között lehet (kétszeres pontosságú lebegőpontos érték)
- *String*: szövegek tárolására alkalmas, $0-2 \cdot 10^9$ karakter terjedelemben.
- *Boolean*: logikai változó, a *TRUE / FALSE* (igaz / hamis) logikai értéket tárolja.
- Objektum11
 - egy létező objektumtípusként deklaráljuk a változót
 - *Object*: általános objektumtípus

A változók deklarációját praktikusán az eljárások vagy modulok elejére helyezzük; lokális (eljárás-szintű) változók deklarálásának utasítása a *Dim*, modul-szintű változókat vezet

11 Ezt a változótípust főleg akkor használjuk, ha helyettesíteni akarjuk vele valamelyik (esetleg több) grafikus objektumot, amely(ek)re hivatkozni hosszú vagy bonyolult, illetve ha még nem létező objektumok bevezetése a célunk.

be a *Private*, globális változókat a *Public* kulcsszó. A deklarálás formája például egy „szov” nevű szöveges változó esetében a következő:

```
Dim szov As String
```

Az eljárások, függvények és változók neve bármilyen karaktersorozat lehet, a következőkre azonban ügyelni kell:

- számmal nem kezdődhet,
- szóköz, írásjel, műveleti jel, szimbólum nem szerepelhet benne, az alulvonás (_) megengedett,
- nem egyezhet meg a VB kulcsszavaival vagy más változók, eljárások, függvények nevével,
- az ékezetes karakterek megengedettek, de biztonsági és kompatibilitási okokból kerülendők,
- kis és nagy betűk között a VB nem tesz különbséget.

2.5.2. Beolvasás, kiírás, értékadás

A 2.2.1-es pontban tárgyalt szekvencia alapstruktúra három utasítástípusa a beolvasás, kiírás és értékadás. Adatok képernyőre való kiírásának legegyszerűbb eszköze a VB-ben az *üzenetablak*. A *Message Box* egy speciális objektum, amelynek a programkódban adjuk meg a paramétereit. Aktiváló parancsa a rövidítése: *MsgBox*, teljes szintaxisa

```
MsgBox ( {üzenet} , {gombbeállítások} , {cím} , {help-fájl} , {kontextus} )
```

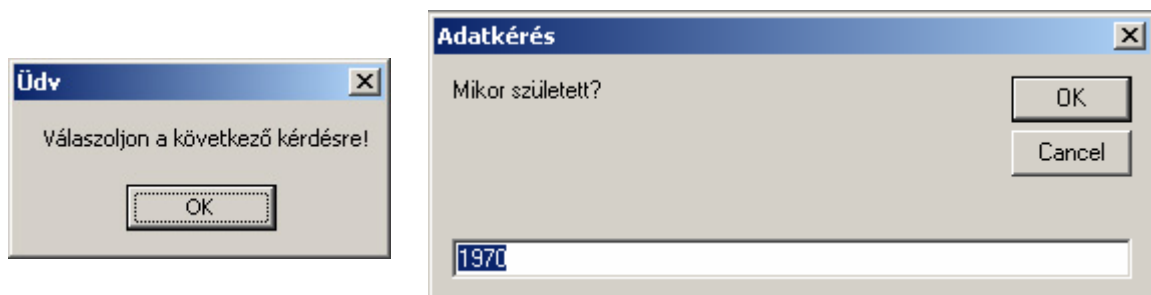
A *Message Box* valójában egy függvény, amelynek felvett értéke attól függ, hogy az OK, CANCEL stb. közül melyik gombot választotta a felhasználó.

Beolvasáskor információt juttatunk egy változóba, ez származhat grafikus objektumokból (a VBA esetében pl. akár az Excel egy cellájából is), de létezik rá egy a *Message Box*hoz hasonló speciális függvény is a VB nyelvben, amelynek neve: *InputBox*, szintaktikája:

```
InputBox ( {üzenet} , {cím} , {kezdőérték} , {X-koord.}, {Y-koord.}, {help} )
```

Például:

```
Dim int_ev As Integer  
MsgBox "Válaszoljon a következő kérdésre!", vbOkOnly, "Üdv"  
int_ev = InputBox ("Mikor született?", "Adatkérés", "1970", 4958, 4665 )
```



2.8 ábra: Message Box és Input Box

A példa utolsó utasításában látható, hogy az értékadás eszköze a VB-ben az egyenlőségjel, az utasítás formája pedig a következő:

{változó} = {érték}

A jobboldalt megadott vagy kiszámított érték kerül a bal oldali változóba (vagy konstansba). A jobb oldalon természetesen nemcsak általunk megadott szám, szöveg vagy dátum állhat, hanem más változó is.

2.5.3 Feltételes elágazások

Az Input Box segítségével megszerzett információt fel kell dolgoznunk, hogy hasznát vehessük. Aszerint, hogy bizonyos feltételeknek eleget tesz-e a kapott érték vagy sem, több szála bonthatjuk a programot. A megfelelőség vizsgálatának helye a program egy csomópontja, egy feltételes elágazás kiindulópontja. Az Input Box példájánál maradva például megvizsgálhatjuk, hogy a felhasználó által megadott évszám elfogadható-e értelmes válaszként. Az ilyen elágazások „Ha... Akkor... Egyébként...”- szerkezetűek, a Visual Basic megfelelő kulcsszavai:

If {feltétel} Then {utasítás} Else {utasítás}

A feltétel maga olyan, mint egy boolean változó, amely az IGAZ vagy HAMIS értékeket veheti fel. Az *Else*-ág elhagyható, ekkor a feltétel nem teljesülése nem jár következményekkel. Amennyiben az egyes ágakon több utasítás követi egymást, akkor többsoros *elágazás-blokk* keletkezik, amelynek lezárása az *End If* utasítás. Előfordulhat azonban, hogy kettőnél több ágra kell bontanunk a programot, ekkor a hamis ágot bontjuk ketté, az *ElseIf* kulcsszó használatával:

```
If {feltétel} Then  
    {utasítások}  
Elseif {feltétel} Then  
    {utasítások}  
Else  
    {utasítások}  
End If
```

Az *ElseIf* ismételt közbeiktatásával még számtalan ilyen alelágazást hozhatunk létre, más kérdés, hogy ez a módszer a probléma túlbonyolításához vezethet. Nem célszerű halmozni az egyezőségi feltételeket, például:

If a = 5 ... ElseIf a = 6 ... ElseIf a = 7 stb.

Erre ugyanis létezik egy másik elágazási forma, a *Select Case*-típus. Ennek lényege pont az, ami az előző, kerülendő példában látható, azaz egy változó vagy kifejezés értékei szerint bontja szálakra a programot (ezért nem minden *If – ElseIf – Else* elágazás írható át *Select Case* formájúra, hanem csak azok, amelyeknél minden egyes feltétel *ugyanarra* a változóra vonatkozik). A *Select Case* szintaxisa:

```
Select Case { változó v. kifejezés }  
    Case { 1. érték }
```

```

        { utasítás(ok) }
    Case { 2. érték }
        { utasítás(ok) }
    [...]
    Case Else
        { utasítás(ok) }
End Select

```

Ha az adott változó aktuális értéke megegyezik az első *Case*-t követő értékkel, akkor a következő sor(ok)ban levő utasítás(oka)t végrehajtja a VB, majd az *End Select* utáni utasításra ugrik. A hét napjai vagy a hónapok a *Select Case* tipikus alkalmazási lehetőségei: ekkor 7, illetve 12 ágból választunk pontosan egyet.

A *Case Else* kulcsszavakkal jelölt ág utasításait akkor hajtjuk végre, ha egyik megelőző ágba sem léptünk be a program futásakor, vagyis ha az egymás után felkínált feltételek egyike sem teljesült. Ez szolgálhat hibajelzésre is: ha egy változónak csak bizonyos konkrét értékeit fogadjuk el (és ezen értékektől függ a program további menete), egy *Case Else*-beli üzenettel jelezhetjük, ha a változó értéke egyik várt értékkel sem egyezik meg – vagyis feltehetően hibásan lett megadva egy bemenő adat.

2.5.4. Ciklusok

Gyakran előfordul, hogy egyes utasításokat egymás után többször is végre kell hajtani a program során. Például ki akarunk írni egy számsort a képernyőre, az egyszerűség kedvéért most álljon ez a számsor a 1 és 100 közötti négyzetszámokból! Külön-külön írjuk ki őket, ezért szükségünk lesz 10 Message Boxra, ami valamilyen trükk nélkül 10 sornyi kódolást jelentene. (Ez még nem tűnik olyan veszélyesen soknak, de 100-szor ugye már nem vállalnánk ugyanannak a begépelését?) Ehelyett egy ciklus szerkesztésével a kiíró utasítást csak egyetlen alkalommal adjuk meg, és a kiírandó helyére rendre a 1 és 10 közötti számok négyzetét helyezve a gépre bizzuk a munka javát.

A Visual Basicben kétféle módon írhatunk ciklust: az első esetben megadjuk, hogy hányszor kell ugyanazt az utasítást végrehajtani, míg a második típusnál az ismétlést logikai feltételekhez kötjük. Az első típust, amelyet gyakran *számlálós ciklus*nak neveznek, **For-ciklus**nak, a másodikat **Do... Loop-ciklus**nak nevezzük. Az újra meg újra végrehajtandó utasítások halmaza a *ciklusmag*.

A For-ciklusok kedvelt olvasata a „For i megy 1-től n-ig”, ahol az *i* a ciklus számlálója, az 1 a számláló kezdeti, az *n* pedig az utolsó értéke. A ciklusmag minden egyes végrehajtása során eggyel növekszik a számláló értéke, amennyiben nem adjuk meg külön a lépésközt – így nyilvánvaló, hogy ekkor a ciklusmag utasításait *n*-szer hajtja végre a program. A For-ciklusok korrekt formája:

```

For {számláló} = {kezdő érték} To {záró érték} Step {lépésköz}
    {ciklusmag}
Next {számláló}

```

A lépésköz megadása nem kötelező, megadható törtszám vagy negatív szám is, utóbbi esetben természetesen a záró érték kisebb vagy egyenlő kell legyen, mint a kezdő. A számlálónak nem kell felvennie magát a záró értéket, de ha az értéke meghaladja azt, akkor a

ciklus véget ér. A fejezet elején említett négyzetszámsor a következőképpen néz ki For-ciklussal:

```
Dim i As Integer
For i = 1 to 10
    MsgBox i^2
Next i
```

A Do... Loop típusú ciklusok esetén nem(csak) az a fontos, hogy hányszor lesznek végrehajtva, hanem az, hogy az általunk szabott feltétel fennáll-e. Ezt a feltételt – mint a 2.2 pontban láttuk – a ciklus elejére és végére is helyezhetjük, ez alapján elől- és hátultesztelő típusokat különböztethetünk meg. Mindkét altípus két további altípusra bontható aszerint, hogy a megadott feltétel teljesülése a ciklus folytatását (amíg teljesül...) vagy befejezését (amíg nem teljesül...) eredményezi.

Az előltesztelő Do... Loop-típusok:

<i>Do While</i> {feltétel}		<i>Do Until</i> {feltétel}
{ciklusmag}	illetve	{ciklusmag}
<i>Loop</i>		<i>Loop</i>

A ciklus folytatódik a While használatakor a feltétel teljesülése esetén, az Until használatakor pedig addig, amíg a feltétel nem teljesül. A hátultesztelő Do... Loop-ciklusok két altípusa:

<i>Do</i>		<i>Do</i>
{ciklusmag}	illetve	{ciklusmag}
<i>Loop While</i> {feltétel}		<i>Loop Until</i> {feltétel}

A következő példa egy hátultesztelő ciklust mutat be, amelynek futása mindaddig tart, amíg a felhasználó le nem állítja. Ehhez kihasználjuk, hogy az üzenőablak egy függvényhívás eredménye, aminek a visszatérési értéke a kiválasztott gombtól függ: ezt az értéket építjük be a ciklus feltételébe. A ciklusmag a példában legyen egy egyszerű adatbekérés:

```
Dim str_nev As String
Do
    str_nev = InputBox("Kérem a nevét:")
Loop While MsgBox("Új bekérés?", vbYesNo, "Folytatás") = vbYes
```

2.5.5. Tömbök

A tömb azonos típusú változók rendezett gyűjteménye, amelyet összességében egy változónév azonosít, az elemeire pedig egy számsorból vett indexszel hivatkozhatunk. E számsornak alsó és felső határa van, a tömb deklarálásakor (kerek zárójelek között) megadható mindkettő, csak a felső vagy egyik sem. Az indexek a két határ közötti egész számokból állnak, akár negatívak is lehetnek, ha meg van adva az alsó határ. Ha csak a felsőt adjuk meg, akkor az alsó határ alapértelmezésben 0. Például:

Dim tomb(10) As Long	'tizenegy (0-10) elemű, Long típusú tömb
Dim tomb(3 To 6) As String	'négy elemű, String típusú tömb
Dim tomb(1 To 10, 2)	'Kétdimenziós tömb, ~ 10 × 3 -as mátrix

Az egydimenziós tömböket egyszerű listaként is elképzelhetjük, ahol a lista minden eleme a memória egy rekesze, a kétdimenziós tömböket legjobban egy mátrixhoz hasonlítani. Több dimenzió esetén az egyes dimenziókhoz tartozó alsó és felső határokat vesszővel elválasztva soroljuk fel a deklarációban.

A tömbök használata nagyszámú (és persze azonos típusú) változó esetén praktikus, mert ciklusok segítségével a határoktól függetlenül könnyen változtathatók az elemek értékei. Tekintsük például a 10×10 –es szorzótáblát, és töltsük be egy kétdimenziós tömbbe:

```
Dim tomb(1 To 10, 1 To 10) As Integer, i As Integer, j As Integer
For i = 1 To 10
    For j = 1 To 10
        tomb(i, j) = i * j
    Next j
Next i
```

Megjegyezzük, hogy tömböket nemcsak változókból, hanem objektumokból is létrehozhatunk, és amennyiben Excelt programozunk, használatuk szükségessége nyilvánvaló: az Excel munkafelülete felfogható egy hatalmas kétdimenziós tömbként, ahol a tömb sor- és oszlopindexei az egyes cellák koordinátái. A cellák közös azonosítója a *Cells*, ami ezúttal maga is objektum, valamennyi cellára érvényes műveletek hajthatók végre a segítségével. Például az adott munkalap "E2" cellájára a következőképpen léphetünk:

```
Cells(2, 5).Select
```

A fenti példákban már egy konkrét objektumra hivatkozunk. A Windows rendszerek használata során néhány elem funkciója az átlagfelhasználó előtt is nyilvánvaló: a gombokra kattintva kezdeményezhető valamilyen utasítás végrehajtása, egyes mezők adatbevitelre szolgálnak stb. Ezen – látásból mindenképpen – ismert elemek használatába nyújt bevezetőt a következő fejezet.

2.5.6 Grafikus objektumok a VBA-ban

Programozói szemmel az *objektumok* osztályként definiált halmazok elemei, amelyek adatokat és eljárásokat egyaránt magukban foglalnak. Az objektumokhoz *tulajdonságok*, *metódusok* és *események* tartoznak:

- A tulajdonságok az objektumot jellemző adatok.
- A metódusok olyan tevékenységek, amelyeket egy objektum végrehajthat.
- Az eseményeket felismerik az objektumok, rájuk reagálva utasításokat adhatunk ki. Az eseményeket előídeázheti felhasználói magatartás (pl. egy billentyű leütése), a programkód vagy az operációs rendszer.

Ez a fejezet néhány gyakran használt vezérlőelem tulajdonságait, azok programkódban történő beállítását, valamint a fontosabb metódusokat és eseményeket mutatja be.

2.5.6.1 Tulajdonságok

A 2.7 ábrán látható fejlesztőkörnyezet *Properties* ablaka tartalmazza az Office programjaiban és makróiban használható grafikus objektumok tulajdonságait. Ezek közül emeljük ki öt fontos kategóriát:

- Megjelenés (Appearance)
 - háttérszín (*BackColor*), határszín (*BorderColor*), karakterszín (*ForeColor*): a *System* fülön 25, a palettán 48 szín közül választhatunk, de megadhatunk egyéni színeket is
 - felirat (*Caption*): alapértelmezésben megegyezik a vezérlő nevével (de nem követi annak módosításait)
- Viselkedés (Behavior)
 - automatikus méretezés (*AutoSize*): TRUE vagy FALSE értéket vehet fel, TRUE érték esetén a felirat, szöveg vagy kép méretéhez igazítja az objektum méretét
 - igazítás (*TextAlign*): balra (1), középre (2) vagy jobbra (3) zárja a felirat szövegét
- Adatok (Data)
 - szöveg (*Text*): szövegdobozok tartalma, listák kiválasztott eleme
 - különféle listabeállítások
- Betűtípus (Font)
 - betűtípus (*Font*): betűtípus, -stílus és -méret beállítására szolgál
- Egyéb (Misc)
 - azonosító (*Name*)
 - engedélyezett (*Enabled*): TRUE vagy FALSE értéket vehet fel, azt adja meg, hogy az objektum válaszolhat-e a felhasználó által generált eseményekre
 - magasság (*Height*), szélesség (*Width*)
 - koordináták (*Left, Top*): az objektum bal felső sarkának távolsága az őt tartalmazó konténer koordináta rendszerének origójától
 - láthatóság (*Visible*): TRUE vagy FALSE értékű lehet
 - cellacsatolás (*Linked Cell*), bemeneti tartomány (*ListFillRange*)
 -

A tulajdonságok – a program futása szempontjából – kezdő értékeit állíthatjuk be a *Properties* ablakban, ugyanezt megtehetjük a programkódban is, de a program futása közbeni módosításra már csak programozással van lehetőség. Az eljárásokban és függvényekben a következő formában hivatkozhatunk a tulajdonságokra:

{objektum azonosítója} . {tulajdonság}

Például tekintsünk egy gombot és egy jelölőnégyzetet:

```
CommandButton1.Width = 2400  
If CheckBox1.Visible = False Then [...]  
    stb.
```

A fentiekben bemutatott „pontosított” forma valamennyi ismertebb objektumorientált programozási nyelvre (így a C++-ra is) jellemző. A pont „operátort” nemcsak objektumok tulajdonságainak és metódusainak megadásakor, hanem objektumok, tulajdonságok hierarchikus láncában is használjuk. Például egy programablak valamely vezérlőjének betűtípusát félkövérre állíthatnánk a következőképpen:

```
Form2.Label1.Font.Bold = True
```

2.6.4.2. Metódusok és események

A metódusokat a programkódban a tulajdonságokhoz hasonló formában hívjuk:

```
{objektum azonosítója} . {metódusnév} ( {paraméterek} )
```

Néhány fontosabb metódus:

- *AddItem*: új elem felvétele a listára
- *Hide*: űrlap vagy dialóguspanel elrejtése
- *Print*: szöveg kiírása az *Immediate* segédablakba
- *Select*: a megadott objektum kiválasztása
- *Show*: űrlap vagy dialóguspanel megjelenítése (és – ha szükséges – betöltése)
-

Például:

```
ComboBox1.AddItem "Schindler"
```

Az utasítás a *ComboBox1* listán elhelyez egy új elemet, a „Schindler” szót.

Amikor egy objektum érzékeli egy esemény bekövetkeztét, automatikusan hívja az eseményhez kapcsolódó eljárást. Az egérekattintás, egy objektum kiválasztása, tartalmának törlése, még az egér céltalan mozgatása is ilyen eseménynek minősül – az eseményorientált programozásban tehát a programok egy részének aktiválása a felhasználói viselkedéstől függ. Az eljárások neve egyfajta kapocs az objektum és a programkód között, mert egyesítik az objektum és az esemény azonosítóit. A két azonosítót az alulvonás karakter köti össze:

```
Sub Label1_Click()  
Private Sub Workbook_Open()
```

Néhány fontosabb esemény:

- *Activate*: dokumentum, űrlap vagy dialóguspanel aktívvá válása
- *Change*: az objektum tartalma (vagy értéke) megváltozik
- *Click*: kattintás az egérrel, az ENTER leütése aktív parancsgomb esetén stb.
- *DoubleClick*: dupla kattintás az egér bal gombjával
- *KeyPress*: egy billentyű leütése

Az alábbi példában a *Click* eseményre reagálva változtatunk a gomb két tulajdonságán és végrehajtunk egy metódust:

```
Private Sub Label1_Click()  
    Label1.BackColor = vbBlue  
    Label1.ForeColor = vbWhite  
    CommandButton1.Select  
End Sub
```

2.6.4.3 Az ActiveX vezérlők

A *címkék* (*Label*) olyan szövegek megjelenítésére használhatók, amelyeken a felhasználó nem módosíthat. Üzeneteket is tartalmazhatnak (például egy makrófolyamat pillanatnyi állapotáról), de feliratként más objektumok azonosítására is használhatók. A háttér- és karakterszín, valamint a keretet megadó tulajdonságokról (*BackColor*, *ForeColor*, *BorderStyle*) korábban elmondottak a címkékre is érvényesek, de legfontosabb tulajdonságuk – természetesen – maga a felirat. A *Caption* hossza elvileg korlátlan, de mindenképpen véges, a megjeleníthető szöveg hossza a címke méretétől és a sortörési beállításoktól függ. Az *AutoSize* tulajdonság IGAZ értékre állítva a szöveg hosszához méretezi a címke szélességét, ha a *WordWrap* is IGAZ értékű (azaz a sortörés a jelenleginél több sorban is engedélyezett), akkor a szélesség és a magasság is megnőhet szükség szerint. (Lásd a 3.3.3 pont példáját!)

A *beviteli mező* (*TextBox*) vezérlő az 2.5 pontból már ismerős lehet: az adatkéréshez használt *InputBox* fehér beviteli sávja is egy ilyen objektum. Háromféle módon kerülhet bele szöveg: a a *Text* tulajdonság beállításával a *Properties* ablakban, közvetlen beírással a munkalapon, illetve egy makró és a felhasználó közreműködésével. Például:

```
Private Sub CommandButton3_Click()  
    TextBox1.Text = "Írjon ide valamit!"  
End Sub
```

A beviteli mezőnek *nincs* felirata, azonosítására legtöbbször egy címkét használunk.

A magyarul *parancsgombként*, *nyomógombként* és egyszerűen *gombként* (nem keverendő az űrlap-vezérlővel!) is emlegetett *CommandButton* vezérlő – ahogy azt a Message Box és Input Box példáján láthattuk – a dialógusablakok állandó kelléke. A formok és dialóguspanelek bezárásán és új események elindításán kívül érvényesítő funkciója is lehet, amennyiben egy felhasználói tevékenység (például szöveges adatbevitel) lezárásakor használjuk. A külseje a Windows-szerthe megszokott „szürke 3D”, ha viszont *Picture* tulajdonságnak értéket adunk, azaz kiválasztunk egy képet, akkor a parancsgomb háttere a kép átméretezett változata lesz. A gomb feliratát a címkékhez hasonlóan a *Caption* tulajdonság adja meg, a karakterek formátuma változtatható.

A *jelölőnégyzetek* (*CheckBox*) az IGAZ / HAMIS, IGEN / NEM választási lehetőséget kínálják fel a felhasználónak. A kipipált vezérlőre kattintva a pipa eltűnik a négyzetből, ismételt kattintáskor pedig újra megjelenik. A pipa meglétét vagy hiányát a váltógombokhoz hasonlóan a *Value* tulajdonság mutatja

A *választókapcsoló* (*Option Button*) nevét nem szó szerint fordították, teljes joggal, mivel a gombokhoz sok köze nincs. Annál több a hasonlóság közte és a jelölőnégyzet között, a tulajdonságaik egytől egyig megegyeznek, mindössze egy dologban térnek el: az egy csoportban levő választókapcsolók közül maximum egy lehet kiválasztva.

A *ListBox* (*listapanel*) egy egyszerű listát jelenít meg, amelyről a felhasználó egy vagy több elemet választhat ki. Ha a listának több eleme van, mint amennyi a méretének megfelelően megjeleníthető, a jobb szélén megjelenik egy gördítősáv. A listát az Excelben csatolt cellák segítségével vagy a makró kódjában az *AddItem* metódus használatával tölthetjük fel elemekkel. A listaelemek sorszámozottak, az első elem sorszáma 0, a másodiké

1 és így tovább. A *Properties* ablakban nem látható és automatikusan beállított *ListCount* tulajdonság azt adja meg, hány elemű a lista. A *ListIndex* tulajdonság értéke megmutatja, hogy a lista hányadik eleme van kiválasztva. A *ListIndex* értéke

- -1, ha egy elem sincs kiválasztva,
- 0, ha az első elem van kiválasztva (stb.),
- *ListCount* - 1, ha az utolsó elem van kiválasztva.

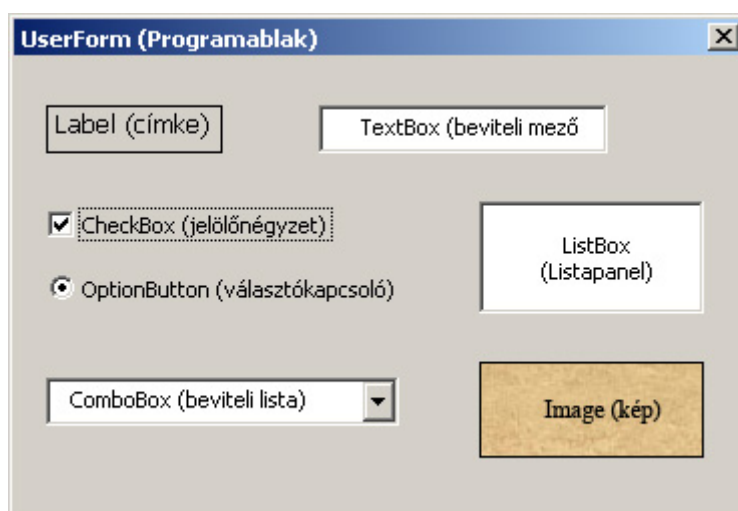
A *ComboBox* (beviteli lista, egyes verziókban *kombi panel*) egy szövegdoboz és egy lista keresztezése, ennek megfelelően a listához képest hangsúlyosabb a *Text* tulajdonsága: ennek tartalma látható a lista fejlécében. Ez eltérhet a lista elemeitől a *Style* tulajdonságtól függően. A *Style* lehetséges értékei:

- 0 – *DropDown Combo*: a szövegdobozban a listelemektől különböző szöveg is szerepelhet
- 2 – *DropDown List*: a listaelemek közül kell választanunk

Az *Image* (ábra, kép) vezérlő legfontosabb tulajdonsága – nevéből adódóan – a *Picture*. Ennek direkt beállítása csak a *Properties* ablak használatával lehetséges, a „3 pont” ikonra kattintva – az így előhívott *LoadPicture* segédablakban megkereshető a megjelenítendő kép. Ha egy makróban akarunk utasítást adni egy kép megnyitására, a *LoadPicture* függvényt kell használnunk:

```
Private Sub CommandButton4_Click()
    Image1.Picture = LoadPicture("C:\kekhalal.jpg")
End Sub
```

Az *Image* vezérlő hatféle formátumot képes fogadni, ezek a következők: bitképek (.bmp), ikonok (.ico), .wmf és .cur képek, .gif és .jpg képek.



2.9 ábra: ActiveX vezérlők

Ellenőrző kérdések

Melyek a programnyelvek generációi?

Mit tartalmaz egy assembly nyelven írt program?

Sorolja fel a programfordítás szakaszait!

Mi a paradigma? Milyen programozási paradigmákat ismer?

Sorolja fel és jellemezze a strukturált programozás alapelemeit!

Mit jelent az, hogy az objektumok fekete doboznak tekinthetők?

Milyen változótípusokat (adattípusokat) ismer?

Mi a különbség az elől- és hátultesztelő ciklusok között?

Mikor célszerű változótömböket alkalmazni?

Mit határoznak meg a Font, Text és Enabled tulajdonságok?

Soroljon fel 4 ActiveX vezérlőt, és adja meg, mikor használjuk ezeket a típusokat!

3. ELEMI ALGORITMUSOK

Az előző fejezetben megismert programozási technikák gyakorlati alkalmazásának illusztrációját adják a következő példaprogramok. Közös bennük, hogy egy-egy összetett probléma, feladat megoldásakor gyakran használt utasítássorozatok, *elemi algoritmusok* – önállóan is futtatható és működő programok tehát, de gyakorlati jelentőségük nagyobb programok komponenseiként fogható meg.

Az algoritmus definíció szerint egy feladat megoldására szolgáló egyértelmű szabályokkal követhető lépések (utasítások) sorozata, amely jól meghatározott bemeneti valamint kimeneti pont(ok)kal rendelkezik. Az elnevezés egy arab tudós, al-Khwarizmi (Bagdadban dolgozott i.sz. 813 – 833 között) nevéből származik; legáltalánosabb jelentése valamilyen *tervszerűség*, így lehet egy épület terve, egy zenemű kottája, vagy ahogy az alfejezetekben látható: egy program leírása.

Példaként keressünk meg egy szót az értelmező szótárban! A lépések a következők:

- Nyissuk ki a szótárt egy véletlenszerű helyen!
- Döntsük el, hogy a keresett szó a most látható oldalakon van-e! Ha igen, akkor készen vagyunk.
- Döntsük el, hogy a keresett szó a most látható oldalak előtt vagy után található! Tekintsük „új” szótárnak az „eddigi” szótár döntésünk által kijelölt részét!
- Ismételjük a lépéseket!

Az alábbi elemi algoritmusokat többféle formában adjuk meg: mondatszerű leírással, illetve egy-egy Visual Basic nyelven írt eljárással, függvénnnyel, néhány esetben folyamatábrával vagy stuktogrammal kiegészítve.

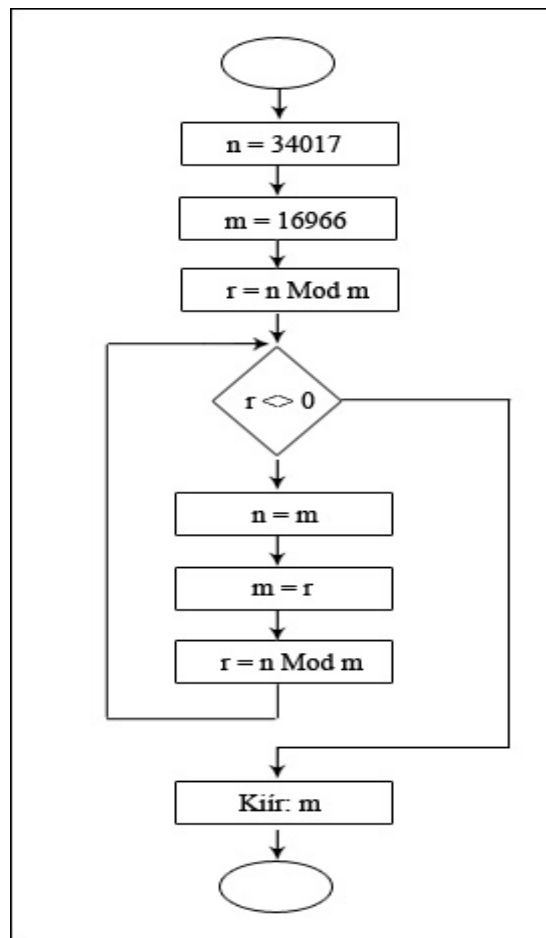
3.1 Az euklideszi algoritmus

Az euklideszi algoritmus két természetes szám legnagyobb közös osztóját keresi meg, osztások sorozatával és azok maradékának vizsgálatával.

- Legyen a két természetes szám m és n (feltételezzük, hogy $m < n$), és legyen r az n/m osztás maradéka.
- Legyen most n új értéke m , m új értéke pedig r . Végezzük el újból az osztást, az új maradék ismét az r lesz.
- Az előző lépést addig ismételjük, amíg a maradék nulla nem lesz. A legutolsó m érték az eredeti két szám legnagyobb közös osztója.

```
Sub Lnko()  
    Dim n As Integer, m As Integer, r As Integer  
    n = 34017  
    m = 16966  
    r = n Mod m  
    Do While r <> 0  
        n = m  
        m = r  
        r = n Mod m  
    Loop  
    MsgBox m  
End Sub
```

'az Lnko kiírása



3.1 ábra: a legnagyobb közös osztó algoritmusa

Példa:

34017 és 16966 legnagyobb közös osztója:

34017 / 16966 maradéka 85

16966 / 85 maradéka 51

85 / 51 maradéka 34

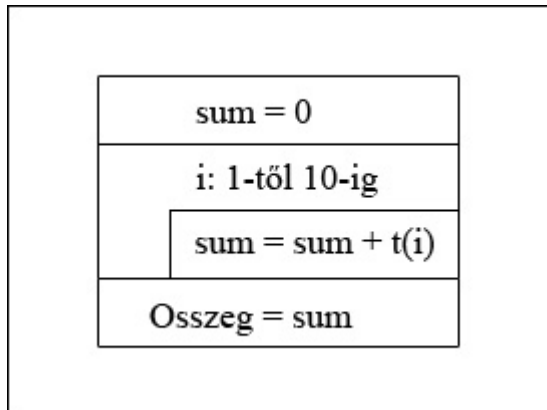
51 / 34 maradéka 17

34 / 17 nincs maradék. A legnagyobb közös osztó: 17.

3.2 Összegzés

Adott egy n elemű numerikus tömb, jelöljük t -vel (a tömb kezdő indexe legyen 1). Feladatunk az elemek összegzése; az összeget egy sum azonosítójú változóban tároljuk. Lépéseink:

- Az összegzés megkezdése előtt 0-ra állítjuk a sum változó értékét.
- Egyenként hozzáadjuk a sum változóhoz a tömb elemeit.
- Az utolsó tömbelem hozzáadása után a sum értéke a keresett összeg.



```

Function Osszeg(t() As Double, n As Integer)
  Dim sum As Double, i As Integer
  sum = 0
  For i = 1 to n
    sum = sum + t(i)
  Next i
  Osszeg = sum
End Function
  
```

3.2 ábra: Összegezés

Mejegyezzük, hogy a 3.1 példával ellentétben ezúttal nem eljárást, hanem függvényt definiáltunk a feladathoz, amely paraméterként kapja a tömb elemeit és elemszámát, és visszatérési értéként szolgáltatja az elemek összegét. A továbbiakban mindkét formát használjuk majd.

