

SELYE JÁNOS EGYETEM

GAZDASÁGTUDOMÁNYI ÉS INFORMATIKAI KAR

Ing. Ondrej Takáč, PhD.,
PaedDr. Ladislav Végh, PhD,
PaedDr. Krisztina Czakóová, PhD.

Vállalati információs rendszerek I.

2026

Ing. Ondrej Takáč, PhD.
PaedDr. Ladislav Végh, PhD.
PaedDr. Krisztina Czakóová, PhD.

Vállalati információs rendszerek I.

(Egyetemi tankönyv)



Univerzita J. Selyeho – Selye János Egyetem
Fakulta ekonómie a informatiky –
Gazdaságtudományi és Informatikai Kar

Ing. Ondrej Takáč, PhD.
PaedDr. Ladislav Végh, PhD.
PaedDr. Krisztina Czakóová, PhD.

Vállalati információs rendszerek I.

(Egyetemi tankönyv)

Komárno, 2026

© **Ing. Ondrej Takáč, PhD.**, Selye János Egyetem
© **PaedDr. Ladislav Végh, PhD.**, Selye János Egyetem
© **PaedDr. Krisztina Czakóová, PhD.**, Selye János Egyetem

Recenzenti - Recenzensek:

Dr. habil. Gombkötő Nóra, Széchenyi István Egyetem
Ing. Ladislav Tóth, PhD., Slovenská poľnohospodárska Univerzita v Nitre
Ing. Norbert Gyurian, PhD., Selye János Egyetem

Univerzita J. Selyeho – Selye János Egyetem
Komárno, 2026

ISBN 978-80-8122-548-2
EAN 9788081225482

Köszönet

Ezúttal szeretnénk kifejezni köszönetünket a recenzenseknek, akik nagymértékben hozzájárultak e tankönyv létrejöttéhez. Köszönjük az észrevételeiket és tanácsaikat.

A szerzők

Előszó

„Az akadályok lesznek azok, amik maradnak, ha nem fogod látni a célodat.”
(Brian Tracy)

Napóleoni háborúk pusztítottak Európában, Angliában egyre több gőzgéppel működő gyár épült, és az európai nagyhatalmak folytatták a gyarmati terjeszkedésüket, különösen Ázsiában és Afrikában. Ezen események árnyékában egy francia szövőmester, Joseph Marie Jacquard 1804-ben Basile Bouchon ötlete alapján tökéletesítette és népszerűsítette a lyukkártyás szövőszékét, amely iparilag is sikeresen alkalmazható volt. Valószínűleg maga sem sejtette, és nem is sejthette, hogy mennyire befolyásolja azt a jövőt, amelyet mi ma az első, a második és a harmadik számítógépek generációinak ismerünk. Jacquard lyukkártyás szövési módszere bonyolult minták létrehozását tette lehetővé, és ez a technika később inspirálta Herman Hollerith-et, aki a lyukkártyákat adatfeldolgozásra használta (1890). A lyukkártyák az informatika korai memóriaeszközeiként is híressé váltak, melyet már az említett három számítógépek generációi alatt adattárolóként használtak. A számítógépek második generációja idején, amikor Jurij Gagarin is először repült az űrbe, megjelentek az 1950-es évekből korábbi adatfeldolgozási rendszerekre épülve az első komplex, számítógépes támogatású modern vállalati információs rendszerek ősei. Megemlíthetünk egy fontos együttműködést. Az 1960-as években, amikor a J. I. Case, egy traktorokat és építőipari gépeket gyártó vállalat együttműködött az IBM-mel, kifejlesztésre került sor az első anyagszükséglet-tervezési rendszerre (MRP). A második generációs számítógépek elég technológiai alapot nyújtottak az ilyen korai vállalati rendszerek létrejöttéhez, amelyek a későbbi – modern vállalati információs rendszerek fejlődését alapozták meg.

A vállalati információs rendszerek alig 70 éves létezésük alatt hatalmas fejlődésen mentek keresztül, mindig beépítve a legújabb felfedezéseket és technológiai trendeket. A fejlődés iránti törekvés ma sem állt meg, amikor a Web 3.0, az Ipar 5.0, online megoldások és a mesterséges intelligencia küszöbén állunk. Egyre nagyobb hangsúly kerül az ember és a technológia közötti együttműködésre, az automatizálásra,

folyamatirányítás támogatására, az előre jelző képesség növekedésére az adatmegosztás decentralizált módjára és jobb adatbiztonságra. Ráadásul, a modern idők megkövetelik a modern webes technológiák használatát annak érdekében, hogy a vállalatok közelebb kerüljenek az ügyfelekhez, vásárlókhoz vagy partnerekhez. A webes technológiák, a közösségi eszközök és az együttműködési platformok használata a vállalati környezetben elősegítheti a tudásmegosztást, az együttműködést az alkalmazottak, a partnerek és az ügyfelek között. A vállalati információs rendszerek manapság kulcsszerepet játszanak nem csak ezekbe a törekvésekben, hanem sok más automatizálási tevékenységben. Segítségükkel a vállalatok hatékonyabban kezelhetik az adatokat, javíthatják a döntéshozatali folyamatokat és gyorsabban reagálhatnak a piaci változásokra. Az ilyen rendszerek lehetővé teszik a különböző részlegek közötti együttműködést és biztosítják az átlátható, egységes adathozzáférést. Ezek a rendszerek támogatják a költségcsökkentést, elősegítik a versenyképesség növelését és az erőforrások optimális kihasználását. Így a vállalatok fenntarthatják piaci pozíciójukat és hosszú távon sikeresek lehetnek. Az információs rendszerek fejlődése az elmúlt évtizedekben nemcsak a vállalatok belső folyamataira, hanem az egész üzleti környezetre is mély hatást gyakorolt. Ezek a rendszerek lehetővé tették az adatok valós idejű gyűjtését és elemzését, így ezen információk pontosabb és gyorsabb felhasználását.

A tankönyvben igyekeztünk komplex bevezetést nyújtani az információs, valamint vállalati információs rendszerekbe, ezek alkalmazásába, elterjedésébe, fejlesztésébe és felépítésébe a modern eszközök és trendek kontextusában. A tankönyvet több fejezetre osztottuk fel a modern technológiai fejlesztések és megoldások összefüggésében, hogy átfogó képet tudjunk nyújtani a vállalati információs rendszerek által jelenleg biztosított lehetőségekről. A vállalati információs rendszerek fejlődése folyamatos és dinamikus folyamat, amelyre nagy hatással vannak a technológiai innovációk. Reméljük, hogy ez a tankönyv ösztönzi majd az olvasót további tanulmányokra és kutatásokra ezen izgalmas területen, amely egyre meghatározóbbá válik a jövőben.

Komárom, 2025

Szerzők

Tartalomjegyzék

RÖVIDÍTÉSEK JEGYZÉKE (I-II. KÖTET)..... 12
A könyvben gyakran használt angol rövidítések jegyzéke 12
A könyvben gyakran használt magyar rövidítések jegyzéke 22
1 BEVEZETŐ, ALAPVETŐ FOGALMAK MEGHATÁROZÁSA.. 24
1.1 Információk és adatok..... 24
1.2 Információs, és vállalati információs rendszer fogalma 28
1.3 A DIKW (Data–Information–Knowledge–Wisdom) modell 33
1.4 Adatok a vállalati információs rendszer hierarchiáján belül..... 35
2 VÁLLALATI INFORMÁCIÓS RENDSZEREK FELADATAI 40
2.1 A vállalati információs rendszerek tulajdonságai 42
2.2 A vállalati információs rendszerek felosztása..... 44
2.2.1 A vállalati információs rendszerek cél szerinti felosztása 44
2.2.2 Az információs rendszerek felosztása méret és használt platform szerint..... 48
2.2.3 Az információs rendszerek felosztása hozzáférés szempontjából..... 49
2.2.4 Az információs rendszerek felosztása területi hatályát tekintve 50
2.3 A vállalati információs rendszerek feladatai..... 50
3 AZ INFORMÁCIÓS RENDSZER ÉS STRATÉGIAI SZEREPE.. 58
3.1 A vállalati információs rendszer stratégiai szerepe 59
3.2 Porter modell 60
3.3 Vállalati információs rendszer infrastruktúrája 66
3.4 Modern megoldások kihívásai és veszélyei..... 68
3.4.1 Web 1.0, Web 2.0 és Web 3.0 68
3.4.2 Internet of Things (IoT)..... 70
3.4.3 BYOD és BYOC..... 71
3.4.4 Big Data és Open Data 74
3.4.4.1 Big Data..... 74
3.4.4.2 Open Data..... 78
3.5 Alkalmazások a kompetitív intelligencia támogatására 79
Ebben a fejezetben használt néhány kifejezés magyarázata 82

4 INFORMÁCIÓS ÉS KOMMUNIKÁCIÓS TECHNOLÓGIÁK, ADATGYŰJTÉS ÉS FELDOLGOZÁS 84
4.1 Hardver 85
4.1.1 A számítógépek típusai és alkalmazási területeik 85
4.1.2 A számítógépek legfontosabb egységei 89
4.1.3 Von-Neuman- és Harvard-architektúrák 91
4.2 Szoftver 94
4.2.1 Alapszoftver 95
4.2.2 Alkalmazásszoftver 101
4.3 Számítógépes hálózatok 102
4.3.1 Internet 109
4.4 Adatbázisok 113
4.4.1 Relációs adatbázisok 117
4.4.1.1 XML, JSON 121
4.4.2 Elosztott adatbázisok 124
4.4.3 In-memory adatbázisok – valós idejű adatfeldolgozás 126
4.4.4 NoSQL adatbázisok 127
4.4.5 Egyéb adatbázisok 128
5 ÜZLETI FOLYAMATOK, VÁLLALATI INFORMÁCIÓS RENDSZEREK ÉS ALRENDSZEREK, JELENTŐSÉGÜK A GAZDASÁGI TERÜLETEN 130
5.1 Dekompozíció / kompozíció elve 132
5.2 A folyamatirányítás 133
5.3 PDCA ciklus – folyamatjavító módszer 136
5.4 Vállalati információs rendszerek és alrendszerek 139
5.4.1 Az információs rendszerek fajtái 140
6 TERVEZÉS, VÁLLALATI INFORMÁCIÓS RENDSZER ÉPÍTÉSE ÉS FEJLESZTÉSE, ALKALMAZÁSA 148
6.1 Cloud Computing 151
6.2 Alkalmazások lehetséges megoldásai 153
6.3 Az alkalmazások fontos jellemzői 157
6.3.1 Üzleti jellemzők 158
6.3.2 Funkcionalitást leíró jellemzők 160
6.3.3 Szolgáltatásokkal kapcsolatos jellemzők 161
6.3.4 Egyéb jellemzők 162
6.4 Információs rendszerfejlesztést támogató szoftverek 165
6.4.1 Computer Aided Systems Engineering 165
6.4.2 A rendszerüzemeltetés irányítása 168
6.5 DevOps (Development Operations) 169
6.6 Az emberek és a siker 172

6.7 A vállalati információs rendszer bevezetésének hatékonysága 174
7 VÁLLALATI INFORMÁCIÓS RENDSZEREK ÉLETCIKLUSA, KARBANTARTÁSA 178
7.1 Vízesés-, V- és agilis módszerek 180
7.2 Karbantartás 183
7.2.1 Vészhelyzeti tervek 185
7.3 Veszélyek feltárása 188
Ebben a fejezetben használt néhány kifejezés magyarázata 190
8 INTEGRÁLT VÁLLALATIRÁNYÍTÁSI RENDSZER 194
8.1 ERP rendszerek 194
8.1.1 ERP rendszerek testre szabása 198
8.1.2 Analitikai alkalmazások és üzleti intelligencia integrációja ..201
8.1.3 ERP rendszerek kategorizálása 202
8.1.4 ERP rendszerek jellemzői 203
8.2 SCM rendszerek 205
8.2.1 Az SCM gyakorlati alkalmazása 207
8.3 CRM rendszerek 207
8.3.1 A CRM gyakorlati alkalmazása 210
8.3.2 CRM rendszerek kategorizálása 210
8.4 Egyéb tranzakciós rendszerek 211
9 INFORMÁCIÓS RENDSZEREK TÁMOGATÁSA A DÖNTÉSHOZATALBAN 212
9.1 Döntéshozatalok az egyes szinteken 214
9.2 Problémamegoldás szisztematikus megközelítése 216
9.3 Modellek használata és tanulás a döntéstámogató rendszereken belül 219
9.3.1 Minták és tudás tárolása 225
9.4 Döntéstámogató rendszerek 226
9.5 Alapvető különbség a vezetői információs rendszer és döntéstámogató rendszer között 233
BEFEJEZÉS 236
FELHASZNÁLT IRODALOM 238
TÁRGYMUTATÓ 246
MELLÉKLETEK 251
Mérési egységek és az információs technológia 252
Hálózattípusok, hálózati modell és hálózati eszközök 257
Alkalmazások az integrált, automatizált termelési és raktározási tevékenységek irányítására 264
A visszacsatolt rendszerek 266
UTF-8 kódtábla (részlet) és Unicode karakterek 268

Rövidítések jegyzéke (I-II. kötet)

A könyvben gyakran használt angol rövidítések jegyzéke

3NF *Third Normal Form (harmadik normálforma)*
ACID *Atomicity, Consistency, Isolation, Durability (atomicitás, konzisztencia, izoláció, tartósság)*
AES *Advanced Encryption Standard (fejlett titkosítási szabvány)*
AGI *Artificial General Intelligence (általános mesterséges intelligencia)*
AI *Artificial Intelligence (mesterséges intelligencia)*
AIIM *Association for Information and Image Management (Információ- és Képkezelési Szövetség)*
AIS *Academic Information System (akadémiai információs rendszer)*
ANSI *American National Standards Institute (Amerikai Nemzeti Szabványügyi Intézet)*
API *Application Programming Interface (alkalmazásprogramozási felület)*
APS *Advanced Planning and Scheduling (fejlett tervezés és ütemezés)*
AS1 *Applicability Statement 1 (alkalmazhatósági nyilatkozat 1)*
AS2 *Applicability Statement 2 (alkalmazhatósági nyilatkozat 2)*
AS3 *Applicability Statement 3 (alkalmazhatósági nyilatkozat 3)*
AS4 *Applicability Statement 4 (alkalmazhatósági nyilatkozat 4)*
ASP *Application Service Providing (alkalmazásszolgáltatás)*
ASW *Application Software (alkalmazásszoftver)*
ATA *Alternative Text Attribute (alternatív szöveg attribútum)*
AWS *Amazon Web Services (Amazon felhőszolgáltatások)*
B2B *Business to Business (vállalatok közötti kereskedelem)*
B2B2C *Business to Business to Consumer (vállalat–vállalat–fogyasztó közötti kapcsolat)*
B2C *Business to Customer (vállalat és végfelhasználó közötti kereskedelem)*
B2G *Business to Government (vállalat és kormányzat közötti kapcsolat)*