3.3 Tulajdonság keresése, kiválasztás, maximum előállítása

Ebben a pontban egy n elemű t tömb elemein értelmezünk egy igaz vagy hamis értéket adó T tulajdonságot. Az első feladat annak eldöntése, hogy van-e a tömb elemei között olyan, amely rendelkezik e tulajdonsággal – ennek meghatározására lineáris keresést használunk. A következő lépésekben már feltételezzük, hogy van ilyen tömbelem, és megállapítjuk, hogy melyik az első ilyen elem, illetve hogy hány elem rendelkezik a T tulajdonsággal. Ez utóbbi feladat módosított változatában a T tulajdonsággal bíró elemeket kigyűjtjük egy új tömbbe. Az utolsó példában numerikus elemek közül kiválasztjuk a legnagyobbat.

3.3.1 Negatív szám lineáris keresése

Legyenek a tömbelemek (darabszámuk tehát n) numerikus típusúak, és a T tulajdonság most legyen az elemek negatív volta. Megvizsgáljuk, hogy találunk-e negatív számot az elemek között. Kétféle módon is eljárhatunk:

(A) változat

- Legyen egy i egész típusú változó értéke 1. Ezt a változót használjuk indexként.
- Az i értékét addig növeljük, amíg az i -edik elem nem negatív, és amíg $i \leq n$.
- A ciklus lefutása után ha $i \leq n$, akkor találtunk negatív elemet. (Ha nincs negatív elem, akkor az i értéke $n+1$!)

(B) változat

- Legyen egy b logikai változó értéke hamis.
- Vizsgáljuk meg az összes elemet! Ha az adott elem negatív, állítsuk a b változó értékét igazra.
- A ciklus végeztével a b változó értéke adja meg, hogy találtunk-e negatív számot.

```

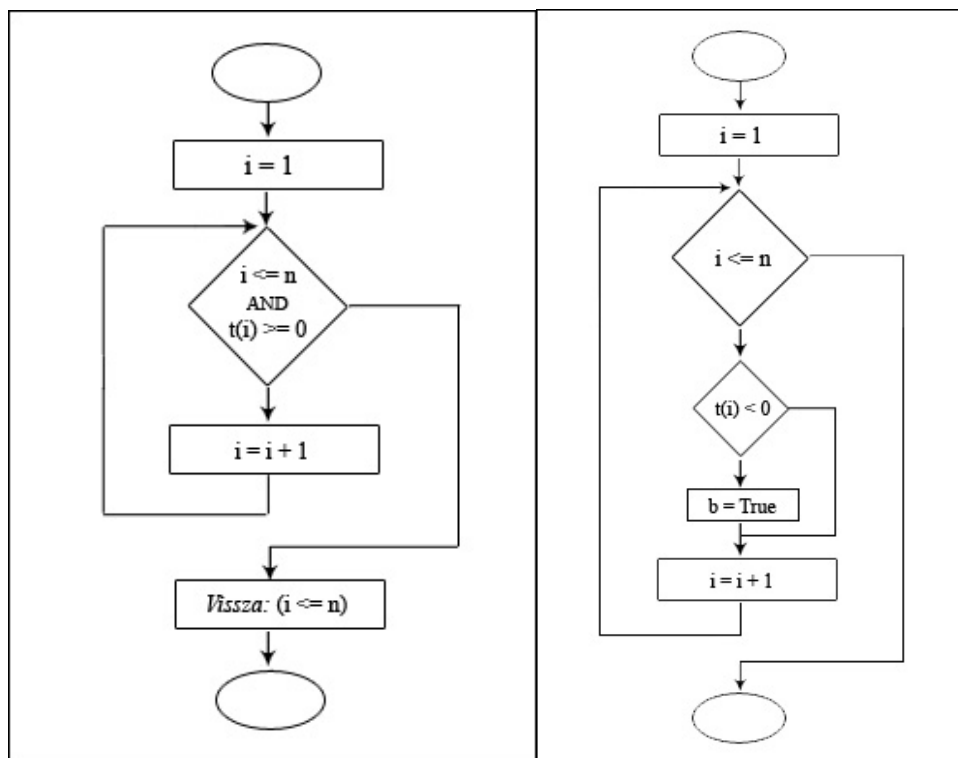
Function NegativA(t() As Double, n As Integer)
  Dim i As Integer
  i = 1
  
```

```

Do While i <= n And t(i) >= 0
    i = i + 1
Loop
NegativA = (i <= n)
End Function

Function NegativB(t() As Double, n As Integer)
    Dim b As Boolean, i As Integer
    For i = 1 to n
        If t(i) < 0 Then b = True
    Next i
    NegativB = b
End Function

```



3.3 ábra: A *NegativA* és a *NegativB* függvény

Megjegyezzük, hogy a függvények mindkét esetben a *True* / *False* (igaz: van negatív / hamis: nincs negatív) logikai értéket adják vissza. Figyeljük meg, hogy a B változatban eggyel több változót használtunk fel, és az A változatban lehetséges, hogy *n*-nél kevesebb ciklusmagismétléssel is eredményre jutunk. Továbbá vegyük észre, hogy a fenti példából minimális módosítással előállítható egy olyan vizsgálat, amely azt mutatja ki, hogy vajon a tömb *valamennyi* eleme rendelkezik-e a *T* tulajdonsággal. Ha a *T* tulajdonság ellentéte nem teljesül egyetlen elemre sem, akkor a *T* valamennyi elemre igaz. A konkrét függvények esetében, ha azt kívánjuk megállapítani, hogy vajon minden elem nemnegatív-e, akkor a függvények utolsó előtti során kell módosítanunk:

NegativA = Not(i <= n)

illetve:

NegativB = Not(b)

A módosítással a függvények az összes elem nemnegativitása esetén a *True* logikai értéket adják vissza.

3.3.2 Kiválasztás és kiválogatás

A *t* tömb elemszáma ezúttal is *n*, de az elemek most karakterlánc (String) típusúak, hosszuk váltakozó. Viszont tudjuk, hogy van olyan elem, amely legalább 5 karakter hosszú. Az első feladatunk a legelső ilyen elem sorszámának megállapítása és tárolása a *sorsz* változóban. Lépéseink a 3.3.1 pontbeli (A) változathoz hasonlóak:

- Létrehozunk egy *i* egész típusú változót, a kezdőértéke legyen 1.
- Az *i*-t indexként használva mindaddig növeljük az értékét, amíg az adott sorszámú tömbelem hossza 5-nél kevesebb.
- Amikor kilépünk a ciklusból, az *i* egy legalább 5 karakter hosszú elem indexe, ez bekerül a *sorsz* változóba.

```
Sub Karakterhossz(t() As String, n As Integer)
    Dim i As Integer, sorsz As Integer
    i = 1
    Do While Len(t(i)) < 5
        i = i + 1
    Loop
    sorsz = i
End Sub
```

A Len({String változó}) Visual Basic-függvény egy String típusú változó karaktereinek számát adja vissza (length = hossz).

A következő példában a legalább 5 karakter hosszú elemeket egy másik, *u* tömbben is rögzítjük. Azonban az új tömb elemszáma előre nem ismert, célszerű tehát először megszámolni, hány elem teljesíti a feltételünket. Erre használhatunk egy külön függvényt is:

- Létrehozunk egy egész típusú, *db* azonosítójú változót, kezdő értéke legyen 0.
- Végigmegyünk a *t* tömb elemein, és a feltételt teljesítő elemek esetén a *db* változó értéke eggyel növekszik.
- A ciklus végén a *db* értéke a keresett darabszám, ez lesz a függvény visszatérési értéke.

```
Function Megszamol(t() As String, n As Integer, hossz As Integer)
    Dim db As Integer, i As Integer
    db = 0
    For i = 1 to n
        If Len(t(i)) >= hossz Then db = db + 1
    Next i
    Megszamol = db
End Function
```

A *Megszamol()* függvény segítségével definiálhatjuk az *u* tömböt, és elhelyezhetjük benne a megfelelő tömbelemeket:

- Létrehozunk az *u* tömböt, illetve egy *j* egész típusú változót, utóbbi kezdő értéke 1.
- Az *u* tömböt a *Megszámol()* függvény eredménye szerint méretezzük.
- Végigmegyünk a *t* tömb elemein, ha a feltételt teljesítő elemre bukkantunk, azt átmásoljuk az *u* tömb *j*-edik rekeszébe, és a *j* indexet léptetjük 1-gyel.

```

Sub Valogat(t() As String, n As Integer)
    Dim i As Integer, j As Integer, u() As String
    ReDim u(1 To Megszámol(t, n, 5)) As String
    j = 1
    For i = 1 To n
        If Len(t(i)) >= 5 Then
            u(j) = t(i)
            j = j + 1
        End If
    Next i
End Sub

```

Mivel a Visual Basic nyelv egy tömb deklarációjakor csak konstans számértékeket fogad el elemszámként, a fenti példában dinamikus deklarációt, újradimenzionálást használtunk.

3.3.3 Maximum kiválasztása

A tömb elemei legyenek numerikus típusúak, feladatunk közülük a legnagyobb elem kiválasztása, és sorszámának megadása.

- Létrehozunk egy *sorsz* egész típusú változót, kezdő értéke legyen 1. Feltételezzük, hogy az 1. elem a legnagyobb.
- Végigmegyünk a tömb elemein, megvizsgáljuk, hogy találunk-e az eddig feltételezettnél nagyobbat. Ha találunk, feltételezzük, hogy az a legnagyobb, és a *sorsz* változót egyenlővé tesszük az indexével.
- A ciklus lefutása után a *sorsz* változó a maximum sorszámát tartalmazza.

```

Function Maxkeres(t() As Double, n As Integer)
    Dim i As Integer, sorsz As Integer
    sorsz = 1
    For i = 1 to n
        If t(i) > t(sorsz) Then sorsz = i
    Next i
    Maxkeres = sorsz
End Function

```

Ha a függvénnyel a maximális értéket akarjuk inkább előállítani, az utolsó előtti sort így kell módosítanunk:

```

Maxkeres = t(sorsz)

```

A legkisebb érték előállítása a fentihez hasonló elvű algoritmussal végezhető, kidolgozását az Olvasóra bizzuk. Megjegyezzük, hogy amennyiben a sorozatban a maximális érték többször is előfordul, a fenti algoritmus csak a legkisebb indexű elem sorszámát képes meghatározni. Ilyen esetben az összes maximum sorszámainak előállításához a függvény kombinálható a 3.3.2 pontban megismert kiválogató algoritmussal.

3.4 Rendezés

Az előző pontokban bemutatott módszerekben közös, hogy valamilyen T tulajdonságot vizsgáltak azonos típusú adatok sorozatán. Számos esetben az ilyen vizsgálatok könnyebben, gyorsabban elvégezhetők, ha rendezett sorozattal van dolgunk – könnyen látható például, hogy egy numerikus, növekvően rendezett számsorozat minimumának megállapítása valóban „minimális” erőfeszítést igényel. A fejezet két rendezési módszert részletesen is ismertet; ezek olyan klasszikus módszerek, amelyek finomításával más, bonyolult, de hatékonyabb rendezési eljárások is alkothatók.

3.4.1 Buborék-rendezés

A buborék-elv a „szomszédos” tömbelemeket hasonlítja össze, és szükség esetén cserét hajt végre. Amennyiben egy n elemű t tömbünk van, legfeljebb $n-1$ lépésre van szükség ahhoz, hogy minden elem a helyére kerüljön. (Gondoljuk végig, hol lehet a sorozat legnagyobb/legkisebb eleme a rendező szempontjából „legrosszabbnak” számító esetben!) Ez alapján tehát $n-1$ db vizsgálatot végzünk a szomszédos elemekre, és ezt a vizsgáldást $n-1$ -szer ismétljük. A lépések:

- A folyamat $n-1$ -szeri ismétlését ciklussal (számlálós típus) végezzük.
- Végigmegyünk a szomszédos elemeken, ez $n-1$ lépés.
- Szükség esetén cserét hajtunk végre.

Két elem cseréjéhez szükségünk lesz egy átmeneti tárolóra. Ezt *seged*-nek nevezve a cserefolyamat a következő:

- A *seged* felveszi az első elem értékét.
- Az első elem felveszi a második elem értékét.
- Végül a második elem felveszi a *seged* értékét.

Tekintsünk egy tízelemű tömböt ($n = 10$), a rendezés növekvő sorba Visual Basicben:

```
Sub Buborek()  
Dim t(1 to 10) As Double, i As Integer, j As Integer, seged As Double  
Const n = 10  
'... a tömb feltöltése  
For i = 1 To n - 1  
    For j = 2 To n  
        If t(i) > t(j) Then  
            seged = t(i)  
            t(i) = t(j)  
            t(j) = seged  
        End If  
    Next j  
Next i  
End Sub
```

3.4.2 Minimum-rendezés

Ez az algoritmus felhasználja a minimumkeresés logikáját: a sorozat egyre csökkenő elemszámú részhalmazainak minimumát megkeresve, elemenként halad. Legyen ismét egy $n = 10$ elemű t numerikus tömbünk.

- Végig kell mennünk a tömb elemein, az elsőtől az utolsó előttiig.
- Az 1. pont ciklusán belül, annak aktuális ciklusindexétől kezdve megkeressük a legkisebb elemet.
- A 2. lépésben megtalált legkisebb elem kerül a ciklusváltozó aktuális értéke által megadott helyre.

```
Sub Minimum()  
Dim t(1 To 10) As Double, i As Integer, j As Integer, seged As Double  
Dim minsorszam As Integer  
Const n = 10  
'... a tömb feltöltése  
For i = 1 To n - 1  
    minsorszam = i  
    For j = i + 1 To n  
        If t(j) < t(minsorszam) Then minsorszam = j  
    Next j  
    If minsorszam <> i  
        seged = t(i)  
        t(i) = t(minsorszam)  
        t(minsorszam) = seged  
    End If  
Next i  
End Sub
```

3.4.3 További rendezési módszerek

Röviden említést teszünk még néhány rendezési elvről – csak a rendezések logikáját ismertetjük, a konkrét algoritmusok az eddigiek alapján kidolgozhatók, vagy pedig bonyolultságuk miatt nem férnek bele a jegyzet kereteibe.

A 3.4.1 pontban bemutatott buborék-elv egy módosított változata lehet a következő: továbbra is szomszédos párokat vizsgálunk, de váltakozó irányban haladunk végig a sorozaton. A ciklus első lefutásakor a maximumot az utolsó helyre tettük, ekkor visszafelé haladunk, és a minimumot elhelyezzük a legelső helyen. Az első és utolsó elemek indexét eggyel növelve, illetve csökkentve egy az előzőhöz hasonló szituációhoz érkezünk – a lépések ismétlése addig tart, amíg már csak két elem marad az egyre szűkülő listán.

Beszűrásos rendezésnek nevezzük azt az algoritmust, amely sorrendbeli hibákat keres, és a hibás elem áthelyezését úgy oldja meg, hogy a közbeeső elemeket eltolja. Ennek a formának egy módosított változata Shell-rendezés néven ismert, ez a rendezés valamilyen t lépésközzel alkalmazza a beszűrásos módszert; a lépésköz folyamatosan csökken, amíg el nem éri az 1-et.

A Shell-módszer nagy elemszám esetén gyorsabb, mint a fentebb bemutatott változatok; az utolsó ismertetett rendezés is nagy mennyiségű elem esetén hatékony: a Gyors rendezés az egymástól távoli indexű elemeket veti össze és cseréli, majd folyamatosan szűkíti az elemek körét.

Irodalomjegyzék

3.1. Rónyai, Ivanyos, Szabó: Algoritmuskok, Typotext, 1998.

Ellenőrző kérdések

Mi az az algoritmus?

Hogyan működik a két természetes szám legnagyobb közös osztóját meghatározó algoritmus?

Mi a tulajdonság keresési algoritmusok működési elve?

Milyen rendezési algoritmusokat ismer? Hasonlítsa össze a hatékonyságukat!

4. AZ ADATTÓL AZ ADATBÁZIS KEZELÉSIG

4.1. Az adatbázis

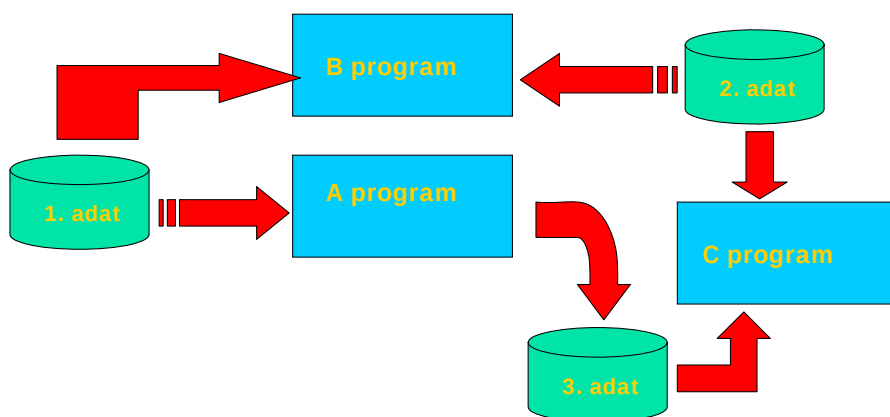
Az adatbázisok ma már az üzleti élet minden területén elterjedtek. A pénzügyi, kereskedelmi folyamatokat adatbázisok alkalmazásai segítik. Megtalálhatóak a kereskedelemben is, ahol a vásárlók és az ügyfelek számára szolgáltatnak adatokat. Adatbázisokat használnak az egészségügyben és a tudományos kutatásokban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

Adat

Adatnak nevezik az objektum azon jellemzőit, amelyek célirányos csoportosítása az információ. Ma már leginkább elektronikus módon tárolják az adatokat. A tárolás rendszerben történik a visszakereshetőség érdekében.

Adatbázis

Az *adatbázis* (DB – Database) definíciószerűen a *program független* adatok *rendezett* halmaza. Ez 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.



4.1. ábra: A program függetlenség követelménye

Mint az előző fejezetben is rögzítettük, az adat az elsődleges, a program csak az adaton értelmezett leképezés. Ez azt is jelenti, hogy ugyanazon adathalmazból több program is dolgozhat (lásd 4.1. ábra 1. adat nevű file-ból mind az **A**, mind a **B** jelű program dolgozik, csak a **B** jelű program egy másik adathalmazt is felhasznál). Az **A** program egy rendezettséget igényel, a **B** program esetleg egy másikat. Az *adatbázis tehát nem mennyiségi, hanem minőségi fogalom*. Csak az tekinthető adatbázisnak, amely a fenti két kritériumnak megfelel.

Az adatbázis-kezelés kétféle adatbázis típust különböztet meg: *operációs- 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.

4.2. Az elemi adatszerkezetek

A bejegyzés, vagy rekord

1. Az adatbázisokban használatos adatszerkezetek alapegysége, egy összetartozó adatsor, ami az adatbázis egy elemének jellemzésére szolgál. A bejegyzésre példa az 4.1. táblázat egy sora. Ez egy adott rendszámú autó néhány fontos jellemzőjét írja le. Miután a bejegyzés a logikailag összefüggő elemeket együtt tartalmazza, *logikai rekordnak* nevezzük, eltérően a tároló közegen (pl. mágneslemez, vagy CD) együtt tárolt adatokkal, amiket *fizikai rekordnak* nevezünk. (A kettő nem feltétlenül esik egybe, mert pl. a logikai rekord olyan hosszú is lehet, hogy az két egységnyi tárolón fér el).

4.1. táblázat: Példa a bejegyzésre

RENDSZÁM	GYÁRTMÁNY	TÍPUS	SZÍN	ÚRTARTALOM	EGYÉB
CYX462	Mitsubishi	Colt GL	fehér	1298	...
DY9680	Ford	Sierra L	ezüstmetál	1996	...
ABC017	Daihatsu	Charade	kék	1298	...
V24034	Ford	Escord D	fehér	1796	...
CZ9404	Volkswagen	Fastback TL	zöld	1594	...
KF9547	Volkswagen	Fastback TL	zöld	1594	...

A mező

A bejegyzés *mezőkből* áll össze, amely a fenti példában az az összetartozó jelsorozat, amelynek még önálló jelentése van (a bemutatott példában a gépkocsi rendszáma, vagy gyártmánya, színe, stb.).

A kulcs

Amely mező, vagy mezőkombináció a bejegyzés egyértelmű azonosítására szolgálhat, azt (*keresési*) *kulcsnak* nevezik.

A bejegyzések elemi adatszerkezeteket alkotnak, amelyeknek három alaptípusával dolgoznak az adatbázis kezelők:

- tömb,
- láncolt lista,
- verem.

A tömb

A mátrix analógiája adatszerkezeteknél, lényegében a programozásnál alkalmazott egy-, két-, vagy több indexes változónak megfelelő elemi adatszerkezet. Olyan *lista*, amelynek elemei (a bejegyzések) szorosan egymás után következnek, s amelynél az n -edik elem az $n-1$ -edik elem után helyezkedik el.

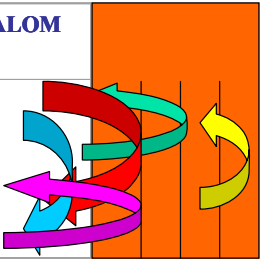
A láncolt lista

A láncolt lista egy eleme, egy T típusú adatrészből, és $1 \dots n$ darab pointerből áll. A pointerok arra a memóriacímre mutatnak, ahol egy újabb listaelem található (rákövetkező). Láncolt lista esetén csak a listaelem helyfoglalása folytonos, a pointer értéke mutatja, hogy a következő elem hol található a memóriában. (Előfordulhat, hogy 1-2 elem, vagy akár az egész lista memória-folytonosan helyezkedik el, de ez nem feltétlenül jellemző). Ezért láncolt listának nevezzük azon listát, amelynek minden eleme egy *pointer* (mutató) alkalmazásával tartalmazza a következő elem helyét.

A láncolt lista fogalma a rendezésekkel kapcsolatban érthető meg a legkönnyebben. Mint már említettük, ugyanazon adathalmazból több, különböző program dolgozhat, de mindegyik saját rendezett szerkezetet igényel. Amennyiben pl. a telefonkönyv az előfizetők betűrendes sorrendjében tartalmazná az adatokat, minden új előfizető esetén át kellene rendezni a milliós nagyságrendű előfizetői listát. A telefonszámok alapján történő kereséshez viszont ugyanazon lista másik kulcs szerint történő rendezése lenne szükséges. Ezt a problémát hídálja át a láncolt lista. Az egyszerűség kedvéért nézzük az előbbi rendszám szerinti nyilvántartás rendezését úgy, hogy a rekordok sorrendjét nem változtatjuk, ennek ellenére azok betűrendes sorrendben érhetőek el. Az első elem megtalálása után annak a végére egy u.n. *pointer mezőt* illesztünk, amely a betűrendes sorrendben következő elemre mutat (4.2. táblázat).

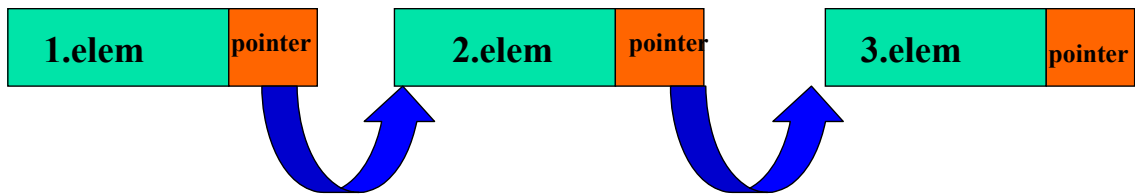
4.2. táblázat: Rendezés pointerok alkalmazásával

RENSZÁM	GYÁRTMÁNY	TÍPUS	SZÍN	ÚRTARTALOM
CAX462	Mitsubishi	Colt GL	fehér	1298
DY9680	Ford	Sierra L	ezüstmetál	1996
ABC017	Daihatsu	Charade	kék	1298
V24034	Ford	Escord D	fehér	1796
CZ9404	Volkswagen	Fastback TL	zöld	1594
KF9547	Volkswagen	Fastback TL	zöld	1594

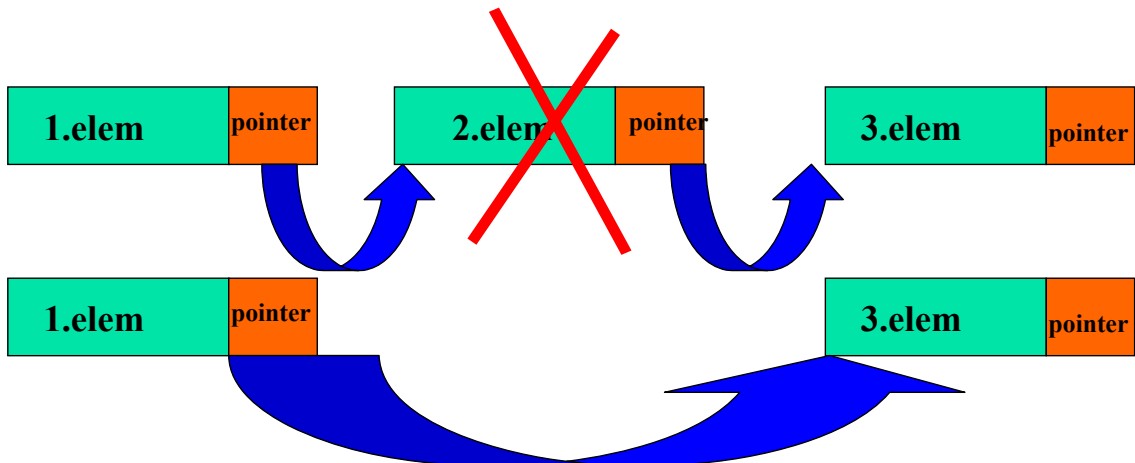


Amennyiben pl. egy további pointer mezőt csatlakoztatunk a rekordjainkhoz, az tartalmazhatja a gyártmány szerinti rendezést is, ami egy másik felhasználó program működését segítené stb. Ily módon lehetővé válik az, hogy ugyanazon adatszerkezet egyidejűleg több szempontból legyen rendezve.

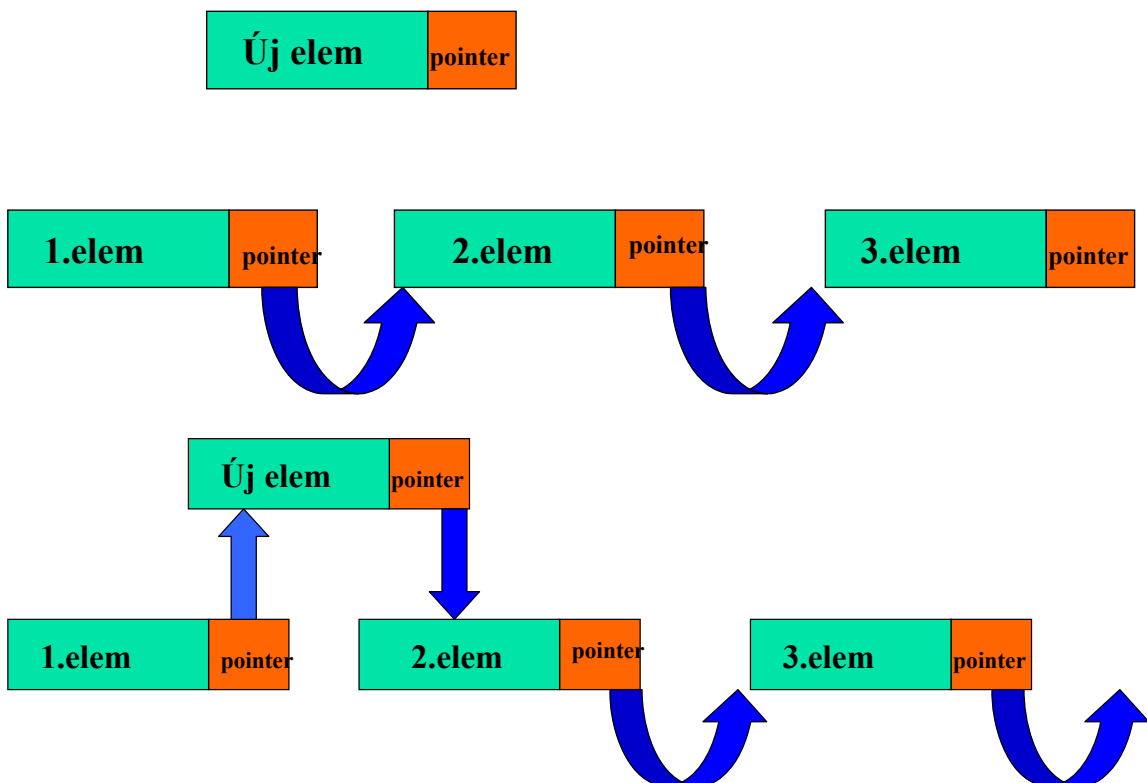
A pointerok alkalmazása nagyon leegyszerűsíti az elemek beírását és törlését a rendezett listákba (l. 4.2., 4.3. és 4.4. ábrák).



4.2. ábra: A láncolt lista alapelve



4.3. ábra: Elem törlése a láncolt listából

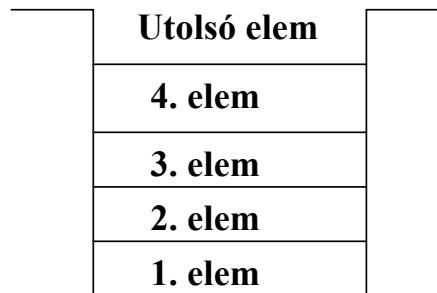


4.4. ábra: Új elem beillesztése a listába

A láncolt listák az alapszoftver leggyakrabban használatos elemei, így oldja meg az operációs rendszer pl. a háttértárolók kezelését. Ebben az esetben a tárolandó adatok láncolt listák formájában kerülnek a tároló elem éppen szabad területeire.

A verem (stack):

A másik leggyakrabban alkalmazott listaforma olyan adattároló, amelyből a legutoljára beírt elemet (top) tudjuk elsőnek elérni, illetve kitörölni.

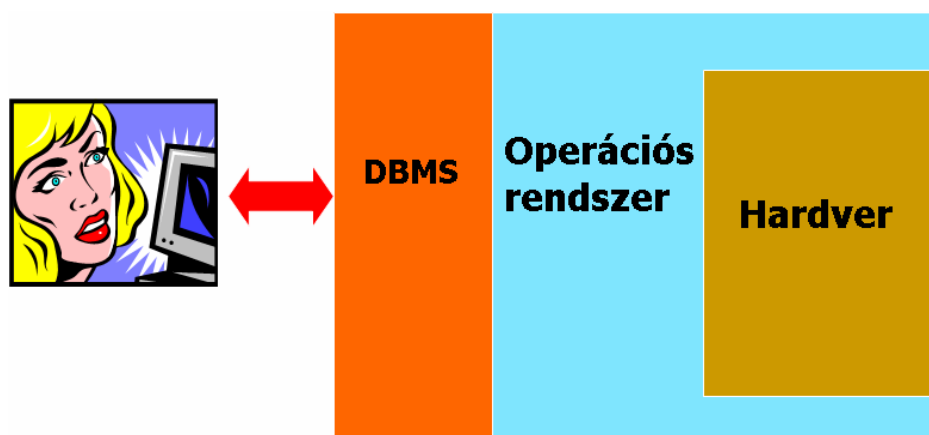


4.5. ábra: A verem alap gondolata

A verem a sorrend megfordítására szolgál (LIFO Last In First Out): a fordító programok a függvények/eljárások hívásakor, az operációs rendszer megszakítás hívásakor (hívás előtti állapot kimentés, majd állapot visszaállítás hívás után) használja, további felhasználási terület például a matematikai képletek kiértékelése a programnyelvekben.

4.3. Adatbázis-kezelő rendszerek

Az adatbázisokhoz tartozik egy nagyméretű programrendszer, amely a fenti alapgondolatokat valósítja meg, s a neve *adatbázis-kezelő rendszer* (Database Managment System - DBMS). Ezzel lehet az adatokat megszervezni a fenti elvek alapján, az adatbázisban tárolt adatokat kezelni: adatokat megkeresni, kiválasztani, karbantartani, módosítani, új adatokat beilleszteni, vagy egyes adatokat törölni.



4.6. ábra: Az adatbázis kezelő helye a számítógépben

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

- 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 Definiton Language, rövidítve: DDL.
- 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.
- 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.
- Egyszerű használat, a különféle felhasználói igények hatékony kielégítése, Az adatok védelme: nehogy illetéktelenek hozzáférhessenek (privacy), és a hozzáférésre jogosultak se ronghassák el lehetőleg az adatbázist (integrity),
- Helyreállíthatóság: bármely esetleges hiba esetén az eredeti állapotot vissza lehessen állítani,
- 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 be illesztéséért, az információ épségéért, *adatbázis-adminisztrátornak* hívják.

4.3.1. Néhány példa az adatbázis-kezelő rendszerre

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 megengedhetők. 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 hirtelen hiba folytán leállna.

A fájlkezelő rendszerekből kialakult 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 érdekességeként említhetők.

Ezekkel a korai modellekkal 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.

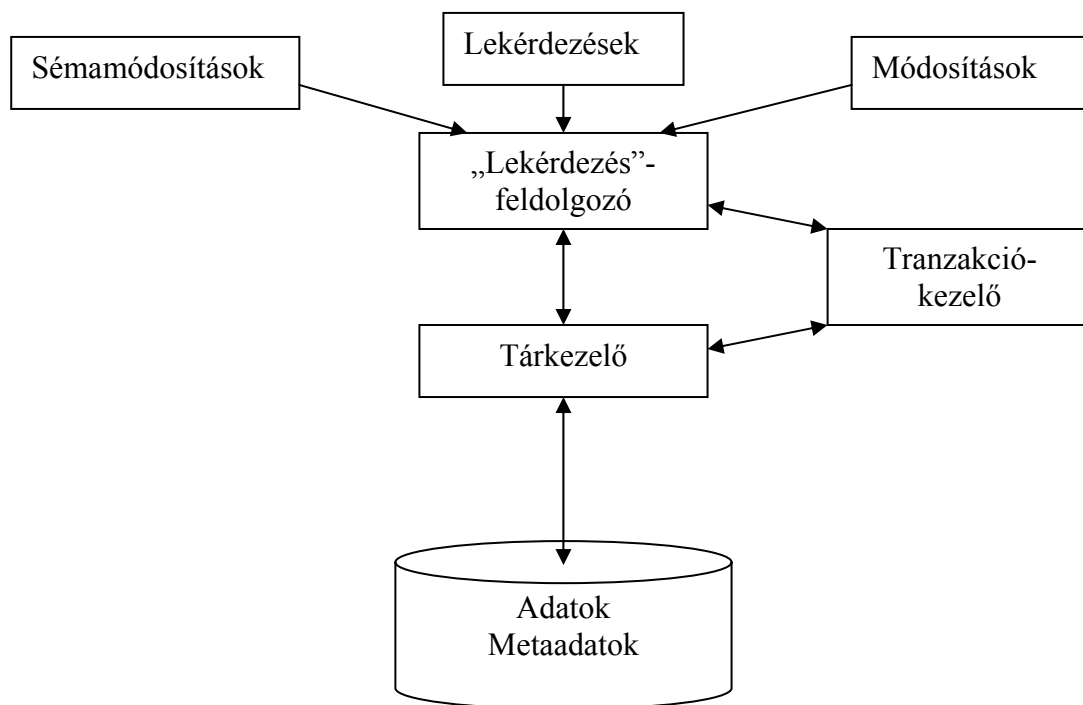
Az IBM volt az első cég, amelyik relációs és a relációs modell előtti modelleket támogató adatbázis-kezelő rendszereket is árusított. Később újabb és újabb cégek alakultak, amelyek relációs adatbázis-kezelők megvalósításával és árusításával foglalkoztak. Ma már ezek a cégek közül néhány a világ legnagyobb szoftverkereskedői közé tartozik.

4.3.2. Adatbázis-kezelő rendszerek felépítése

Ebben a szakaszban felvázoljuk, hogyan néz ki általában egy adatbázis-kezelő rendszer felépítése. Az 4.7. 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.



4.7. 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 lemezzől a fájlkezelő segítségével olvassa be, és kiválaszt egy memória területet, ahol az adatblokkokat tárolja.

Tranzakció kezelő: felelős a rendszer sérthetetlenségéért. 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 lemezre 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*: 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, de a tranzakció-kezelőnek kell biztosítania, hogy a tranzakciók befejeződése után az adatbázis ismét következetes állapotba kerüljön, vagyis elégítse ki 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 veszhet 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 adatételt. 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 feljegyzi 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:

- Létrejöhetnek *általános lekérdező-interfészen* keresztül. Ekkor a rendszer megengedi a felhasználónak, hogy szabványos lekérdező nyelv (SQL) segítségével lekérdezéseket adjon meg, amelyeket a lekérdezés-feldolgozó kap meg és végrehajt.
- 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.

4.4. Adatmodellek

Az alábbi főbb adatmodell típusokat fogjuk vizsgálni:

- hierarchikus adatmodell,
- hálós adatmodell,
- relációs adatmodell,
- objektum-orientált adatmodell

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 modell között kicsi a különbség. A kiterjesztett adatmodell a CODASYL-hálós és a relációs modellek előnyös tulajdonságait egyesíti.

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

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 fagráfok 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 gyökekből 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 ősszülő, amit másképpen egy fagráf gyökerének neveznek. A felhasználó az adatok olvasását mindig a gyökérben kezdi, és innen halad lefelé a kívánt adatig. Az utódok nélküli szülőket a fagráf leveleinek (leaf) nevezik.

Az egy gyökérhez tartozó fát adatbázis-rekordnak (database record) nevezik. 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 modellben 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 szülőtábla egy létező rekordjához kapcsolódik, és a szülőtáblából 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.

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.

Az alapvető kapcsolattípusok:

- A két elem független egymástól;
- Kölcsönösen egyértelmű kapcsolat (pl. jó kódrendszerben a kód és a megnevezés között),
- Egyik irányban egy, a másikban többértelmű (1:N) kapcsolat (pl. szülőanya és gyerekei),
- Több-több értelmű (M:N) kapcsolat (pl. férfiak és nők).

Hálós modellek jellemzői

- Minden adategység tetszőleges más adategységgel lehet kapcsolatban.
- A modellben 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.

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).

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.

4.5. Relációs adatmodell

A relációs adatmodell napjaink legelterjedtebb adatmodellje. 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 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 a modell számos új fejlesztési projekt alapját képezi. Az adatmodell mindenki számára fontos előnye az egyszerűség mellett a modell rugalmassága.

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.

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 tartalmazzak-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.

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.

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.

Kapcsolattípusok

A táblák közötti kapcsolatoknak három típusa van: 1:1, 1:N, N:M.

1:1 kapcsolat

Két tábla közötti 1:1 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.

1:N kapcsolat

1:N 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.

N:M kapcsolat

N:M 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.

Kapcsoló táblákkal alakítható ki ilyen kapcsolat. A kapcsolótábla segíti a két tábla rekordjainak az összekapcsolását. A kapcsolótábla a kapcsolatban álló táblák elsődleges

kulcsaiból hozható létre, ezek az elsődleges kulcsmezők együtt adják a kapcsolótábla összetett elsődleges kulcsát, másrészt idegen kulcsok is.

Normalizálás. Normálformák

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

Az egyedek optimális tulajdonságainak kialakításában a *normalizálás* matematikai eljárás segít. Az egyedek legjobb *normálformája* (*NF*) több lépésben alakítható ki. A normálformák egymásba skatulyázhatók. 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.

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 eredetinel 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. Az egyedek külső szerkezetére is figyelni kell. A tulajdonságoknál fel kell tárni a rejtett kapcsolatokat. Az egyedekből ki kell küszöbölni a többszörös tartalmú ismereteket. A normalizálás igazán akkor kezdődhet meg, ha a modellben nincs ismétlődést mutató tábla. Tehát legelőször a normalizálási alapot kell előállítani, azaz előnormalizálást kell végezni. A tervezett egyedekben az ismétlődések nem mindig láthatóak. Részekre kell bontani az összetett tulajdonságtípusokat, olykor a tulajdonságértékeket is.

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 fölérendelt táblában.

4.6. A relációs adatbázis kezelő rendszerek a gyakorlatban

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.
- *Űrlap*: az adatok rekordjainak bevitelét, módosítását vagy törlését kényelmessé teszi.
- *Lekérdezés*: segítségével a táblában 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.
- *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.

4.6.1. 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 sztring, alapértelmezett hosszúsága 50 karakter.
- *Feljegyzés*: legfeljebb 64000 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 64000 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 kijelzett 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*: a megadott mező szerint történik a tábla rendezése, ami gyorsítja a későbbi adatkeresést.
- *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 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. Ha módosítás közben vissza kell vonni a változtatásokat, akkor ezt az ESC billentyű megnyomásával lehet elérni.

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 **Szerkesztés\Rekord törlése** menü funkcióval, vagy az eszköztár **Rekord törlése** ikonnal is. Hasonlóan törölhető oszlop (mező) is.

Másolás, áthelyezés

Ezek a műveletek teljesen hasonlóan elvégezhetők, mint a Wordben vagy az Excelben.

Tábla rendezése

A táblában először ki kell jelölni azt a mezőt, amely szerint rendezni kell. Majd a **Rekord\Rendezés** menü 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\Szűrő\Szűrés** kijelöléssel menüpontot kell választani. A feltételnek nem megfelelő rekordok megkeresése a **Rekordok\Szűrő\Szűrés** kizárással funkcióval kezdeményezhető.

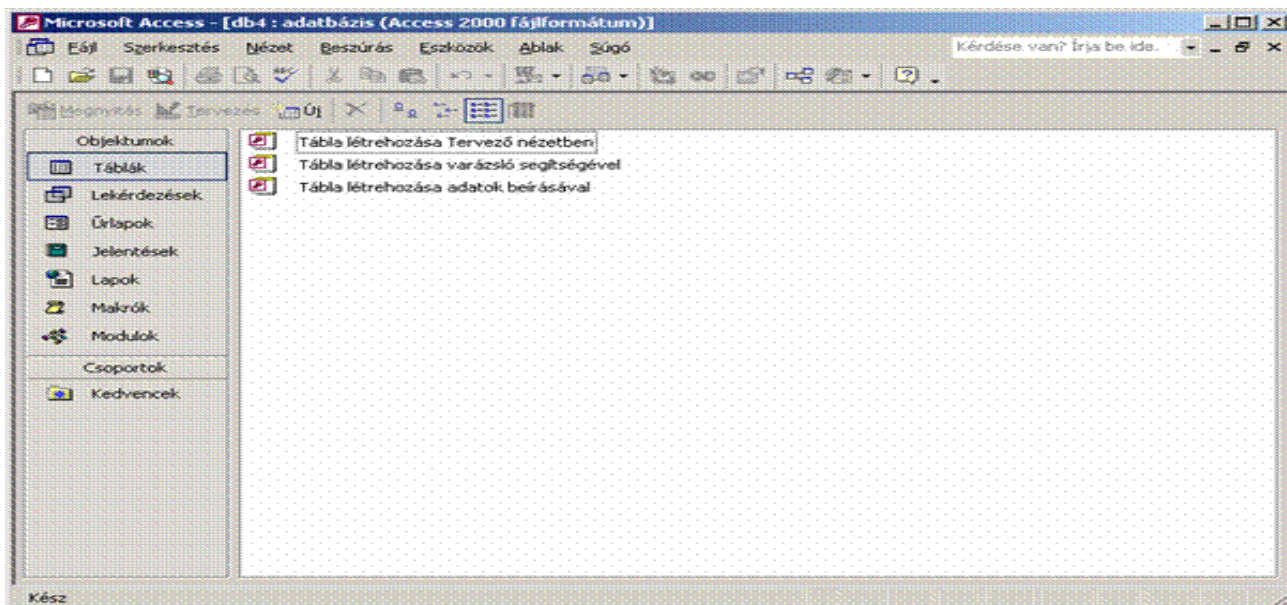
Az úrlappal történő szűréskor egyszerre több mező tartalmára is megadható feltétel vagy egy mezőre több feltételt lehet beállítani. A **Rekordok\Szűrő\Szűrés** úrlappal menüpont választása után megjelenő adatlapon az egyes mezőkhöz tartozó szűrőértékeket legördülő menüből lehet kiválasztani. Az összes feltétel beállítása után a **Szűrő\Szűrés/rendezés** funkció választásával megkaphatóak a táblában a feltételeknek megfelelő rekordok. Még összetettebb feltétel adható meg az **Írányított szűrés** eljárással. Ez a **Rekordok\Szűrő\Írányított szűrés/rendezés** menüponttal kezdeményezhető.

A kiszűrt vagy a rendezett rekordok helyett újból az összes rekord megjelenítését a **Rekordok\Rendezés törlése** menüpontból lehet kérni.

4.6.2 Adatbázis tervezése (relációs adatbázis tervezése). Adatbázis létrehozása

Adatbázis létrehozása

Új adatbázis létrehozása vagy meglévő adatbázisfájl megnyitása után megjelenik az Adatbázis ablak (4.7. ábra).



4.7. 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:

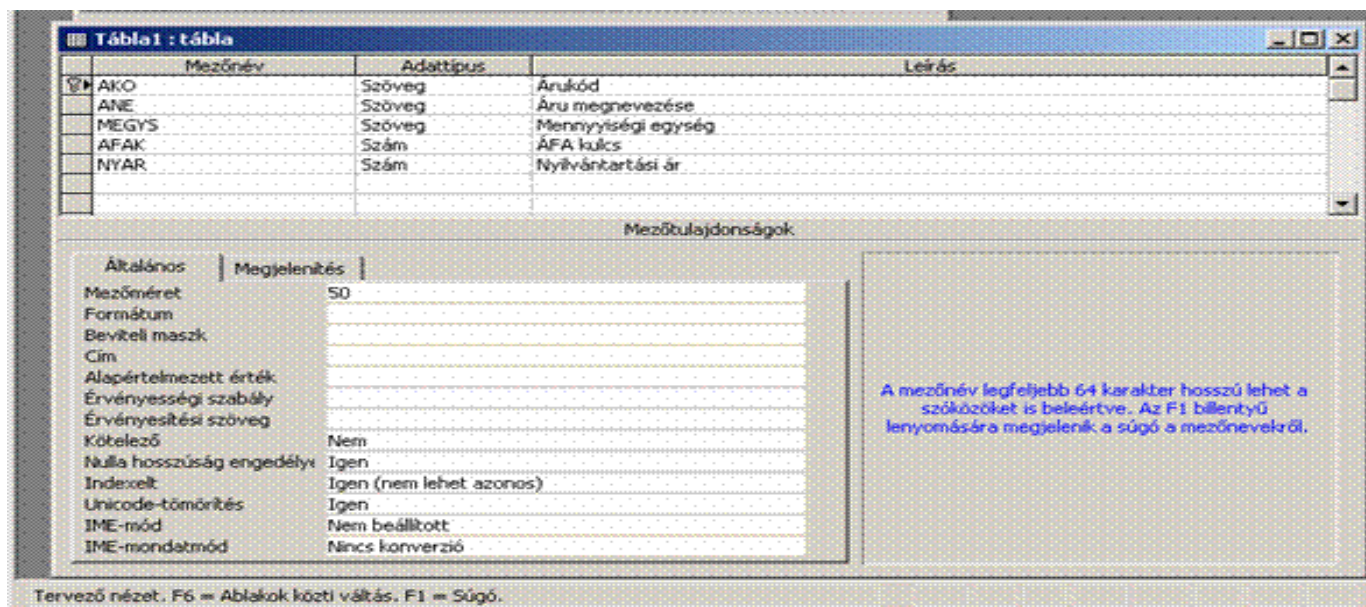
- *Automatikus*: ekkor az Access automatikusan elkészíti az objektum szokásos, legcélszerűbb változatát.
- *Varázsló segítségével*: a szokásos varázsló-technikával végigvezet a tervezési folyamaton.
- *Tervező nézetben* teljesen manuális a tervezés, itt a rendszer valamennyi lehetősége elérhető.

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 (4.8. ábra).



4.8. 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.

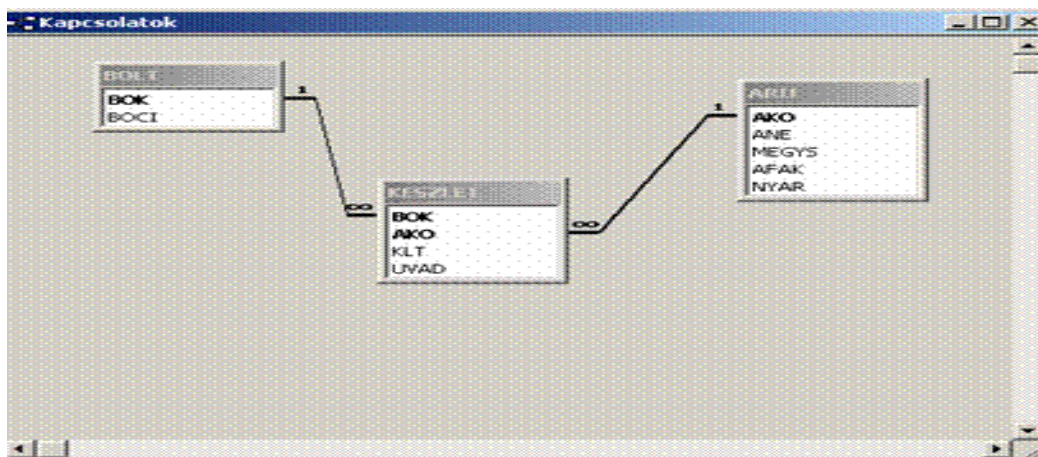
Elsődleges kulcs létrehozása

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

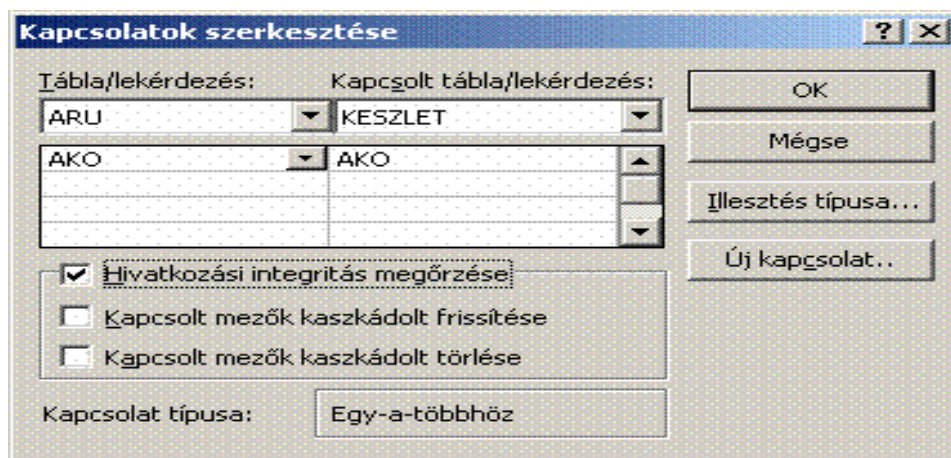
Kapcsolat létrehozása az **Eszközők\Kapcsolatok** menüpontból kezdeményezhető. Megjelenik a **Tábla hozzáadása** panel, ahol ki kell választani azokat a táblákat, amelyek részt vesznek a kapcsolatok kialakításában, majd a kapcsolatot biztosító mezőt az elsődleges táblából egérrel át kell húzni a másik tábla megfelelő mezőjéhez. A **Létrehozás** gomb megnyomásával befejeződik a kapcsolat készítése (4.9. ábra).

4.9.



ábra: Kapcsolatok az Access-ben

A kapcsolat jellemzőinek módosítását vagy törlését a kapcsolati vonalra kattintva lehet elvégezni (4.10. ábra).



4.10. ábra: Kapcsolatok szerkesztése ablak

A táblák közötti kapcsolatok típusai

Az Access *elsődleges táblának* nevezi az 1:N kapcsolatok 1-oldalán álló táblát, *illesztőtáblának* 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 állandóan ellenőrzi a kulcsfeltételeket. Az üres sztring megadását a rendszer általában Null (definiálatlan) értéknek tekinti.

4.6.3 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,
- keresztátlás lekérdezés,
- táblakészítő lekérdezés,
- frissítő lekérdezés,

- hozzáfűző lekérdezés,
- törlő lekérdezés.

Lekérdezés elkészíthető Tervező nézetben vagy varázslóval.

Lekérdezés készítése Tervező nézetben

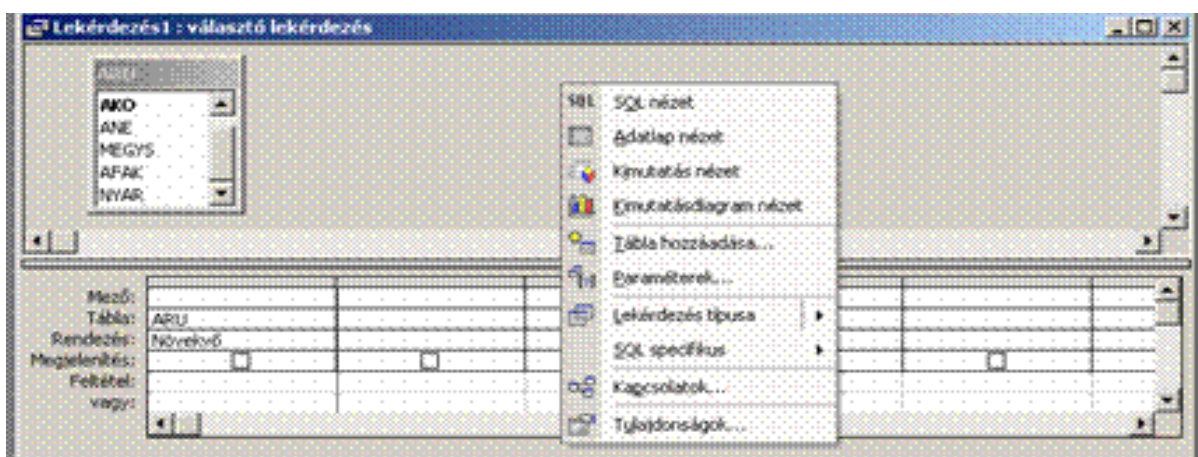
Az Adatbázis-ablakban a lekérdezés objektumtípust, majd az Új gombot, és végül a Tervező nézetet kell választani. A Tábla hozzáadása ablakban ki kell jelölni a lekérdezéshez szükséges táblákat. Utána megjelenik a Tervező ablak, amelynek beállítási lehetőségei:

- 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 (4.11. ábra).



4.11. ábra: Választó lekérdezés

- *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 **Felhasználó** vagy **Tulajdonos** értéknek megfelelően az adathozzáférési jog vagy a felhasználó 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 jelezzon időtúllépést az Access.
- *Szűrő:* szűrőfeltétel adható meg, hasonlóan az **SQL WHERE** záradékban megszokott formában. A **WHERE** kulcsszót nem kell használni.

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 lekérdezés**hez 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 törölhetőnek 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.

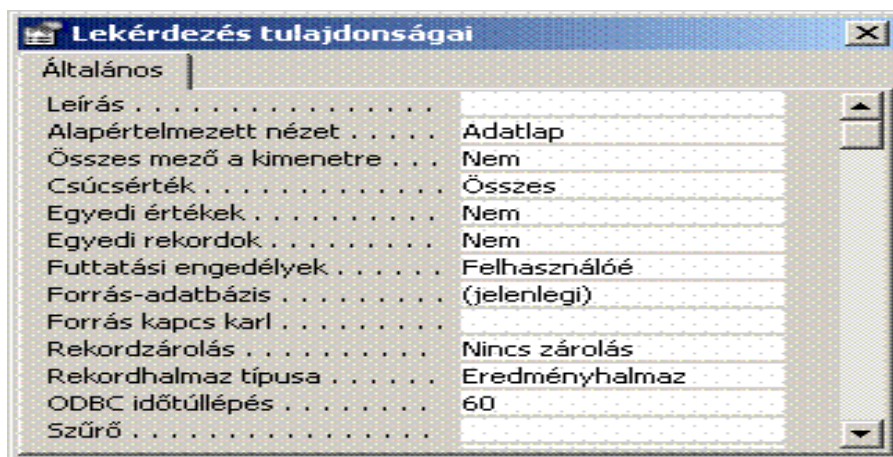
Lekérdezések többféle módon hozhatók létre.

- Lekérdezés panelen az Új gombot választva.
- Beszúrás\Lekérdezés menü funkció választásával.

Lekérdezések létrehozása

4.12. ábra:
lekérdezés

Az Új

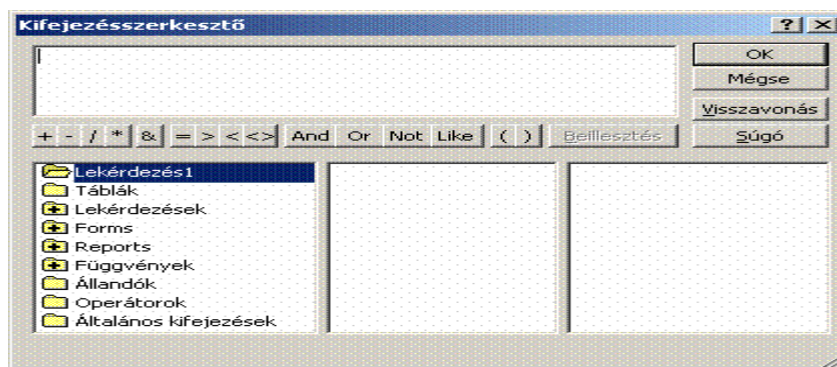


Új
ablak

lekérdezés ablakban lehet választani, milyen módon készüljön a lekérdezés (4.12. ábra). A varázslók használatával gyorsabban hozhatók létre lekérdezések, de végül még tervező nézetben meg kell adni a feltételeket.

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 adattáblából mely mezők tartalma szerepeljen az egyes oszlopokban. A tervezőrács utolsó két sorában a feltétel szerkesztéséhez alkalmazható a **Kifejezésszerkesztő**, amely a **Tervezőrács Feltétel sorából** a gyorsmenüből elérhető (4.13. ábra).



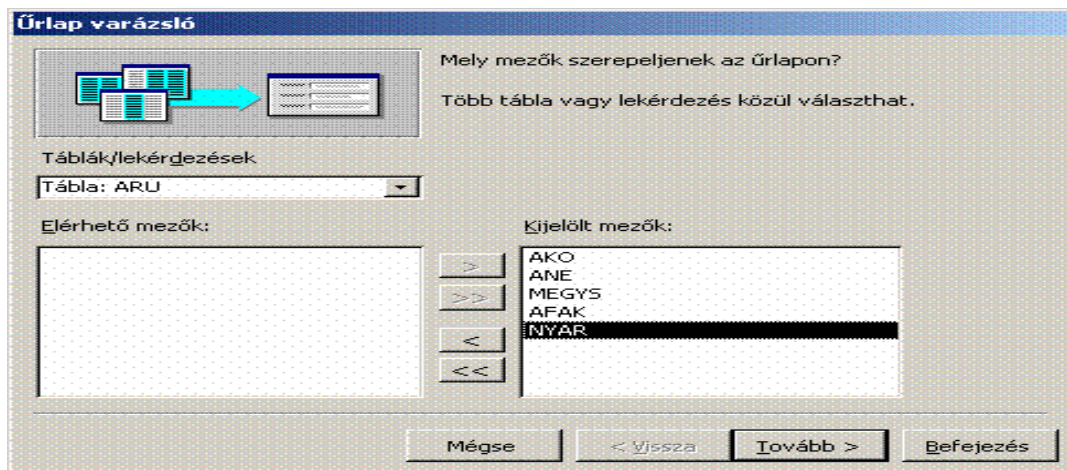
4.6.4. Űrlapok készítése

Űrlap létrehozása

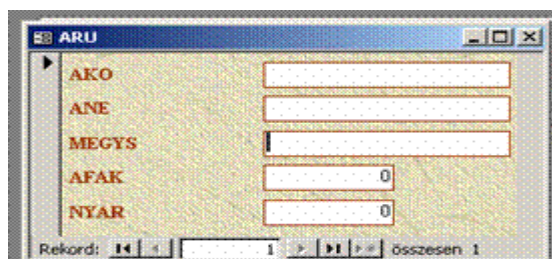
Az űrlap létrehozása előtt meg kell tervezni, hogy mi legyen annak tartalma és milyen szakaszban. Az Accessben az űrlapnak öt szakasza van:

- *Űrlapfejléc*: tartalmazhatja az űrlap címét, rövid kezelési tájékoztatást vagy parancsgombot. Az űrlap kinyomtatásakor ez csak a legelső oldal tetején jelenik meg.
- *Oldalfejléc*: a nyomtatott oszlopok fejrészét tartalmazza.
- *Törzs*: adatbeviteli mezőket, vezérlőelemeket tartalmaz.
- *Oldallábléc*: az 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 táblákból, lekérdezésekből készüljön az űrlap és milyen mezőket tartalmazzon (4.14. 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 (4.15. ábra).



4.14. ábra: Űrlap varázsló ablak



4.15. 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.

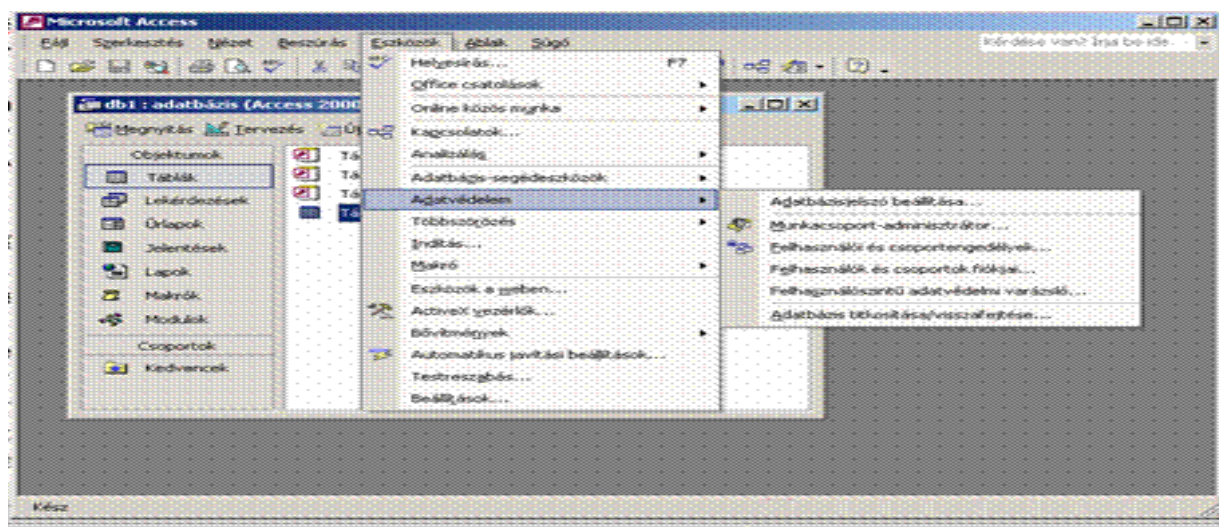
4.6.5 Jogosultság, biztonság, adatvédelem

Az Access-ben az adatvédelemmel foglalkozó szolgáltatások az **Eszközök\Adatvédelem** menüpontból érhetők el (4.16. ábra).

Adatvédelem jelszóval

Az adatvédelem egyik lehetősége, hogy jelszóval lehessen hozzáférni az adatbázishoz. A jelszó beállítását csak akkor lehet elvégezni, ha a **Fájl\Megnyitás** parancsnál a **kizárólagos** jelölő négyzet be van jelölve. Ezután az **Eszközök\Adatvédelem\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.



4.16. ábra: Adatvédelmi lehetőségek

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 (4.17. ábra).



4.17. ábra: Munkacsoport adminisztrátor ablak

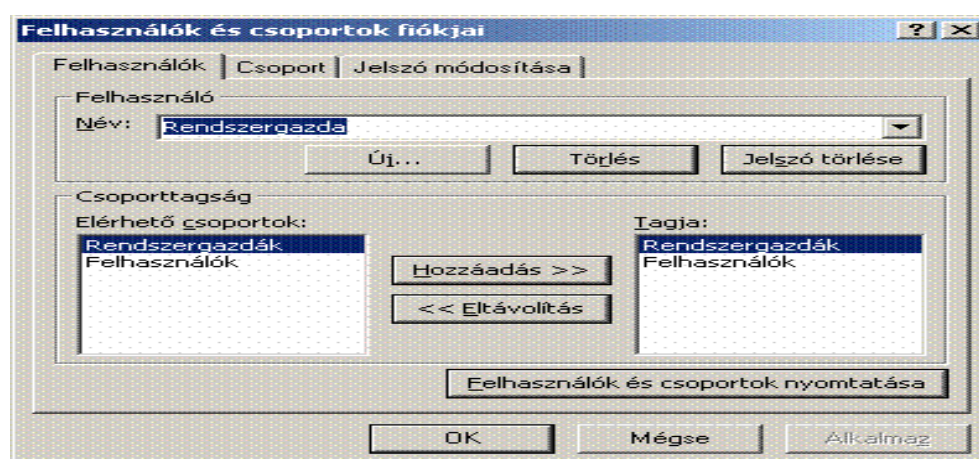
Az Eszközök\Adatvédelem\Munkacsoport adminisztrátor menü funkcióval lehet munkacsoportot létrehozni. A Létrehozás gomb megnyomása után meg kell adni a nevet, a szervezet nevét és a munkacsoport kódját.

Ha olyan adatbázist kell használni, amelyik rendelkezik az előzőekben megadott védelemmel, akkor először csatlakozni kell a hozzá beállított munkacsoport fájlhoz. Ez az Eszközök\Adatvédelem\Munkacsoport adminisztrátor funkció választása után a Csatlakozás gombot kell megnyomni (4.17. ábra).

Az adatvédelem használatához be kell kapcsolni a bejelentkezési eljárást. Ehhez a rendszergazda felhasználói jelszavát kell megváltoztatni.

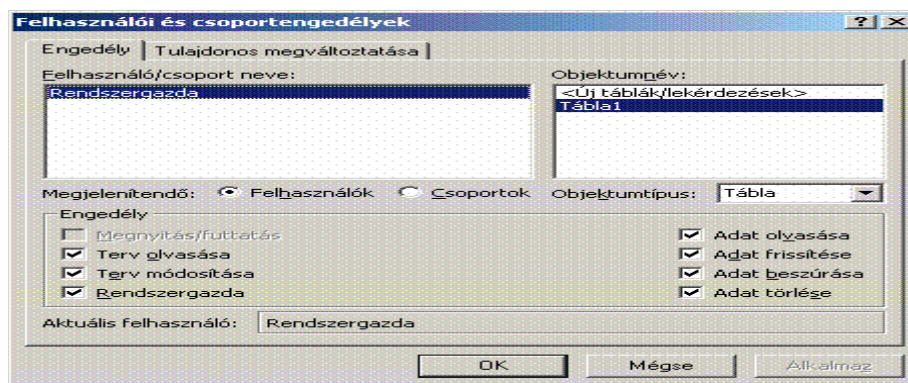
A jelszó módosítása az **Eszközők\Adatvédelem\Felhasználók és csoportok fiókjai** funkció választásával történik. A **Jelszó módosítása** fül kiválasztása után meg kell adni az új jelszót, az ellenőrzést és OK. Ezután az Access program elindítása után a megadott munkacsoporthoz csak a bejelentkezési párbeszédablak adatainak helyes kitöltése után lehet kapcsolódni. Ezután meg lehet adni a felhasználókat és a csoportokat az **Eszközők\Adatvédelem\Felhasználói adatvédelmi varázslóval**.

Az adatvédelem jó használatához meg kell adni az adatbázis felhasználóit és felhasználói csoportjait. Ez megtehető az **Eszközők\Adatvédelem\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 (4.18. ábra).



4.18. ábra: Felhasználók és csoportok fiókjai ablak

A felhasználóknak és a csoportoknak különböző engedélyeket lehet adni, amelyekkel hozzáférhetnek az adatbázis-objektumokhoz. Ezt az **Eszközők\Adatvédelem\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 (4.19. ábra). A **Felhasználói és csoportengedélyek** párbeszéd ablakban a **Tulajdonos megváltoztatása** párbeszéd ablakban van lehetőség az objektumok tulajdonosának a megváltoztatására.



4.19. ábra: Felhasználói és csoportengedélyek

Adatbázisok strukturált lekérdező nyelve az SQL

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*).

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.