B2E Business to Employee (vállalat és alkalmazott közötti kapcsolat)
BAM Business Activity Monitoring (üzleti tevékenységfigyelés)
BI Business Intelligence (üzleti intelligencia)
BIS Business information system (vállalati információs rendszer)
BIOS Basic Input/Output System (alapvető bemeneti/kimeneti rendszer)
BMIS Business Model for Information Security (üzleti modell az információbiztonsághoz)
BOM Bill of Materials (darabjegyzék)
BPEL Business Process Execution Language (üzleti folyamat-végrehajtási nyelv)
BPM Business Process Management (üzleti folyamatmenedzsment)
BPMN Business Process Model and Notation (üzleti folyamatok modellezése és jelölésrendszere)
BPMS Business Process Management System (üzleti folyamatmenedzsment-rendszer)
BPR Business Process Reengineering (üzleti folyamatok újratervezése)
BPSS Business Process Specification Schema (üzleti folyamatspecifikációs séma)
BR Barcode Recognition (vonalkód-felismerés)
BYOD Bring Your Own Device (hozd a saját eszközöd)
BYOC Bring Your Own Cloud (hozd a saját felhőszolgáltatásodat)
C2B Consumer to Business (fogyasztó és vállalat közötti kapcsolat)
C2C Consumer to Consumer (fogyasztók közötti kapcsolat)
C3 Chrysler Comprehensive Compensation System (Chrysler átfogó kompenzációs rendszere)
CA Certificate Authority (hitelesítésszolgáltató)
CaaS Communication as a Service (kommunikáció mint szolgáltatás)
CAD Computer-Aided Design (számítógéppel segített tervezés)
CAM Computer-Aided Manufacturing (számítógéppel segített gyártás)
CAP Consistency, Availability, Partition Tolerance (konzisztencia, rendelkezésre állás, partíciótűrés)
CAP Computer-Aided Planning (számítógéppel segített tervezés)
CASE Computer Aided Systems Engineering (számítógéppel támogatott rendszertervezés)
CASE Computer Aided Software Engineering (számítógéppel támogatott szoftvertervezés)
CD Continuous Deployment (folyamatos telepítés)
CD Continuous Delivery (folyamatos leszállítás)
CEO Chief Executive Officer (vezérigazgató)
CFO Chief Financial Officer (pénzügyi igazgató)

CHRO Chief Human Resources Officer (humánerőforrás-igazgató)
CI Competitive Intelligence (versenytársfigyelés/versenyhírszerzés)
CI Continuous Integration (folyamatos integráció)
CI Customer Intelligence (ügyfélintelligencia)
CIM Computer Integrated Manufacturing (számítógéppel integrált gyártás)
CIO Chief Information Officer (információs vezető / informatikai igazgató)
CLS Cumulative Layout Shift (halmozott elrendezés-eltolódás)
CMS Content Management System (tartalomkezelő rendszer)
CNC Computer Numerical Control (számítógépes számjegyvezérlés)
COBIT Control Objectives for Information and Related Technologies (ellenőrzési célok az információ és a kapcsolódó technológiák számára)
COO Chief Operations Officer (működési igazgató)
COTS Commercial Off-The-Shelf (kereskedelmi forgalomban kapható, dobozos szoftver)
CPA Collaboration Protocol Agreement (együttműködési protokoll-megállapodás)
CPC Central Product Classification (központi termékosztályozás)
CPU Central Processing Unit (központi feldolgozóegység, processzor)
CRC Cyclic Redundancy Check (ciklikus redundanciaellenőrzés)
CRM Customer Relationship Management (ügyfélkapcsolat-kezelés)
CSO Chief Sales Officer (kereskedelmi igazgató)
CSS Cascading Style Sheets (lépcsőzetes stíluslapok)
CSS Customer Service and Support (ügyfélszolgálat és támogatás)
CV Customer Value (vásárlói érték / ügyfélérték)
DaaS Desktop as a Service (asztali környezet mint szolgáltatás)
DAM Digital Asset Management (digitális eszközkezelés / multimédiakezelő rendszer)
DAX Data Analysis Expressions (adat-elemzési kifejezések nyelve)
DBMS Database Management System (adatbázis-kezelő rendszer)
DCL Data Control Language (adatvezérlő nyelv)
DDL Data Definition Language (adatdefiníciós nyelv)
DDoS Distributed Denial of Service (elosztott szolgáltatásmegtagadás)
DevOps Development Operations (fejlesztési és üzemeltetési együttműködés)
DIKW Data–Information–Knowledge–Wisdom (adat–információ–tudás–bölcsesség)
DHCP Dynamic Host Configuration Protocol (dinamikus állomáskonfigurációs protokoll)

DL Deep Learning (mélytanulás)
DLP Data Loss Prevention (adatvesztés-megelőzés)
DMAIC
Define-Measure-Analyze-Improve-Control (meghatározás–mérés–elemzés–fejlesztés–szabályozás)
DML Data Manipulation Language (adatkezelő nyelv)
DMS Document Management System (dokumentumkezelő rendszer)
DMZ Demilitarized Zone (demilitarizált zóna)
DNS Domain Name System (tartománynévrendszer)
DOLAP
Desktop OLAP (asztali OLAP)
DoS Denial of Service (szolgáltatásmegtagadás)
DSA Data Staging Area (adat-előkészítő terület)
DSS Decision Support System (döntéstámogató rendszer)
DV Domain Validated (tartományhitelesített)
DW Dataware (adatok)
DWH Data Warehouse (adattárház)
E-E-A-T
Experience, Expertise, Authoritativeness, Trustworthiness (tapasztalat, szakértelem, hitelesség, megbízhatóság)
EAI Enterprise Application Integration (vállalati alkalmazásintegráció)
ebXML Electronic Business XML (elektronikus üzleti XML)
ECC Elliptic Curve Cryptography (elliptikus görbéken alapuló kriptográfia)
ECM Enterprise Content Management (vállalati tartalomkezelő rendszer)
EDA Event-Driven Architecture (eseményvezérelt architektúra)
EDI Electronic Data Interchange (elektronikus adatcsere)
EDP Electronic Data Processing (elektronikus adatfeldolgozó rendszer)
EDW Enterprise Data Warehouse (vállalati adattárház)
EIS Executive Information System (felsővezetői információs rendszer)
ELT Extract, Load, and Transform (kivonás, betöltés és átalakítás)
EMA Enterprise Marketing Automation (vállalati marketingautomatizálás)
ENIAC Electronic Numerical Integrator and Computer (elektronikus numerikus integrátor és számítógép)
ERD Entity Relationship Diagram (entitás-kapcsolat diagram)
ERP Enterprise Resource Planning (vállalati erőforrás-tervezés)
ES Expert System (szakértői rendszer)

ESN Enterprise Social Network (vállalati közösségi hálózat)
ETL Extract, Transform, and Load (kivonás, átalakítás és betöltés)
EULA End-User License Agreement (végfelhasználói licencszerződés)
EV Extended Validation (kiterjesztett hitelesítés)
exFAT Extended File Allocation Table (kiterjesztett fájltáblás rendszer)
FAT32 File Allocation Table 32 (32 bites fájltáblás rendszer)
FDI Foreign Direct Investment (közvetlen külföldi befektetés)
FID First Input Delay (első bevitel késleltetése)
FLS Fuzzy Logic System (fuzzy logikai rendszer)
FMCG Fast-Moving Consumer Goods (gyorsan forgó fogyasztási cikkek)
FNC Federal Networking Council (Szövetségi Hálózati Tanács)
FTP File Transfer Protocol (fájlátviteli protokoll)
FTPS FTP Secure (biztonságos FTP)
G2B Government to Business (kormányzat és vállalat közötti kapcsolat)
G2C Government to Citizen (kormányzat és állampolgár közötti kapcsolat)
G2G Government to Government (kormányzatok közötti kapcsolat)
GA Genetic Algorithms (genetikus algoritmusok)
GAN Global Area Network (globális kiterjedésű számítógépes hálózat)
GB Gigabyte (gigabájt)
GDPR General Data Protection Regulation (Általános Adatvédelmi Rendelet)
GDSS Group Decision Support System (csoportos döntéstámogató rendszer)
GHz Gigahertz (gigahertz)
GPU Graphics Processing Unit (grafikus feldolgozóegység, grafikus kártya)
GraphQL Graph Query Language (gráf alapú lekérdezési nyelv)
GUI Graphical User Interface (grafikus felhasználói felület)
HCM Human Capital Management (emberi tőke menedzsment)
HDD Hard Disk Drive (merevlemez-meghajtó)
HIS Hospital Information System (kórházi információs rendszer)
HOLAP Hybrid OLAP (hibrid OLAP)
HR Human Resources (emberi erőforrások)
HRMS Human Resource Management System (humánerőforrás-menedzsment rendszer)
HTAP Hybrid Transactional / Analytical Processing (hibrid tranzakciós/ analitikus feldolgozás)
HTML HyperText Markup Language (hipertext jelölőnyelv)

HTTP HyperText Transfer Protocol (hiperszövegátviteli protokoll)
HTTPS HyperText Transfer Protocol Secure (biztonságos hiperszövegátviteli protokoll)
HW Hardware (hardver)
IaaS Infrastructure as a Service (infrastruktúra mint szolgáltatás)
IAM Identity and Access Management (személyazonosság- és hozzáférés-kezelés)
IASW Individual Application Software (egyedi alkalmazási szoftver)
IBM International Business Machines (Nemzetközi Üzleti Gépgyártó Vállalat)
ICMP Internet Control Message Protocol (internetvezérlő üzenetprotokoll)
ICR Intelligent Character Recognition (intelligens karakterfelismerés)
IDC International Data Corporation (Nemzetközi Informatikai Piackutató Vállalat)
IDE Integrated Development Environment (integrált fejlesztőkörnyezet)
IE Inference Engine (következtető motor)
IEC International Electrotechnical Commission (Nemzetközi Elektrotechnikai Bizottság)
IEEE Institute of Electrical and Electronics Engineers (Villamos- és Elektronikai Mérnökök Intézete)
IFRS International Financial Reporting Standards (Nemzetközi Pénzügyi Beszámolási Standardok)
IMAP Internet Message Access Protocol (internetes üzenet-hozzáférési protokoll)
IMAPS IMAP Secure (biztonságos IMAP)
IMS Inventory Management System (készletkezelő rendszer)
IP Internet Protocol (internetprotokoll)
IRR Internal Rate of Return (belső megtérülési ráta)
IS Information System (információs rendszer)
ISACA Information Systems Audit and Control Association (Információs Rendszerek Ellenőrzési és Irányítási Egyesülete)
ISO International Organization for Standardization (Nemzetközi Szabványügyi Szervezet)
IT Information Technology (információs technológia)
ITGI IT Governance Institute (IT irányítási intézet)
ITIL Information Technology Infrastructure Library (információs technológiai infrastruktúra-könyvtár)
ITSM Information Technology Service Management (információs technológiai szolgáltatásmenedzsment)

ITU International Telecommunication Union (Nemzetközi Távközlési Egyesület)
IoT Internet of Things (dolgok/tárgyak internete)
JDBC Java Database Connectivity (Java adatbázis-kapcsolati felület)
JIT Just-in-Time (éppen időben)
JSON JavaScript Object Notation (JavaScript objektumjelölés)
KB Knowledge Base (tudásbázis)
KM Knowledge Management (tudásmenedzsment)
KMS Knowledge Management System (tudásmenedzsment-rendszer)
KPI Key Performance Indicator (kulcs teljesítménymutató)
LAN Local Area Network (helyi hálózat)
LCP Largest Contentful Paint (legnagyobb tartalmi festés)
LMS Learning Management System (tanulásmenedzsment-rendszer)
LPA Least Privilege Access (legkisebb jogosultság elve szerinti hozzáférés)
M2M Machine to Machine (gép-gép közötti kommunikáció)
MAC Media Access Control (médiah hozzáférés-vezérlés)
MAN Metropolitan Area Network (városi hálózat)
MAP Manufacturing Automation Protocol (gyártásautomatizálási protokoll)
MDDB Multidimensional Database (többdimenziós adatbázis)
MDN Message Disposition Notification (üzenetkézbesítési értesítés)
MES Manufacturing Execution System (gyártásirányítási rendszer)
MIME Multipurpose Internet Mail Extensions (többcélú internetes levelezési kiterjesztések)
MIS Management Information System (vezetői információs rendszer)
MIT Massachusetts Institute of Technology (Massachusettsi Műszaki Egyetem)
ML Machine Learning (gépi tanulás)
MOLAP Multidimensional OLAP (többdimenziós OLAP)
MPP Massively Parallel Processing (tömeges párhuzamos feldolgozás)
MR Model Repository (modell-adattár / modell-repozitórium)
MRP Material Requirements Planning (anyagszükséglet-tervezési rendszer)
MRP II Manufacturing Resource Planning (gyártási erőforrás-tervezés)
MS Messaging Service (üzenetküldő szolgáltatás)
MSA Microservices Architecture (mikroszolgáltatás-architektúra)
MX Mail eXchanger (levélcserélő)
NIC Network Interface Card (hálózati csatolókártya)
NIST National Institute of Standards and Technology (amerikai Nemzeti Szabványügyi és Technológiai Intézet)

NLP Natural Language Processing (természetes nyelvfeldolgozás)
NN Neural Networks (neurális hálózatok)
NPV Net Present Value (nettó jelenérték)
NTFS New Technology File System (új technológiájú fájlrendszer)
OCR Optical Character Recognition (optikai karakterfelismerés)
OData Open Data Protocol (nyílt adatszolgáltatási protokoll)
ODBC Open Database Connectivity (nyílt adatbázis-kapcsolati szabvány)
ODS Operational Data Store (működési adatbázis / operatív adattár)
OECD Organisation for Economic Co-operation and Development (Gazdasági Együttműködési és Fejlesztési Szervezet)
OEE Overall Equipment Effectiveness (berendezés-összhatékonyság)
OLAP Online Analytical Processing (online analitikus feldolgozás)
OLRT On-Line Real-Time (valós idejű online feldolgozás)
OLTP Online Transaction Processing (online tranzakciófeldolgozás)
OMR Optical Mark Reading (optikai jelölésolvasás)
OS Operating System (operációs rendszer)
OSI Open Source Initiative (nyílt forráskódú kezdeményezés)
OSI Open Systems Interconnection (nyílt rendszerek összekapcsolása)
OSS Open Source Software (nyílt forráskódú szoftver)
OTP One-Time Password (egyszer használatos jelszó)
OV Organization Validated (szervezethitelesített)
OW Orgware (szervezeti tényező)
P2P Point-to-Point (pont–pont közötti)
PaaS Platform as a Service (platform mint szolgáltatás)
PB Petabyte (petabájt)
PC Personal Computer (személyi számítógép)
PCI Peripheral Component Interconnect (perifériakomponens-csatoló)
PCS Production Control System (termelésirányítási rendszer)
PDA Personal Digital Assistant (személyi digitális asszisztens)
PDCA Plan-Do-Check-Act (tervezés–végrehajtás–ellenőrzés–beavatkozás)
PDF Portable Document Format (hordozható dokumentumformátum)
PLM Product Lifecycle Management (termék-életciklus kezelés)
POP3 Post Office Protocol 3 (postahivatal protokoll 3)
POP3S POP3 Secure (biztonságos POP3)
PPP Point-to-Point Protocol (pont–pont protokoll)
PR Public Relations (közönségkapcsolatok)
PW Peopleware (emberi tényező)
QoS Quality of Service (szolgáltatásminőség)
QR Quick Response (gyors válasz)

RAD Rapid Application Development (gyors alkalmazásfejlesztés)
RAM Random Access Memory (véletlen elérésű memória)
RDBMS Relational Database Management System (relációs adatbázis-kezelő rendszer)
REST Representational State Transfer (reprezentációs állapotátvitel)
RM Record Management (nyilvántartás-kezelés)
ROI Return on Investment (megtérülés / befektetésarányos megtérülés)
ROLAP Relational OLAP (relációs OLAP)
ROM Read-Only Memory (csak olvasható memória)
RPA Robotic Process Automation (robotikus folyamatautomatizálás)
RPC Remote Procedure Call (távoli eljáráshívás)
RSA Rivest–Shamir–Adleman (Rivest–Shamir–Adleman-algoritmus)
RTOS Real-Time Operating System (valósidejű operációs rendszer)
RUP Rational Unified Process (racionális egységes folyamat)
SaaS Software as a Service (szoftver mint szolgáltatás)
SAP Systems, Applications and Products in Data Processing (rendszerek, alkalmazások és termékek az adatfeldolgozásban)
SCM Supply Chain Management (ellátási lánc menedzsment)
SCOR Supply Chain Operations Reference (ellátási lánc műveleti referenciamodell)
SE Staging Environment (tesztelési / előzetes éles környezet)
SEO Search Engine Optimization (keresőmotor-optimalizálás)
SFA Sales Force Automation (értékesítési folyamatok automatizálása)
SFTP Secure/SSH File Transfer Protocol (biztonságos/SSH fájlátviteli protokoll)
SGML Standard Generalized Markup Language (szabványos általános jelölőnyelv)
SI International System of Units (nemzetközi mértékegységrendszer)
SIS Strategic Information System (stratégiai információs rendszer)
SMTP Simple Mail Transfer Protocol (egyszerű levéltovábbítási protokoll)
SOA Service-Oriented Architecture (szolgáltatásorientált architektúra)
SOAP Simple Object Access Protocol (egyszerű objektum-hozzáférési protokoll)
SOHO Small Office / Home Office (kis iroda / otthoni iroda)
SPI Stateful Packet Inspection (állapottartó csomagellenőrzés)
SQL Structured Query Language (strukturált lekérdezőnyelv)
SRM Supplier Relationship Management (beszállítói kapcsolatok menedzsmentje)
SSD Solid State Drive (félvezető-alapú meghajtó, szilárdtest-meghajtó)
SSH Secure Shell (biztonságos parancsértelmező / biztonságos shell)

SSL Secure Sockets Layer (biztonságos foglalati réteg)
SSO Single Sign-On (egyszeri bejelentkezés)
SVC Service Value Chain (szolgáltatás-értéklánc)
SVS Service Value System (szolgáltatásérték-rendszer)
SW Software (szoftver)
TASW Type Application Software (típusalkalmazási szoftver)
TB Terabyte (terabájt)
TCP Transmission Control Protocol (átvitelvezérlő protokoll)
TCP/IP Transmission Control Protocol / Internet Protocol (átvitelvezérlő protokoll / internetprotokoll)
TDD Test-Driven Development (tesztvezérelt fejlesztés)
TLS Transport Layer Security (szállítási réteg biztonság)
TMS Transportation Management System (szállítmányozási / fuvarozási menedzsment rendszer)
TPS Transaction Processing System (tranzakciófeldolgozó rendszer)
UDDI Universal Description, Discovery, and Integration (univerzális leírás, keresés és integráció)
UDP User Datagram Protocol (felhasználói datagram protokoll)
UEBA User and Entity Behavior Analytics (felhasználói és entitásviselkedés-elemzés)
UEFI Unified Extensible Firmware Interface (egységes bővíthető firmware interfész)
UI User Interface (felhasználói felület)
UML Unified Modeling Language (egységes modellezési nyelv)
UPS Uninterruptible Power Supply (szünetmentes áramellátás)
USB Universal Serial Bus (univerzális soros busz)
UTP Unshielded Twisted Pair (árnyékolatlan csavart érpár)
UX User Experience (felhasználói élmény)
VAN Value-Added Network (értéknövelt hálózat)
VDW Virtual Data Warehouse (virtuális adattárház)
VM Virtual Machine (virtuális gép)
VMI Vendor Managed Inventory (szállító által kezelt készlet)
VoIP Voice over Internet Protocol (hangátvitel internetprotokoll felett)
VPN Virtual Private Network (virtuális magánhálózat)
W3C World Wide Web Consortium (Világháló Konzorcium)
WAN Wide Area Network (nagy kiterjedésű hálózat)
WCM Web Content Management (webtartalom-kezelés)
WCS Warehouse Control System (raktárirányítási rendszer)
WfMS Workflow Management System (munkafolyamat-kezelő rendszer)
WMS Warehouse Management System (raktárkezelő rendszer)
WSDL Web Services Description Language (webszolgáltatás-leíró nyelv)

WSS Web Services Security (webszolgáltatás-biztonság)
WWW World Wide Web (világháló)
XML eXtensible Markup Language (kiterjeszthető jelölőnyelv)
XP eXtreme Programming (extrém programozás)
ZB Zettabyte (zettabájt)
ZTS Zero Trust Security (nulla bizalmon alapuló biztonság)

A könyvben gyakran használt magyar rövidítések jegyzéke

ENSZ Egyesült Nemzetek Szervezete
IKT Információs és kommunikációs technológia
IR Információs rendszer
IT Információs technológia
MI Mesterséges intelligencia
TI Természetes intelligencia
VIR Vállalati információs rendszer

1 Bevezető, alapvető fogalmak meghatározása

1.1 Információk és adatok

Napjainkban az élet szinte minden területén jól láthatóan vagy éppen rejtetten jelen vannak a számítógépek és az őket működtető szoftverek. Laptop, PC, biztonsági rendszerek, gyártószalagok, autóink, okosTV-k, stb. Az információ tehát nélkülözhetetlen eleme az életünknek, pl. az orvostudományban, közlekedésben, kommunikációban, de mindennapi életünkben is. Az *informatio* (ami latin *Informare* szóból származik, és azt jelenti, hogy alakot adni, formázni) kifejezés először 1274-ben jelent meg és bizonyítékként használták a bűncselekmények feltárása során.

Norbert Wiener szerint az információ nem materiális jellegű (munkájában rámutatott arra, hogy az információ érzékelése szorosan összefügg a visszacsatolásokkal és vezérlőrendszerekkel, különösen az élő szervezetekben és a gépekben[1]).

Claude Shannon az információt valószínűségi alapon definiálta (1949). Meghatározása szerint, minél ritkább egy esemény bekövetkezése, annál nagyobb információ tartalommal bír, és annál értékesebb lehet a felhasználó számára. Az információ tehát a bizonytalanság csökkentésével áll összefüggésben. Tegyük fel, hogy az információ egy elementáris információkból álló halmaz (A_n), és minden egyes A_n eseménynek van egy hozzárendelt előfordulási valószínűsége (P_n). Például, egy hatoldalú dobókocka esetén az egyes kimenetelek azonos valószínűséggel következnek be:

$$P(A_1) = P(A_2) = \cdots = P(A_6) = 1/6$$

[1] Különösen a *Cybernetics: Or Control and Communication in the Animal and the Machine* (1948) című könyvében hangsúlyozta az információ fontosságát a visszacsatolási folyamatokban és a vezérlőrendszerekben. Wiener rámutatott, hogy az információ kulcsszerepet játszik a folyamatok szabályozásában és irányításában mind az élő szervezetekben, mind a gépekben.

Azonban a természetes nyelv szimbólumainak előfordulási valószínűsége eltérő. A beszélt vagy írott szövegekben egyes betűk vagy szavak gyakrabban fordulnak elő, míg mások ritkábbak. Ez azt jelenti, hogy $P_1 \neq P_2 \neq \ldots \neq P_n$. Ugyanakkor, minden lehetséges esemény együttes valószínűsége mindig összeadva 1-et ad ki: $P_1+P_2+\ldots+P_n = 1$, tehát 100%. Shannon információelméletében kulcsfontosságú fogalom az **entrópia**. Az entrópia az információ mennyiségének mértéke, és azt fejezi ki, hogy mennyi bizonytalanságot csökkenthet egy üzenet. Tegyük fel, hogy létezik egy rendszer, és benne egy független esemény zajlik le, amely a rendszer új állapotba való átmenetét eredményezi. Legyen a rendszernek *n* darab kölcsönösen kizáró állapota (S_i), és minden állapothoz rendeljünk egy előfordulási valószínűséget $P(S_i)$. Ekkor a rendszer entrópiáját Shannon a következőképpen határozta meg:

$$H(S) = -\sum_{i=1}^{n} P(S_i) \log_2 P(S_i)$$

Ez azt mutatja, hogy az entrópia függ az események valószínűségi eloszlásától. Az entrópia akkor maximális, ha az események egyforma valószínűséggel következnek be (például 50% – 50% esetén egy kétállapotú rendszerben). Az entrópia akkor minimális (azaz = 0), ha az egyik esemény 100%-os bizonyossággal következik be, míg a többi 0%-os valószínűséggel fordul elő. Ez azt jelenti, hogy ha egy esemény kimenetele előre ismert, akkor az nem hordoz jelentős információt. Ellenben, ha egy esemény kimenetele nagyon bizonytalan, akkor az információtartalma magasabb.
Shannon az információ mennyiségének mérésére kétféle egységet javasolt: **Bitek** (bináris logaritmus alapján – $\log_2$), vagy **Nat** (természetes logaritmus alapján – ln). Az informatikában a bit egységet használjuk, amely az információ mennyiségét fejezi ki. **Az informatikában a bit az információ mennyiségének alapegysége**. Egy bit egy bináris döntést (igen/nem, 0/1, stb.) fejez ki, vagyis két lehetséges állapot közötti különbséget rögzít: $\log_2 2 = 1\ bit$.

Adat és információ kifejezést a hétköznapi szóhasználatban gyakran keverjük, sokszor egymás szinonimájaként használjuk a két kifejezést. Pedig informatikai értelemben nagyon fontos különbséget tenni közöttük. **Adatnak** nevezünk olyan tényeket, fogalmakat vagy utasításokat, amelyek alkalmasak emberi vagy gépi feldolgozásra, értelmezésre, illetve kommunikációra – ISO 27001 szabvány. Robert Glazer az információt úgy definiálja, mint **olyan adat, amely elrendeződött vagy meghatározott szerkezetet kapott, azaz kontextusba került, és így meghatározott**

jelentést kapott. Az információ a bizonytalanságot csökkentő új ismeret, a vállalatok működését integráló folyamatok egyik összetevője. A vállalat működésének minden szintjén kulcsszerepe van az időben meghozott helyes döntéseknek, amelyekben a döntéshozó a saját megérzésén kívül leginkább az általa elérhető információkra támaszkodik. Az információ tehát a vállalati döntések „nyersanyaga", egy erőforrás, amely a hagyományosan erőforrásként elismert tényezők (munkaerő, állóeszközök, pénztőke) mellett meghatározó szerepet játszik a vállalat tevékenységében. Kapcsolatban lévő és megértett információk **tudássá** alakulnak. Tapasztalat vagy tanulmányok révén szerzett megértést képvisel, érthető és alkalmazható a problémamegoldásban vagy a döntéshozatalban. Információkat feloszthatjuk több, különböző szempont alapján. Gála Libor (2009) szerint ismerünk:

- **Technikai információkat**
  - Ide tartoznak a műszaki folyamatokkal, gépekkel, szoftverekkel és technológiákkal kapcsolatos információk. Például szoftverkonfigurációk, gyártósorok futási információi, gépteljesítmény-statisztikák.
    - Kezdeteknél fontos volt az adatátvitel – maximális üzenetszám átvitel minimális idő alatt és biztonságosan. Ez fontos volt a gépek irányítására.
- **Biológiai információkat**
  - Ezek az információk az élő szervezetekkel, azok működésével és biológiai folyamataival kapcsolatosak.
  - A modern informatikában a biológiai információ különösen fontos a bioinformatika, a genetika vagy az egészségügy területén, ahol nagy mennyiségű biológiai adatot (pl. DNA-szekvenálás) kezelnek.
    - Például a génszekvencia információk, az agyműködésre vonatkozó adatok vagy az információfeldolgozás az informatikai neurális hálózatokban.
- **Társadalmi információkat**
  - Ezek az információk az emberi társadalom szociális és kulturális aspektusaihoz kapcsolódnak.
    - Tudományos – pl. a szabadalmakba foglalt információk, a legújabb tudományos kutatások eredményeire vonatkozó információk.
    - Szociális – pl. a lakosság társadalmi státuszára vonatkozó információk, a szakképzett személyek számára vonatkozó információk, a lakhatás típusára (ház vagy lakás) vonatkozó információk.

- **Gazdaságtudományi** – pl. információk a piaci trendekről, gazdasági elemzések.
- Politikai – pl. információk a választási eredményekről, közvélemény-kutatásokról.
- Művészeti – pl. a társadalmi szokásokra vagy hagyományokra vonatkozó információk.

A gazdaságtudományban az információ háromféle értelmezésével találkozhatunk: **általános értelemben** az adatok átadására gondolunk, **társadalmi értelemben** gondolunk a médián keresztül gyakorolt hatásra (pl. TV, rádió, sajtó, reklámok stb.), **kibernetikai értelemben** pedig az irányításra és visszacsatolásra – beleértve a gazdaságtudományi rendszereket is.