Irodalomjegyzék

- 4.1. Bakó Sándor (1999): Adatbáziskezelés, Pedellus Novitas Kft, Debrecen.
- 4.2. Békéssy András - Demetrovics János (2005): Adatbázis-szerkezetek, Akadémiai Kiadó, Budapest.
- 4.3. Halasy Béla (2002): Adatmodellezés, Nemzeti Tankönyvkiadó Rt, Budapest.
- 4.4. Hector Garcia-Molina – Jeffrey D. Ullman – Jennifer Widom (2001): Adatbázisrendszerek megvalósítása, Panem Könyvkiadó Kft, Budapest.
- 4.5. Jeffrey D. Ullman – Jennifer Widom (1998): Adatbázisrendszerek, Alapvetés, Panem Könyvkiadó Kft, Budapest.
- 4.6. Michael J. Hernandez (2004): Adatbázis-tervezés, A relációs adatbázisok alapjairól földi halandóknak, Kiskapu Kiadó, Budapest.
- 4.7. Rajtik János (2004): Adatbázis-kezelés MS ACCESS 97, Pedellus Tankönyvkiadó Kft, Debrecen.

Ellenőrző kérdések

Milyen elvárások vannak az adatbázis kezelő rendszerrel szemben?
Az adatbázis kezelő rendszerek felépítése! Az egyes elemek feladatait ismertesse!
Mit értünk metaadatok alatt?
Milyen adatmodellek ismertek?
Jellemezze a hierarchikus adatmodellt!
Jellemezze a hálós adatmodellt (kétszintű hálós adatmodell, Codasyl-hálós)!
Jellemezze a relációs adatmodellt!
Mit értünk redundancián, inkonzisztencián, normalizáláson, normálformán?
Mit jelent a funkcionális függés?
Milyen objektum típusok érhetők el az Access Adatbázis ablakában?
Mit tartalmaz és milyen szerkezetű a tábla?
Mire használhatók a lekérdezések és az űrlapok?
Milyen eszközökkel védhetők az Accessben az adatbázisok?
Milyen engedélyek adhatók a felhasználóknak és a csoportoknak az Accessben?
Milyen adattípusok adhatók meg az Accessben?
Milyen mezőtípusok vannak az Accessben és mik a jellemzőik?
Milyen mező tulajdonságok adhatók meg a táblákban?
Milyen típusú lekérdezések készíthetők?
Mi a lényege a keresztáblás lekérdezésnek?
Hogyan kezdődik a lekérdezés létrehozása?
Milyen grafikus eszköz segíti a lekérdezés tervezői munkát?
Milyen szakaszok különíthetők el az űrlapon?
Mi az SQL?

5. SZÁMÍTÓGÉP-HÁLÓZATOK

5.1. 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:

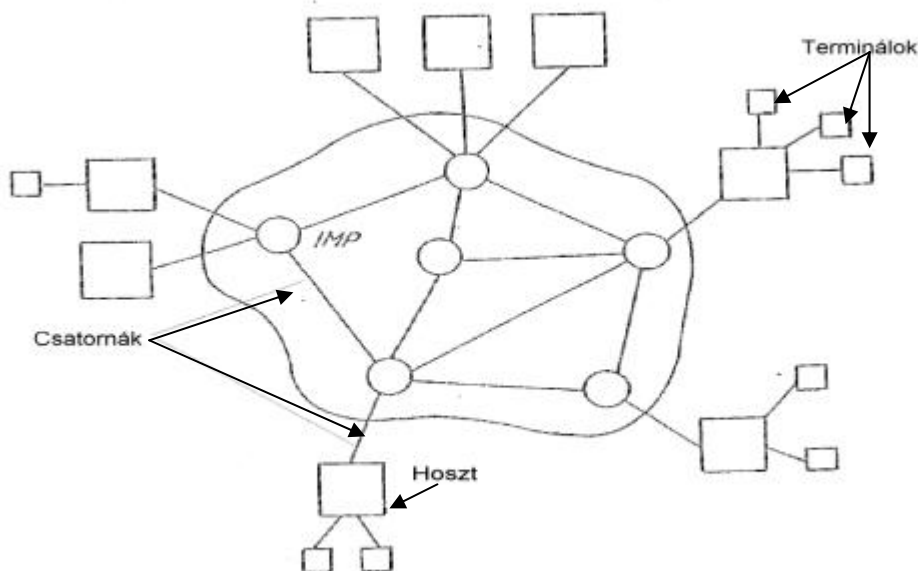
- **Erőforrás megosztás:** az eszközök (nyomtatók, nagy kapacitású tárolók), programok, adatok a felhasználók számára azok fizikai helyétől függetlenül bárki számára elérhetők legyenek.
- **Nagyobb megbízhatóság:** alternatív erőforrások alkalmazása (pl. fájlok több gépen való tárolása, egyszerre több 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.

5.1.1 Hálózati struktúrák

A hálózati struktúrák és a fogalmak meghatározásában a mai napig érvényesek az ARPA (Advanced Research Project Agency) által kidolgozott elvek.

- **Hosztoknak** (host) nevezzük azokat a gépeket, amelyekben a felhasználó program fut.
- **Kommunikációs alhálózat** (communication subnet) köti össze a hosztokat. Az alhálózat az összeköttetést biztosító csatornákból és kapcsológépekből áll. A csatornát (channels) szokás még vonalnak, áramkörnek, vagy több vonal esetén trónknak is nevezni. Figyeljük meg, hogy a hoszt nem része az alhálózatnak!
- **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).



5.1. ábra Kommunikációs alhálózat

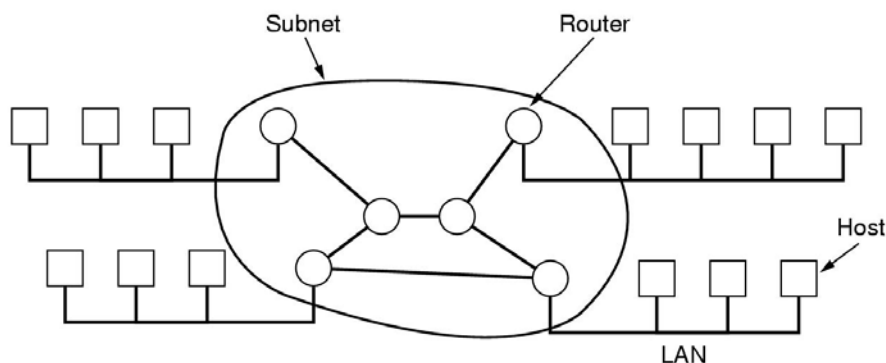
Egy IMP-hez egy hoszt is tartozhat, de egy IMP-hez több hoszt is kapcsolódhat. Az egy IMP – több hoszt a nagytávolságú hálózatokra jellemző.

5.1.2. Hálózati hardver

A számítógép-hálózatok osztályozásának szempontjai közül két fontos szempont az átviteli technológia és a hálózat mérete (kiterjedése) szerinti osztályozás.

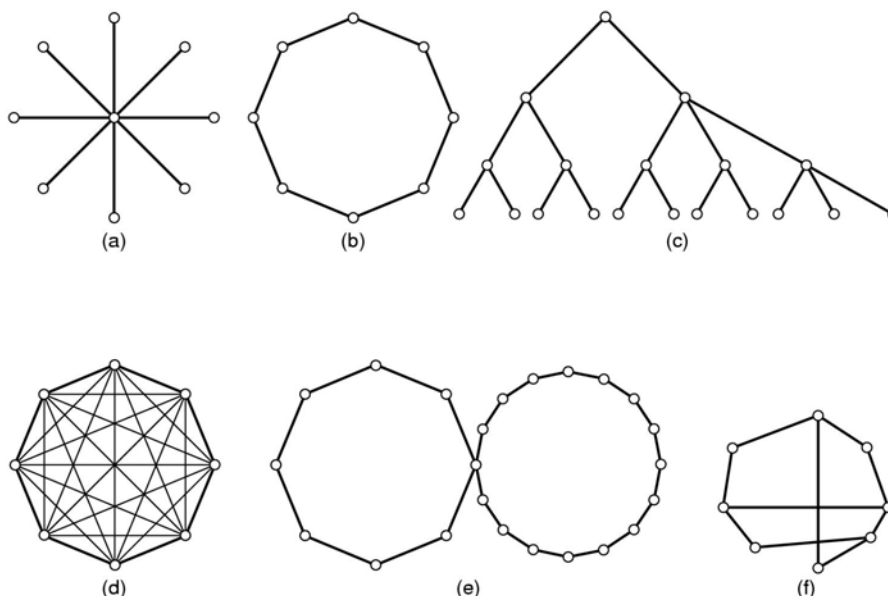
- Az átviteli technológia szerinti két fő csoport a következő:
 - **Adatszóró hálózatok.** Egyetlen kommunikációs csatorna, ezen osztozik a hálózat összes gépe. Ha bármelyik gép elküld egy rövid üzenetet, azt az összes többi gép megkapja. Ekkor a címezési eljárás úgynevezet csoportcímezés.
 - **Kétpontos 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.
- Méret:
 - Lokális hálózatok (Local Area Network, LAN). Az ilyen helyi hálózatok egy intézményen, vállalton 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). Ország, földrész. 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).



5.2. 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.



5.3. ábra Router kapcsolódási topológiák

(a) Csillag. (b) Gyűrű. (c) Fa. (d) Teljesen összekötött. (e) Egymást metsző gyűrűk
(f) Szabálytalan.

A nagy kiterjedésű hálózatok másik nagy csoportja a műholdas vagy földi rádiós rendszerek. Ezek adatszóró rendszerek.

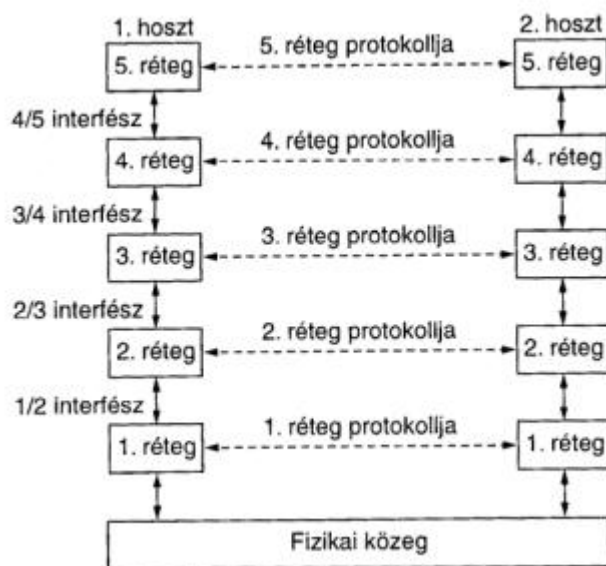
Összekapcsolt hálózatok. Egymástól különböző, sokszor nem kompatibilis hálózatok összekapcsolása, mely általában egy **átjárónak** nevezett (**gateway**) számítógép segítségével történik. Elnevezés: **internetwork**, **internet** (ilyen az Internet is).

5.1.3. 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 bulgár 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:



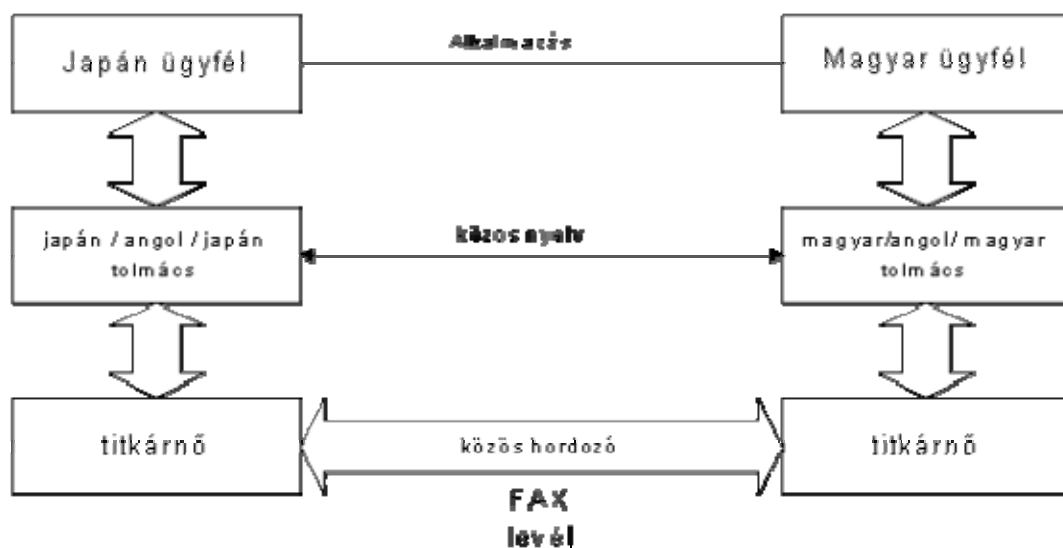
5.4. ábra Réteg szemléletű kommunikáció

Minden réteg formálisan a vele azonos szintű réteggel társalog. Az adat valójában végighalad az alatta lévő összes rétegen, de ezt a rendszer elfedi. Az 1. 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, amint az az 5.5 ábrán látható.



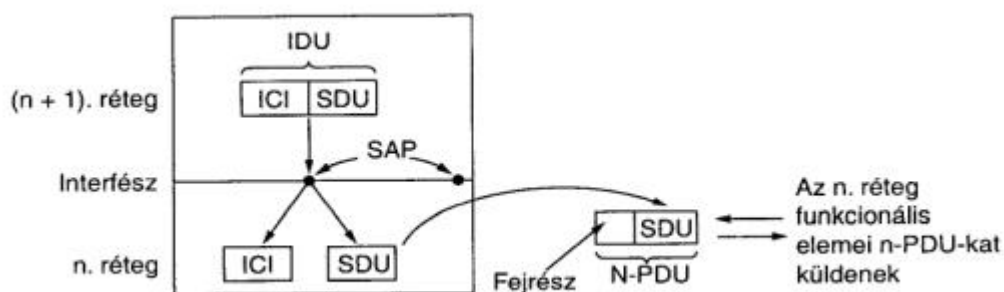
5.5. ábra Az emberi kommunikáció modellje

A rétegek és protokollok halmazát **hálózati architektúrának** nevezzük.

Egy hálózatban sok számítógép van, tehát 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 (l. 5.6. ábra), 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.



5.6 ábra. Rétegek kapcsolatai

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	Procol Data Unit	((protokoll adatelem)

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 azt. (telefon).
- **Összeköttetés nélküli szolgálat:** Minden egyes üzenet rendelkezik egy teljes címmel, és minden üzenet az összes többitől független útvonalon továbbítódik. (levél).

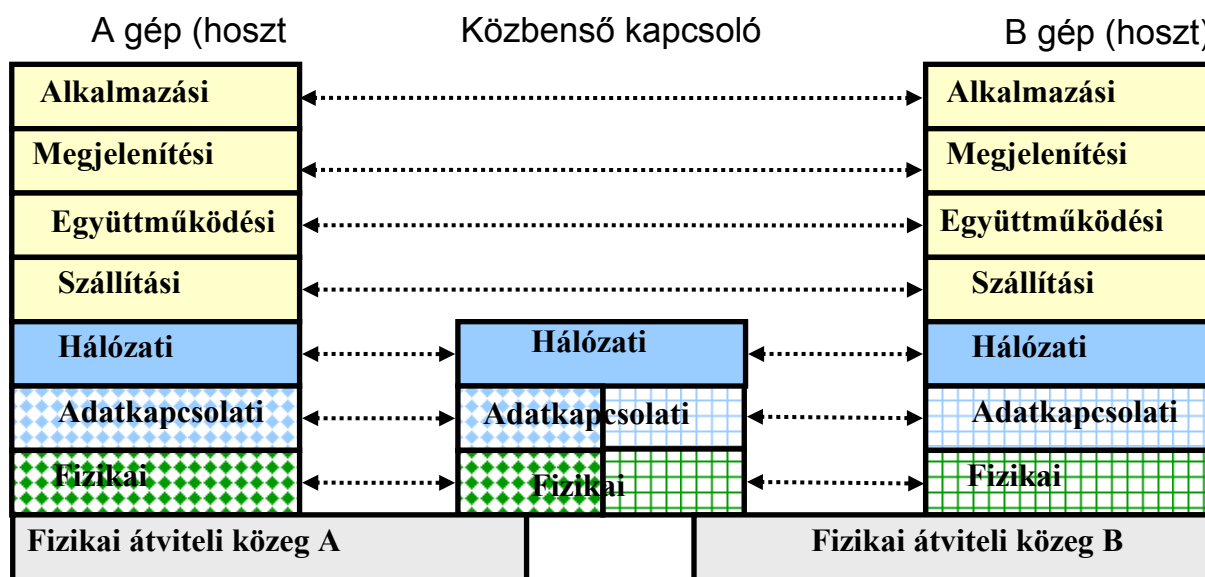
A szolgálatokat olyan szolgálatprimitívek, azaz elemi műveletek halmazával írhatjuk le, amelyek a szolgálatokat elérhetővé teszik a felhasználók számára. A protokollok olyan szabályhalmazok, melyek megadják, hogy a társ entitások az adott rétegen belül milyen formátumban, milyen jelentéssel küldenek 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.

5.2 Hivatkozási modellek

A gyakorlatban előforduló hálózati architektúrák valamennyien megfelelnek a rétegekből álló modellnek. Különbségek a rétegek számában, és a megvalósított funkciókban vannak.

5.2.1 Az OSI hivatkozásai modell

Az Open System Interconnection az ISO (International Standards Organization - Nemzetközi Szabványügyi Szervezet) ajánlása. A Nemzetközi Szabványügyi Szervezet az 1980-as évek elején fogadta el az OSI referenciamodellt, mely a számítógépek közötti adatkommunikációval foglalkozik. Az OSI modellnek hét rétege van.



5.7. 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-át 0-nak, az 1-et, 1-nek). 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. 5.7. 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 megérkezzék a másik oldalra. Továbbá, mind ezeket hatékonyan kell végrehajtania, ráadásul oly módon, hogy a viszonyréteg elől el kell fednie a hardvertechnikában elkerülhetetlenül bekövetkező változásokat.

Viszony réteg (más néven **együttműködési réteg**): A különböző gépek felhasználói viszonyt létesítenek egymással, 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 program 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.

5.2.2 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). A hálózat legyen alkalmas a fájl átviteltől a beszéd átvitelig minden feladatra.

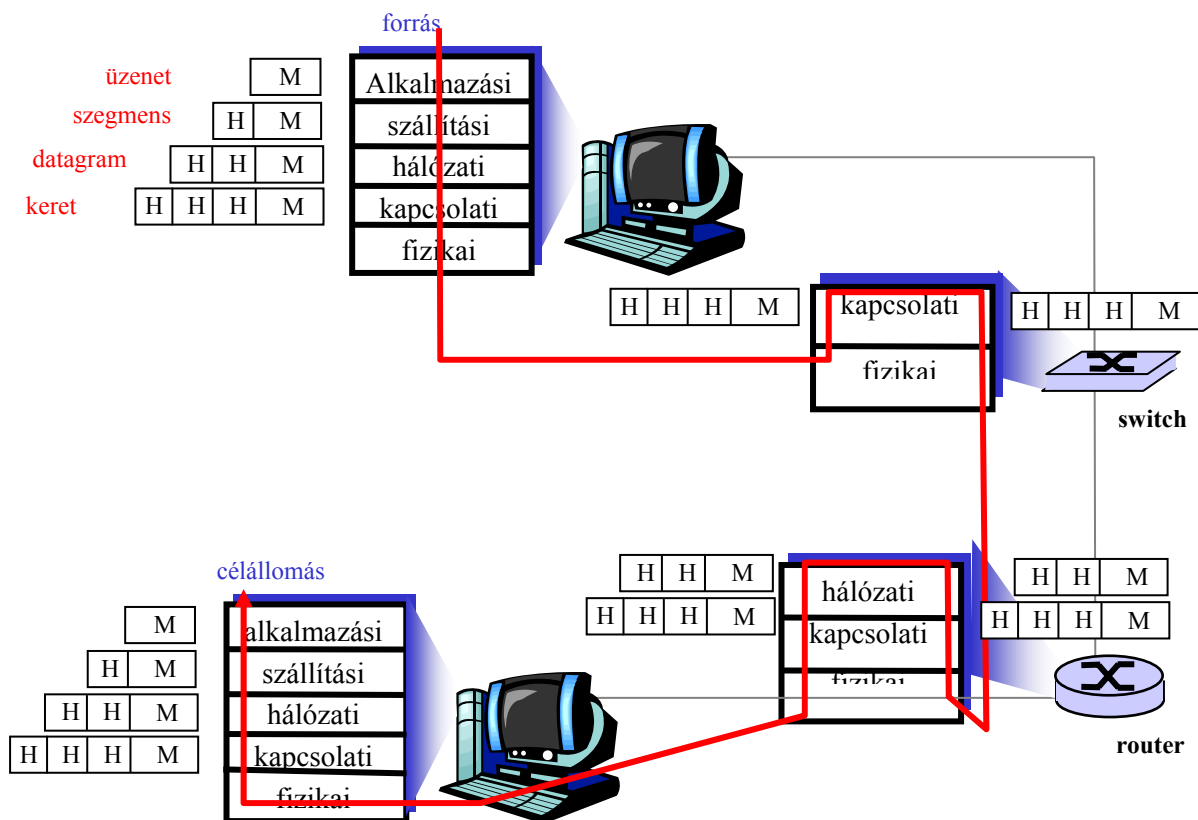
A megoldás egy 4-rétegű modell, ami természetesen különbözik az OSI ajánlástól (amely több mint 10 évvel korábbi). A TCP/IP és az OSI rétegek megfeleltetése vázlatosan az alábbi:

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	

5.8. ábra TCP/IP hivatkozási modell

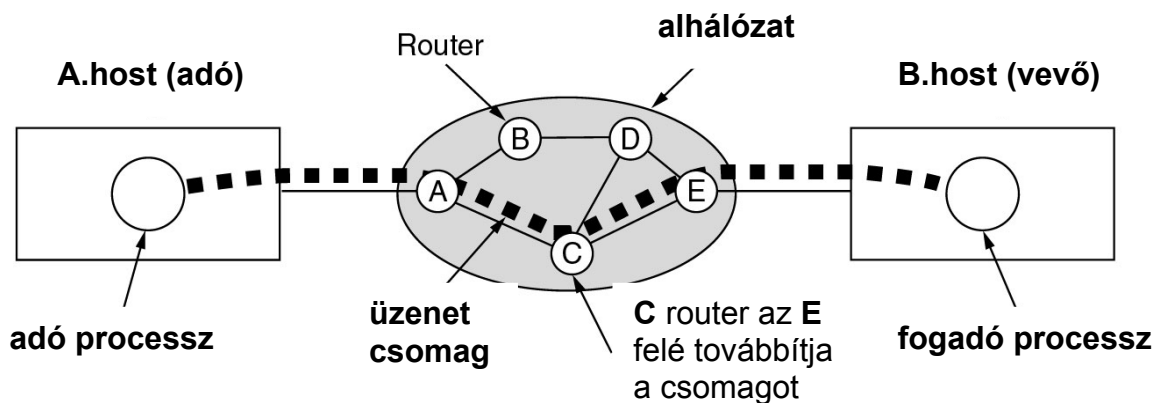
A megfeleltetés nem pontos. A rétegek feladatai azonban erősen hasonlóak.

- 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.
- 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).
- A szállítási réteg az Internet réteg felett helyezkedik el. Lehetővé teszi a társentitások 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 sorba-rendezést sem. Általában a kliens-szerver alkalmazásokban használatos, vagy ahol a gyors válasz fontosabb a garantált válasznál (címfeloldás, beszéd, videó). Pl.: egy címfeloldási kérésre nem kapunk választ, akkor megismétljük a kérést. Lényegtelen, hogy miért nincs válasz.
- 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:
 - Elektronikus levelezés (SMTP),
 - Domain Name Service (DNS),
 - hipertext (HTTP),
 - fájl transzfer FTP (FTP),



5.9. 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.

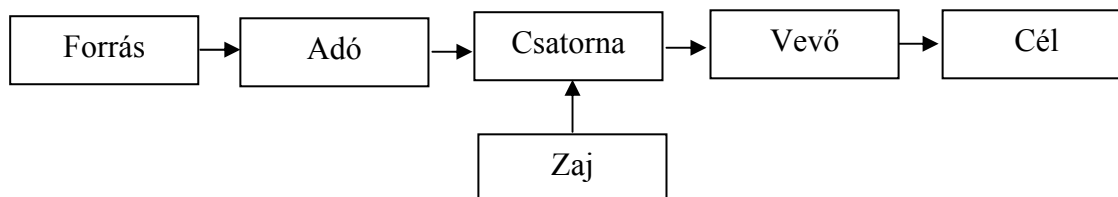


5.10. ábra Csomagtovábbítás a hálózaton

5.3. A hálózat fizikai megvalósítása

5.3.1. 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.



5.11. ábra Adatátviteli modell

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 átviteli módszerek a következők:

- vezetékes adatátvitel,
- vezeték nélküli adatátvitel.

A fontosabb fizikai átviteli közegek a következők:

Csavart érpár

Két spirálszerűen egymás köré tekert szigetelt rézhuzal. A két erezet 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.



5.12. ábra Csavart érpár

Az érpárok egymásra hatása csökkenthető, ha az érpárok távolságát növeljük. A kábel erek között távtartó van, mint az alábbi ábrán látható. A következő 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.



5.13. ábra: Koaxiális kábel

- **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.
- **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.
- **Üvegszál kábel**
Laboratóriumi körülmények között már a 100 Gb/s-os sebességet is elérték. A fényvezető szál adatátviteli rendszernek három fő komponense van: a fényforrás (LED vagy lézerdíóda) , az átviteli közeg és a fényérzékelő (fotodíóda vagy fototranzisztor). A fény terjedésének módja:
 - teljes visszaverődés (többmódusú szál, több különböző szögű fénysugár),
 - a szál átmérőjét néhány hullámhossznyiira lecsökkentjük (8-10 μ m, egymódusú szál, drágább, nagyobb távolságra használható).

A fényvezető kábelben általában több szálat fognak össze. Az ilyen vezetéknek 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 (1012-1014 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).
 - Látható fénycsugár átvitel. Pl. két épület lokális hálózatát a tetőjükre 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).

Az adatátviteli közegeken megvalósított átvitel típusa lehet az

- Analóg és
- 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 lokális hálózati kábel 11 nagyságrenddel jobb (3-4 nagyságrendnyi

sebességkülönbség, 7-8 nagyságrendnyi hibaaránybeli különbség) mint egy telefonkábel.

Digitális átvitel

A folyamatos jelek helyett 0-kból és 1-ekből álló sorozatok haladnak a vonalakon.

Előnyei:

- Hibákra érzéketlenebb (csak két állapotot kell megkülönböztetni).
- Tetszőleges jel átvihető (hang, kép, ...)
- A jelenlegi vonalakon jóval nagyobb adatátviteli sebességet lehet elérni.
- Olcsóbb (nem szükséges az analóg hullámformát pontosan helyreállítani).

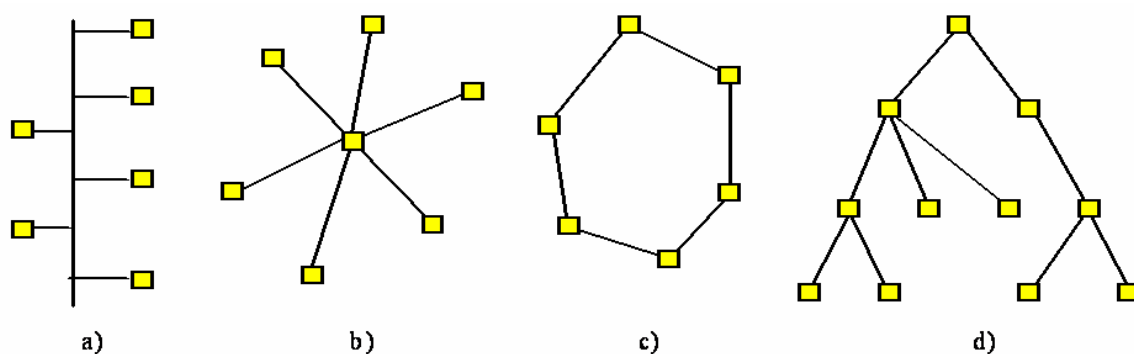
Modem

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, hiba javító és adattömörítő protokollok.

Az átvitel során mindig biteket viszünk át, de mivel eleinte szövegátvitelt valósítottak meg, ezért az átvitt információ egysége a bitsorozat volt, amely a szöveg egy karakterét kódolta. Az ilyen, bitsorozatokat átvivő módszert szokták **karakterorientált átviteli** eljárásnak nevezni. A hálózatok elterjedésével a szöveges jellegű információk mellett más jellegű információk átvitele is szükségessé vált, ezért a bitsorozat átvitel helyett a tetszőleges bitszámú üzenetátvitel került előtérbe, ezek a **bitorientált** eljárások.

5.3.2. Hálózat felépítés (topológiájuk)

A hálózat felépítését, topológiáját a kábelek elrendeződése, a csomópontok fizikai elhelyezkedése határozza meg. Ez a "hálózat alakja". Az Ethernet lineáris, vagy sín hálózati, ill. csillag topológiát alkalmaz (5.14. ábra).



5.14. ábra: Hálózati topológiák

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. Minden csomópontnak egyedi címe

van. Olcsó, kevés kábel kell hozzá. Hiba esetén az egész hálózat működésképtelen lesz. (sin-busz)

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, sok kábel kell hozzá ezért drága. (ábra)

Gyűrűs :

a csomópontokat közvetlenül egymáshoz csatlakoztattják, 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. (ábra)

Busz:

egyetlen kábelre felfűzött gépek alkotják a hálózatot. Gyakori a helyi hálózatokban, mivel olcsó a kialakítása. Hátránya, hogy a kábel megbontása, azaz bármely gép kivevése a hálózathoz a hálózat működésképtelenségét eredményezi (ábra)

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.

5.3.3. 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



5.15. 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 újrarendőztetését végzi.

router:

egy intelligens eszköz, amely meghatározza a hálózaton áramló adatcsomagok 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 koaxiális kábel nem azonos a TV készülékekhez használt koax kábelrel.
- **T elágazó:** a koaxiális 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 segédtábla, 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 csatlója (kártya) közötti összeköttetést biztosítja.

vezérlő egység:

ide tartoznak a hálózatot vezérlő számítógépek (server) - több is lehet egy hálózatban.

működtető program:

a hálózat működését, az őt működtető program, a határozati 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, kábeleztől, a hálózat bonyolultságától függően kell megválasztani.

5.4 Hálózatok összekapcsolása

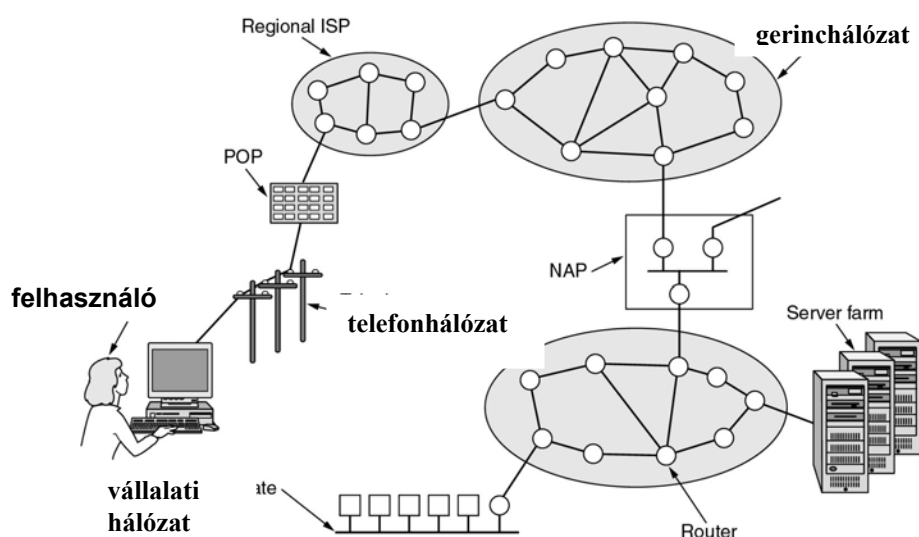
A hálózatok, hálózatrészek összekapcsolására szolgáló eszközök:

- **Ismétlők (repeater):** Egyedi biteket másolnak kábelszegmense közt. Alacsony szintű eszközök (1. réteg), amelyek csak erősítik vagy újragenerálják 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 továbbítanák a másik hálózaton.
- **Többprotokollos routerek:** Eltérő hálózatok között továbbítanak csomagokat. Elvükben hasonlóak a hidakhoz, kivéve, hogy a hálózati (3.) rétegben működnek. Az egyik vonalon veszik a beérkező csomagot és egy másik vonalon továbbítják őket, mint ahogy azt minden router teszi, de a vonalak különböző hálózatokhoz tartozhatnak és különböző protollokat 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.)

A hozzáférés az Internethez (kapcsolt vonali, illetve közvetlen vonali hozzáférés):

- **közvetlen kábelkapcsolat:** helyi hálózatoknál a leggyakoribb forma. Olcsó, könnyen kivitelezhető és a fenntartása nem jár költségekkel.
- **nagy távolságú kábelezés:** ritkán alkalmazzák. A kábelezés bonyolultsága miatt igen költséges.
- **optikai kábelezés:** a nagytávolságú kábelezéshez hasonló tulajdonságokkal rendelkezik. Költséges, de megbízható és rendkívül gyors adatátvitelt tesz lehetővé
- **mikrohullámú lánc:** olyan földrajzi viszonyok között alkalmazzák, ahol a kábelezés nem megoldható. Költséges ugyan, de gyors adatátviteli forma.

- **műholdas kapcsolat:** a mikrohullámú lánc műholdas változata. A már meglévő műholdas rendszerekhez igen könnyen hozzákapcsolható új végpontok kialakításával.
- **ISDN vonal:** viszonylag gyors, de hosszú idejű adatátvitelnél költséges a használata, hiszen a szolgáltatást biztosító telefontársaságok a normál telefonvonalhoz hasonlóan kezelik, ill. számlázzák.
- **bérelt vonali összeköttetés:** ott alkalmazzák, ahol fontos az állandó kapcsolat. Sebessége a kiépítésnél meghatározott sávszélesség függvénye. Fenntartási költsége az adatátviteltől független, állandó.
- **kapcsolt vonali összeköttetés:** egyszerűbben, a telefonvonalon modemeken keresztül létesített adatátvitel. Mivel a kapcsolat normál telefonvonalon történik, az adatátvitel minden egyes percéért fizetni kell.