Minden információra három különböző nézetet alkalmazhatunk:

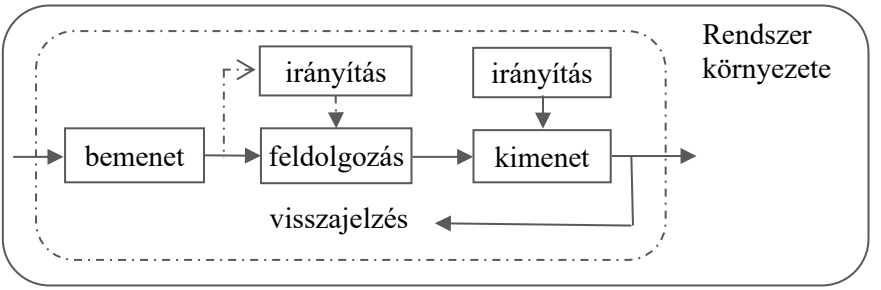
- **Szintaktikai nézet** – üzenetek belső rendszerei, jelek, karakterek… – nyelv.
- **Szemantikai nézet** – üzenet megértése.
- **Pragmatikai nézet** – információ praktikus felhasználhatósága.

A pragmatikus állítások közül kiemelkedően provokatívnak tekinthető **Peter Drucker** véleménye, aki azt állítja, hogy az Információ az egyetlen jelentőségteljes forrás a vállalkozásban (Drucker ezt a kifejezést a modern társadalmak átalakulásának hangsúlyozására használta, ahol az információ különösen a tudásalapú munkavállalókkal és a döntéshozatallal összefüggésben jelentős hatást gyakorol a gazdaságra). **Marcel Pourbaix**, belga informatikai és vállalatirányítási szakértő szerint egy vállalat jó irányítása a jövőjének irányítását jelenti, és a jövő irányítása az információkkal való munkát jelenti. Információ értékével tisztába voltak a jól ismert nagyvállalkozók is mint **Tomáš Baťa**, **Henry Ford** vagy **Gerard Philips** is, akik szerint az információ ugyanolyan fontos forrás, mint a munka és munkaerő vagy a tőke. A Ford például futószalagokat vezetett be, és adatokat használt fel a gyártási folyamatok optimalizálására, ami jelentősen csökkentette a költségeket és növelte a hatékonyságot. Tomáš Baťa a termelés és az üzleti folyamatok rendszerszintű irányítását támogatta vállalatában olyan szervezeti újításokkal, mint a decentralizált irányítási rendszerek[2] és az alkalmazottak előtérbe helyezése, ezáltal hatékony belső kommunikációs és ellenőrzési rendszert hozva létre.

[2] A döntéshozatali jogkörök nem a szervezet vezetésének legmagasabb szintjén összpontosulnak, hanem az alsóbb vezetési szintek, illetve a különböző osztályok, részlegek vagy ágazatok között oszlanak meg.

Az említettek alapján elmondhatjuk, hogy az információ olyan üzenet/adat a megtörtént jelenségről, amely a felhasználó tudatlanságának fokát csökkenti erről a jelenségről. Az adatok például lehetnek számok, de csak akkor válnak információvá, ha értelmezzük őket (pl. hőmérséklet, árbevétel, stb.). (Gála, L.; pour, J.; Šedivá, Z. 2009) (Drucker, P. F. 1998) (Rosický, A. 2010) (Basl, J.; Blažíček, R. 2012) (Bencsík, A.; Tóbiás Kosár, S. 2016)

1.2 Információs, és vállalati információs rendszer fogalma



1.1. ábra: Rendszer klasszikus értelmezése
Forrás: (Gála, L.; pour, J.; Šedivá, Z. 2009)

A rendszert úgy definiáljuk, mint egy bizonyos céllal létrehozott, nem üres halmazt, amely elemekből és köztük lévő kapcsolatokból áll. Ezek tulajdonságai határozzák meg az egység tulajdonságát. Az elem alatt egy elemi meghatározott egységet értünk, amelyet nem lehet tovább szétválasztani (például egy műszaki komponens, egy folyamat vagy egy személy, a rendszer típusától függően). Így definiált rendszert a következők alapján azonosítjuk be: **rendszer célja**, rendszer **összetétele** (struktúrája – **struktúra** alatt azt értjük, ahogyan a rendszer elemei és azok kapcsolatai összeállnak, elrendeződnek és felépülnek), **elemek tulajdonságai**, rendszer **környezete**[3] (azon elemek meghatározása, amelyek már nem részei a rendszernek, de amelyek tulajdonságai és a rendszer ezen környezeti elemekkel való kapcsolatai jelentősen befolyásolják a rendszer viselkedését, pl. vevő, szállító, de politikai vagy szociokulturális tényezők is) és rendszer **részletei** (néha a rendszer egészének vizsgálata túl bonyolult,

[3] A vállalat az outputjaival más szereplőkkel - konkurensekkel - versenyez. Nemcsak a kimeneti oldalon, hanem a bemeneti oldalon is versenyez, azaz a beszállítókért is versenyez.

és a rendszert kisebb, viszonylag elkülönült és zárt egységekre kell bontani a rendszeren belül). Az attribútumok egy adott időpontban érvényes értékei alakítják a rendszer **állapotát**. A rendszer **viselkedése** akcióval, reakcióval és a rendszer válaszával (általában a külső ingerekre) reprezentálható. Informatikában az ilyen rendszereket információs rendszereknek nevezzük. Tehát az információs rendszert definiálhatnánk, **mint elrendezett halmaz, melyben a komponensek az összegyűjtés, feldolgozás, átvitel és információkódolás érdekében együttműködnek**. A vállalati információs rendszerek a döntéshozatalt és a folyamatok optimalizálását támogató adatkezelést és adatelosztást is biztosítják. Általánosságban tehát emberekből, megfelelő eszközökből és módszerekből áll, amelyek három alapvető összetevőbe csoportosíthatók:

- **Bemenet**: magában foglalja azokat az elemeket, amelyek lehetővé teszik az információk és egyéb bemenetek rögzítését, illetve a bemenetek összekapcsolását (érzékelők, űrlapok vagy kézi adatbevitel).
- **Feldolgozás**: magában foglalja azokat az elemeket, amelyek biztosítják a bemenetek átalakítását a kívánt kimenetté (adatelemzés, transzformáció, számítások vagy algoritmusok).
- **Kimenet:** magában foglalja azokat az elemeket, amelyek képesek az információkat és egyéb kimeneteket a címzett, azaz a felhasználó számára átadni (a kimenetek lehetnek jelentések, grafikonok vagy más rendszerek számára készített adatok megfelelő formában).

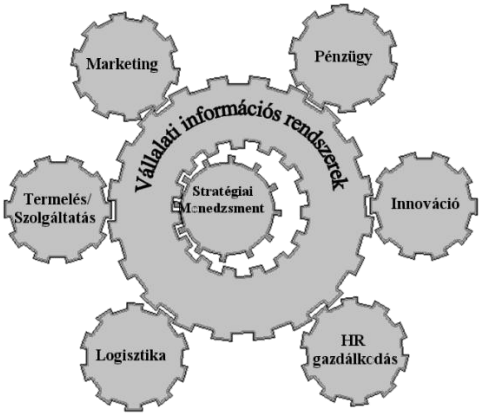
A leírtakat az 1.1. ábra szemlélteti. Az időbeli válasz tekintetében a rendszerek statikus és dinamikus rendszerekre oszthatók. A **statikus** rendszer olyan rendszer, amelynek pillanatnyi állapota a pillanatnyi bemenetektől vagy külső ingerektől függ, és nem függ a korábbi állapotoktól. A **dinamikus** rendszer olyan rendszer, amelynek pillanatnyi állapota függ a korábbi állapotoktól és a pillanatnyi bemenő jelektől vagy külső ingerektől, és amelyet tehetetlenség és válaszkésleltetés egyaránt jellemez. A rendszer viselkedésének vagy kimenetének kiszámíthatósága szerint megkülönböztetünk **determinisztikus** és **sztochasztikus** rendszereket. A determinisztikus rendszer olyan rendszer, amelynek viselkedését az állapota, a bemeneti értékek vagy az ingerek egyértelműen meghatározzák. A sztochasztikus rendszer olyan rendszer, amelyben sztochasztikus (véletlenszerű) folyamatok játszódnak le, például az elemei között, ami a rendszer sztochasztikus viselkedését okozza. Az általános rendszerelmélet és a struktúra komplexitása alapján a rendszerek komplexitásuk szerint **monostrukturális** és **multistrukturális**

rendszerekre oszthatók. A monostrukturális rendszer viszonylag egyszerű rendszer. Ebben a rendszerben sem nagy komplexitás, sem több rész közötti összetett kölcsönhatás nincs jelen. Általában kevés egymással összekapcsolt részt tartalmaz. A multistrukturális rendszer egy összetettebb rendszer, amely több struktúrából vagy alrendszerből áll, amelyek egymással kölcsönhatásban vannak. Az egyes alrendszerek különböző funkciókat láthatnak el, ezért az egész rendszer összetettebb. Egy vállalati környezetben ilyen például egy ERP (Enterprise Resource Planning) rendszer. A multistrukturális rendszerek több, egymással összekapcsolt részt tartalmaznak, amelyek növelik az általános összetettségüket. Attól függően, hogy a rendszer valamely eleme kölcsönhatásba lép-e a környezettel vagy sem, **nyílt** vagy **zárt** rendszerekről beszélünk. A környezettel kölcsönhatásban lévő elemet **határelemnek** nevezzük, és ezen elemek halmaza alkotja a **rendszer határát**. Az ilyen rendszert általában **irányítás**[4] és **visszajelzés** egészíti ki. Ezek a mechanizmusok lehetővé teszik a rendszer számára, hogy az eredmények vagy a környezet változásai alapján alkalmazkodjon és optimalizáljon. Az irányítás magában foglalhatja például a feldolgozás beállítását szabványok szerint, a szabványoktól való eltérés mértékének mérését és az eltérés minimalizálására irányuló intézkedés elindítását. A visszajelzés például egy olyan mechanizmust jelent, amellyel a kimenet értékelése alapján érvényesítjük a feldolgozás jövőbeli inputjait. Befolyásolhatja magát a feldolgozást vagy annak irányítását is. A visszacsatolási mechanizmus az alapja egy döntéstámogató rendszernek, ahol a kimenet (azaz a döntés) korrigálja a következő jövőbeli döntést. Egy komponens egy vagy több elemből áll össze[5]. Az információs rendszer elemei között az emberek (az információ használói) és az információs források kulcsszerepet játszanak, azonban az eszközök, szoftverek, adatok és szervezeti szabályok elengedhetetlenek a rendszer teljes működéséhez. A **vállalati információs rendszer** olyan integrált szoftverrendszer, amely támogatja és kezeli a szervezetek kulcsfontosságú üzleti folyamatait és műveleteit. A rendszer az információk gyűjtésére, feldolgozására, tárolására és a különböző részlegek közötti elosztására szolgál, lehetővé téve a hatékonyabb üzleti irányítást. Vállalati információs rendszer komponensei a következők:

[4] Az angol nyelv többféle kifejezést használ az „irányítás” kifejezésre. A „control” általában a műszaki rendszerek irányításával kapcsolatban használatos. A szervezetek esetében gyakrabban használják a „management” kifejezést. A „governance” az irányítás egy speciális típusa vállalaton belül.

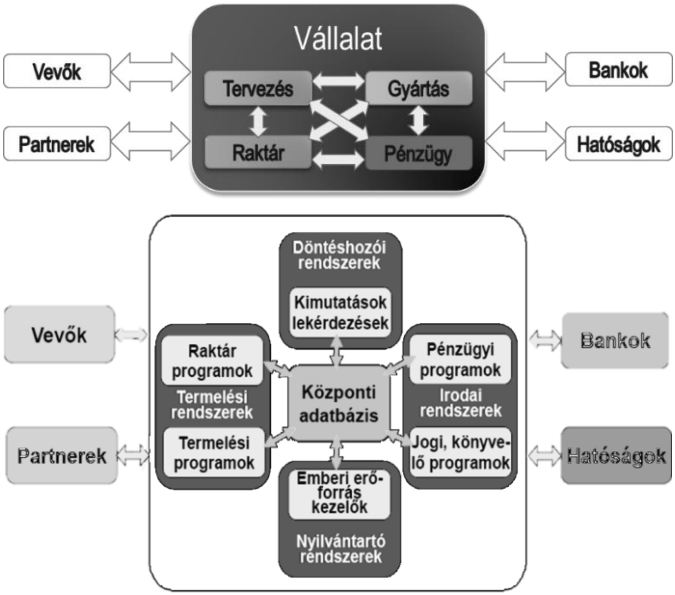
[5] Pl. a hardver, mint komponens, olyan elemekből épül fel, mint a laptop, tablet, hálózat, hálózati eszközök vagy mobiltelefon stb. Lásd a vállalati információs rendszer komponenseit.

- **HW (hardware**) – Szerverek, PC-k, nyomtatók, kommunikációs eszközök, hálózati eszközök, stb.



1.2. ábra: Vállalati információs rendszer szerepe Kotálik Gábor szerint (2003)
Forrás: https://slideplayer.hu/slide/2997068/

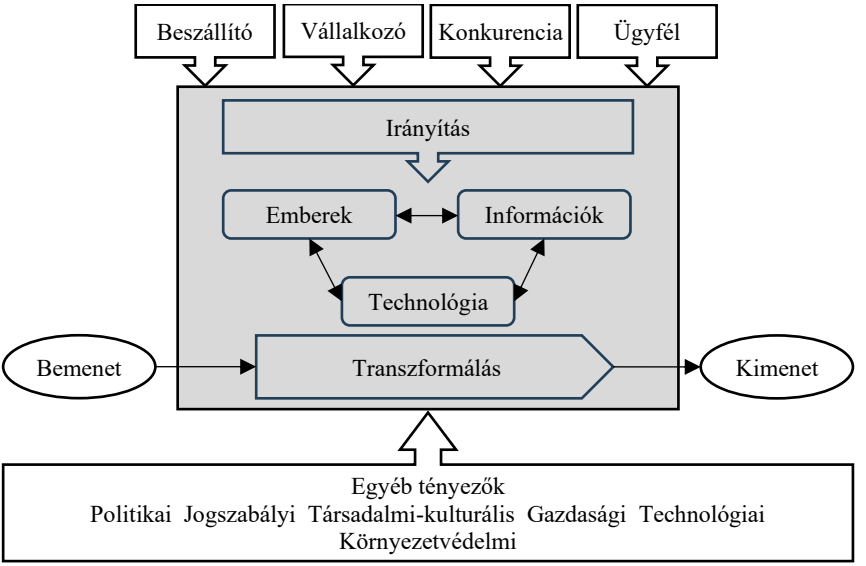
- **SW (software)** – Operációs rendszerek, adatbázis kezelő rendszerek, alkalmazások, specializált programok, stb.
- **DW (dataware)** – Adatbázisok, adattárházak, ETL (Extract, Transform, Load) eszközök, stb.
- **PW (peopleware)** – felhasználók, informatikai szakemberek, vezetők, alkalmazottak, stb.
- **OW (orgware) –** vállalati irányelvek, előírások, szabványok, folyamatok, stb.



1.3. ábra: Fent: A vállalaton belülii kapcsolat ábrázolása. Lent: A vállalati információs rendszer lehetséges összetétele és információkezelés központi adatbázis segítségével
Forrás: https://slideplayer.hu/slide/11194702/, https://slideplayer.hu/slide/11140712/

Nézzünk néhány példát a komponensek közötti kapcsolatokra:

- **Szoftver és hardver:** A vállalati információs rendszer (szoftver) szervereken (hardveren) fut, és adattárolást használ a rendelések feldolgozásához.
- **Adatok és szoftverek:** Az ügyféladatokat adatbázisrendszerben tárolják, és a CRM (Customer Relationship Management) alkalmazásban használják fel az ügyfelek interakcióinak feldolgozására.
- **Emberek és szoftverek:** A vezetők jelentéskészítő eszközöket (szoftvert) használnak a pénzügyi adatok elemzésére és az üzleti stratégiával kapcsolatos döntések meghozatalára.
- **Orgware és humán erőforrás:** Belső szabályozás szabályozza a munkavállalók érzékeny információkhoz való hozzáférését, biztosítva azok védelmét a megállapított szabályok szerint.
- **és egyéb kapcsolatok.**



1.4. ábra: Vállalat, mint rendszer
Forrás: (Gála, L; Pour, J; Šedivá, Z. 2015)

A vállalati információs rendszerek segítenek a döntéshozatali folyamatok javításában, a költségek optimalizálásában és a hatékonyság növelésében. Egy vállalati információs rendszer olyan területeket fedhet le, mint például:

- gyártásirányítás (a termelési folyamatok tervezése és ellenőrzése),
- számvitel és pénzügyi irányítás,

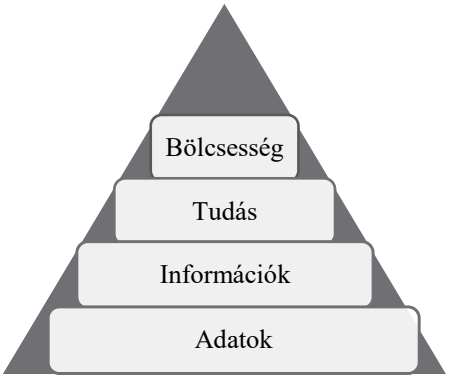
- humánerőforrás-gazdálkodás (bérszámfejtési rendszerek, személyzeti menedzsment) ellátási lánc és logisztika,
- ügyfélkapcsolat-kezelés (CRM).

A legismertebb vállalati információs rendszerek közé tartozik a SAP, az Oracle, a Microsoft Dynamics vagy az Odoo. Ezek a rendszerek a megoldások széles skáláját kínálják, az egyszerű moduloktól a komplex integrált rendszerekig, amelyek a vállalkozás minden aspektusát támogatják. (Gála, L.; pour, J.; Šedivá, Z. 2009, 2015) (Drucker, P. F. 1998) (Rosický, A. 2010) (Basl, J.; Blažíček, R. 2012)

1.3 A DIKW (Data–Information–Knowledge–Wisdom) modell

„Hol van a bölcsesség?
Elveszett a tudásban.
Hol van a tudás?
Elveszett az információban.
Hol van az információ?
Elveszett az adatokban.”

(Anonymus)



1.5. ábra: Az adat-információ-tudás-bölcsesség (DIKW) hierarchia - mint piramis

A DIKW (Data–Information–Knowledge–Wisdom) modell az adatok, információ, tudás és bölcsesség közötti hierar-chiáról beszél. Ez a hi-erarchia azon az elgondoláson alapul, hogy a nyers adatok információvá válnak, ha rendszerezik vagy kontextusba helyezik őket, és ezáltal értelmet és értéket nyernek a felhasználó számára. A DIKW koncepció szerint az információt strukturált adatként fogjuk fel, amely a befogadó számára kontextussal és így jelentéssel bír. Ezt a koncepciót a vezetők és az információmenedzsment szakemberei arra használják, hogy leírják, hogyan jön létre az információ, és milyen értékkel bír egy szervezeti kontextusban. Minden szint több értéket, összefüggést és jelentést ad, ami jobb döntésekhez vezet. Ez a modell leírja, hogyan alakulnak a nyers adatok információvá, majd tudássá és végül bölcsességgé. Ezt a hozzáállást és az információk átalakulását **információs**

hierarchiának is nevezik. (Ackoff, R. L. 1999) (Baskarada, S.; Koronios, A. 2013) (Bontis, N. 2002) (Bencsík, A.; Tóbiás Kosár, S. 2016) Nézzünk egy példát egy fiktív online ruhaboltról:

- **Adatok** (*Data*): nyersek, feldolgozatlanok, amelyeknek nincs konkrét jelentésük vagy kontextusuk.
    - *A vállalat nagy mennyiségű adatot gyűjt a web áruházából:*
        - *oldallátogatási idők,*
        - *mely termékeket nézték meg,*
        - *az egyes termékekre történő kattintások száma,*
        - *a vásárlások dátuma és időpontja,*
        - *a vásárlók címei.*

    Ezek a nyers adatok azonban még nem nyújtanak közvetlen értéket, mert nem rendszerezettek.
- **Információ** (*Information*): strukturált vagy elemzett adatok, amelyek valamilyen jelentéssel bírnak.
    - *Az adatfeldolgozás után a ruhabolt információkat szerezhet, például:*
        - *a legkeresettebb termékek egy adott időpontban (például őszi ruházat októberben),*
        - *azok az időszakok, amikor a webhely a legnagyobb forgalmat bonyolítja (pl. esti órák),*
        - *mely termékek kerülnek leggyakrabban a kosárba, de nem vásárolják meg őket (potenciális érdeklődés, de nincs elég motiváció a vásárláshoz).*

    Ezen a szinten az adatokat strukturálják és elemzik, és érdemi információkat szolgáltatnak a marketingcsapat számára.
- **Tudás** (*Knowledge*): Megértett és egy adott helyzetre alkalmazott információ.
    - *Ezen információk alapján a vállalat tudást szerez, például:*
        - *A vásárlók bizonyos termékek iránt bizonyos időszakokban nagyobb érdeklődést mutatnak (pl. télikabátok novemberben).*
        - *A vásárlók bizonyos napokon (pl. hétvégén) hajlandóak többet költeni.*

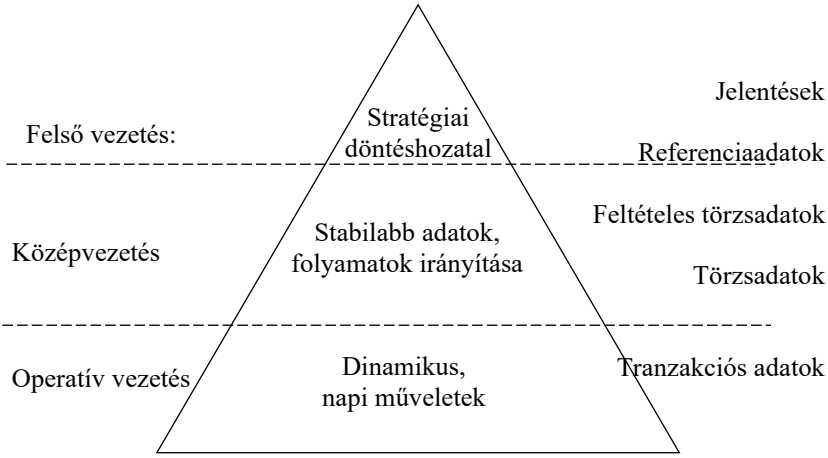
    A vállalat most már tudja, hogyan célozhatja meg hatékonyabban (reklámmal és kedvezménnyel) a vásárlókat. Például kedvezményeket kínálhat olyan termékekre, amelyeket gyakran néznek meg, de nem vásárolnak, vagy marketing-kampányokat folytathat a csúcsidőszakokban.

- **Bölcsesség** (*Wisdom*): Stratégiailag alkalmazott tudás a hosszú távú célok eléréséhez vagy összetett problémák megoldásához.
    - *A cég a tudás alapján hoz stratégiai döntéseket:*
        - *Úgy dönt, hogy a kosarat vásárlás nélkül elhagyó látogatókat úgy motiválja, hogy személyre szabott e-maileket küld további kedvezményekkel vagy ajánlatokkal.*
        - *Előrejelző analitikát használ a jövőbeli trendek és a vásárlói magatartás előrejelzésére.*

    Optimalizálja a készletszinteket a várható vásárlói igények alapján, minimalizálva a túlkínálatot és a készlethiányt.

1.4 Adatok a vállalati információs rendszer hierarchiáján belül

Ebben a részben tovább osztjuk az adatokat kategóriákra, hierarchikus kapcsolatot alakítunk ki a kategóriák között, és rámutatunk arra, hogy az egyik kategória adatait egy másik kategória adataival együtt használjuk fel. Az 1.5. ábra szemlélteti az egyes adatkategóriákat és azok hierarchiáját.



1.5. ábra: Adatkategóriák hierarchia piramisa

- A legalacsonyabb szinten a **tranzakciós adatok** vannak, amelyek a legszámosabbak és legdinamikusabbak. Ezek az adatok a napi műveleteket és tevékenységeket rögzítik. Ezek az adatok a

szervezetben végrehajtott műveleteket vagy tranzakciókat képviselik. Olyan eseményeket rögzítenek, mint a megrendelések, eladások, vásárlások, készletmozgások és kifizetések. Ezek dinamikusabbak, mint a törzsadatok, mivel folyamatosan változnak és frissülnek. Az egyes műveletek részletei képezik az alapját minden magasabb szintnek. A tranzakciós adatok nagyon időérzékenyek, mivel gyakran valós időben rögzítik az eseményeket, ami fontos a feldolgozási sebesség szempontjából.

- A **törzsadatok** magasabb szintűek, mint a tranzakciós adatok. Ezek a szervezet objektumaira (ügyfelek, termékek, alkalmazottak, beszállítók stb.) vonatkozó alapvető információk. Ezek az adatok stabilabbak, kevésbé változnak, és referenciapontként szolgálnak a tranzakciók és műveletek számára. Kulcsfontosságúak a szervezet működéséhez, mivel lehetővé teszik a tranzakciók pontos és hatékony feldolgozását. A törzsadatok tágabb fogalmat alkotnak, és az egész szervezet számára kulcsfontosságúak. Ezeket folyamatosan használják a különböző műveletekben. A jó törzsadatkezelés elengedhetetlen a hatékony üzleti folyamatokhoz.
- A következők a **feltételes törzsadatok**. Ezek az adatok hasonlóak a törzsadatokhoz, de felhasználásuk meghatározott feltételekhez vagy szabályokhoz köthető. Ezek lehetnek például olyan egyedi adatok, amelyek csak egy bizonyos időszak alatt vagy bizonyos feltételek mellett érvényesek, például olyan kedvezmények, amelyek bizonyos időszakokban bizonyos ügyfelekre érvényesek, vagy olyan kedvezmények, amelyek csak bizonyos termékkategóriákra vagy ügyfelekre vonatkoznak. Bár ezek a feltételek szerint folyamatosan frissülnek, még mindig kevésbé dinamikusak, mint a tranzakciós adatok.
- A hierarchikus piramis csúcsa alatt találhatók a legfontosabb **referenciaadatok**. Ezek az adatok a legfontosabbak, és az egész szervezet számára az „igazság aranyforrásaként” szolgálnak. Ezek tartalmazzák a legstabilabb és legjelentősebb adatokat, amelyekre minden fontos döntéshozatali folyamat épül. Ezek konzisztenciája és pontossága elengedhetetlen az egész szervezet működéséhez. Alapvető, változatlan vagy kevéssé változó adatokról van szó. Ide tartoznak például az ügyfelekre, termékekre, beszállítókra és alkalmazottakra vonatkozó információk. Ezeket az adatokat különböző rendszerekben és folyamatokban használják. Ki kell emelnünk a szabványosítás és az adatintegritás fontosságát az egész rendszerben, lehetővé téve a szervezet számára az adatok konzisztens feldolgozását.

- A piramis csúcsán az alacsonyabb szintű adatokból származó **jelentések és elemzések** állnak. Ezeknek a jelentéseknek a célja, hogy betekintést és információt nyújtsanak a döntéshozatalhoz és a stratégiai tervezéshez. A törzs- és tranzakciós adatokat kombinálhatják, hogy információkat generáljanak az üzleti teljesítményről, a folyamatok hatékonyságáról, a pénzügyi eredményekről stb. Ezek olyan kimenetek, amelyek különböző forrásokból, köztük törzs- és tranzakciós adatokból gyűjtött adatokat tartalmaznak, és betekintést vagy elemzést nyújtanak az adatok felett. Ezek számos célra generálhatók, például pénzügyi jelentések, értékesítési teljesítményjelentések vagy logisztikai jelentések készítése. A jelentések automatikusan vagy kézi kérés alapján is készíthetők, és ezek a jelentések képezik az üzleti intelligencia és a stratégiai döntéshozatal alapját.

A tranzakciós adatok példájaként említhetjük a vevői megrendelések kiszolgálásához kapcsolódó adatokat. Maga a rögzített megrendelés tranzakciós adat, amely olyan adatok gyűjteménye, mint a szállítási dátum, a megrendelt tételek száma, stb. A törzsadatok kategóriája a vállalkozás átalakítási folyamatához kapcsolódó adatokból áll. Ezek egyrészt termék- és ügyféljellemzők, másrészt anyag-, alkatrész- és beszállítói jellemzők (például termékadatok). A feltételes törzsadatok az adatok egy kategóriája, amelyeket feltételeknek nevezhetünk. Ezeket különböző helyzetekben alkalmazzák, és segítségükkel alakítják az üzleti szabályokat. Erre példa a termékár képzési feltételei az ügyféljellemzők függvényében. A kulcsfontosságú referenciaadatok kategóriája a vállalat erőforrásaira és képességeire vonatkozó adatokból áll. Az adatok meghatározzák a vállalkozást és annak elemeit, beleértve a korlátozó jellemzőket is. Ilyenek például a vállalkozás emberi erőforrásaira vonatkozó adatok, ahol az adatok a személy jellemzőit (tudás, készségek, tapasztalat, stb.), a hatalmi hierarchiát meghatározó szervezeti struktúrát, az átalakítási folyamathoz kapcsolódó tevékenységek végzésének helyét (osztályok, egységek, stb.) jellemzik. Egy másik példa ebben a kategóriában: lehetnek a gépek és berendezések tulajdonságait jellemző adatok (pl. konfigurációjuk, kapacitásuk, elhelyezkedésük, stb.) vagy a térbeli adatok (pl. a raktárak földrajzi elhelyezkedése és kapacitása, stb.). A jelentések kategória olyan adatokat képvisel, amelyek állapot- vagy jelentési adatoknak tekinthetők. Ezeket a tranzakciós adatokból képezik az összes alsóbb adatkategóriából származó adatok felhasználásával. Ezek lehetnek egyszerű összefoglalók egy adott időszakra vonatkozóan (pl. készletszintek, kiállított számlák összefoglalása, stb.), melyeket (kimeneti)

jelentéseknek nevezzük. Emellett fejlett analitikus feldolgozás kimenetei is lehetnek.