5.16. ábra Hálózatok Internet hálózatba kapcsolása

A tűzfalak

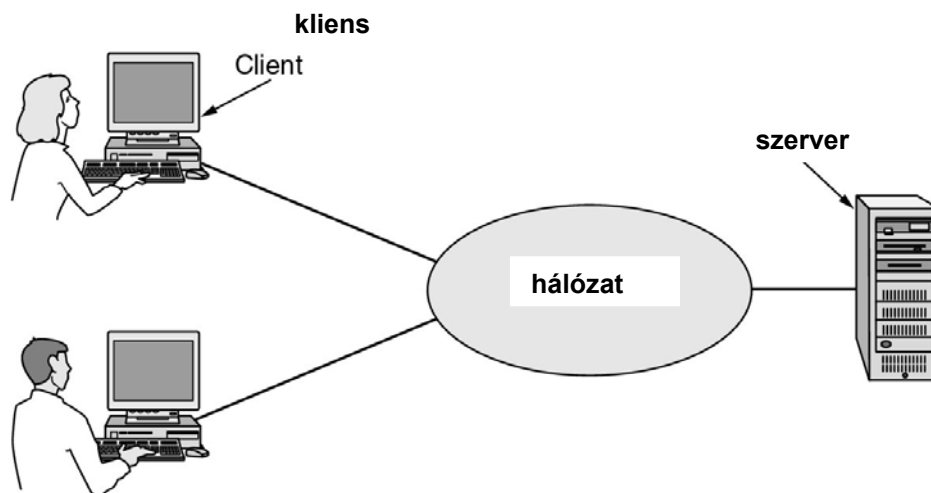
Védik a lokális hálózatot és a kimenő forgalmat is szűrik. A tűzfalak elemei:

- **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.

5.5. 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 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 értelem szerűen 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.



5.19. ábra A kliens-szerver architektúra

Sok számítógépes rendszer és hálózati szolgáltatás működik ezen architektúrával.

Irodalomjegyzék

- 5.1. Tannenbaum, A.S.: Számítógép hálózatok, Novotrade, Budapest, 1992.
ISBN 963 585 162 6
- 5.2. Kovács Péter: Számítógép-hálózatok Praktikus útmutató hálózatépítéshez,
Internet kapcsolatok beállításához, vezetékes és vezeték nélküli technológiákhoz,
Computerbooks, 2003

Ellenőrző kérdések

- Mi a számítógépi hálózatok célja?*
- Milyen hálózati struktúrákat ismer?*
- Mi az a hálózati architektúra?*
- Sorolja fel miért előnyös a számítógépeket hálózatba kapcsolni!*
- Melyek a számítógépes hálózatok legfontosabb jellemzői?*
- Mi a topográfia és a topológia közötti különbség?*
- Határozza meg a protokoll fogalmát!*
- Mik azok a hálózati rétegek?*
- Mi a hálózati architektúra?*
- Miért fontos a hálózatok szabványosítása? Mik azok a de-jure és a de-facto szabványok?*
- Fogalmazza meg az OSI-modell filozófiáját!*
- Milyen rétegekből épül fel az OSI modell?*
- Mi a fizikai réteg feladata?*
- Mi az adatkapcsolati réteg szerepe?*
- Mit biztosít a hálózati réteg?*
- Milyen célokat valósít meg a szállítási réteg?*
- Mire használják az együttműködési rétegeket?*

*Milyen feladatokat lát el a megjelenítési réteg?
Ismeresse az alkalmazási réteg szerepét az OSI modellben!
Milyen rétegekből áll az OSI hivatkozási modell?
Mi az egyes rétegek feladata?
A TCP/IP hivatkozási modell és az ISO/OSI összehasonlítása.
A hálózatok hardver elemei.
A hálózatok összekapcsolási lehetőségei.
Mire szolgál a kliens-szerver modell?*

6. AZ INTERNET

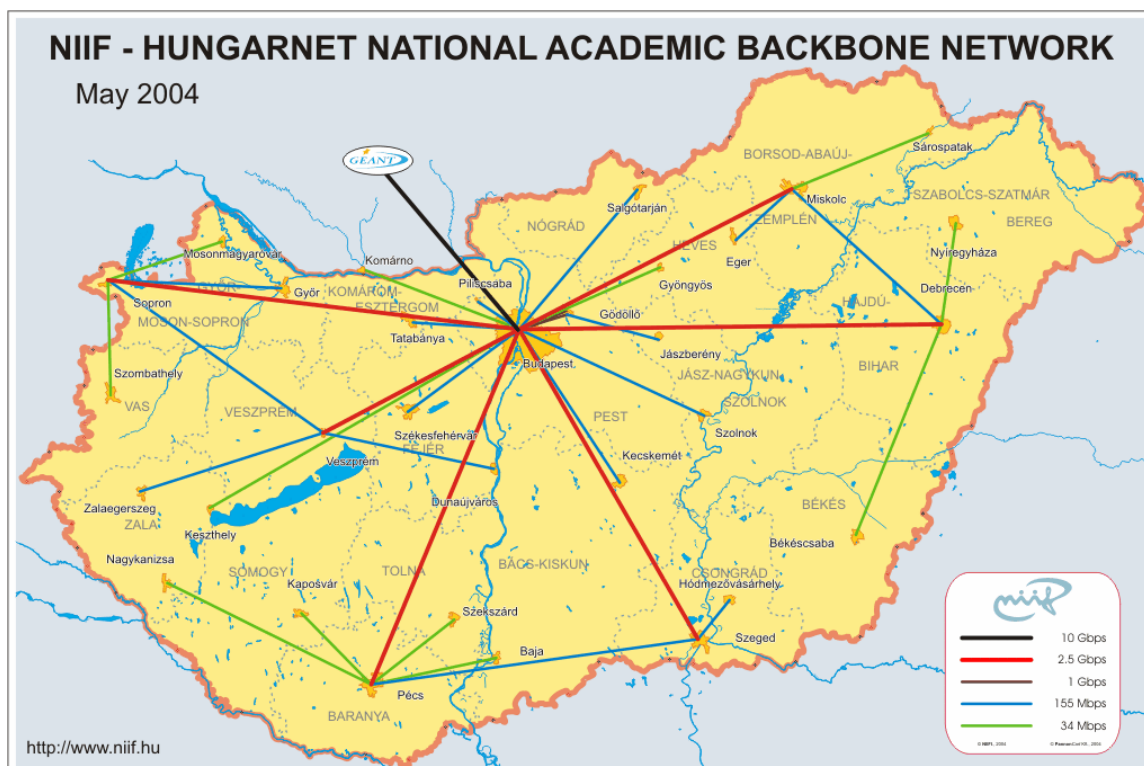
A világot behálózó rendszerben nagyszámú különböző hálózat áll egymással összeköttetésben átjárók (gateway) segítségével. Ezeken a hálózatokon lévő bármely számítógépet vagy erőforrást a felhasználónak el kell tudnia érni. Az adatsomagok esetleg több tucat hálózaton is keresztülmehetnek mielőtt a célállomásra 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.

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 címzetthez. Figyel arra, hogy mi került át, és ami nem jutott el a címzetthez, 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 protokolloknak 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 protokollok (pl. levelezés);
- a TCP-hez hasonló protokollok, amelyek rengeteg alkalmazás számára biztosítanak szolgáltatásokat;
- IP, amely a datagrammok célba juttatását biztosítja;
- a felhasznált fizikai eszközök kezeléséhez szükséges protokollok (pl. Ethernet)

6.1. A hazai internet hálózat

A hazai oktatási és tudományos célú Internet felépítését a 6.1. ábrán láthatjuk. A legfontosabb része a nagy adatátviteli sebességű, általában optikai kábelekből, és műholdas kapcsolatokból álló gerinchálózat (backbone), amely az ide kapcsolódó hálózatok információit szállítja.



6.1. ábra: A Nemzeti Informatikai Infrastruktúra (NIIF) gerinchálózata

A csomagokat routerek irányítják a különféle útvonalakon. Azonban kevés felhasználónak adatik meg a gerincre 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.

6.1.1. Telefonvonalon való kapcsolódás

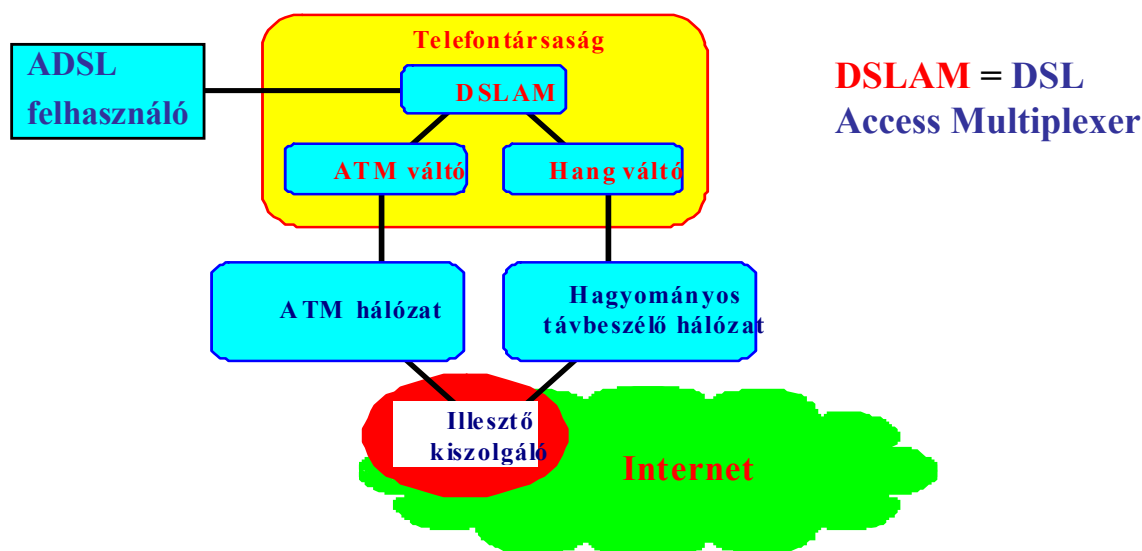
A megfelelő hálózati teljesítmény eléréséhez csak nagyteljesítményű gépekkel lehet a gerincvonalakra csatlakozni. Az átlagos felhasználók ezért a helyi hálózati kapcsolataikat használhatják fel, míg egyéni felhasználók számára az Internet szolgáltatók (providerek) által üzemeltett nagyteljesítményű gépeken keresztül való csatlakozás a megoldás.

A legegyszerűbb szolgáltatás a levelezés: ez lényegében hálózati kapcsolatot nem igényel. Általában egy Internet szolgáltató számítógépén elhelyezett postaládát használunk: ennek tartalmát modemcsatlakozáson keresztül kezelhetjük. A modemcsatlakozáson keresztül kapcsolódunk a szolgáltató gépére és a leveleket egy menetben fel-, illetve letöltjük, terminálként (vagy a szolgáltató speciális szoftverén keresztül) bejelentkezünk a szolgáltató gépére, és arról böngésszük a hálózatot. Az egyszerű akusztikus modemmel való csatlakozás nagy hátránya a lassúság (max. 52 Kbit/sec), valamint az, hogy a szolgáltató modemjével a telefonvonalon – közönséges tárcsázással – pont-pont kapcsolatot tud létrehozni, s ez rendkívül költséges.

6.1.2. 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.



6.3. ábra: Az ADSL technika alap gondolata

Miután az ADSL a hagyományos telefonvonalat használja, csak más frekvenciatartományban, fontos tudni, hogy az infrastruktúra kiépítettségétől függ a szolgáltató által biztosított adat letöltési sebesség (milyen távol van a csatlakozó egység a felhasználótól). Erre példa a 6.1. táblázatban látható.

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.

6.1. táblázat: Az ADSL technikával elérhető sebességek

Adatátviteli távolság	Elérhető adatssebesség
2km	8Mbit/s
3,6km	6Mbit/s
5,5km	1,544Mbit/s

6.1.3. 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.

A 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, a 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.

6.1.4. 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 internet hozzáférés. Míg a 802.16-os szabvány (a WiMAX) akár 50 kilométeres körzetben is adhat hálózati hozzáférést, a WiFi (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.

6.1.5. 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 – alapjában 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.

6.1.6. 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.

GPRS (General Packet Radio Service)

Általános csomagkapcsolt rádiószolgáltatás. Csomagkapcsolt adatátviteli technológia, amely nagy sebességű (115 kbps vagy gyorsabb) vezeték nélküli internetes és egyéb adatkommunikációt tesz lehetővé.

UMTS (Universal Mobile Telecommunications Service)

A harmadik generációs (3G) mobiltelefon szolgáltatás európai változatának elnevezése. Az **UMTS** rendszer egy a GSM-nél nagyságrendekkel gyorsabb és fejlettebb mobil hálózat

6.2. 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 csomópontra van szükség. A hálózatazonosító az összekapcsolt hálózatok között, a csomópont-azonosító 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.

Az Internetben a rétegeknek megvan az egyedi azonosítója a címzéshez:

Réteg	Címzési módszer
Alkalmazási	Hoszt neve, portja
Internet	IP cím
Hálózatelérési	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 csatlakozója van, és mindegyik más (helyi) hálózathoz csatlakozik. Az útválasztó csomagokat fogad az egyes, hozzá csatlakozó hálózatok számítógépeitől, és továbbítja őket egy másik hálózati csatlakozóján. Az, hogy melyik hálózati csatlakozót kell használni a csomag elküldéséhez, a memóriájában lévő útválasztási tábla (routing table) alapján dönti el. A tábla bejegyzései hálózatok felé vezető útvonalakat (route) képviselnek.

6.3. 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) nevezik.

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

6.4. 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. Leveletet tud küldeni és fogadni.
- 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 lemezegegye között. Pl.: fájlokat vihet át, törölhet, átnevezhet fájlokat.
- GOPHER Hierarchikusan felépített információban kereső protokoll
- HTTP HyperText Transport Protocol

A következőkben ezen protokollok és ezeket használó szolgáltatások közül a legfontosabbakat tárgyaljuk.

6.4.1. 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 tekintsük át. Tegyük fel, hogy a **edu.ensmp.fr** nevű számítógép szeretné az alábbi üzenetet elküldeni.

Date: Fri, 14 Apr 2006 17:15:17 +0200

From: Marie-Francoise GUENEGAN <Marie-Francoise@edu.ensmp.fr>

User-Agent: Mozilla Thunderbird 1.0.6 (Windows/20050716)

X-Accept-Language: fr, en

To: gipsz@alpha.uni-corvinus.hu

Subject: International Conference: HSIMP

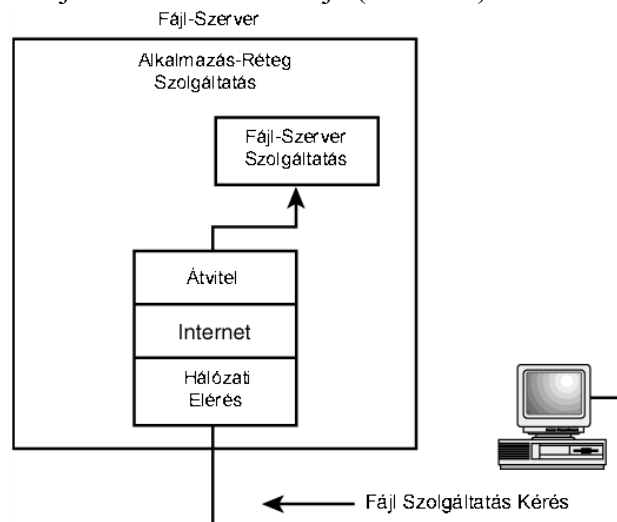
In view of your international scientific reputation, I have the pleasure of inviting you to join the Advisory Committee of a new Conference that we are organizing next year....

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 **gipsz@alpha.uni-corvinus.hu**. 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 **alpha.uni-corvinus.hu** 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. A levelezési listák, illetve az olyan általános nevek, mint "postmaster" vagy "operator" kezelésére is felkészült a rendszer.

6.4.2. Állományok átvitele - FTP - File transfer protokoll

Az FTP protokoll a hálózatban lévő gépeken megtalálható fájlok átvitelére használható. Használata az Email-el 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 (6.4. ábra).



6.4. 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, azaz a fájl nevében tetszőlegesen lehetnek kis és nagybetűk.

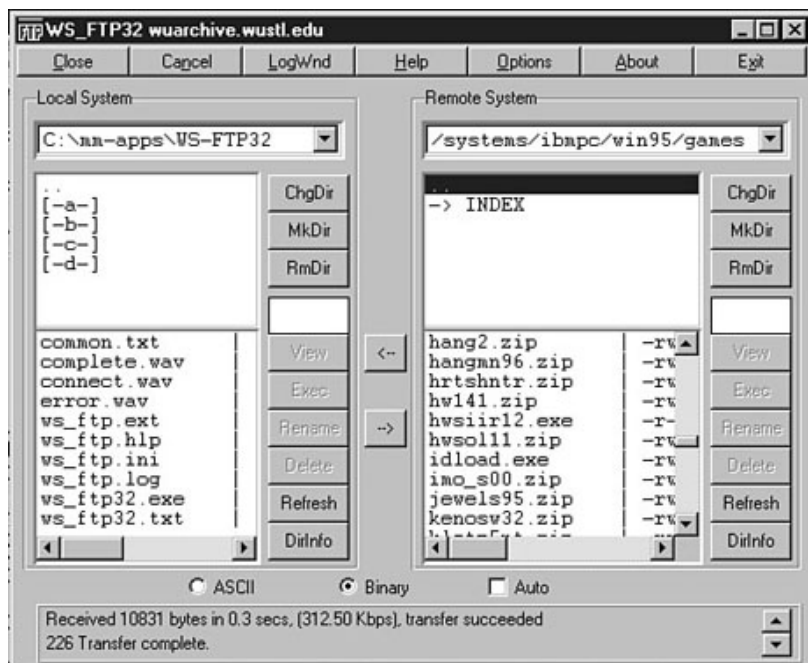
A felhasználó általában akkor tud egy távoli gépről/gépre másolni, ha a távoli gépen is rendelkezik felhasználói jogosultsággal (account-tal).

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 6.5. ábra.

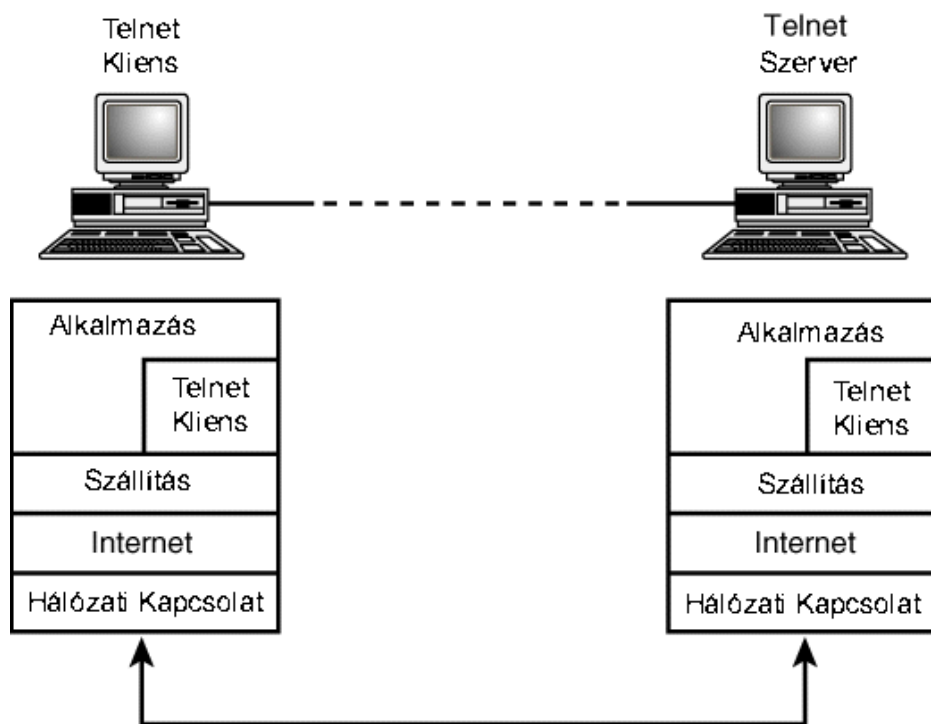


6.5. ábra: File átvitel Windows-ban

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.

6.4.3. 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 (6.6. ábra).

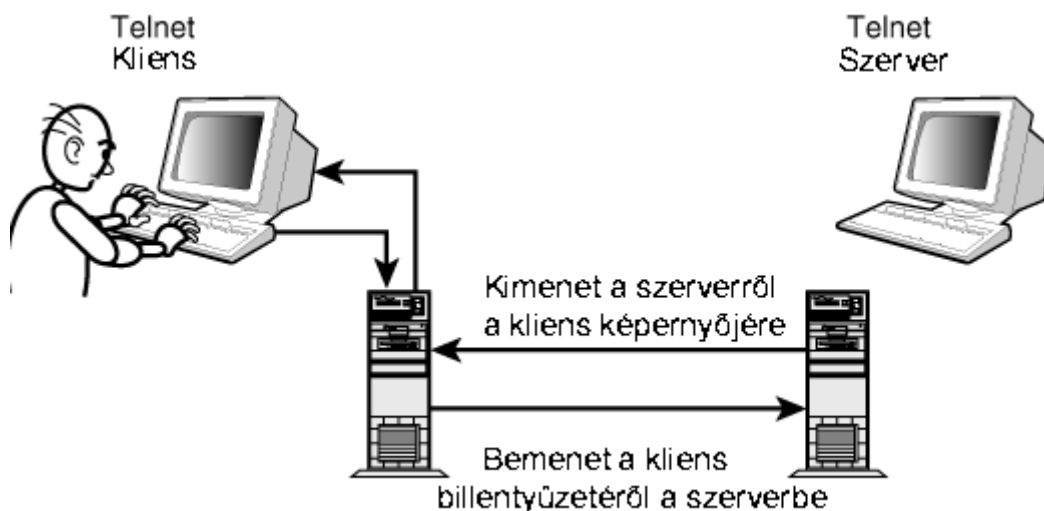


6.6. ábra: A TELNET kliens-szerver szolgáltatás

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 az távoli operációs rendszer hajtja végre (6.7. 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) lépünk 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.



6.7. ábra: A kliens-szerver kommunikáció

6.4.4. A World Wide Web

A WWW az Internet világban forradalmi változást hozott. Hatására az Internet akadémiai, kutatói hálózattól 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 helyszínre, 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 WWW-hez javasolt 80-as port címet használják akkor ezt az URL-ben a kiszolgáló nevéhez vagy címéhez kettősponttal (:) kapcsolva kell megadni.

A forrás helyét a kiszolgáló lemezegeységének 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) direktíváival 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 Hyper Text Transport 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 fájl-rendszerén, 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.

6.4.5. A WWW alkalmazások fejlesztésének eszközei

A HTML

A dokumentumok logikai struktúráját a HTML (Hyper Text 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, olyan módon hogy 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 mozgóképek (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, illetve a Windows alatt futó programok súgói. Amennyiben a szöveg mellett más objektum is megjelenik, akkor hypermédiáról 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, lejátszsa). 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.

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 programokkal, 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 több előnyt is eredményezett. Egyik előny az, hogy tehermentesítik a szervert, 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.

A nyelv általános jellemzői:

- **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 interpreterét kell használnunk, ez egy olyan speciális program, amely a Java virtuális kódját processzor-specifikus bináris utasításokká fordítja. Az így futtatott alkalmazásoknak olyan képességeik vannak, amelyekkel az appletek nem rendelkeznek, például fájlműveleteket tudnak végezni.

6.4.6. Valós idejű kapcsolattartás

6.4.6.1. 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 sávszélességet 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.

6.4.6.2. IPv6

Az IPv6 vagy másnéven Internet Protokoll 6 az internet új generációs protokollja. Az IETF (Internet Engineering Task Force) - testület, amely felügyeli és ellenőrzi az Internetet - kifejlesztette az IPv6-ot ami a már 20 éves IPv4-et fogja felváltani. Az IPv4 egy ideig rugalmasnak bizonyult, azonban mára már tapasztalhatjuk a korlátait. Különösen kritikus probléma az, hogy az új internetes eszközök (a 3G technológia és a "mobil" internet) következtében "kifogyunk" az internet címekből. A jövő ugyanis az hogy minden technikai eszközt irányítani tudunk majd valamilyen kezelőfelületről, és ehhez az kell hogy minden eszköznek legyen egy címe. Az IPv6 128 bit-es címet használ a megszokott 32 bites IPv4-es címek helyett. Ez 2^{96} - (durván $7,92 \times 10^{28}$) szorosára növeli az elméletben rendelkezésre álló címtartományt. Elméletileg a Föld minden egyes homokszemcséjének jutna egy IPv6-os cím. Lényeges, hogy megszűntek különböző méretű hálózatokon alapuló tartományok (A, B és C tartományok). Az IPv6 továbbá olyan lehetőségeket is nyújt, mint például a hitelesítés és biztonság, Alapjában az IPv6 megőrzi az IPv4 és protokoll alaptulajdonságait. Továbbra sem változott az alapelv, hogy az IP-címeket nem gépekhez, hanem hálózati interfacekhez rendelik. Egy interfácnek viszont több címe is lehet, és ezt az új szolgáltatások ki is használják.

Irodalomjegyzék

- 6.1. Tanenbaum, Andrew S.: Számítógép-hálózatok, Budapest, Novotrade Kiadó Kft., 1992
- 6.2. Novosel, Gary - Hudson, Kurt – Stewart, James Michael: TCP/IP (MCSE, adaptív teszt változat), Budapest, Kiskapu Kft., 1999
- 6.3. Pitter, Keiko – Amato, Sara – Callahan, John – Kerr, Nigel – Tilton, Eric: Egyszerűen Internet, Budapest, Panem Kft., 1996
- 6.4. Revoly András – Tarr Bence: Egyszerűen középhaladó Internet, Budapest, Panem Kft., 1998
- 6.5. Lengyel, V.: Az Internet világa, Computerbooks 1998
- 6.6. Moncur, M.: Tanuljuk meg a Javascript programozását, :Kiskapu „002, ISBN:9639301426

Ellenőrző kérdések

Hogyan használható a telefonvonal az internet eléréshez?

Mi az az ADSL?

Hogyan működik a mobil internet kapcsolat?

Melyek az internet szolgáltatások?

Mi az FTP?

Mi a World Wide Web?

Hogyan programozható a Web?

Hogyan működik az internet telefonálás?

7. INFORMÁCIÓS RENDSZEREK

7.1. Információs rendszer fogalma

Az információs rendszer definiálására több megfogalmazás született, pl

„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” [7.1]

Davis és Olson definíciója szerint [7.2]

„A 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 [7.3] együttesen vezetést (menedzsmentet) támogató rendszereknek (Management Support Systems - MSS) nevezte.

7.2. Az információrendszerek fő típusai

- Felsővezetői információrendszerek - executive support systems (ESS)
- Vezetői információrendszerek - management information systems (MIS)
- Döntéstámogató rendszerek - decision support systems (DSS)
- Tudásalapú rendszerek - knowledge work systems (KWS)
- Irodáautomatizálás - office automation systems (OAS)
- Tranzakció feldolgozó rendszerek - transaction processing systems (TPS)

A vezetést 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 jelennek meg, de néha ezek egymásba integráltak.

A vezetési 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 vezetési 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.

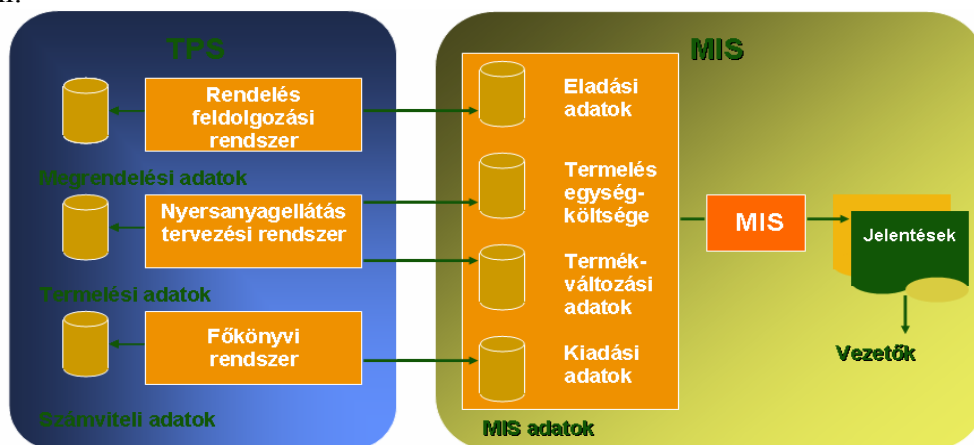
Azonban a környezet amelyben a vezetésnek dolgoznia kell változik. Mivel az információ és a mikroelektronika korszakában élünk ezért ez a technológia életstílusunk meghatározójává válik. Az üzleti élet és környezete sokkal komplexebb ma mint volt bármikor ezelőtt és a fejlődési irány a komplexitás irányába mutat.

A döntés és döntés-előkészítés Turban szerint három ok miatt nehezebb [7.3]:

- A lehetséges alternatívák száma sokkal nagyobb ma mint volt bármikor ezelőtt a fejlettebb technológiai és kommunikációs rendszereknek köszönhetően.
- A döntések jövőbeni következményeit nehezebb megítélni a megnövekedett bizonytalansági tényezők miatt.
- A hibaelkövetés költsége nagyon nagy lehet a tevékenységek, az automatizálás és a láncreakció komplexitásának és terjedelmének következtében, amelyet egy hiba okozhat a szervezet számos részében.

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 vezetés tudománynak 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 alkalmazza nagymennyiségű 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.



7.1. ábra: Tranzakciós adatok a vezetői információrendszerek számára

Döntéstámogatás

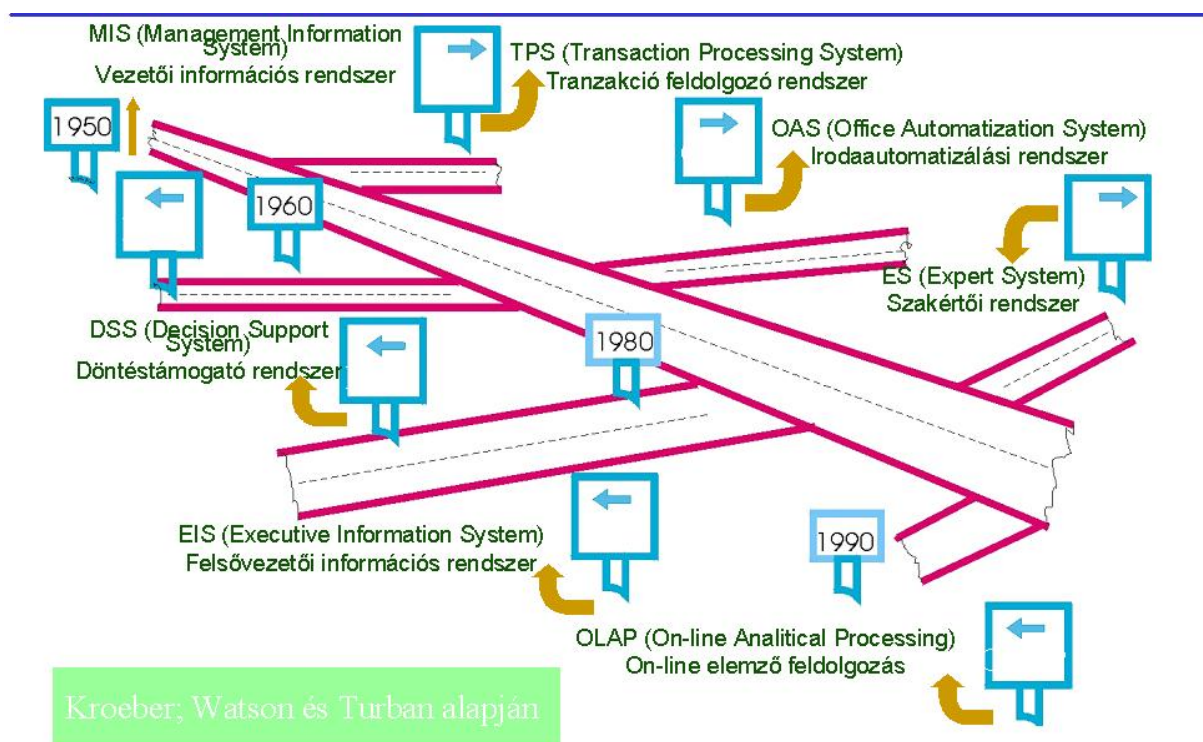
Simon szerint [7.4] a döntés-előkészítési folyamatok egy összefüggő sorozatba esnek amely a nagymértékben strukturált (gyakran programozottnak nevezett) döntésektől a nagymértékben strukturálatlan (nem programozott) döntésekig terjednek. *A strukturált folyamatok* a rutin ismétlődő feladatok amelyekre 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 három fázisra osztható:

- feltételek keresése amelyek a döntéshez szükségesek
 - a lehetséges tevékenységek felderítése, kidolgozása, elemzése
 - a lehetséges akcióprogram kiválasztása
- Intelligencia,
Tervezés,
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 strukturálnak nevezi [7.5].

A számítógépeket már több mint két évtizede használják eszközként a vezetői döntés-előkészítések támogatására. A számítógépesített eszközök vagy döntési segítségék Kroeber és Watson szerint hat kategóriába sorolhatók [7.6].



7.2. ábra: Információs rendszerek fejlődési útja [7.6]

E rendszerek (számítógépre alapozott információs rendszerek) fejlődési útját az 7.2. ábra szemlélteti. A TPS, MIS, DSS, EIS és ES attribútumai néhány dimenzió szerint csoportosíthatók, amelyek közül a legjellemzőbbeket az 7.1. táblázat foglalja 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.
-

7.3. 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 címzetre, 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 döntés-előkészítő tevékenységekben -, másrészt magában a döntéshozóban rejlik.