Az adatok hierarchia szerinti felosztása, tükrözi a vállalat irányítási struktúráját, és azt, hogy a különböző vezetési szintek hogyan használják a különböző típusú adatokat saját igényeikre. Ezt a fogalmat a szakirodalomban gyakran **információs piramisnak** vagy az **információs igények hierarchiájának** nevezik. Ez lehetővé teszi, hogy a különböző vezetési szintek a döntéshozatalukhoz leginkább megfelelő adatokhoz férjenek hozzá. Az **operatív irányítás** a napi folyamatokra és műveletekre összpontosít. Ezek a folyamatok dinamikus tranzakciós adatokat igényelnek, amelyek tényleges eseményeket rögzítenek, például eladásokat, vásárlásokat, készleteket, stb. A **középvezetés** felelős a szervezeten belüli folyamatok koordinálásáért és ellenőrzéséért, ezért gyakran használ törzsadatokat és feltételes törzsadatokat. A **felső vezetés** az egész szervezet hosszú távú döntéshozatalára és stratégiájára összpontosít. Ehhez stabil és pontos referenciaadatokra van szüksége, amelyek az egész szervezet számára „igazságként" szolgálnak.

Könnyű belátni, hogy mindez összefügg a DIKW modellel. (Gála, L.; pour, J.; Šedivá, Z. 2009) (Drucker, P. F. 1998) (Rosický, A. 2010) (Basl, J.; Blažíček, R. 2012)

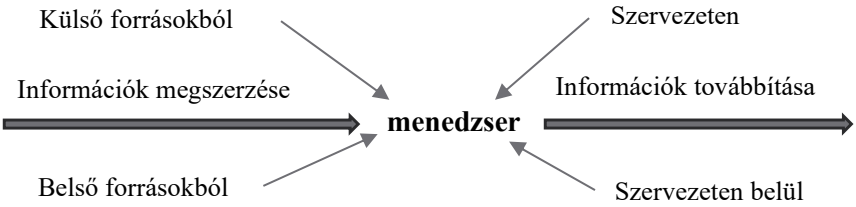
2 Vállalati információs rendszerek feladatai

A vállalati információs rendszer fő feladata annak biztosítása, hogy a döntéshozatalhoz szükséges releváns, időszerű és pontos információk megfelelő időben és formában rendelkezésre álljanak. A szakirodalomban és az információs rendszerek elméletében gyakran megkülönböztetik az információ különböző funkcióit aszerint, hogy milyen szerepet játszanak a döntéshozatali folyamatban. Általánosságban, ezek a funkciók a következőképpen oszthatók fel:

- **Diagnosztikai funkció** - tájékoztatást nyújt a vizsgált jelenség aktuális állapotáról, lehetővé téve a tényeken alapuló értékelést.
    - *Pl. A len magasságnövekedésének mérése három hét alatt: ez a szakasz tényszerű adatokat gyűjt a növény aktuális növekedési állapotáról.*
- **Magyarázó funkció** - elemzi a jelenségeket, azok összefüggéseit és okait, és ismereteket nyújt, amelyek segítségével a jelenségek mélyebb megértése válik lehetővé, tapasztalatokat teremt.
    - *Pl. különböző éghajlati viszonyok közötti növekedési ütemek kimutatása: ez a funkció az okokat és összefüggéseket elemzi, például azt, hogy a különböző éghajlati viszonyok (napsütéses napok vs. esős napok) hogyan befolyásolják a len növekedését.*
- **A prediktív funkció** - a diagnosztikai és magyarázó funkció eredményeit felhasználva előrejelzéseket nyújt a jövőbeli fejleményekről, hozzájárulva a stratégiai tervezéshez.
    - *Pl. A len növénymagasság előrejelzése az előttünk álló hétre: a múltbeli adatok (diagnosztikai funkció) és az elemzés (magyarázó funkció) alapján megbecsülhető a növény jövőbeli növekedése.*
- **A döntési funkció** - az előző funkciók eredményei és a rendelkezésre álló adatok alapján segíti a különböző megoldási

lehetőségek kidolgozását és az optimális megoldás kiválasztását, támogatva a megalapozott döntés-hozatalt.

- *Pl. Döntés a len betakarításáról a textilfeldolgozásra alkalmas érettségi állapotban: ez a végső döntés az előző lépések (diagnosztika, magyarázat, előrejelzés) alapján.*



2.1. ábra: Információk irányítása a vezetői gyakorlatban
Forrás: (Kučera,M.; Fiľa, M.; Látečková, A.; Stuchlý, P. 2017)

A fentiekre példa a mobiltelefonok tervezésében és gyártásában korábban világelső finn Nokia vállalat megszűnése. Bár sok éven át piacvezető volt Európában és a világon, és példátlan fejlődést hozott a mobilkommunikáció világában (pl. a világ legsikeresebb beépített antennával rendelkező telefonja, a Nokia 3310), a mintegy 20 éves piaci dominancia az elmúlt években nagyon gyorsan elveszett. Bár a Microsoft technológiai bravúrja azt eredményezte, hogy a Nokia "nem tudott lépést tartani a korral, és kikerült a fogyasztói igények és hangulatok főáramából". Ugyanakkor az Apple (iPhone) és a Samsung új megoldásainak erőteljes megjelenése magával a Microsofttal való összeolvadással magával hozta a Nokia bukását. A márka teljesen eltűnt a piacról, a jelenlegi visszatérési kísérlet éppen csak elkezdődött a 3310 feltámasztásával (2000 és 2017).

Az Apple ezzel szemben megértette a mobil megoldások jelenlegi trendjeit és azt a tényt, hogy a mobiltelefon már nem csak egy telefon, hanem egy mobil iroda is, amely a csatlakoztathatóság révén lehetővé teszi a hatékony munkavégzést. Emellett a mobiltelefon a szórakoztatás eszközévé is vált, lehetővé teszi a szabadidő lerövidítését a közösségi hálózatokon keresztül történő kommunikációval. Mindez az információs rendszerek különböző eszközeivel való aktív kapcsolat révén lehetséges. Ugyanakkor a mobiltelefon státuszszimbólum is a tulajdonosa számára.

Ez az eredmény, valamint a mobiltelefon-piac vezető szereplőinek gyors változása részben a verseny következménye, részben pedig a Nokia rossz döntéseinek eredménye. A vállalat olyan információk és források alapján hozta meg döntéseit, amelyek nem tükrözték pontosan a piac és a vásárlók – különösen a fiatalok és az üzleti vezetők új

> *generációjának – igényeit és irányultságát, ellentétben az Apple stratégiájával.*

A belső és külső információk szisztematikus felhasználása a mai viharos időkben a sikeres vállalatirányítás alapjává válik. Az olyan információk, amelyek nem látják el a fenti funkciók egyikét sem, valószínűleg nem fogják betölteni céljukat a vezetők döntéshozatali munkájában. A menedzserek különböző információforrásokat használnak, amelyek függnek a vállalkozás méretétől, az üzleti területtől, a piaci környezettől, a rendelkezésre álló adat- és információforrásoktól stb. Az egyes vállalkozások információs igényei ezekhez és hasonló tényezőkhöz kapcsolódnak. (Kučera,M.; Fiľa, M.; Látečková, A.; Stuchlý, P. 2017)

2.1 A vállalati információs rendszerek tulajdonságai

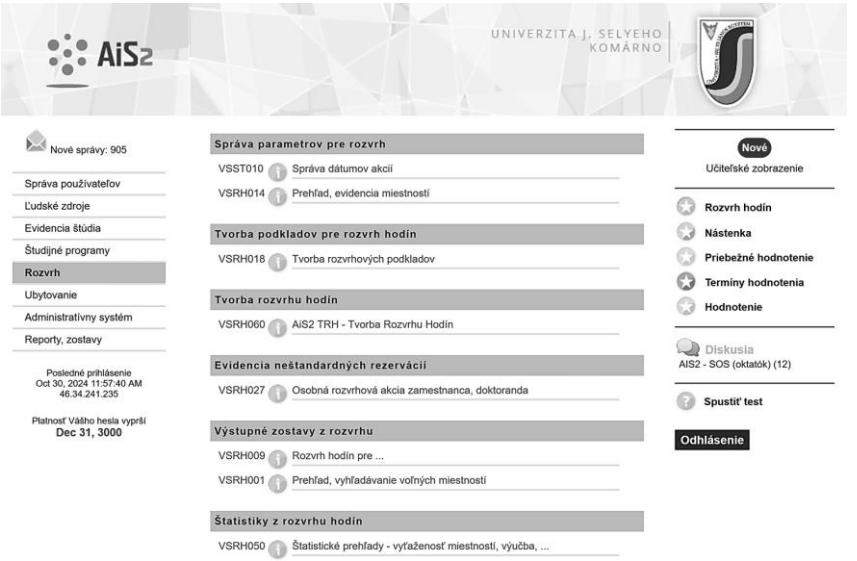
A vállalati információs rendszerek számos tulajdonsággal rendelkeznek. A következő felosztásban kiemeljük a legfontosabbakat.

- A különféle bemeneti adatokból olyan információkat szereznek, amelyeket más módon nagyon nehéz lenne megszerezni.
    - *Pl. az adatbányászat,*
    - *eszközök az online vásárló ügyfelek információinak összekapcsolására a vevők mozgásával és vásárlásaival kapcsolatos információkkal a hagyományos üzletekben,*
    - *vagy értékesítési előrejelzés az időjárás alapján.*
- A munkakör besorolása szerint különböző nézeteket biztosítanak az adatokról.
    - *Más adatokat kell lekérnie az információs rendszerből a vállalkozás tulajdonosának, másokat a kereskedőnek, másokat a raktárosnak.*
        - *A raktáros szempontjából szükséges, hogy lássa az egyes termékek egyes darabjait egy adott raktárban, élelmiszerek esetében érdekelheti a szavatossági időre vonatkozó információ, stb.*
        - *A kereskedő számára elegendő lehet a százra kerekített darabszám, amivel a vállalat jelenleg rendelkezik.*
        - *A tulajdonos pedig elégedett lesz, ha egy egyszerű táblázatban globális áttekintést lát.*
- Lehetővé teszik a fejlett jelentéskészítést.

- *Mesterséges intelligenciát használnak, grafikonokat és adatmegjelenítéseket terveznek. Így az információkról teljesen új perspektívát kaphatunk. Például, hogy egy adott termékkel együtt mely kiegészítő termékek fogytak a legjobban bizonyos vásárlói csoportoknál.*

- Automatizálják a munkát és a folyamatokat.
  - *Pl. az ügyfél megrendelésének beérkezése után a rendszer maga is meg tudja rendelni az árukat a szállítótól, kiállítja a számlát és elküldi az elégedettségi kérdőívet.*
- Moduláris felépítésűek.
  - *Pl. egyes webáruházaknak rendelési rendszerre lehet szükségük, más kiskereskedőknek automatikus számlázásra és importálásra a könyvelési rendszerükbe, más vállalkozásoknak pedig egy futárcéggel való kapcsolatra, hogy ne kelljen kézzel kiírniuk a csomagcímkéket. Az információs rendszerek moduláris felépítésével ez nem jelent problémát.*
  - *Másik példa lehet a Selye János Egyetem AIS[6] rendszere.*
- Támogatják a mobileszközökön végzett munkát is.
- Könnyű felhasználókezelést és speciális hozzáférési jogosultságbeállításokat tesznek lehetővé.
- Rugalmasak a módosítások és bővítések szempontjából.
  - *Pl., ha szükségünk van arra, hogy a rendszert más vállalati eszközökkel összekapcsoljuk, vagy valami különlegeset fejlesszünk, akkor az információs rendszerek készítői általában olyan dokumentációt bocsátanak rendelkezésünkre, amely ezt lehetővé teszi. Ez nemcsak dokumentum, hanem például videó formájában is megjelenhet. Ezek különböző oktatóanyagok, telepítési útmutatók vagy demók*.
- Kifinomult felhasználói felülettel rendelkeznek.
  - *Minél összetettebb a vállalati információs rendszer, annál fontosabb ez a funkció. Segítenie kell a felhasználóknak a munkájuk megkönnyítésében, a gyors munkavégzésben és a gyors tájékozódásban.*

[6] Az Akadémiai Információs Rendszer (AIS) egy átfogó információs rendszer, amelyet elsősorban a felsőoktatás mindhárom szintjének irányítására, támogatására, valamint a felsőoktatási intézményekben folyó tudományos és kutatási tevékenység támogatására terveztek. Az AIS fejlesztését a kassai P. J. Šafárik Egyetem Természettudományi Karának Alkalmazott Informatikai Központja biztosítja.



2.2. ábra: Az AIS rendszer moduljai (a legfrissebb 2-es verzió)
Forrás: Saját szerkesztés

2.2 A vállalati információs rendszerek felosztása

Mai modern világunkra tekintve könnyen beláthatjuk, hogy számos szempont szerint tudnánk a vállalati információs rendszereket felosztani. Következő felsorolásban kiemeljük az irodalmakban is gyakran említett felosztásokat, melyek jó áttekintést nyújtanak.

2.2.1 A vállalati információs rendszerek cél szerinti felosztása

- **Általános rendszerek (pl. ERP)**
  - Széles körű funkciókat fednek le, és modulárisak.
    - Az **ERP** (*Enterprise Resource Planning*) rendszerek olyan általános üzleti rendszerek, amelyek az üzleti folyamatok széles körét fedik le különböző területeken, például a pénzügy, a gyártás, az emberi erőforrások, az ellátás és a logisztika területén.

  - Ezeket a rendszereket úgy tervezték, hogy rugalmasak legyenek, ami azt jelenti, hogy különböző típusú vállalkozásokhoz és iparágakhoz igazíthatók.
  - Jellemzők:
    - Számos iparágban alkalmazható.
    - Moduláris és a vállalat igényeihez igazítható.
    - Az üzleti funkciók széles körével foglalkozik.
  - Példák: SAP, Oracle, Microsoft Dynamics.
- **Specifikus rendszerek (pl. CRM, SCM)**
  - Konkrét vállalati területekre koncentrálnak.
  - A specifikus rendszerek az üzleti tevékenység meghatározott területeire vagy funkcióira összpontosítanak.
  - Például:
    - A **CRM** (*Customer Relationship Management*) egy specifikus ügyfélkapcsolat-kezelő rendszer, amely az értékesítésre, a marketingre és az ügyfélszolgálatra összpontosít.
    - Az **SCM** (*Supply Chain Management*) rendszerek az ellátási lánc irányítására összpontosítanak.
  - Jellemzők:
    - Szűken egy vagy néhány üzleti folyamatra összpontosítanak.
    - Segítenek bizonyos területek, például az ügyfélkapcsolatok, a logisztika vagy az ellátás racionalizálásában.
  - Példák: Salesforce (CRM), Zoho CRM, Oracle SCM Cloud.
- **Specializált rendszerek (pl. ingatlanügynökségek, kórházak, iskolák számára)**
  - Adott iparágak számára vagy specifikus típusú vállalkozásra készültek.
  - Ezek a rendszerek az adott szektor egyedi igényeit szolgálják ki, így olyan funkciókat tartalmaznak, amelyek az iparág specifikus követelményeihez igazodnak.
  - Például:
    - **Kórházi információs rendszerek (HIS)** az egészségügyi ellátásra és a betegkezelésre összpontosítanak.

  - Oktatási rendszerek iskolák és egyetemek számára, amelyek a hallgatók, kurzusok és tanulmányi eredmények kezelését támogatják.
- Jellemzők:
  - Az adott iparág számára fejlesztett rendszerek.
  - Speciális eszközöket és megoldásokat nyújtanak az iparág egyedi igényeihez igazodva.
- Példák: AppFolio (ingatlanügynökségek), Epic Systems (kórházak), Moodle.

- **Hibrid rendszerek**
  - Több terület funkcióit kombinálják.
  - Például:
    - olyan rendszerek, amelyek egyesítik az ERP és CRM funkciókat egyetlen megoldásban, vagy
    - amelyek széles funkcionális spektrumot fednek le, miközben iparág-specifikus modulokat is tartalmaznak.
  - Példák: NetSuite – ERP és CRM funkciókat nyújt egy rendszerben.
- **Felhő alapú és SaaS rendszerek**
  - Rugalmas, interneten keresztül elérhető megoldások.
  - A felhő alapú rendszerek vagy SaaS (Software as a Service) rendszerek modern megközelítést jelentenek, ahol a vállalati információs rendszereket szolgáltatásként kínálják az interneten keresztül.
  - A vállalatok saját szerverek és infrastruktúra nélkül is hozzáférhetnek ezekhez a rendszerekhez, csökkentve az IT menedzsment költségeit.
  - Jellemzők:
    - Rugalmasság és alacsonyabb üzemeltetési költségek.
    - Hozzáférés webböngészőn keresztül.
  - Példák: Salesforce, Workday, Odoo.
- **Nyílt forráskódú rendszerek**
  - A nyílt forráskódú rendszerek gyakran rugalmasak, olcsók, és lehetővé teszik a vállalat igényeihez való testre szabást. Ideális azoknak a cégeknek, amelyek testreszabott rendszert szeretnének, de nem kívánnak kereskedelmi megoldásokba beruházni.
    - A „nyílt forráskódú” (angolul „open source”) azt jelenti, hogy egy szoftver forráskódja szabadon elérhető és bárki számára megtekinthető,

módosítható, illetve terjeszthető. Ilyenek például a Linux operációs rendszer, Apache webszerver, Mozilla Firefox böngésző vagy LibreOffice irodai programcsomag.

- Példák: Odoo, ERPNext, Dolibarr.

Az általános rendszerek a robusztusabb megoldások kategóriájára utalnak, amelyeket jellemzően közepes és nagyvállalatok használnak. Ezek az úgynevezett all-in-one rendszerek. Gyakran egyedi modulokból vagy alrendszerekből állnak. A moduláris rendszerek előnye, hogy csak azt aktiválhatja, amit ténylegesen használni fog. Az általános információs rendszerek általában mindent tartalmaznak az anyagbeszerzéstől kezdve a raktárkezelésen és a könyvelésen át az értékesítésig és a jelentéstételig. A legismertebb kategória az ERP (Enterprise Resource Planning) rendszerek. Egy vállalat számára fontos, hogy egy ilyen rendszer képes legyen több gyártó különböző eszközeit egyetlen egységbe cserélni. Ez időt, pénzt és a több eszköz kezelésével járó gondokat takarít meg. Ajánlott felhagyni a különböző, csak egy feladatot megoldó alkalmazások használatával, és elkezdeni az új rendszer teljes körű használatát. Ez az adatokkal kapcsolatos. E rendszerek különböző összetettebb feladatainak, mint például az értékesítési előrejelzés, az ügyfélmagatartás, az adatbányászat, csak nagy mennyiségű, jó minőségű adatokkal van értelme. Minél több adat áll a rendszer rendelkezésére, annál jobb.

> *Pl., ha egy családi vállalkozásról beszélünk, akkor lehet, hogy a vállalat vezetője nem bánja, ha Excelben állítják ki a számlákat és kézzel írják ki a nyugtákat. Ezzel ellentétben egy másik megoldás lehet egy olyan rendszer használata, ahol az első megrendeléstől számítva mindez automatikusan történik, ahol a számla és a nyugta kiállítása elvégezhető néhány egérkattintással.*

A specifikus rendszerek főként a nagy megoldások egyes alrendszereit foglalják magukban. A vállalkozók gyakran találkoznak két rövidítéssel - ERP és CRM. Az ERP (Enterprise Resource Planning) egy általános információs rendszer, mivel sokféle területet érint. Ezzel szemben a CRM (Customer Relationship Management) már egy igen specifikus rendszer. A CRM segítségével a vállalat egy helyen gyűjtheti és értékelheti az összes ügyfélkapcsolati tevékenységre vonatkozó információt. Ez magában foglalja a megkereséseket, árajánlatokat, e-maileket, telefonhívásokat, értékesítési találkozókat stb. A különböző specifikus eszközök gazdag funkcionalitást kínálnak, rengeteg termék közül lehet választani, és gyakran ingyenes verziókat is kínálnak. Különösen

ajánlhatók olyan kis kezdő vállalkozások számára, amelyek még csak most kezdik az információs rendszerekkel való ismerkedést. Rendet teremthetnek a vállalkozásban, segíthetnek az értékesítés és a hatékonyság növelésében.

A fenti eszközökön kívül ismerünk speciális információs rendszereket is. Ezek lehetnek általánosak és specifikusak is. Ebben az esetben egy adott ágazatban használható alkalmazásokra gondolunk. Ezek nem alkalmasak különböző üzleti szegmensekre, hanem szűken egy adott ágazatra specializálódtak. Kiváló példa erre a CENKROS 4 szoftver az építőipari költségvetés készítői számára vagy a Realsoft szoftver az ingatlanügynökök számára. Mindkettő iparágspecifikus, és minden szakember ismeri őket, mert megkönnyítik a munkájukat.

> *Képzeljük el, ha ugyanazt az ingatlanhirdetést kézzel kellene feltölteni 15 különböző ingatlanportálra, majd azokat időről időre frissíteni kellene. Hasonló a helyzet az építőiparban is, az építési költségvetések frissítése, az árak kiigazítása vagy a különbségek kitalálása órákig vagy akár néhány kattintásig tarthat.*

2.2.2 Az információs rendszerek felosztása méret és használt platform szerint

A vállalati információs rendszerek feloszthatók méret és platform szerint, amelyen működnek, ami hasznos lehet az üzleti méret vagy a technológiai követelmények szempontjából. Ismerünk:

- **Nagy rendszereket** → és **többplatformos rendszereket**
    - Ezek a rendszerek nagy szervezetek vagy nemzetközi vállalatok számára készültek. Fő jellemzőjük, hogy különböző platformokon (pl. különböző operációs rendszereken) képesek futni, és átfogó megoldásokat kínálnak.
    - Pl.: SAP R/3, Microsoft Dynamics AX, stb.
- **Közepes rendszereket → Unix, Windows Server, stb.**
    - A közepes méretű rendszerek olyan középvállalkozásokat céloznak meg, amelyeknek hatékonyan kell menedzselniük folyamataikat, de nem rendelkeznek olyan nagy infrastruktúrával, mint a nagyvállalatok. Ezek a rendszerek gyakran Unix vagy Windows szerver-platformokon futnak.
    - Pl.: Intentia Movex, JBA system 21, Helios, stb.
- **Kis rendszereket → MS Windows, iOS vagy Novell platform**

- A kis információs rendszereket olyan kisebb vállalkozások számára tervezték, amelyeknek alapvető funkciókra van szükségük az üzleti folyamatok kezeléséhez. Ezek a rendszerek gyakran olyan népszerű asztali platformokon futnak, mint például az MS Windows, sőt néhány iOS vagy Novell platformon is fut.
- Pl.: MRP, Omega, Pohoda, stb.

2.2.3 Az információs rendszerek felosztása hozzáférés szempontjából

Ez alatt a felosztás alatt főleg a helyi, hálózaton elérhető, felhő alapú és hibrid információs rendszerekre való felosztást értjük. Ezek a következők lehetnek:

- **Helyi információs rendszerek**
  - Az ilyen típusú információs rendszerek csak egy fizikai helyen működnek, általában egyetlen szervezeten belül egy szerveren vagy számítógépen.
  - Pl. Hagyományos könyvelő programok vagy adatbázis-rendszerek, amelyek internet- vagy hálózati kapcsolat nélkül telepíthetők a számítógépre.
- **Hálózaton elérhető - hozzáférés a szervezeten belül**
  - Az ilyen típusú információs rendszerek a szervezet belső hálózatán (pl. intranet) érhetők el a szervezeten belüli felhasználók számára.
  - Ez a fajta rendszer nem érhető el külső hálózatról, ami nagyobb vezérlést és biztonságot jelent.
- **Hálózaton elérhető - hozzáférés kívülről is**
  - Ez a fajta rendszer a hálózaton keresztül a szervezeten belülről, és külső forrásokból, például az internetről is elérhető.
  - A felhasználók távolról is elérhetik a rendszert, ami különösen fontos a több fiókkal rendelkező cégek vagy a távolról dolgozó alkalmazottak számára.
- **Cloud-based rendszerek**
  - Ezek a rendszerek felhőszolgáltatók külső szerverein működnek, és bárhonnan elérhetők az interneten keresztül. Ez a modell magas rendelkezésre állást, skálázhatóságot és

rugalmasságot biztosít. Tipikus példa erre a modern SaaS (Software as a Service) megoldások.
    - Pl. Google Workspace, Microsoft 365, Salesforce, stb.
- **Hibrid rendszerek**
    - Pl. olyan rendszerek, ahol az infrastruktúra egy része a felhőben működik, de a kulcs adatokat helyben tárolják a nagyobb biztonság érdekében.

Ez a felosztás jól tükrözi, hogy a szervezetek az igényeiknek és biztonsági követelményeiknek megfelelően hogyan választhatnak információs rendszert, figyelembe véve a hozzáférhetőséget és a skálázhatóságot.

2.2.4 Az információs rendszerek felosztása területi hatályát tekintve

Az információs rendszereket területi hatályuk szempontjából kettő nagy kategóriába sorolhatjuk:

- **Lokálisak**
    - individuálisak,
    - helyi – pl. Anyakönyv,
    - regionálisak.
- **Globálisak**
    - állami – pl. Szlovákia,
    - nemzetközi – IBM, Samsung, Az EU-n belüli schengeni információs rendszer személyek, áruk felkutatására, tiltott termékek és anyagok regisztrálására.

Számos információs rendszer, különösen a helyi és regionális önkormányzatokban, illetve az állam- és közigazgatásban, mostanra komplex, globális és nemzeti rendszerré alakul át, amely az ország egész területén hozzáférést biztosít az arra jogosult személyek számára. Szlovákiában erre egy jó példa a Központi Kormányzati Portál (Ústredný portál verejnej správy).

2.3 A vállalati információs rendszerek feladatai

A vállalati információs rendszereknek a feladatait két nagy kategóriába sorolhatjuk. Első kategóriába a kulcsfeladatok, a másikba a

támogató feladatok tartoznak. Ez a felosztás átláthatóvá teszi, hogy melyek a rendszerek kritikus feladatai a mindennapi működéshez (kulcsfeladatok), és melyek azok, amelyek segítik a hosszú távú növekedést és alkalmazkodást (támogató feladatok).

Kulcsfeladatok – a vállalkozás működéséhez elengedhetetlen feladatok (pl. automatizálás, folyamatintegráció):

- **Döntéstámogatás**
  - A vállalati információs rendszer releváns, naprakész és pontos információkat szolgáltat, amelyekre a vezetőknek szükségük van a megalapozott döntések meghozatalához. Segít elemezni a pénzügyi helyzetre, a termelési teljesítményre, a készletekre és más területekre vonatkozó adatokat.
- **Folyamatautomatizálás és optimalizálás**
  - Egyszerűsíti és automatizálja az olyan ismétlődő és időigényes tevékenységeket, mint a rendelésfeldolgozás, a számlázás, a bérszámfejtés vagy a készletgazdálkodás. Ez csökkenti a hibák számát és időt takarít meg.
- **Üzleti funkciók integrálása**
  - A vállalati információs rendszer egyetlen rendszerbe integrálja a különböző részlegeket és folyamatokat, ami lehetővé teszi a termelés, a logisztika, a pénzügy, az értékesítés és más területek közötti jobb koordinációt. A különböző részekből származó információk összekapcsolódnak és valós időben állnak rendelkezésre.
- **A kommunikáció és az együttműködés javítása**
  - A vállalati információs rendszer központosított adatainak köszönhetően a munkavállalók és a vezetők jobban tudnak kommunikálni és koordinálni az együttműködést a szervezeten belül. Mindenki hozzáférhet a releváns információkhoz a jogosultságainak megfelelően.
- **Erőforrás-gazdálkodás**
  - Segít hatékonyan kezelni a szervezet erőforrásait, például az anyagokat, a pénzügyeket, a humán erőforrásokat és a technológiát. Az erőforrás-tervezés és -gazdálkodás kulcsfontosságú a termelés és a szolgáltatások zökkenőmentes működésének eléréséhez.
- **Az adatok minőségének és rendelkezésre állásának javítása**
  - A vállalati információs rendszer biztosítja, hogy minden adatot egy helyen gyűjtsenek össze, csökkentve ezzel a

duplikációkat és az adathibákat. A rendszer lehetővé teszi a valós idejű hozzáférést a naprakész adatokhoz, ami jobb betekintést és gyorsabb döntéshozatalt eredményez.