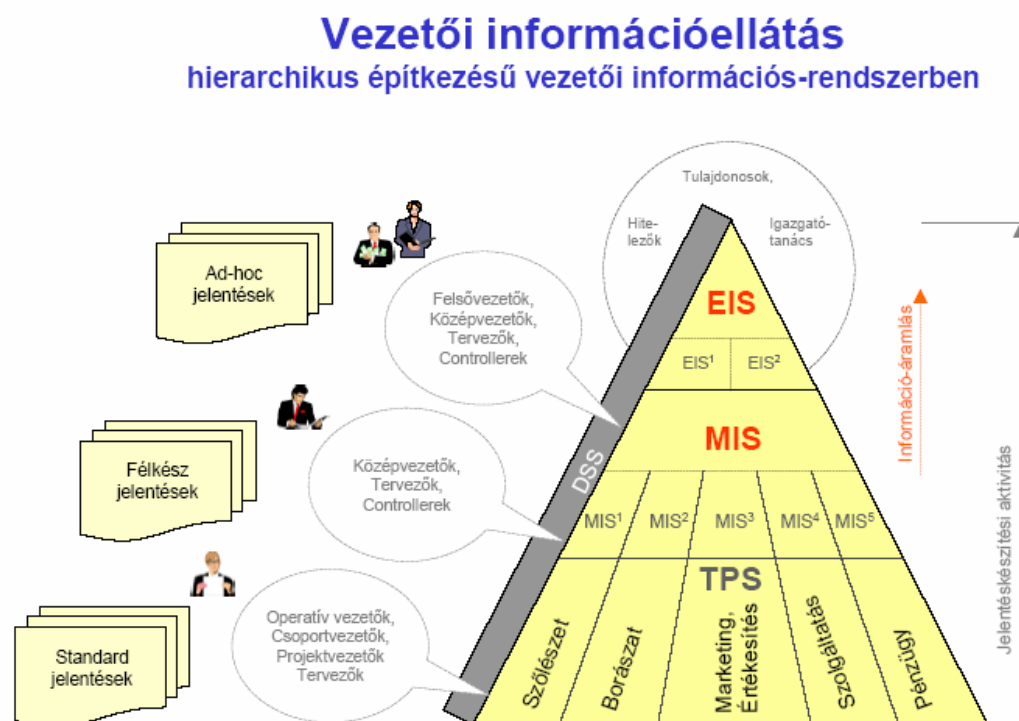
7.1. táblázat Számítógépesített rendszerek attribútumai (Forrás : Turban [7.3])

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

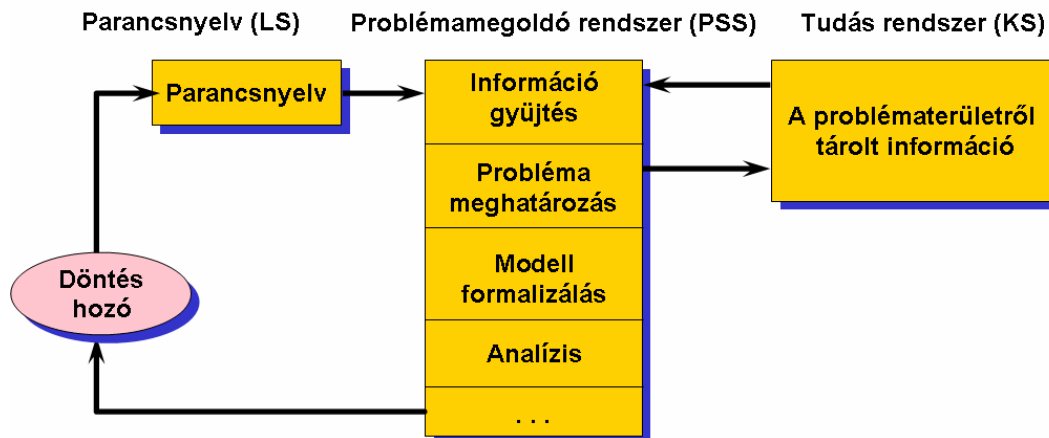
A döntési problémák

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 tartalmaznak 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 rendszerezük*: piaci, kereskedelmi, termelési és/vagy szolgáltatási, gazdálkodással, termelési tényezőkkel, és pénzügyekkel kapcsolatos döntési problémák. Hangsúlyozzuk itt a probléma *döntésorientált* megközelítését, *a folyamatszempelésre 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.



7.3. ábra: A vezetői információellátás



7.4. ábra: Döntéstámogató rendszerek (Decision Support Systems - DSS) tipikus szerkezete

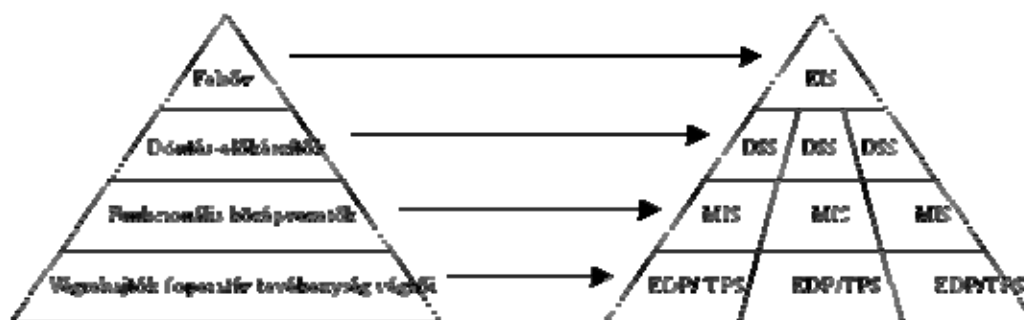
7.4. Vállalati információs rendszerek

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 középvezetői 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 igénytel nyújtottak főként a móltra vonatkozó, aggregált információkat a vállalatvezetésnek. A 80-as évek vállalati információrendszereit jól szemlélteti a 7.5. ábra.



7.5. 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 vállalati 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. 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 terabájnyi 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 alakíthatók 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 még 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 vállalaton 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:

- szükség volt egy összvállalati integrált IT-környezet kialakítására, mely biztosítja az alkalmazások közötti kommunikációt, adatcserét, és amely nyitott a külső adatok, különböző formátumok bevonására is;
- hatékonyabbá kellett tenni a vezetők információellátását és döntéshozói igényeiknek megfelelően kialakított vezetői információs rendszerek kifejlesztése lett a cél;
- a funkcionális szemléleten túl - az előzőekkel szoros összefüggésben - a vállalat valódi működésének megjelenítése érdekében a szervezeti határokon átnyúló folyamatokkal, az értékteremtés dimenzióinak tudatos kezelésével kezdtek el foglalkozni.

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.

ERP-rendszerek

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 összvállalati 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 operacionalizálására. 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

konceptiók, 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 alakíthatók a vállalat sajátosságainak megfelelően.

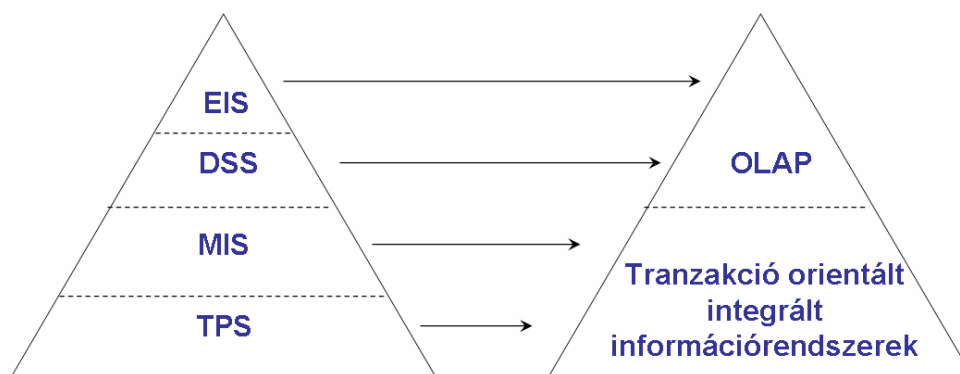
Az átszervezések végrehajtása, illetve az eredmények értékelése szintén a kontrolling feladata volt. A beruházások és a projektek kontrollingja is rendelkezik informatikai támogatással: a személyi teljesítménynövelő szoftverek néhány formája (például: MS Projekt), illetve az ERP-rendszerek önálló moduljai említhetők itt (például SAP IM, illetve PS modulok).

SCM és CRM

A folyamatszemplélet 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 gondolaton alapul, hogy a vállalati értékteremtés folyamata nem ér véget a vállalat határainál, valamint, hogy a vállalat csak akkor lehet sikeres, ha megfelelő módon tudja kielégíteni vevőköreinek 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** (On Line Analytical Processing – a tranzakciók közvetlen hatásának elemzése) szoftverek (7.6. ábra). Ezekkel a technológiákkal újraértelmezték a vezetői döntések támogatását; az egyes szakterületek információigényét kielégítve új informatikai területet nyitottak: az üzleti intelligencia világát.



7.6. ábra: Tipikus vállalati információrendszer napjainkban

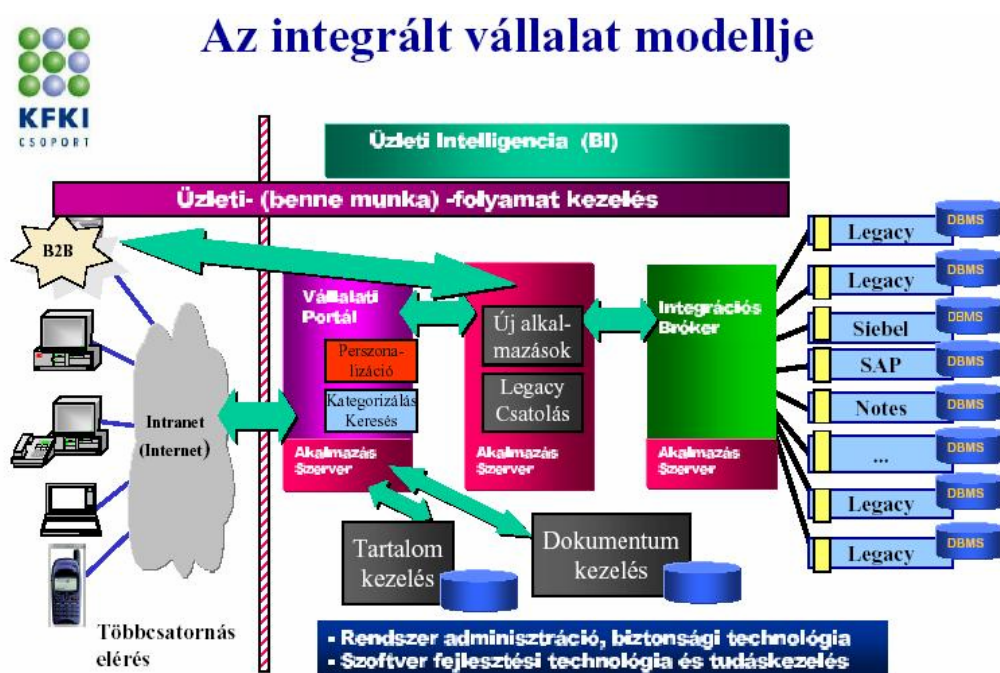
A folyamat-újrászervezések keretében többnyire értékelték a cégek saját informatikai egységüket, rendszerüket is -- az előző évek, évtizedek tapasztalatainak és a megszerezhető benchmarkoknak a tükrében. Az információtechnológia széles körű alkalmazása ugyanis nem mindig eredményezett hatékonyságjavulást, a nagy IT befektetések gyakran nem hozták meg a várt üzleti sikereket. A 90-es években ezért újfogalmazták 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-előkészítési munkájuk során. Az operatív controlling mellett ekkortájt kialakuló stratégiai controlling is épít az IT-alkalmazásokra, a modern ERP-rendszerek bázisán kialakított Üzleti intelligencia megoldások támogatására.

7.5. Integrált vállalati rendszer felépítése

Az integrált vállalatirányítási rendszer egy vállalat **valamennyi** feldolgozását (komplett üzleti folyamatok) megvalósító, **egységes** információs rendszer, amelyben minden adat csak **egyszer** szerepel (mindenki ugyanabból az adatbázisból dolgozik), és az adat a keletkezése helyén kerül rögzítésre.



7.7. ábra: Az integrált vállalati modell [7.8]

A 7.7. ábra egy tipikus vállalati integrált információrendszert mutat be, ahol a vállalat az új, korszerű eszközei mellett támaszkodik a korábbi évek rendszereire (Legacy), s azok eredményeire. Látható, hogy az üzleti folyamatok kezelésére használt kliens-szerver architektúra mögött a dokumentációkezelés és a tartalom (tudás) kezelése is segíti a vállalati szervezeti hierarchia hatékony működését.

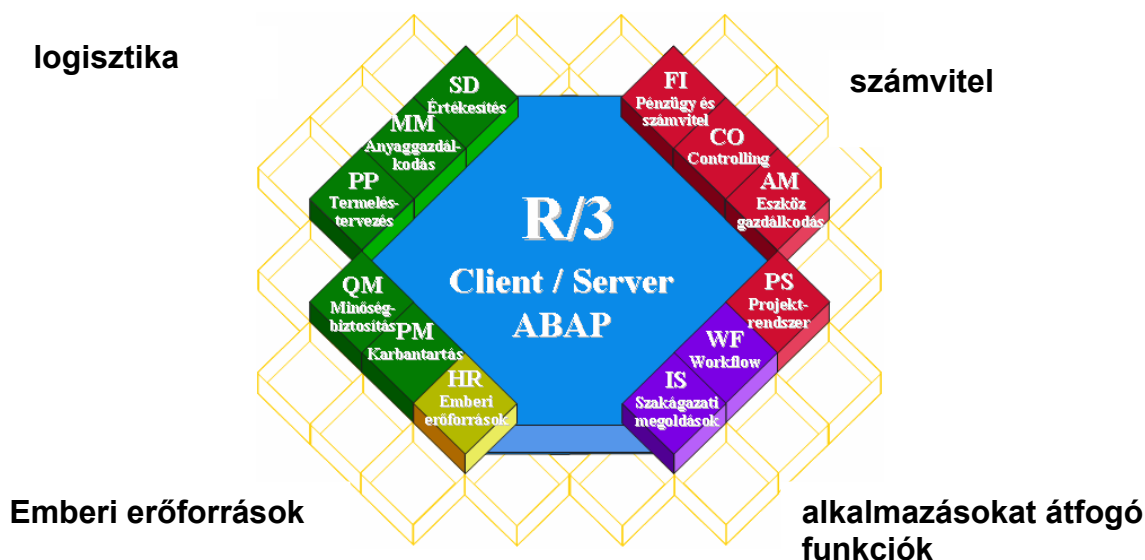
A vállalat korábbi rendszereinek adattartalma a később bevezetett rendszerek mellett fontos részét képezi a vállalati információ- és tudásmenedzsmentnek. Kizárólag az u.n. „zöldmezős” beruházások vannak abban a szerencsés helyzetben, hogy információs rendszerüket a semmiből a legkorszerűbb elvek alapján alakíthatják ki.

Az integrált rendszerek előnyei:

- kevesebb idő, erőfeszítés az információkhoz való hozzájutás,
- gyorsabb érthetőbb prezentációs technikák,
- a hibák aránya jelentősen csökken többek közt az automatizmusoknak és az egyszerű adatbevitelnek köszönhetően,
- jobban nyomon követhető a vállalat tevékenysége,
- alacsonyabb munkaerőigény,
- optimalizálható a beszerzés, termelés, értékesítés,
- javul a szervezeten belüli kommunikáció,
- csökken a földrajzi távolságok jelentősége.

Mindez stratégiai előnyt jelenthet. Magyarországon az integrált vállalatirányítási rendszerek piacának mérete meghaladja a 12 milliárd Ft-ot, s a piac még mindig nem telített. Három piaci szegmens különböztethető meg:

- nagyvállalatok (bevezetés költségigénye: 600 - 2000 millió Ft, időigénye: 1-2 év)
- közepes vállalatok (bevezetés költségigénye: 10 - 300 millió, időigénye: 3-9 hónap)
- kis- és mikrovállalatok (bevezetés költségigénye: 10 - 1000 ezer Ft, időigénye: elhanyagolható)



7.8. ábra: Egy integrált vállalatirányítási rendszer: SAP R/3 [7.10]

Az integrált vállalatirányítási rendszerek hazai piacát az SAP vezeti, piaci részesedése az IDC felmérése alapján meghaladja a 60 százalékot, 2000. végéig több, mint 160 magyarországi vállalatnál vezették be az SAP szoftverrendszerét [7.10].

7.6. 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:

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.

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.

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.

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.

Irodalomjegyzék

- 7.1. Burt P., Kinnucan M.: Models and Modeling Techniques for Information-Systems Review of Information Science and Technology 25: 175-208, 1990
- 7.2. Davis, G., Olson, M.: Management Information Systems: Conceptual Foundations, Structure and Development. McGraw-Hill, New York, New York, 1985. Turban, McLean & Wetherbe, Information Technology for Management John Wiley & Sons, 1999. Simon, H.: The New Science of Management Decision, Harper & Row, 1960. Gorry, G.A., Scott-Morton, M.S.: A Framework for Management Information Systems, Sloan Management Review, 1989.
- 7.6. Kroeber, D., Watson, H.J.: Computer-based Information Systems, Prentice Hall International Paperback Editions (Mar 1991)
- 7.7. Gábor A.: Információmenedzsment, AULA 1997.
- 7.8. IQSyS Szakmai Nap az adatbányászat alkalmazásáról, 2005.
- 7.9. Fajsz B., Cser L.: Üzleti tudás az adatok mélyén, IQSYS-BME, 2005.
- 7.10. www.sap.com

Ellenőrző kérdések

Ismertesse az információs rendszer fogalmát!

Mire szolgál a vezetői információs rendszer?

Az információrendszer típusok közötti kapcsolatok leírása

A stratégiai információrendszerek meghatározása

Miként támogathatják az információrendszerek az üzleti stratégiát?

Mi a döntés előkészítő rendszerek feladata?

Mi az az integrált vállalatirányítási rendszer?

Hogyan változott az integrált információs rendszerek fejlesztési filozófiája?

Mik az ágazati információs rendszerek?

8. AGRÁRINFORMÁCIÓ-RENDSZEREK

8.1. Az Európai Unió agrárinformáció rendszerei

Az Európai Unió információs rendszerének jogi szabályozását a Bizottság és a Tanács rendeletekkel alapozza meg. A rendeletek a tagországok számára kötelezőek, ezektől a nemzeti jogszabályok nem térhetnek el. A rendeletek mellett az információs rendszerek működését irányelvekkel is befolyásolja az EU. Ezek ugyancsak kötelezőek, de abban az értelemben, hogy az irányelvekben megfogalmazott szabályozást a tagállamoknak saját jogrendjükben kell megvalósítaniuk. Az Európai Tanács és az Európai Bizottság határozatokkal is és ajánlásokkal is szabályozza az információs rendszereket. A határozatok szintén kötelező érvényűek, de csak azokra a tagországokra vonatkozóan, akikre azt címezték. Az ajánlások nem kötelező erejűek, jelentőségük sokkal inkább morális.

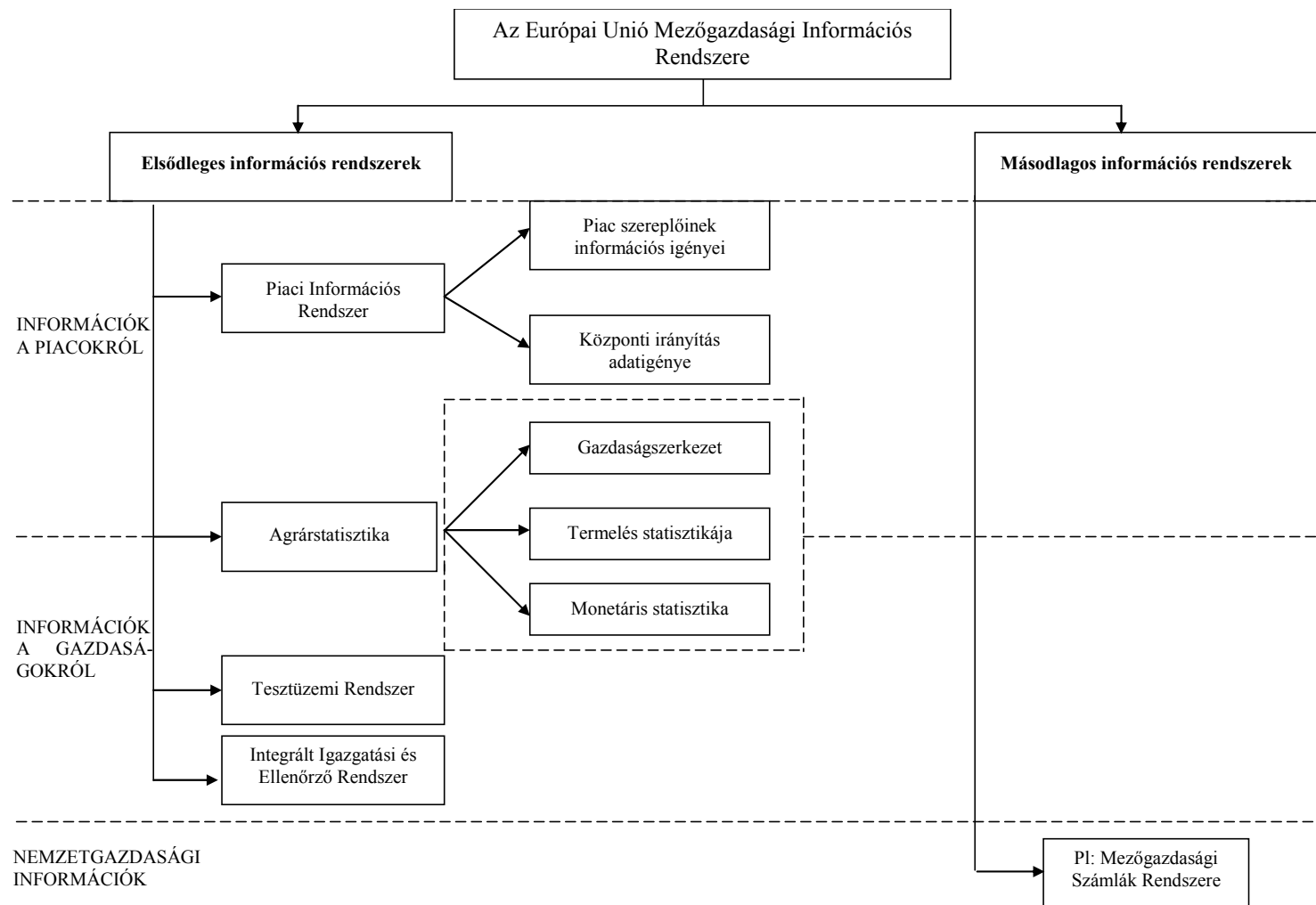
Az EU agrárgazdaságának irányítása hatalmas mennyiségű pontosan egyeztetett előírásoknak megfelelő információ szabályozott áramlásán alapszik. Az információs csatornák kölcsönösen összekötik a tagországokat a szervezet döntéshozó központjaival. Az ezen információk alapján hozott döntések komoly előnyöket, 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 szándékozók számára. Így Magyarország agrárinformatikai rendszerének is meg kell felelnie az EU elvárásainak. Az agrár nyilvántartási rendszer harmonizálásánál azonban tekintettel kell lenni a magyar agrárstruktúra hosszú távon is fennmaradó sajátosságaira, nevezetesen a nagy-, közép és kisüzemek létre és az ebből fakadó igényekre (többek között a nagyobb pontosság és részletesség iránti követelményekre, amelyek a jogi személyiségű vállalatoknak kötelező is betartaniuk).

Az EU informatikai struktúrája viszonylag stabilabb elem az uniós mechanizmusban, mint a Közös Agrárpolitika /KAP/. A KAP ugyanis 5-7 évenként jelentős átalakuláson megy keresztül attól függően, hogy milyen tendenciák érvényesülnek a tagországok mezőgazdaságában és ennek milyen szabályozási konzekvenciáit kell érvényre juttatni. Mindezek a változások mellett a KAP működését segítő információs rendszer egyes elemei viszonylag stabilak maradtak. Ez természetesen nem egy „megmerevedett” állapotot jelent, hisz beléphetnek új elemek /mint pl. a kompenzációs támogatásokkal kapcsolatos Integrált Rendszer/, de a fejlődés sokkal kiegyensúlyozottabb és kiszámíthatóbb, mint az agrárpolitikában.

Az EU szerteágazó informatikai struktúrája két jól elkülöníthető csoportba sorolható. Ezek:

- a primer, vagy elsődleges információs rendszerek,
- valamint a másodlagos, vagy szekunder információs rendszerek.

Az EU agrárinformációs rendszerét a 8.1. ábra szemlélteti.



Forrás: Kapronczai, 2003

8.1. ábra: Az Európai Unió Mezőgazdasági Információs Rendszere

Az elsődleges, vagy primer információs rendszerek az EU nagy adatgyűjtő és feldolgozó struktúrái. Lényegében ezekre épül az Unió agrárinformációs rendszere. Négy meghatározó eleme a következő:

- Az agrárstatisztika, amely az EUROSTAT által koordinálva számos területen nyújt statisztikai jellegű információkat az Unión belüli folyamatokról, a főbb tendenciákról.
- 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.
- 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.
- Végül a primer információs csoport negyedik eleme a támogatások elnyerését szolgáló információs elemek összessége. Ezek közül elsősorban 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 rövid távú előrejelzése, illetve agrárpolitikai intézkedések várható hatásának prognosztizálása.

Ehhez az agrárinformációs struktúrához kell alkalmazkodni Magyarországnak is, mégpedig úgy hogy megfeleljen az EU követelményeinek, ugyanakkor az Európai Unió nyilvántartási rendszerével összhangban álló saját rendszert és információs hálózatot alakítson ki.

8.2. Agrárstatisztikai rendszer

Az Európai Unió az agrárinformációs rendszerrel szemben megkülönböztetett igényeket és követelményeket támaszt, mivel a közösségi költségvetésből mezőgazdasági támogatások a legnagyobb arányban részesülnek. 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. A legfontosabb dokumentumok az Acquis communautaire és a Compendium, ezen túlmenően a módszertani kézikönyvek - melyek száma meghaladja a tízet - nyújtanak útmutatást. Az EU jogszabályok, megállapodások és ajánlások a közösségi Agrárpolitika (CAP) igényeinek megfelelően folyamatosan változnak.

A harmonizációs feladatok teljesítése érdekében a KSH és az EUROSTAT 1994-ben az un. „Közös Nyilatkozatban” határozta meg az együttműködés kereteit. Ezt követően

került sor a 2000. évig terjedő stratégia kidolgozására. A csatlakozási tárgyalások során a Statisztikai Fejezet átvilágítása 1998. júliusában multilaterális és bilaterális keretek között Brüsszelben zajlott. A magyar küldöttség kijelentette, hogy Magyarország a statisztika területén átveszi a közösségi vívmányokat és nem tart igényt átmeneti időszakokra.

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 Európai Bizottságon belüli működő 23 főigazgatóság (Directorate General) közül a VI. számú főigazgatóság felelős a közös agrárpolitika megvalósításáért, "vezényléséért" és továbbfejlesztéséért. Az ehhez szükséges információk két csatornán keresztül áramlanak hozzá:

- az Európai Unió statisztikai hivatalán, az EUROSTAT-on keresztül, és
- közvetlenül a tagországoktól.

A tagországoktól közvetlenül érkező adatok főleg operatív irányítási célokat szolgálnak, elsődlegesen a piac helyzetéről adnak pontos, naprakész tájékoztatást.

Az agrárstatisztikai információk – amelyeket alapvetően agrárpolitikai, közgazdasági elemzési és általános információs célokra hasznosítanak – az EUROSTAT útján jutnak el a 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. Az EUROSTAT természetesen nem pusztán egy adatátvevő és továbbító szervezet, hanem az EU hivatalos statisztikai szolgálataként feladata a közösségi információs rendszerek kialakítása, továbbfejlesztése, az adatgyűjtések módszertani megalapozása, a tagországok statisztikai munkájának segítése, koordinálása. Az EUROSTAT-on belül a különböző szakterületeket Igazgatóságok irányítják, a mezőgazdaság, az erdészet, a halászat és a környezet statisztikáiért külön igazgatóság (Directorate F) a felelős.

A Statisztika alrendszereinek meghatározásakor azzal kell számolni, hogy a közösségi agrárstatisztika lényegében három pillérre épül. A magyar agrárstatisztika talán legnagyobb elmaradása az EU követelményeitől a mezőgazdasági üzemek (mindenek előtt a különböző méretű és profilú magángazdaságok) működésének és teljesítményeinek rendszeres számbavételében és ezzel szoros összefüggésben a monetáris statisztika területén van. Különösképpen hiányosak az ökonómiai elemzést szolgáló információs alrendszerek. A monetáris statisztika alrendszerei közül az árstatisztika, illetve a munkaerő- és keresetek statisztikája területén kisebb-nagyobb kiigazításokra van "csupán" szükség ahhoz, hogy az EU követelményeinek meg tudjon felelni az ország.

1997. óta üzemszerűen működik a műholdas távérzékelésen alapuló szántóföldi növény monitoring. A térinformatika és a távérzékelés ezzel az agrárirányítás döntéselőkészítő 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 előállíthatóak így pl. időszakosan vízzel elborított terület, növénybetegség, stb. A távérzékelés technikai hasznosítása a mezőgazdasági statisztika rendszeréhez fontos ellenőrző szerepet tölthet be a teljes lefedettség területén (vetésterület növényenként, a termés mennyiség egyeztetésében), továbbá az előrejelzések, prognózisok elkészítésében.

8.3. Tesztüzemi rendszer - A FADN (Farm Accountancy Data Network)

A Tesztüzemi Rendszer lényegét tekintve egy reprezentatív adatgyűjtésre épülő üzemgazdasági információs rendszer. A meghatározott szempontok szerint kiválasztott üzemek önkéntesen csatlakoznak a hálózathoz, egységes könyvelést vezetnek, s adataikat agrárpolitikai célokra rendelkezésre bocsátják. A legfontosabb eredményeket és az azokból levezethető végkövetkeztetéseket évente beszámoló jelentésben összegzik.

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 -: 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.

Az FADN megfigyelési körén belül nagy eltérések vannak az üzemek között. Ahhoz, hogy a kiválasztott minta ezen változatosságot visszatükrözze, a mintavétel előtt el kell végezni a sokaság rétegezését. A rétegezés olyan statisztikai eljárás, amelyet a mintavétel hatékonyságának növelésére használnak, abban az értelemben, hogy a sokaság megfelelő képviseléséhez szükséges üzemek száma minél kisebb legyen. A rétegezés során az üzemeket megközelítőleg homogén csoportokba sorolják, s minden csoportból

választanak üzemeket a mintába. A Bizottság széles körben használja ezt a technikát, s három kritériumot alkalmaz a rétegezésnél:

- a földrajzi fekvést /társág/,
- a gazdasági méretet,
- és a gazdálkodás típusát.

Magyarországon az 1996. évben kezdődött meg a Tesztüzemi Rendszer szervezése és működtetése német szakemberek részvételével. A Földművelésügyi és Vidékfejlesztési Minisztérium Magyarországon az Agrárgazdasági Kutató és Informatikai Intézetet jelölte ki a tesztüzemi információs rendszer bázisintézményeként. Az intézet feladata a hálózat működésének szervezése, szabályozása, az érintett intézmények – Európai Bizottság, FVM, Központi Statisztikai Hivatal, érdekképviselői szervezetek és kamarák – közötti együttműködés fenntartása, az adatok ellenőrzése, feldolgozása, elemzése, továbbá az eredmények közzététele. Más szervezetekkel együttműködve az intézet végzi a rendszer teljes mértékű EU harmonizációjából adódó módszertani fejlesztéseket is.

A tesztüzemek – vagyis információs szolgáltatásra önkéntes alapon kiválasztott üzemek – adatainak gyűjtése és elemzése az alábbi területekre terjed ki:

- az üzemek méretében és a tulajdoni viszonyokban bekövetkező változások (életképes gazdaságméret, a földtulajdon és a földhasználat kérdései);
- a termelés anyagi műszaki feltételeinek változása (a beruházások alakulása, a termelőkapacitások cseréje, korszerűsítése);
- a termelés és a termelési szerkezet alakulása (a termelés racionális mértékű növelése: a természeti adottságok jobb kihasználását, az üzemi jövedelmek növelését eredményező termelésszerkezeti változások);
- a foglalkoztatottság, a munkatermelékenység, a mezőgazdasági jövedelmek alakulása;
- a termékek a termelési folyamatok hazai és nemzetközi versenyképességének alakulása, a mezőgazdasági termékek előállításához felhasznált természetes ráfordítások, valamint a termelés költségének és jövedelmezőségének bemutatásával;
- az agrártámogatások hatása a versenyképességre és a jövedelmezőségre.

A feldolgozott adatokat az AKII egy kiadvány formájában évente megjelenteti, amely nagy mértékben igazodik az EU Bizottság „Eredmény-beszámolójához”, lényegében minden olyan információt tartalmaz, amellyel a vizsgált üzemek gazdálkodása jellemezhető.

A vizsgált üzemkör az 5 hektárnál nagyobb területen gazdálkodó, vagy 5 számosállatnál többet tartó egyéni gazdaságokból és gazdasági szervezetekből áll. Az üzemek a mezőgazdasági tájörzetek szerinti elhelyezkedésük, méretük és termelési profiljuk figyelembevételével kerültek kiválasztásra.

A KSH korábbi adatai szerint a magyar mezőgazdaságban mintegy 70 ezer olyan üzem található, amelynek mezőgazdasági területe 5 hektárnál nagyobb, vagy 5 számosállat-egységnél több állatot tart. Ezek többsége részmunkaidős családi gazdaság, de mivel ezek súlya összességében jelentős, a mintából való kihagyásuk nem engedhető meg. A 70 ezer gazdaságban a termőterület hozzávetőlegesen 72 százalékát használják és az állatállomány 68 százalékát tartják. Ha Magyarországon – szemben az EU rendszerén belüli 1,5 százalékos átlagos kiválasztási aránnyal – egyelőre 3 százalékos mintavétellel számolunk, akkor a tesztüzemi hálózat teljes kiépülését feltételezve

- 350-400 jogi személyiségű gazdasági társaság és szövetkezet, valamint
- 1600-1800 egyéni gazdaság, vagyis összesen mintegy
- 2000-2100 üzem

kiválasztására, illetve az adatszolgáltatásban történő önkéntes közreműködésre van szükség.

Magyarországon a területi munkát könyvelőirodák végzik a megyékben. Ez a struktúra a német modellnek felel meg, ahol ugyancsak a minisztérium és könyvelőirodák működnek együtt. Ezt a szisztémát követi Spanyolország és Luxemburg is. Ugyanakkor az EU többi tagországában meglehetősen változatos, sokszínű a különböző szervezetek közötti együttműködés a tesztüzemi adatgyűjtésben. A munkában résztvesznek minisztériumi területi szervezetek, gazdaszövetségek, agrárkamarák, agroökonómiai irányítási központok, egyetemek, szaktanácsadó szervezetek is. Végeredményben a 15 tagországban 9 féle koordinációs forma létezik.

8.4. A Piaci Információs Rendszer

A PÁIR egy olyan államilag működtetett szolgáltatás, amely széles körben forgalmazott mezőgazdasági termények árairól és mennyiségeiről rendszeresen, megfelelő módon adatokat gyűjt a vidéki, nagybani és fogyasztói piacokon és ezt az információt időben és rendszeres időközönként a különböző médiumokon keresztül eljuttatja a gazdálkodókhoz, a kereskedőkhöz, a kormányzati tisztviselőkhöz, illetve a fogyasztókhoz.

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 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 két rendszer csoport között van bizonyos átjárhatóság és ezért több EU tagállamban a két rendszer szoros kapcsolatban működik egymással. Erre példa Franciaország, ahol a SNM /Services des Nouvelles des Marchés - Piaci Információs Szolgálat/ közvetlenül a

Mezőgazdasági Minisztériumhoz tartozik és információival mind az országos és az EU hatóságok, mind pedig a gazdasági résztvevők igényeinek megfelel. Ettől gyökeresen eltérő a szervezeti struktúra a Német Szövetségi Köztársaságban, ahol a piaci szereplők információs igényeinek kielégítésére külön szervezet jött létre Bonnban, a ZMP /Zentrale Markt-und Preisberichtsstelle für Erzeugnisse der Land-, Forst- und Ernährungswirtschaft GmbH - Mező-, Erdő- és Élelmiszergazdaság Termékeinek Központi Piac- és Árfigyelő Szolgálat/. Megint más a helyzet például Hollandiában, ahol szakmai szervezetek /Produktschappen/, szövetségek és értékesítési társulások foglalkoznak a piaci- és árinformációs rendszerek működtetésével.

A brüsszeli adminisztráció a tagországoktól pontosan meghatározott mechanizmus szerint igényel piaci /statisztikai/ információkat. Az adatközlési kötelezettséget termékpályánként eltérően írja elő a 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. Az információs rendszer főbb célcsoportjai, információi és jellemzői a következők:

Az elmúlt években végrehajtott Phare koordinációban megvalósított fejlesztés eredményeként kiépült egy EU konform Piaci Információs Rendszer a búza, kukorica, vágósertés, vágómarha és tej ágazatokra. Az adatgyűjtés kéthetente /a tej esetében havonta/ történik. Az információgyűjtés és feldolgozás, illetve a közzététel közti idő maximum 3 nap. A termékpályák piaci helyzetéről az adatgyűjtéssel megegyező gyakorisággal jelentés készül, amely a médiumokon keresztül, valamint bulletin formájában is azonnal közzétételre kerül. Az adatszolgáltatók – a nemzetközi gyakorlatnak megfelelően – a feldolgozó-, felvásárló szervezetek. A rendszer adatköre kiterjed a felvásárlási /input/ árakra és mennyiségekre minőség szerinti részletezettséggel, valamint néhány feldolgozott /output/ termék termelői áraira és a fogyasztói árakra. Verbális és adatszerű információk is begyűjtésre kerülnek a főbb piaci folyamatokról és a várható tendenciákról. Mindemellett a kiadott bulletinnek tartalmazzanak hazai és nemzetközi tőzsdei árakat, valamint a főbb nyugat-európai országok piaci tendenciáinak egy részéről is tájékoztatnak.

A rendszer egy a kilencvenes évek első felében kialakított információs eleme a zöldség-gyümölcs, a vágott virág fogyasztói és nagybani piacokon folyó kereskedelméről tájékoztat. A Magyarországon működő fogyasztói piacok közül 50 jelentősebb tartozik megfigyelési körébe, a nyolc nagybani piac közül pedig öt. Az ország összes zöldség-gyümölcs, illetve vágott virág termelésének mintegy 40 százalékát kínálják a rendszerhez tartozó piacok. Az árak naponta, illetve hetente két alkalommal kerülnek bevitelre az AKII központi számítógépébe modemén keresztül. Az adatok feldolgozása folyamatos, és a kész táblázatok már a bevitel napjának délelőttjén megjelennek a magyar televízió képűségében. A heti összefoglaló adatokat nyomtatott médiumokon keresztül teszik közzé. Adott a lehetőség a modemén keresztül történő közvetlen csatlakozásra is. Ez az alrendszer heti gyakorisággal közöl adatokat a bécsi, a milánói, illetve a főbb németországi piacokról, valamint a környező országok néhány nagyobb határ menti városából is gyűjt piaci árakat.

A Piaci Információs Rendszer részeként működik a sertésállományra vonatkozó előrejelző rendszer. A prognózisok készítéséhez számítógépes program készült, ami a KSH négyhavonta megjelenő tényadatai mellett figyelembe veszi a tenyésztés technikai-technológiai paramétereit, valamint a piaci folyamatok alakulását is.

Az élelmiszergazdaság piaci tevékenységét bemutató információk zöme az AKII mellett a KSH-ból, a Kopint-Datorg-tól, az Agrárintervenciós Központtól /AIK/ és az Agrárrendtartási Hivaltaltól /ARH/ származik. A Földművelésügyi és Vidékfejlesztési Minisztérium /FVM/ keretei közt működő ARH saját adatgyűjtést nem végez. A törvényi előírásból adódóan fokozottan támaszkodik a terméktanácsoktól gyűjtött, piaccal kapcsolatos információkra

8.5. Integrált Igazgatási és Ellenőrző Rendszer

Az Integrált Igazgatási és Ellenőrző Rendszer /Integrated Administrative and Controll 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 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.

Az Integrált Rendszer keretében begyűjtött adatokat az EU adminisztráció szigorúan ellenőrzi, ami súlyos terheket ró az igazgatási szervekre is. Az EU munkatársainak a helyszínen kell ellenőrizni a területtámogatási kérelmek 5 százalékát, illetve az állattartásra vonatkozó kérelmek 10 százalékát. Az arányt növelni kell az egyes régiókban, ha a korábbi ellenőrzések során ott jelentősebb szabálytalanságokat állapítottak meg. „Tévedések” esetén pedig szigorúak a szankciók. Büntetést kap a termelő, és az az ország is amelynek termelői valótlan adatokat közöltek. Az a gazdálkodó, aki hibázik az adatgyűjtő tömb kitöltésénél és ennek következtében többlet támogatást vesz fel, ha a

tévedés 3 és 20 százalék közötti, a támogatási többlet kétszeresét kell büntetésként megfizetnie. Ha a „tévedés” 20 százalék feletti, nem fizetnek támogatást és a gazdát a következő naptári évre kizárják a kompenzációs támogatási rendszerből. A tagországokat pedig úgy szankcionálja a brüsszeli adminisztráció, hogy olyan mértékben csökkenti az országnak adott kompenzációs támogatási keretet, amilyen arányban hibát találtak a tételes ellenőrzéseknél a minta esetében.

8.6. Speciális szakterületi információs rendszerek

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-, szakhatósági-, a gazdálkodás eredményességére vonatkozó összehasonlító információk, valamint a gazdálkodók szakmai felkészültségének fejlesztésére vonatkozó információk értendők.

Az EU tagságtól függetlenül egyre sürgetőbbé vált az agrárkormányzat döntéseit támogató új információs rendszer kialakításának szükségessége.

Az elmúlt időszakban felmerült új követelményeknek az információs szolgáltatás jelenleg nem tud minden elemében megfelelni. Kormányzati célú információs rendszereinket szakmai, tartalmi, adatgyűjtési kör és a fegyelem, valamint a nyújtott szolgáltatások és az azokhoz való hozzáférhetőség tekintetében is alakítani szükséges. A magyar agrárgazdaságnak az európai követelményekkel összhangban álló nyilvántartási rendszerekre és információs hálózat kialakítására illetve bővítésére van szüksége. Elkészültek a már Uniós igényeket is kielégítő információs rendszerek minnt pl. a MePAR vagy a TakarNet. A Mezőgazdasági Parcella Azonosító Rendszer (MePAR) az agrártámogatások eljárásainak kizárólagos országos földterület-azonosító rendszere mivel a földterülethez kapcsolódó részben vagy egészben európai uniós támogatások igénylése során csak ennek az azonosítási rendszernek az adatait lehet használni. A TakarNet (TAKARos NETwork) egy olyan földhivatalokat összekötő hálózati rendszer, amely az ingatlan-nyilvántartási adatok távoli, elektronikus elérését biztosítja. A földhivatali adatok (tulajdoni lap adatai) számítógépes kezelését a TAKAROS (Térképen Alapuló KATaszeri Rendszer Országos Számítógépesítése) rendszerrel már korábban megoldották.

Irodalomjegyzék

- 8.1. Herdon M. et. al. (2000): Szemelvények az EU agrár szak- és közigazgatási képzéséhez.VIII. Agrár- és közigazgatási információs rendszerek az Európai Unióban. Debreceni Egyetem ATC AVI. ISBN 963 9274 16 X.
- 8.2. Kapronczai I. (2001): Agrárinformációs rendszerek fejlesztésének megalapozása. PhD tézis. <http://miau.gau.hu/miau/57/akiitezis.doc>. Ipargazdasági Kutató és Tanácsadó Kft. (2001) A mezőgazdaság infokommunikációs eszközökkel történő támogatása

Ellenőrző kérdések.

Hogyan kell értelmezni az Európai Bizottság és az Európai Tanács agrárinformációs rendszerekre hozott rendeleteit, irányelveit, határozatait és ajánlásait a tagországoknak? Az EU informatikai struktúráját, vagy a Közös Agrárpolitikát tartja e stabilabbnak? Hogyan csoportosítaná az EU informatikai rendszerét? Mik a Primer Információs Rendszer elemei?

Mi a feladata az Unió Agrárstatisztikai Rendszerének?

Mi a Tesztüzemi Rendszer feladata, honnan gyűjt adatokat és mire használja azokat?

Kinek szolgáltatnak információkat a Piaci Információs Rendszerek?

Mi a fő feladata az Integrált Igazgatási és Ellenőrző Rendszernek?

9. TÉRINFORMATIKA

9.1. 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 fejlesztve, 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 Institut (ESRI)* nevű szoftverház, mely 1982-ben első verzióban kibocsátott ARC/INFO nevű **GIS** szoftverével az első általános és 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, azaz grafikus (térbeli) és nem grafikus (leíró) adatokat együtt, integráltan tud kezelni.

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-hozatalban; 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:

- Mi van egy adott helyen?
- Hol található vagy nem található egy bizonyos jellemző?
- Mi és hol változott egy bizonyos idő óta?
- Melyik az optimális?
- Mi lenne, ha ...? Modellezés.

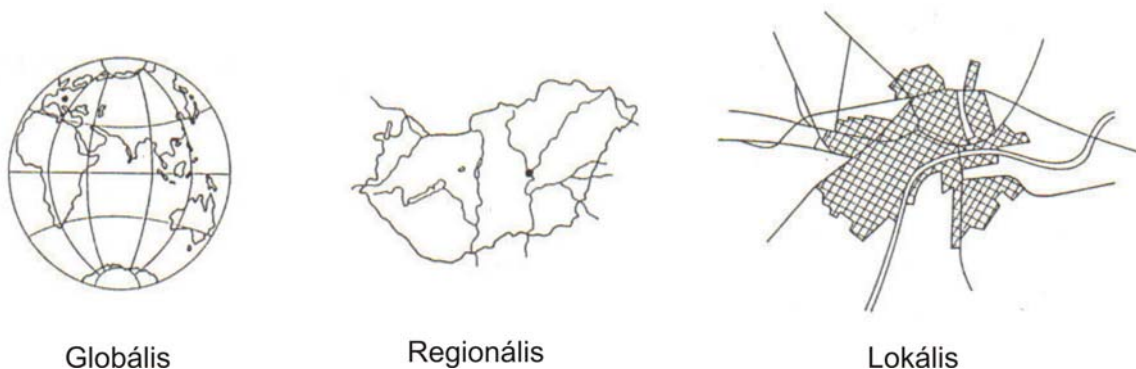
A térképre alapozott rendszerekre is a "térbeli" jelzőt alkalmaztuk jóllehet e rendszerek az euklidesi tér 3 dimenziójából valójában csak két dimenziót alkalmaztak, ily módon talán helyesebb lett volna, ha a nemzetközi gyakorlattal ellentétben térbeli sík rendszereknek nevezzük őket. Kivételt ezek közül 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 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.

9.2. Térinformatikai rendszerek típusai

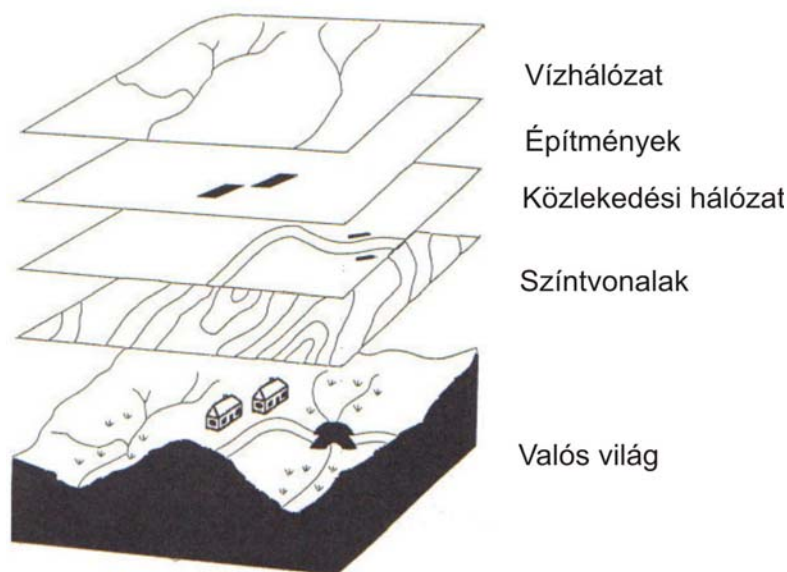
A geoinformációs rendszereket területi kiterjedésük szerint lokális, regionális és globális kategóriákba soroljuk.



9.1. ábra: Térinformatikai rendszerek területi kiterjedése [9.1]

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-, bánya-, erdő-, földmérő-, építő-, építész-, közlekedési mérnöki rendszerek).

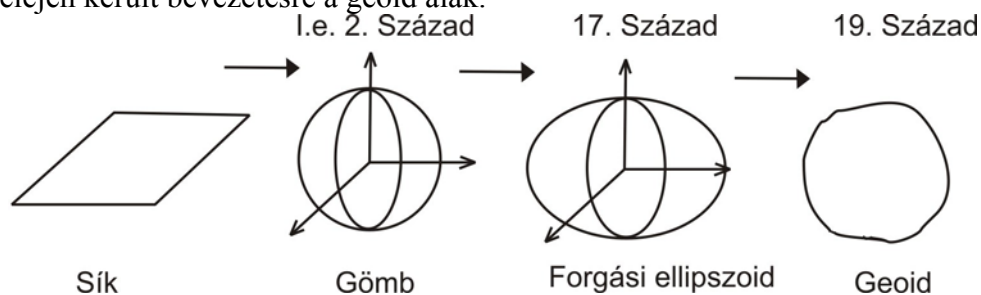


9.2. ábra A valós világ fedvényekkel történő ábrázolása.
Az ábrán a tematikus dimenzió száma négy [9.1.]

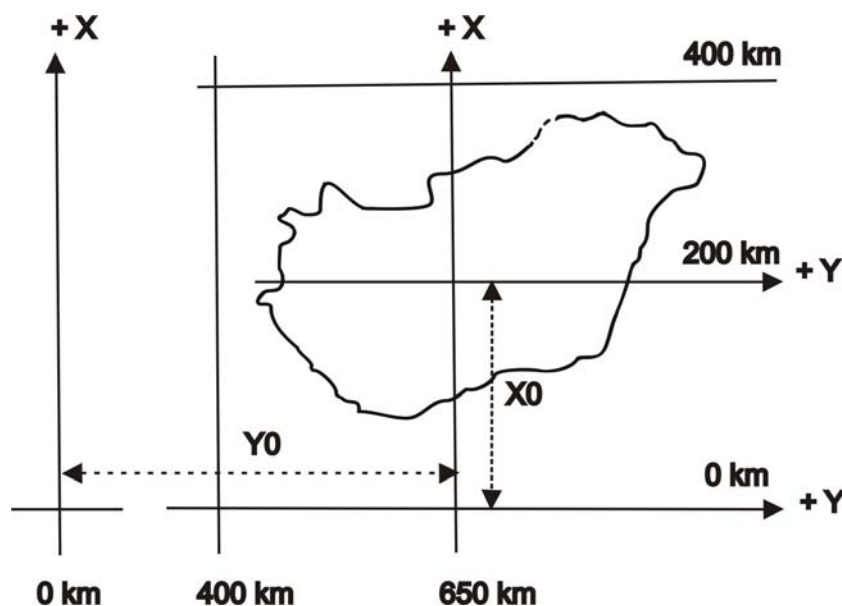
Az objektumok osztályba sorolásához a jobb áttekinthetőség kedvéért gyakran az egyes objektumosztályokat objektumoknak tekintik, ebből komplex objektumosztályokat hoznak létre. Például, ha a településekből hoznak létre objektum osztályt, akkor a komplex objektum osztályt a megyék, az ennél magasabb komplex objektumosztályt a régiók alkotják. Az objektumok osztályainak megválasztásakor gyakran alkalmazzák a fedvényekkel (layers) történő leírást. A fedvények alkalmazását szemlélteti a 9.2. ábra.

9.3. 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 földalakat. A 19. század elején került bevezetésre a geoid alak.



9.3. ábra: Az elméleti földalak fejlődése [9.1.]

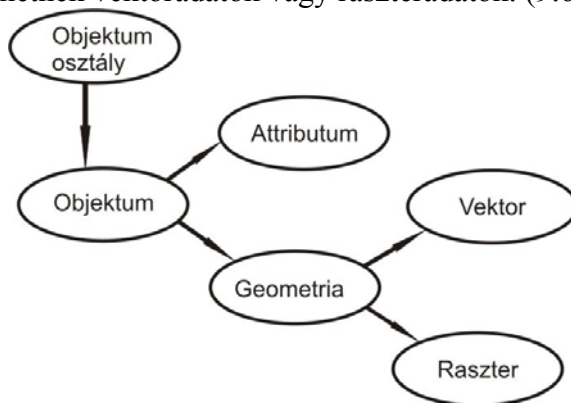


9.4. ábra Egységes Országos Vetületi Rendszer (EOV) koordináta-rendszere [9.1.]

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.

9.4. 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. (9.6.ábra).



9.5. Raszteres és vektoros adatok, adatkezelés, alapműveletek

A térbeli objektumok (pont, vonal, felület, test) leírására történhet raszteres és vektoros adatmodellel. A térbeli adatok hagyományos ábrázolási formája a vektor modell. A vektoros rendszereknél az objektum leírása jellegzetes pontjainak helyvektoraival történik. A jellegzetes pontok (pl. szakasz esetén a két végpont) koordinátáin kívül meg kell adni az összekötés szabályszerűségét is, ez általában egyenes, de lehet ív is.

A **raszter modell** esetén a fényképek és a számítógép monitorok ábrázolásmódját választották mintaként a grafikus adatok modellezéséhez. Raszteres rendszerek esetén a vizsgált térszt összefüggő idomokkal (2D modell esetén legtöbbször négyzettel, 3D modell esetén kockával) fedjük le. Négyzettel történő lefedés esetén a legkisebb geometriai objektum a pixel. A lefedés valamennyi pixelének állapotát (a hozzá tartozó információt) tárolni kell, ami egy raszter-mátrixban történik. A mátrixelem szokásos elnevezései képpont, pixel, cella.

Képi információ tárolása esetén a kép típusát alapvetően a bit-per-pixel érték határozza meg, vagyis az hogy egy képponthoz tartozó információt hány biten tárolunk.

Néhány jellemző típus:

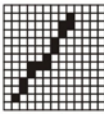
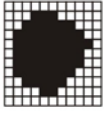
- 1 bites pixelek: bináris kép.
- 8 bites pixelek: monochrom kép, 256 szürkeárnyalat.
- 24 bites pixelek: színes kép, ahol a színek a három alapszín (piros, zöld, kék) keverékeként kódoltak, mindegyik színt komponensnél 256 árnyalattal (3*8 bit).
- Multispektrális műholdkép: például 7 sávban, sávonként 8-bites (vagy 16-bites) pixelek.

Előfordulhat, hogy a rasztermátrix nem képi információt hordoz (pl. talajtérkép, hozamtérkép, tápanyagtérkép), ilyenkor a pixelenkénti bitek száma is a fentitől tetszőlegesen eltérő lehet.

Felbontás (geometriai): megadja, hogy egy pixel mekkora területnek felel meg a valóságban (pl. 10 x 10 méter)

A térinformációs rendszerek szoftverei a térbeli adatokkal különböző műveleteket hajtanak végre. A műveletek egy jelentős részénél az alfanumerikus illetve grafikus adatbázisokkal kapcsolatos lekérdezések egyszerű vagy összetett alkalmazásairól van szó. Más feladatok 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		

9.6. ábra: Vektor- és raszteradatok grafikus jellemzése

Például a fedvénymetszési műveletek (overlay) a raszteres adatmodellel igen egyszerűen végezhetők, míg a vektormodellben igen sok számítást és rendezést igényelnek, ugyanakkor a távolság és kerületszámítások pontosan és egyszerűen csak a vektoros adatmodellben hajthatók végre.

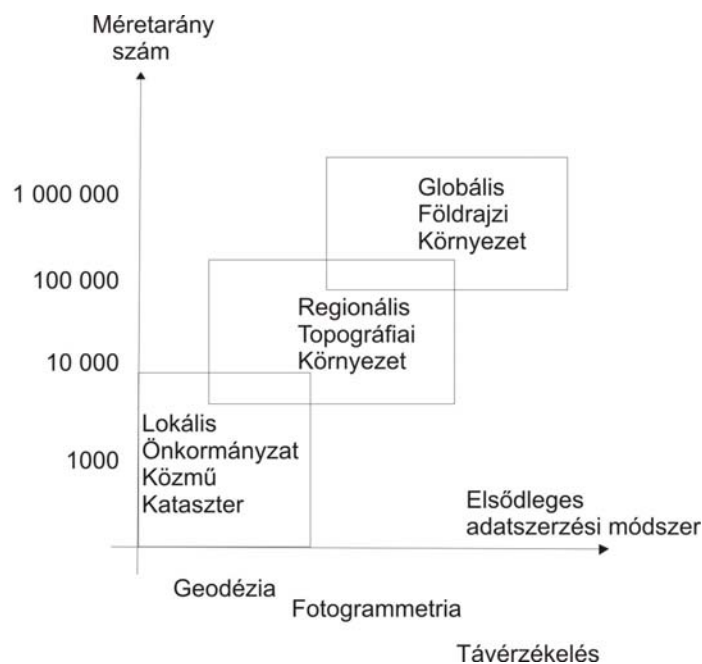
9.6. A térinformatika fontosabb eszközei, alkalmazása

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 (9.7. 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 térképezp rendszer	- Számítógép - Mágneslemez tárolók - Mágnesszalagos tárolók - Optikai tárolók	- Számítógép - Nagyfelbontású grafikus képernyő - Egér - Tablet	- Számítógép - Printer - Plotter

9.7. ábra: A térinformatikai rendszerek hardver eszközei

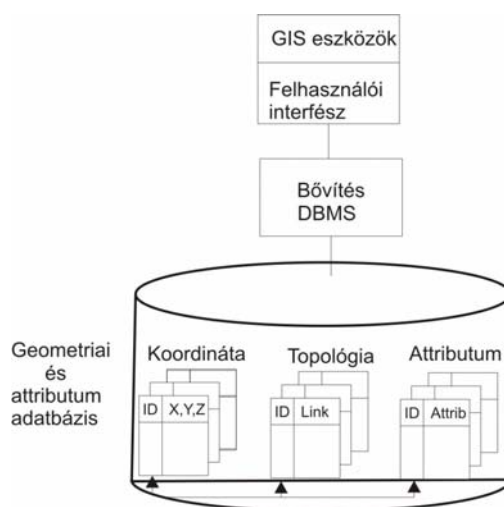
A térinformatikai rendszerek fontos feladata az adatgyűjtés. A geometriai és az attribútumadatok különböző módszerekkel nyerhetők. Az adatnyerés módja elsősorban a térinformációs rendszer alkalmazási területétől, felépítési elvétől, a rendelkezésre álló adatforrásoktól és az adatsűrűségtől függ. A térinformatikai rendszerek létrehozásakor a geometriai adatnyerési módszer függ a rendszer területi kiterjedésétől is. (9.8. ábra.)



9.8. ábra: Térinformációs rendszerek területi kiterjedésének és az adatnyerési módszerek kapcsolata [9.1]

9.7. 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 (9.9. ábra)



9.9. ábra: Geometriai és szakadatok kezelésének integrált rendszere

9.8. Precíziós gazdálkodás

A szántóföld, ahol a gazdálkodás folyik, egy rendkívül bonyolult változatos rendszer, eltérő talajtípusú és- féleségű talajfoltok találhatók akár egy táblán belül is, és ezeknek a foltoknak gyakran eltérő a tápanyag tartalma, más a gyomflórája és másként művelhetők, más a vízelvezető képességük stb. A talaj fizikai, kémiai és biológiai jellemzői együttesen alakítják ki a talaj legfontosabb tulajdonságát a termékenységet. A talajok termékenysége természetesen térben és időben is állandóan változik, így értelemszerűen egy mezőgazdasági táblán belül is elkülöníthetünk termékeny és kevésbé termékeny talajfoltokat. A termékenység leginkább a termés mennyiségével és minőségével jellemezhető.

A hagyományos növénytermesztési technológiák figyelmen kívül hagyják táblán belüli változásokat a talajművelés, a tápanyag-utánpótlás, a növényvédelem, az öntözést során, így a területek egy része kap csak optimális kezelést, az eltérő tulajdonságú hányad pedig többet vagy kevesebbet.

A technikai fejlődés, a globális helyzetmeghatározó navigációs rendszerek (GPS), a mezőgazdasági gépek nagyfokú automatizálásának lehetősége, illetve a térinformatikai szoftverek (GIS) megjelenése napjainkban már lehetővé teszik a talajok térbeli változatosságát is figyelembe vevő agrotechnikai beavatkozásokat, azaz a precíziós gazdálkodás megvalósítását.

A hely-meghatározó készülék, a GPS (Global Positioning System) vevőkészülék feladata a szántóföldön a táblán belüli hely meghatározása. A vevőkészülék a NAVSTAR (Satellite Timing and Ranging) műholdrendszer jeleit veszi, ennek eredményeként alkalmas a háromdimenziós hely-meghatározásra, valamint a sebesség és az idő mérésére. A hely-meghatározás pontossága függ a mérési módszertől. Ez lehet abszolút, vagy differenciál módszer.

A hely-meghatározás abszolút módszere azt jelenti, hogy a mozgó GPS vevőkészülék a NAVSTAR műholdak jelét veszi és hibakorrekciónak nem történik. Az a jel 10 m alatti hely-meghatározásra alkalmas.

Differenciális hely-meghatározás esetén, a mozgó vevőn kívül egy másik, egy ismert koordinátájú ponton elhelyezett vevő is dolgozik azonos időben. Az ismert koordinátájú ponton elhelyezett, úgynevezett bázis vevő pontossága a mozgó vevőhöz viszonyítva legalább egy nagyságrenddel jobb. A bázis vevő meghatározza a GPS hibájának nagyságát és irányát és az erre vonatkozó információt közli a mozgó vevővel. A mozgó vevő a hibajel nagyságának és irányának ismeretében korrigálja saját helymeghatározását. A hiba így 1 cm alatt is lehet.

9.8.1. A precíziós gazdálkodás eszközei

A központi számítógép és a rajta futó GIS

Az fedélzeti számítógép, - összekötve egy műholdas helyzet-meghatározó berendezéssel (GPS) -, ami alkalmas a megfelelő információk (földrajzi koordináták, hozameredmények, kiszórt műtrágya-mennyiség) mérésére, rögzítésére és munkagép-vezérlésre, így ezen adatok (hozamtérképek) és más információk (talajtérképek,

táblatorzskönyvi- és távérzékelési adatok, stb.) alapján hozott döntések megvalósítására (tápanyag-, növényvédő szer- és vetőmag kijuttatás, talajminta-vételezés, stb.).

Adatátvitel kártya a terepi és a központi számítógépek között.

9.8.2. Adatgyűjtés

A talaj jellemzésének kiinduló pontja a tábla digitalizált nagyléptékű genetikai talajtérképe, ha ez nem áll rendelkezésre, akkor ennek a légi-, vagy űrfelvétele. A talaj tápanyag tartalmának a megállapítása céljából talaj mintavétel szükséges. A mintavétel történhet a genetikai talajtérkép alapján, a tábla poligonokra való felosztása után, vagy a táblának valamilyen rácsozat szerint parcellákra történő felbontása után. Ilyenkor a poligonok, ill. a rácsokkal határolt parcellák talajából veszünk mintát. A mintavétel történhet előre kijelölt pontokban is.

Szükség van még számos más információra, ilyenek lehetnek például a digitalizált 1:10.000-es léptékű domborzati terepmodellek. Az adatgyűjtésnek része az elért hozam táblán belüli helyspecifikus felvétele, amit a betakarítás, ill. célszerűen több betakarítás során kell elvégezni az aratócséplő gépbe beépített hozammérővel, amelyhez műholdas helymeghatározó készülék csatlakozik. A hozammérők általában a magfelhordóhoz kapcsolódnak. Két alapvető típusuk használatos, az egyiknek alapelve a térfogat mérése, a másiknak a tömeg mérése. Mindegyik rendszerű hozammérőhöz illeszthető nedvességmérő, a mérés pontosságának javítása céljából.

A szakmai tapasztalatok azt mutatják, hogy az évjáráthatás miatt legalább három-öt év hozamtérképe - különböző növénykultúrákkal - lehet az alapja a precíziós gazdálkodás indításának. Ha elegendő hozamtérképpel rendelkezünk, akkor ezek alapján ki lehet jelölni az adott táblán azokat a foltokat, ahol nagy valószínűséggel azonos termést várhatunk, és amelyek azonos kezelést igényelnek. További adatok is szükségesek a teljes képhez, így a megelőző talajművelés, tápanyag-utánpótlás, meteorológiai adatok stb, és nem szabad elfeledkeznünk a gazdálkodó helyi tapasztalatáról sem.

A térképek és a gyűjtött információk feldolgozásával tehát kijelölhetők a táblán viszonylag homogén egységek, ahol - maradvány a műtrágyázás példájánál - elvégezhető a talajminta-vételezés. A talajmintavételt a tápanyag-szaktanácsadás követi, majd elkészül a területre adaptált műtrágyázási térkép. Ezt a - kezelési tervet- egy erre a célra kialakított chipkártyára írva és azt a műtrágyaszóró gép fedélzeti számítógépébe helyezve megvalósítható a műtrágya adagoknak a terület változatosságának megfelelő, differenciált kijuttatása. A műtrágyázás közben ezután a fedélzeti számítógép gyűjti a ténylegesen kijuttatott műtrágya mennyiségét.

9.9. Országos mezőgazdasági térinformatikai rendszerek

A térinformációs rendszerek mindegyikéhez szükség van földfelszíni geodéziai pontokban vagy szatellit-pályák által fizikailag megvalósuló koordináta keretrendszerre, vetületi keretre és a helyfüggő információ helyzetének megítélését viszonyítási alapon lehetővé tévő, a legáltalánosabban használt térbeli objektumokat tartalmazó térképekre, a térbeli referenciát biztosító alapadat körökre.

Az információtechnológia világában alapkövetelmény, hogy ezek az alapadat-körök digitálisan álljanak rendelkezésre, hogy tartalmukban és méretarány (felbontás) kiterjedésüket illetően teljeskörűek legyenek. További alapkövetelmény, hogy az alapadat-körökre 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 alaptérképek (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 légi fényképekkel és űrfelvételekkel biztosítja.

Az állami térképeknek alkalmasaknak kell lenniük:

- A hatósági nyilvántartások,
- Térinformatikai rendszerek,
- A honvédelmi és rendvédelmi tevékenység,
- A helyi önkormányzatok feladatai, illetve településfejlesztési és településrendezési, vagyon-nyilvántartási, információs és településirányítási tevékenység,
- A közlekedési, a hírközlési, a vízgazdálkodási tevékenység,
- Az infrastruktúra-fejlesztés,
- Az agrár-és térségfejlesztési tevékenység,
- A természet-és környezetvédelmi tevékenység,
- A bányászati szakigazgatás által elrendelt térképészeti tevékenységek, geológiai nyilvántartások,
- Kutatás-fejlesztési tevékenység,
- Adózási célú feladatok térbeli térképi referenciájaként.
- A meglévő térképállományt folyamatosan fel kell újítani, a fel nem újítható térképek helyett újakat kell készíteni.
- Új állami földmérési alaptérképet számítógépen kezelhető módon, számszerű (numerikus) meghatározással kell készíteni. Ugyanúgy kell eljárni térképfelújítás esetén is.

Irodalomjegyzék

- 9.1. Detrekői Á. – Szabó Gy. (2000): Bevezetés a térinformatikába. Nemzeti Tankönyvkiadó Rt. Budapest. ISBN 963 19 0783 X.
- 9.2. ESRI: (2006) About GIS for Agriculture.
<http://www.esri.com/industries/agriculture/>
- 9.3. Fekete A. (2000) Precíziós termesztés, Gyakorlati Agrofórum 11. évf. 14. sz..
<http://www.agrarin.hu/>
- 9.4. Földmérési és Távérzékelési Intézet (FÖMI) (2006): <http://www.fomi.hu/>
- 9.5. Sárközy F. (2006): Térinformatika
http://www.agt.bme.hu/tutor_h/terinfor/tbev.htm

Ellenőrző kérdések

Mi a GIS?

Mire kaphatunk választ egy földrajzi információs (GIS) rendszertől?

Milyen felhasználási területeit ismeri a földrajzi információs rendszereknek?

Miért szükséges a térbeli objektumok osztályba sorolása?

Mi a vetületi rendszer? Mi az EOV?

Mi a raszteres adatkezelés lényege?

Mi a vektoros adatkezelés lényege?

Mi a hibrid adatmodellű GIS szoftver koncepciója?

Milyen hardvereszközök használhatók adatgyűjtésre a térinformatikai rendszereknél?

Hogyan történik a térinformatikai rendszerek adatainak tárolása?

Mikor beszélünk a precíziós gazdálkodásról?

Milyen eszközök szükségesek a precíziós gazdálkodáshoz?

Mire használhatók a hozamtérképek?

Milyen követelményeket kell kielégíteni az állami térképeknek?

10. ADATBIZTONSÁG ÉS AZ ELEKTRONIKUS KERESKEDELEM

10.1. Az adatbiztonság kérdései

Az adatok értékét az általuk hordozott információ adja. Ezen értékek biztonsága egyéni felhasználók esetében és egy nagyvállalat információs rendszerében egyaránt fontos szempont. Legyen szó személyes adatokról vagy üzleti érdekeket képviselő, versenyelőnyt jelentő adatokról, gondoskodnunk kell azok megóvásáról, mind hozzáférhetőségüket, mind fizikai integritásukat tekintve.

Az informatikai rendszerek biztonságának szempontjai összetettek, a rendszer fizikai összetevőire, a hardverelemek működőképességére és a rendszerben tárolt adatok biztonságára egyaránt vonatkoznak. Az adatbiztonság fogalma: az adattároló eszközök védelme az ártó szándékú hozzáféréstől, változtatástól, rombolástól. A továbbiakban elsősorban az adatbiztonságra összpontosítva áttekintjük a lehetséges veszélyforrásokat, a rendszerek sebezhetőségének kérdéseit, az ezekből adódó károkat, illetve a védelmi stratégiákat.

10.1.1. Veszélyforrások és lehetséges károk

A veszélyforrás a biztonsági szint negatív változását okozó tényező. Tehát minden olyan tényező, ami az információk bizalmasságát, sértetlenségét és rendelkezésre állását veszélyezteti. A veszélyforrások az alábbi kategóriákba sorolhatók [10.1]:

- Személyekhez kapcsolódó külső és belső veszélyforrások
 - szabályozatlanság,
 - bosszúállás,
 - szakképzetlenség,
 - gondatlanság,
 - megvesztegetés
- tápáramellátás, villámvédelem
 - üzemzavar,
 - villámcsapás
- tűz- és vagyonvédelem
 - tűz,
 - illetéktelen behatolás,
 - robbanás, baleseti károsodás
- épület, szerverszoba, számítógépközpont
 - katasztrófa,
 - fizikai behatolás
- hardver, hálózat, rendszerszoftver
 - illetéktelen rácsatlakozás,
 - illetéktelen hozzáférés
 - szakszerűtlen üzemeltetés, karbantartás
 - beszerzési politika hiánya
- alkalmazott szoftver
 - vírus,

- illetéktelen szoftver installáció,
- szakszerűtlen tervezés
- dokumentumok
 - illetéktelen használat,
 - illetéktelen másolás

Az adatokhoz kapcsolódó veszélyforrások tehát az alábbiak lehetnek:

- Fizikai jellegűek: természeti-környezeti veszélyforrások (árvíz, földrengés, áramkimaradás) vagy szándékos / véletlen emberi tevékenység okozta kár a rendszer fizikai összetevőiben, amelyek következtében elveszhetnek vagy sérülhetnek az adatok.
- Előre nem látható meghibásodások a hardver- vagy szoftverelemekben. Ide sorolhatjuk a számítógépes vírusokat is mint a szoftverekre vonatkozó veszélyforrást.
- A adatok hozzáférhetőségével kapcsolatos veszélyforrások:
 - külső behatolók, hackerek,
 - belső, hozzáféréssel nem rendelkező munkatársak „betörései”,
 - professzionális bűnszövetkezetek.

A veszélyforrásokkal kapcsolatban megjegyezzük, hogy a támadások sokszor nem közvetlenül az adatokat érik, hanem a rendszer hardver- és szoftverelemeinek sérülésén keresztül áttételesen okoznak károkat.

Az okozott kár lehet:

- számítógépek, hálózatok, szolgáltatások ideiglenes kiesése (az adatok rendelkezésre állása kerül veszélybe),
- a rendszer erőforrásainak illetéktelenek általi használata,
- az adatok ellopása, illetéktelenek általi módosítása, törlése, ehhez kapcsolódóan a vállalat titkos belső információnak, üzleti titkoknak illetéktelen kezekbe jutása – sérülhet az adatok bizalmassága és hitelessége.

10.1.2. Védelmi intézkedések

Az informatikai rendszerek védelmi stratégiájának tervezésekor a fenti kockázati tényezőket és az esetlegesen okozható kár nagyságát egyaránt tekintetbe kell venni. A védelmi költségeket tekintve nemcsak azok összege, hanem a ráfordítás módja és mértéke is lényeges, amelyet a még elviselhető kockázat mértéke határoz meg. A rendszerek védelemhez kapcsolódó két fontos fogalom az adatbázisokra vonatkozó *privacy* és az *integrity*: a *privacy* az illetéktelenek hozzáféréseinek megakadályozását takarja, míg az *integrity*, azaz az adatok integritásának követelménye azt jelenti, hogy a hozzáférésre jogosultak se „ronthassák el” az adatbázist például adatok véletlen törlésével – lépéseket kell tehát tenni az ilyen hibák helyreállíthatósága érdekében.

A fizikai károktól való védelem lehetőségei a védeni kívánt rendszer jellegétől függően alakulnak: egy otthon használt személyi számítógép esetében a körültekintő elhelyezés és használat, esetleg szünetmentes áramforrás alkalmazása is elegendő lehet, míg például egy nagyvállalat szerverparkja egészen más nagyságrendű védelmet igényel: ezeket általában olyan zárt – és őrzött – helyiségekben helyezik el, ahol a hardveregységek stabil hőmérsékleti viszonyok és áramszolgáltatás mellett üzemelhetnek. A szoftverkárokat

okozó vírusok és az illetéktelen behatolók azonban mérettől függetlenül minden informatikai rendszerre fenyegetést jelentenek.

A számítógépes vírusok adathordozókon vagy számítógép-hálózatokban áramló állományokon keresztül támadnak meg rendszereket, és rendszerint a továbbításra felhasznált közeget megfertőzve „szaporodnak”. Ezek a rosszindulatú programok rendszerint nehezen észlelhetők és gyorsan terjednek, adatokat rombolva és a számítógép működését megzavarva. Néhány vírushajtó (a teljesség igénye nélkül):

- programférgek: önálló programként terjednek, futtatható állományok tartalmát cserélik a sajátjukra
- trójai programok: hasznos programnak álcázott vírusok, ide tartoznak a kémprogramok is
- boot-vírusok: a merevlemezrel kapcsolatos alapvető információkat tároló ún. boot szektor tartalmát helyezik át vagy írják felül
- makrovírusok: irodai alkalmazások (pl. MS Word) belső programnyelvét használják fel terjedésükhöz.

A vírusveszély semlegesítésére szolgáló antivírus programok manapság minden rendszer kötelező elemei. Ezek olyan szoftverek, amelyek képesek észlelni és irtani a számítógépet fertőzni kívánó vírusokat; fejlettebb verziók a memóriában futnak, hogy a gép működése közben (valós időben, „real-time”) megvédelmezzék a feldolgozást, védelmet nyújtsanak a hálózatról bejövő fájlokban levő vírusoktól.

Az adatok hozzáférhetőségét különféle jogosultságokkal szabályozzák, e szabályozást játsszák ki a hackerek. A hackerek fogalmának többféle értelmezése is lehetséges, egy általános meghatározás szerint a hacker olyan személy, aki hozzáférést teremt a számítógéphez hasznoszerzés, bűncselekmény elkövetése, vagy személyes szórakozás céljából. A hackerekkel szembeni védekezést segítő eszközök közé tartoznak a tűzfalak.

Az Internetre csatlakozó számítógépek a csatlakozással egyidejűleg számos csatornát nyitnak meg a külvilág előtt, amelyeken keresztül adatáramlás történhet. Ennek kockázatosága nyilvánvaló, ezért minden ilyen csatornát, kaput zárva kell tartanunk, kivéve azokat, amelyekre éppen szükségünk van. Ehhez szükséges segédeszköz a tűzfal, ami alatt egy nagyobb belső hálózat külvilág felé kapuként funkcionáló, elkülönített számítógépet vagy (többnyire) egy speciális védelmi szoftvert is érthetünk. A tűzfalak két fő csoportba sorolható: léteznek csomagszűrő eszközök és ún. applikációs tűzfalak. Az első csoportba tartozó eszközök a hálózati forgalom minden csomagjának fejlécét ellenőrzik, és szükség szerint blokkolják azokat (ez az ún. csomagszűrés). Az applikációs tűzfalak nemcsak a fejléceket, hanem a teljes adatfolyamot ellenőrzik.

10.2. E-aláírás

Az információs társadalom kialakulásához vezető úton mérföldkönek 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 átültetésére, 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 (gépeli), 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 lefedí. 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.

10.3. 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 sajátkezű 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. egy cégbejegyzésnél a cégbíróságnál a cég vezetőjének aláírási címpéldányt kell készíteni közjegyző előtt, amelyet saját kézjegyével 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

10.3.1. 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 kézjegy 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 - akárcsak 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 aláíró. 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 aláírás mintákkal és a hasonlóságot százalékos értéként adja meg. Hátrányaként talán azt említhetjük meg, hogy az aláírás rögzítéséhez digitalizáló tábla szükséges.

10.3.2. 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.

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.

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.

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 árnevet, 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 az aláíró más személy (szervezet) képviselőjében történő aláírásra jogosítsa fel, de ebben az esetben a tanúsítvány kibocsátását megelőzően a képviselői jogosultságot igazolni kell. A képviselői jogosultság meglétét a hitelesítés-szolgáltató köteles ellenőrizni. A tanúsítvány kibocsátásáról a képviselt személyt (szervezetet) haladéktalanul tájékoztatni kell.

10.4. 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

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 [10.3]

10.4.1. 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”

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” [10.3]

Meghatározás 3

„Minden üzleti tevékenység, amely részben vagy egészen digitális úton zajlik.” [10.3]

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.

10.4.2. 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 kereskedelembé. Az új üzleti helyzetek, mint amilyen az e-commerce, újfajta gondolkodást igényelnek” .

Az elektronikus kereskedelem előnyei:

A kereskedelemben régóta ismert és elfogadott 6M szabály szerint tevékenységünk akkor lehet sikeres, ha

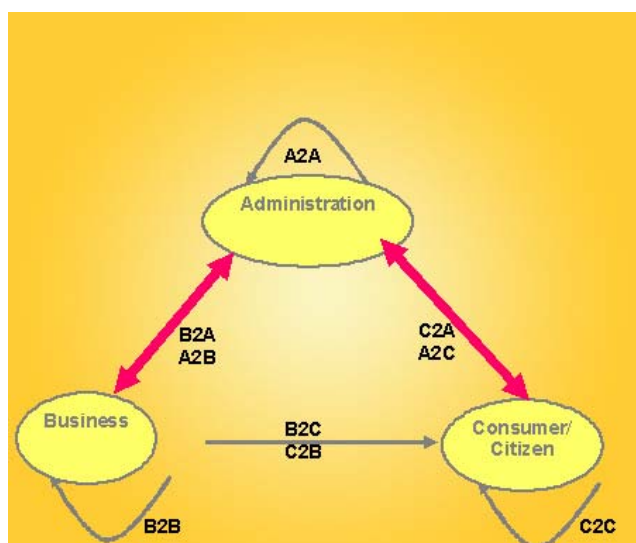
- a megfelelő anyagot, energiát, információt, személyt;
- a megfelelő mennyiségben;
- a megfelelő minőségben;
- a megfelelő időpontban;
- a megfelelő (minimális) költséggel juttatjuk el;
- a megfelelő helyre.

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.

10.4.3. 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 aktorok minősége alapján foglalják rendszerbe az elektronikus kereskedelmet. A fontosabb szereplők közötti kapcsolatokat és modell típusokat a 10.1. ábra mutatja.



10.1. ábra: A modellek kapcsolatai

10.4.3.1. Administration to Administration (A2A)

Az A2A az elektronikus közigazgatás meghatározása, melynek során az egyes kormányzati intézmények elektronikus úton cserélnek 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.

10.4.3.2. 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, mint a

- Business to Administration (B2A)
- Elektronikus ügyintézés. Egyes vélemények szerint nem tartozik az e-kereskedelem körébe.

10.4.3.3. Business to Business (B2B)

A B2B, az elektronikus vállalkozói 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 vállalkozói 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

10.4.3.4. 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 vállalkozói

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 várható, hogy egy újabb boom kezdődjön az B2C kereskedelemben, és legalább megközelítse a B2B fejlődését, ha a hozzá kapcsolódó költségek drasztikusan csökkennének.) [10.10].

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;
- ugyanezen a helyeken folytatott ismeretterjesztő tevékenység (dokumentumfilmek, riportok, passzív weblapok).

Fontos rámutatni, hogy a felsorolt tevékenységek mindegyike passzív, vagyis egyutas. A hirdető csak információ 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 általa 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-businessnek 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.

10.4.3.5. 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 „lejáró” más formában nehezen értékesíthető árucikkek is.

A C2C kereskedelem lényege a kritikus tömeg, vagy kritikus méret elérése (LAUDON, TRAVER, 2002) Mind a vevők, mind az eladók szívesebben választanak olyan elektronikus piacteret, 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-piacterek versenyképesek lehetnek a globális nagy elektronikus piacterekkel.

10.4.3.6. 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 félnél 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.

Irodalomjegyzék

- 10.1. www.standard-team.hu/prg/cikkek.php
- 10.2. Bögel, GY. (2000): Verseny az elektronikus üzletben. Műszaki kiadó, Budapest..
- 10.3. <http://www.webopedia.com>
- 10.4. Coltman COLTMAN, T. és társai (2002): E-business: Revolution, Evolution, or Hype? California Management Review. Vol. 44. No. 1.
- 10.5. Nemeslaki A. és társai (2004): E-business. Mentor Kiadó. Marosvásárhely
- 10.6. Barnes, S., Hunt, B. (2001): E-Commerce&V-Business. Butterworth-Heinemann, Oxford.
- 10.7. Bock, W. H., Senné, J. N. (1997): Jövedelmező internet. Bagolyvár Kiadó, Budapest
- 10.8. Kondricz P., Tímár A. (2000): Az elektronikus kereskedelem jogi kérdései. KJK KERSZÖV. Budapest.
- 10.9. Laudon, K. C., Traver, C. G. (2002): E-commerce. Addison-Wesley. Boston
- 10.10. Bowden, S. és társai (2000): Adoption and implementation of e-business in New Zealand. Preliminary Results. Hamilton

Ellenőrző kérdések

Melyek az információs rendszer biztonságát veszélyeztető tényezők?

Mely tényezők kapcsolatosak a kiszolgáltató személyzettel?

Melyek az adatokhoz kapcsolódó veszélyforrások?

Milyen védelmi intézkedéseket célszerű fogatosítani egy vállalati integrált információs rendszerben?

Mire szolgál az elektronikus kereskedelem? Melyek az előnyei a hagyományos formákhoz képest?

Melyek az elektronikus aláírás módszerei?

Mi a hitelesítés?

Mire szolgál az időbélyegző?

Az elektronikus kereskedelemnél hol jelenik meg az állam, mint szereplő?

11. 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:

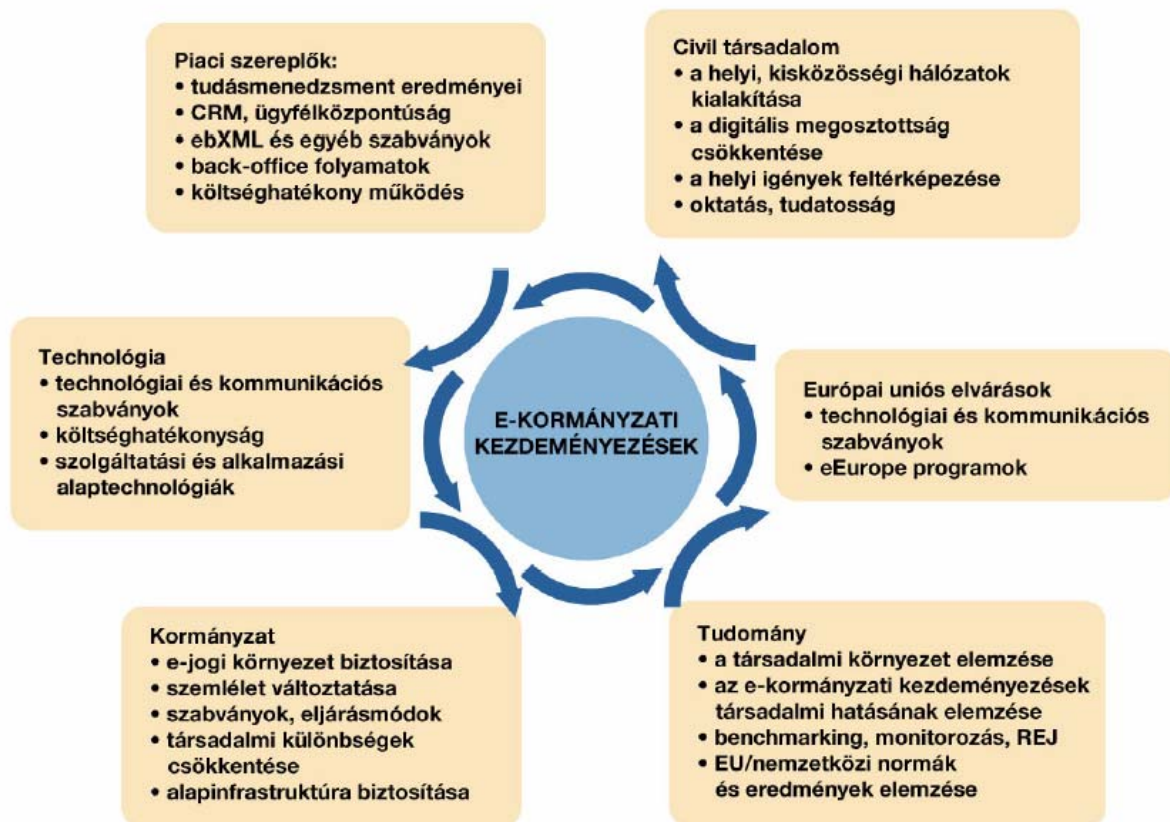
- A fiatalok beléptetése a digitális korszakba
- Olcsó Internet-hozzáférés
- Az elektronikus kereskedelem terjedésének gyorsítása
- Gyors Internet a kutatók és a diákok számára
- Intelligens kártyák a biztonságos elektronikus hozzáféréshez
- Kockázati tőke (kis és középvállalatok) számára
- Elektronikus 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 demokráciamodellel 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.

11.1. E-kormányzat fogalma

Az elektronikus kormányzat (e-kormányzat) kifejezés mára mind a köznyelvben mind az állami zsargonban egy univerzális jövőképjavító tényezővé nőtt 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 információszolgáltatás és egyszer kommunikáció, információ-kérés iránt mutatkozik érdeklődés. Ezekre az elvárásokra épülnek az e-kormányzás működését is meghatározó alapelvek, követelmények:



11.1. ábra: Az e-kormányzati fejlesztésekre ható erőtér [11.1.]

- **nyitottság:** a közigazgatás legyen kész fogadni és feldolgozni a nyilvánosság, a társadalom és gazdaság szereplőinek elvárásait és javaslatait;
- **részvétel:** az állampolgárokat, vállalkozásokat, közösségeket érintő kérdések megvitatásába, a döntések előkészítésébe minél szélesebb körben be kell vonni az érintetteket. A központi és a helyi kormányzatnak olyan politikai környezetet kell biztosítania, amely részvételre, bekapcsolódásra ösztönzi a polgárokat, közösségeket. A közigazgatásnak folyamatosan jeleznie kell ezirányú elkötelezettségét;
- **számmonkérhetőség:** a döntéshozatali folyamatoknak átláthatóbbakká kell válniuk, lehetőséget kell biztosítani a bekapcsolódásra; a nyilvánosság számára egyértelművé kell tenni, hogy ki, miért felel;
- **visszacsatolás:** az érintettek számára lehetőséget kell biztosítani a vélemények, javaslatok, észrevételek visszacsatolására a döntéshozók, politikusok, köztisztviselők felé;
- **hatékonyság:** legyen hatékony a döntéshozatali mechanizmus, az elfogadott határozatok, jogszabályok végrehajtása;
- **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.

11.1. táblázat: On-line szolgáltatási elvárások

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 – útlevel, 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

Az *e*-kormányzati kezdeményezések egy olyan bonyolult, összefüggő társadalmi erőterben értelmezhetők, 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.

11.2. 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 szervezettsége, 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 és az információminőség.

11.2.1. 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

11.2.2. 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

11.3. 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:

13.2. táblázat: E-ügyintézés: az önkormányzati ügyintézési szolgáltatások elektronizálása

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)

Az e-ügyintézés megvalósítja az EU e-Europe programjának „Common List of Basic Public Services” ajánlásában megfogalmazott mind a négy szolgáltatási szintet, s biztosítja az ügyfelek részére is a függőben lévő ügyek 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 érték; 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 ügýtípusok milyen arányban szerepelnek a négy szint valamelyikében; milyen arányban érhetők el a különböző szint; 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 kistelepülések 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 (földhivatalok, közüzemi szolgáltatók stb.).

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 közhasznú információszolgáltatás, ügyfélfórum, -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 a közigazgatási alkalmazások, az adminisztratív (kölségvetési) modulok, vagy az irodaautomatizálás és kommunikáció, vezetői információk és döntéstámogatás, illetve tudásmenedzsment. Az önkormányzati információs rendszer (jövőkép) funkcionalitását a megoldási térkép (11.2. ábra) foglalja össze.

1.1.3. 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 e-Europe 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 használta és rendelkezzenek azok használatához szükséges tudással.

Magyarországon ma majdnem minden régiónak és megyének, valamint egyre több kistérségnek, város, falunak elfogadták az intelligens térségi-települési stratégiáját, valamint operatív programját, amelyekben szinte kivétel nélkül kiemelt feladat az e-közigazgatás bevezetése és elterjesztése. Evvel a területfejlesztési tervezés szintjén jórészt lezajlott a közigazgatási paradigmaváltás előkészítése.

Az e-közigazgatás: digitális közigazgatás. De nem(csak) a régi közigazgatás elektronizálása, hanem egyúttal újfajta közigazgatás. 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 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 üzleti- és civil szféra részére jobb minőségű szolgáltatást nyújtanak.

11.3.2. Szolgáltató közigazgatás

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 átszerveződésre 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.

E-ügyintézés (elektronikus ügyintézési szolgáltatások)	On-line információk a közigazgatási szolgáltatásokról (ügyleírások)	Nyomtatványok, űrlapok letöltése (az ügyintézéshez)	Nyomtatványok, 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, ügyféllevezés		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	Vagyonkezelés, vagyongazdálkodás	Település-fejlesztés, beruházások	Település-üzemeltetés	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, humánerőforrás-gazdálkodás,		Anyag-, tárgyeszköz-gazdálkodás,	Projektok és programok (beruházások, fejlesztések)		Elektronikus iktatás, 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	Önkorm. intelligencia (döntéstámogatás, adattárház, adatbányászat)		Dokumentáció-menedzsment		Csoportmunkaszoftverek (groupware),		Internet, intranet, extranet (mint információforrás)		Mesterséges intelligencia

11.2. ábra Az önkormányzati információs rendszer funkciói [11.2]

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 egyszerű,
- alkalmazásával megvalósul a „mindent egy helyen intézni” elv,
- egyszerű és hatékony információáramlást biztosít,
- interaktív,
- biztosítja a tapasztalatcserét,
- bárki részére, bárhol, bármilyen eszközzel lehetővé teszi az elérhetőséget
- 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 tudnak 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 vezetés és a termelési rendszerek hatékonyságának növelése érdekében, valamint az üzleti tevékenység elektronikus szervezésére. Hálózati kapcsolattrendszer 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.

11.4. 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.

11.4.1. 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 WORK FLOW alapú fejlesztésnek. A kistelepülések 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 funkción 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.

11.4.2. Rendszerfejlesztés

A rendszerfejlesztés célja első fázisban korszerű tájékoztatás és szakszerű ügyintézés segítése. Második fázisban a hivatalban működő WORK FLOW 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.

Irodalomjegyzék

- 11.1. Kormányzat Stratégia és Programterv 2005. MeH Elektronikus Kormányzat Központ
- 11.2. IHM Elektronikus Önkormányzat Stratégia 2003

Ellenőrző kérdések

Mi az elektronikus kormányzat? Melyek a feladatai?
Melyek az e-közigazgatás és az e-vállalatok kapcsolódási pontjai?
Hol kapcsolódik az e-kormányzat programja az EU irányelveihez?
Hogyan működik az elektronikus közbeszerzés?
Mi az az egyablakos ügyintézés?
Hol jelentkezik a back-office és a front-office integrációja?
Melyek az önkormányzatok feladatai az e-ügyintézés területén?
Mire szolgálnak az ügymeneti modellek?
Miben nyilvánul meg az e-demokrácia?

Ellenőrző kérdések

A kérdésre kattintva megkapja a választ

1. Fejezet

- 1.1. *Melyek a gazdaságinformatika főbb részterületei?*
- 1.2. *Mit értünk az alatt, hogy a gazdaságinformatika interdiszciplináris tudomány?*
- 1.3. *Jellemezze a virtuális vállalatot!*

2. Fejezet

- 2.1. *Melyek a programnyelvek generációi?*
- 2.2. *Mit tartalmaz egy assembly nyelven írt program?*
- 2.3. *Sorolja fel a programfordítás szakaszait!*
- 2.4. *Mi a paradigma? Milyen programozási paradigmákat ismer?*
- 2.5. *Sorolja fel és jellemezze a strukturált programozás alapelemeit!*
- 2.6. *Mit jelent az, hogy az objektumok fekete doboznak tekinthetők?*
- 2.7. *Milyen változótípusokat (adattípusokat) ismer?*
- 2.8. *Mi a különbség az elől- és hátultesztelő ciklusok között?*
- 2.9. *Mikor célszerű változótömböket alkalmazni?*
- 2.10. *Mit határoznak meg a Font, Text és Enabled tulajdonságok?*
- 2.11. *Soroljon fel 4 ActiveX vezérlőt, és adja meg, mikor használjuk ezeket a típusokat!*

3. Fejezet

- 3.1. *Mi az az algoritmus?*
- 3.2. *Hogyan működik a két természetes szám legnagyobb közös osztóját meghatározó algoritmus?*
- 3.3. *Mi a tulajdonság keresési algoritmusok működési elve?*
- 3.4. *Milyen rendezési algoritmusokat ismer? Hasonlítsa össze a hatékonyságukat!*

4. Fejezet

- 4.1. *Milyen elvárások vannak az adatbázis kezelő rendszerrel szemben?*
- 4.2. *Az adatbázis kezelő rendszerek felépítése! Az egyes elemek feladatait ismertesse!*
- 4.3. *Mit értünk metaadatok alatt?*
- 4.4. *Milyen adatmodellek ismertek?*
- 4.5. *Jellemezze a hierarchikus adatmodellt!*
- 4.6. *Jellemezze a hálós adatmodellt (kétszintű hálós adatmodell, Codasyl-hálós)!*
- 4.7. *Jellemezze a relációs adatmodellt!*
- 4.8. *Mit értünk redundancián, inkonzisztencián, normalizáláson, normálformán?*
- 4.9. *Mit jelent a funkcionális függés?*
- 4.10. *Milyen objektum típusok érhetők el az Access Adatbázis ablakában?*
- 4.11. *Mit tartalmaz és milyen szerkezetű a tábla?*
- 4.12. *Mire használhatók a lekérdezések és az űrlapok?*
- 4.13. *Milyen eszközökkel védhetők az Accessben az adatbázisok?*
- 4.14. *Milyen engedélyek adhatók a felhasználóknak és a csoportoknak az Accessben?*

- 4.15. Milyen adattípusok adhatók meg az Accessben?
- 4.16. Milyen mezőtípusok vannak az Accessben és mik a jellemzőik?
- 4.17. Milyen mező tulajdonságok adhatók meg a táblákban?
- 4.18. Milyen típusú lekérdezések készíthetők?
- 4.19. Mi a lényege a keresztábrlás lekérdezésnek?
- 4.20. Hogyan kezdődik a lekérdezés létrehozása?
- 4.21. Milyen grafikus eszköz segíti a lekérdezés tervezői munkát?
- 4.22. Milyen szakaszok különíthetők el az űrlapon?
- 4.23. Mi az SQL?

5. Fejezet

- 5.1. Mi a számítógépi hálózatok célja?
- 5.2. Milyen hálózati struktúrákat ismer ?
- 5.3. Mi a hálózati architektúra?
- 5.4. Sorolja fel miért előnyös a számítógépeket hálózatba kapcsolni!
- 5.5. Melyek a számítógépes hálózatok legfontosabb jellemzői?
- 5.6. Mi a topográfia és a topológia közötti különbség?
- 5.7. Határozza meg a protokoll fogalmát!
- 5.8. Mik azok a hálózati rétegek?
- 5.9. Miért fontos a hálózatok szabványosítása? Mik azok a de-jure és a de-facto szabványok?
- 5.10. Fogalmazza meg az OSI-modell filozófiáját!
- 5.11. Milyen rétegekből épül fel az OSI modell?
- 5.12. Mi a fizikai réteg feladata?
- 5.13. Mi az adatkapcsolati réteg szerepe?
- 5.14. Mit biztosít a hálózati réteg?
- 5.15. Milyen célokat valósít meg a szállítási réteg?
- 5.16. Mire használják az együttműködési rétegeket?
- 5.17. Milyen feladatokat lát el a megjelenítési réteg?
- 5.18. Ismertesse az alkalmazási réteg szerepét az OSI modellben!
- 5.19. Milyen rétegekből áll az OSI hivatkozási modell?
- 5.20. Mi az egyes rétegek feladata?
- 5.21. A TCP/IP hivatkozási modell és az ISO/OSI összehasonlítása.
- 5.22. A hálózatok hardver elemei.
- 5.23. A hálózatok összekapcsolási lehetőségei.
- 5.24. Mire szolgál a kliens-szerver modell?

6. Fejezet

- 6.1. Hogyan használható a telefonvonal az internet eléréshez?
- 6.2. Mi az az ADSL?
- 6.3. Hogyan működik a mobil internet kapcsolat?
- 6.4. Melyek az internet szolgáltatások?
- 6.5. Mi az FTP?
- 6.6. Mi a World Wide Web?
- 6.7. Hogyan programozható a Web?
- 6.8. Hogyan működik az internet telefonálás?

7. Fejezet

- 7.1. *Ismertesse az információs rendszer fogalmát!*
- 7.2. *Mire szolgál a vezetői információs rendszer?*
- 7.3. *Az információrendszer típusok közötti kapcsolatok leírása*
- 7.4. *A stratégiai információrendszerek meghatározása*
- 7.5. *Miként támogathatják az információrendszerek az üzleti stratégiát?*
- 7.6. *Mi a döntés előkészítő rendszerek feladata?*
- 7.7. *Mi az az integrált vállalatirányítási rendszer?*
- 7.8. *Hogyan változott az integrált információs rendszerek fejlesztési filozófiája?*
- 7.9. *Mik az ágazati információs rendszerek?*

8. Fejezet

- 8.1. *Hogyan kell értelmezni az Európai Bizottság és az Európai Tanács agrárinformációs rendszerekre hozott rendeleteit, irányelveit, határozatait és ajánlásait a tagországoknak?*
- 8.2. *Az EU informatikai struktúráját, vagy a Közös Agrárpolitikát tartja e stabilabbnak?*
- 8.3. *Hogyan csoportosítaná az EU informatikai rendszerét?*
- 8.4. *Mik a Primer Információs Rendszer elemei?*
- 8.5. *Mi a feladata az Unió Agrárstatisztikai Rendszerének?*
- 8.6. *Mi a Tesztüzemi Rendszer feladata, honnan gyűjt adatokat és mire használja azokat?*
- 8.7. *Kinek szolgáltatnak információkat a Piaci Információs Rendszerek?*
- 8.8. *Mi a fő feladata az Integrált Igazgatási és Ellenőrző Rendszernek?*

9. Fejezet

- 9.1. *Mi a GIS?*
- 9.2. *Mire kaphatunk választ egy földrajzi információs (GIS) rendszertől?*
- 9.3. *Milyen felhasználási területeit ismeri a földrajzi információs rendszereknek?*
- 9.4. *Miért szükséges a térbeli objektumok osztályba sorolása?*
- 9.5. *Mi a vetületi rendszer? Mi az EOVS?*
- 9.6. *Mi a raszteres adatkezelés lényege?*
- 9.7. *Mi a vektoros adatkezelés lényege?*
- 9.8. *Mi a hibrid adatmodellű GIS szoftver koncepciója?*
- 9.9. *Milyen hardvereszközök használhatók adatgyűjtésre a térinformatikai rendszereknél?*
- 9.10. *Hogyan történik a térinformatikai rendszerek adatainak tárolása?*
- 9.11. *Mikor beszélünk a precíziós gazdálkodásról?*
- 9.12. *Milyen eszközök szükségesek a precíziós gazdálkodáshoz?*
- 9.13. *Mire használhatók a hozamtérképek?*
- 9.14. *Milyen követelményeket kell kielégíteni az állami térképeknek?*

10. Fejezet

- 10.1. *Melyek az információs rendszer biztonságát veszélyeztető tényezők?*
- 10.2. *Mely tényezők kapcsolatosak a kiszolgáló személyzettel?*
- 10.3. *Melyek az adatokhoz kapcsolódó veszélyforrások?*
- 10.4. *Milyen védelmi intézkedéseket célszerű fogantatni egy vállalati integrált információs rendszerrel?*

- 10.5. Mire szolgál az elektronikus kereskedelem? Melyek az előnyei a hagyományos formákhoz képest?*
- 10.6. Melyek az elektronikus aláírás módszerei?*
- 10.7. Mi a hitelesítés?*
- 10.8. Mire szolgál az időbélyegző?*
- 10.9. Az elektronikus kereskedelemnél hol jelenik meg az állam, mint szereplő?*

11. Fejezet

- 11.1. Mi az elektronikus kormányzat? Melyek a feladatai?*
- 11.2. Melyek az e-közigazgatás és az e-vállalatok kapcsolódási pontjai?*
- 11.3. Hol kapcsolódik az e-kormányzat programja az EU irányelveihez?*
- 11.4. Hogyan működik az elektronikus közbeszerzés?*
- 11.5. Mi az az egyablakos ügyintézés?*
- 11.6. Hol jelentkezik a back-office és a front-office integrációja?*
- 11.7. Melyek az önkormányzatok feladatai az e-ügyintézés területén?*
- 11.8. Mire szolgálnak az ügymeneti modellek?*
- 11.9. Miben nyilvánul meg az e-demokrácia?*