- **A teljesítmény ellenőrzése és nyomon követése**
  - A vállalati információs rendszer egy szervezet teljesítményét, például a pénzügyi mutatókat, a termelési hatékonyságot vagy az üzleti célok elérését figyeli. Lehetővé teszi a vezetők számára, hogy nyomon kövessék a kitűzött célok megvalósulását és a folyamatok hatékonyságát.
- **Ügyfélszolgálati támogatás**
  - A vállalat információs rendszerébe integrált CRM (Customer Relationship Management) rendszerek lehetővé teszik az ügyfélkapcsolatok javítását, az ügyfélrendelések nyomon követését, a panaszok feldolgozását és a magas szintű kiszolgálás biztosítását.
- **A jogszabályoknak és szabványoknak való megfelelés biztosítása**
  - A vállalati információs rendszer segíti a vállalatokat a törvényi, számviteli előírásoknak és egyéb szabályozási követelményeknek való megfelelés fenntartásában, automatizálva a megfelelőségi folyamatokat különböző területeken, például az adózás, a bérszámfejtés és az adatkezelés terén.

Támogató feladatok – az egyedi követelmények és az üzleti fejlődés szempontjából fontos feladatok (pl. skálázhatóság, kiberbiztonság).

- **Skálázhatóság és testreszabhatóság**
  - A vállalati információs rendszer szerepe, hogy elég rugalmas és skálázható legyen ahhoz, hogy alkalmazkodni tudjon a vállalat növekedéséhez és a változó üzleti struktúrához. A rendszernek képesnek kell lennie arra, hogy jelentős beavatkozás nélkül bővítse a funkcionalitást, például új piacokra való belépés vagy egy másik vállalattal való egyesülés esetén.
- **Analitika és prediktív analitika**
  - A fejlett analitikával rendelkező rendszerek mélyebb betekintést nyújthatnak a trendekbe és előrejelzéseket tehetnek az üzlet különböző területein, például az értékesítés, a készlet vagy a vásárlói magatartás alakulására vonatkozóan. Ez a hagyományos

teljesítményfigyelés és -ellenőrzés kiterjesztése, ahol a hangsúly a jövőre és a lehetséges forgatókönyvekre helyeződik.

- **Adatbiztonság és kiberbiztonság**
  - Az adatvédelem és az információ biztonság támogatása a jogosulatlan hozzáféréssel vagy kiberfenyegetésekkel szemben fontos szerepe az internetszolgáltatónak, bár hagyományos értelemben nem tekinthető központi szerepkörnek. A rendszereket mind a belső, mind a külső támadások ellen védeni kell.
- **Integráció külső rendszerekkel és partnerekkel**
  - A belső üzleti folyamatok mellett az internet szolgáltatónak gyakran külső rendszerekkel (pl. beszállítókkal, ügyfelekkel, pénzintézetekkel) is kapcsolatba kell lépnie. Az e rendszerekkel való rugalmas integráció biztosítja a zökkenőmentes együttműködést és adatcserét harmadik felekkel.
- **Az innováció és az új technológiákhoz való alkalmazkodás elősegítése**
  - A vállalkozások folyamatosan szembesülnek az olyan új technológiai trendekkel, mint a mesterséges intelligencia (AI), a tárgyak internete (IoT) vagy a blokklánc. Az internetszolgáltatóknak támogatniuk kell ezen innovációk átvételét annak érdekében, hogy a vállalkozások versenyképesek maradhassanak.
- **Mobil platformok és távoli hozzáférés támogatása**
  - A mobiltechnológiák térhódításával az internetszolgáltatóknak lehetővé kell tenniük a különböző eszközökről, köztük mobiltelefonokról és táblagépekről történő hozzáférést, és támogatniuk kell a távmunkát, ami ma már sok vállalkozás számára elengedhetetlen.
- **Környezeti fenntarthatóság és felelősségvállalás**
  - Vállalati információs rendszer támogathatja a vállalat ökológiai lábnyomának csökkentésére irányuló kezdeményezéseket is az erőforrás-fogyasztás optimalizálásával, a szén-dioxid-kibocsátás nyomon követésével és a fenntartható üzleti stratégiák megvalósításának segítésével.
- **Felhasználói oktatás és támogatás**
  - Fontos támogatói szerepet tölt be a rendszer felhasználóinak képzése és folyamatos támogatása. Ez

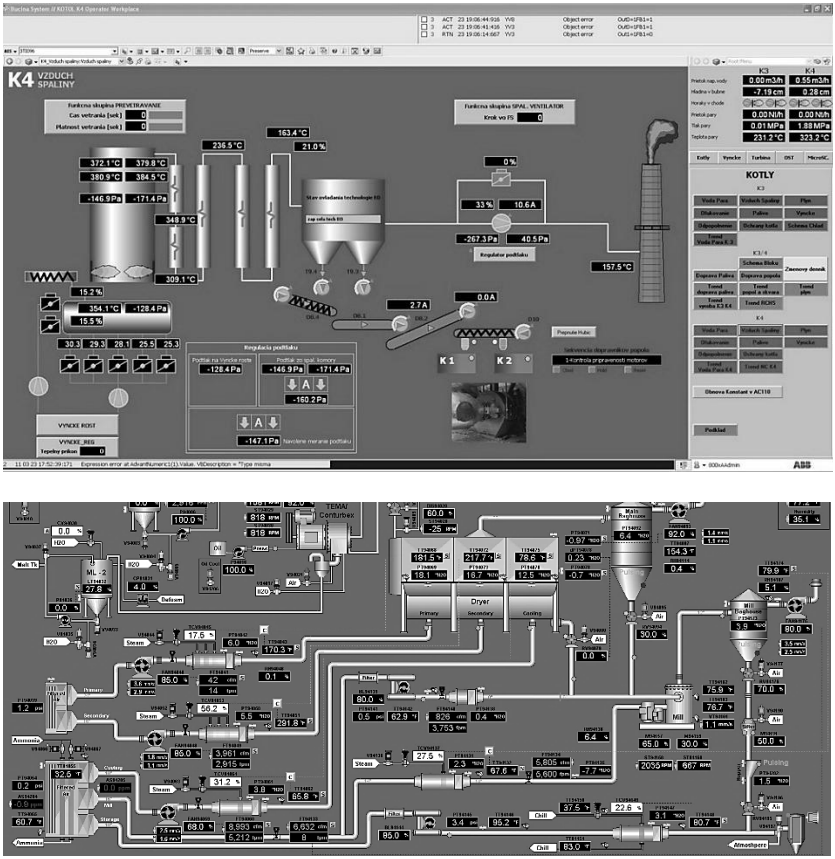
magában foglalja a kézikönyvek, a képzés és a technikai támogatás biztosítását annak érdekében, hogy a felhasználók képesek legyenek hatékonyan kihasználni a rendszer képességeit.

- **Jelentés és megfelelés a nemzetközi szabványoknak**
  - Az internetszolgáltatónak olyan eszközöket kell biztosítania, amelyekkel részletes jelentéseket lehet készíteni, és amelyek megfelelnek a különböző nemzetközi szabványoknak és normáknak, például az IFRS-nek (Nemzetközi Pénzügyi Beszámolási Standardok), az ISO-nak (Nemzetközi Szabványügyi Szervezet) vagy a GDPR-nak (Általános Adatvédelmi Rendelet).
- **A többnyelvűség és a nemzetközi terjeszkedés támogatása**
  - A globális vállalkozásokban használt BIS-nek (Business Information System) képesnek kell lennie a különböző nyelvek, valuták és adórendszerek támogatására, megkönnyítve ezzel a nemzetközi üzleti tevékenységet.

Az automatizálás vagy digitalizálás napjaink modern módszerévé vált, amellyel az informatikai ágazat teljesen új lehetőségeket nyit meg az információval való munkavégzésre. Ugyanakkor megteremti az alapját a hatékonyabb döntéshozatalnak a szervezetekben. Az automatizálás mértékét tekintve az információs rendszereket feloszthatjuk:

- **Nem automatizált rendszerek** - például fizikai nyilvántartások az orvosi rendelőben (pl. papír alpúak) vagy papír alapú raktárkészlet-nyilvántartás (minden beérkező/kimenő tételt kézzel vezetnek).
- **Részben automatizált rendszerek** - az IKT-eszközöket a napirend egy részéhez használják, pl. egy kisvállalkozás bérnyilvántartása.
- **Automatizált rendszerek** - a teljes napirend teljes mértékben IKT-források felhasználásával kerül feldolgozásra, például egy egyetemi rendszer vagy banki online tranzakciók teljes körű automatizálása.

Ebben a szakaszban szándékosan az IKT kifejezést használtuk az információs rendszer helyett, ami szintén helyes, de tisztában kell lennünk azzal, hogy az automatizálás számos technológiai folyamatot is magában foglal. Ezek a fizikai változók érzékelésétől kezdve a döntéshozatalon át a szabályozásig és az automatizált rendszerek működtetőinek vezérléséig terjedhetnek. Ilyen például az automatizált gyártás.



2.3. ábra: Fent: Ipari automatizálás SCADA rendszerben. Lent: részletesebb példa. Forrás: https://siccontrols.com/services/hmi-scada/

Az **Ipar 4.0 (negyedik ipari forradalom)** szorosan kapcsolódik a vállalatirányításhoz, mivel célja, hogy a gyártási folyamatokat és a vállalati működést integrált, digitális rendszerek segítségével optimalizálja és hatékonyabbá tegye. Az Ipar 4.0 főbb elemei, mint az IoT (Internet of Things), a big data elemzés, a mesterséges intelligencia és a robotika, nagyban hozzájárulnak a modern vállalatirányítás fejlődéséhez. Az Ipar 4.0 egyik kulcsfontosságú célja az automatizált, önállóan működő rendszerek fejlesztése. Ez csökkenti az emberi beavatkozást, és pontosabb, gyorsabb folyamatokat eredményez. Az automatizált munkafolyamatok és robotizált megoldások szintén nagyban megkönnyítik az erőforrások elosztását a vállalat irányításában. Nézzünk néhány példát.

> *Az Ipar 4.0 eszközei, mint a szenzorok és a beágyazott rendszerek, folyamatos adatokat szolgáltatnak a gyártási és logisztikai folyamatokról.*

Ezek az adatok a vállalatirányítási rendszerekbe, például az ERP (Enterprise Resource Planning) és a MES (Manufacturing Execution System) rendszerekbe kerülnek, így valós idejű döntéshozatalra adnak lehetőséget.
Az Ipar 4.0-ban a Big Data elemzés és a mesterséges intelligencia lehetővé teszi a prediktív karbantartás alkalmazását, amely az érzékelők adatait felhasználva előre jelzi, mikor lesz szükség egy gép vagy rendszer karbantartására. Segít gyorsabb és pontosabb döntéseket hozni a termelésben, készletezésben és karbantartásban. Ez minimalizálja az állásidőt és csökkenti a karbantartási költségeket, ami közvetlenül hozzájárul a vállalati hatékonysághoz.

Az Ipar 4.0 a vállalatirányítást magasabb szintre emeli azáltal, hogy a gyártási és adminisztratív folyamatokat összekapcsolja és digitalizálja. Muszáj szintén elmondanunk, hogy már létezik az **Ipar 5.0** koncepciója, és egyre több figyelmet kap napjainkban. Az Ipar 5.0 célja, hogy emberközpontúbb legyen, és az emberek és gépek közötti együttműködésre összpontosítson. Az **Ipar 6.0** jelenleg inkább csak egy elméleti koncepció, amellyel tovább a tárgyunk szempontjából nincs értelme részletesebben foglalkozni. A teljesség igényét nézve tekintsük át az egyes ipari forradalmakat alább:

- **Ipar 1.0 (első ipari forradalom)**: a 18. század végén kezdődött, és a gőzgép feltalálásával robbanásszerű fejlődést hozott. A gépek bevezetésével a gyártás jelentősen felgyorsult, és megjelentek az első gyárak. Az emberek kézzel és egyszerű eszközökkel dolgoztak, és a termelés kisebb közösségekben, kézműves módszerekkel történt. Jelen volt a mechanizáció.
- **Ipar 2.0 (második ipari forradalom)**: a 19. század végén jelent meg, a villamosenergia és az összeszerelő sorok bevezetésével. Ez lehetővé tette a tömegtermelést, így nagyobb mennyiségben és alacsonyabb költséggel gyárthattak termékeket. Ebben az időszakban a gyárak még nagyobb mértékben automatizálódtak, és kialakult a modern munkamegosztás.
- **Ipar 3.0 (harmadik ipari forradalom)**: a 20. század közepén indult, és az elektronika, az informatika és az automatizálás jelentette a központi elemeit. Ebben a korszakban megjelentek a számítógépek és a programozható gépek, amelyek lehetővé tették az automatizált gyártást és a termelési folyamatok precíz vezérlését.

3 Az információs rendszer és stratégiai szerepe

A stratégiát definiálhatnánk, mint a vállalati célok rendszerét és az elérhetőség lehetőségeit. **A definiált vállalati stratégia a hierarchia struktúrába állított vállalati stratégiák csúcsát képviseli**. Definiálja és meghatározza a vállalati alrendszerek stratégiáit, melyekből egyik az információs stratégia. Az információs stratégia meghatározza, hogyan lesznek az információs rendszer és információs technológia segítségével elérve a vállalati célok.

A stratégiai irányítás a felső vezetés feladata. Egyik szerepe a nagy stratégiai döntések meghozatala. A stratégiai irányítási és döntéshozatali rendszer stratégiai gondolkodást igényel. A stratégiai gondolkodás azt jelenti, hogy megvizsgáljuk a lehetőségeinket, elemezzük a külső környezet összes tényezőjét, a változatok tervezésén keresztül figyelembe vesszük a változások állandóságát, rugalmasan reagálunk a változásokra és rugalmasan befolyásoljuk azokat, átgondoltan átcsoportosítjuk az erőforrásokat és kreatívan megtanuljuk leküzdeni a problémákat, kihasználva az összes munkatársunkban rejlő lehetőségeket. A felső vezetés szerepe:

- az általános stratégiai célok meghatározása,
- az üzletpolitika megfogalmazása,
- a vállalati politika szabályozása,
- az üzletben bekövetkező főbb változások szabályozása,
- a piaci szegmensek meghatározása,
- a vállalat pénzügyi politikájának és költségvetésének megtervezése és jóváhagyása,
- a vállalat főbb forrásainak megszerzéséről és felhasználásáról, valamint a beruházások elosztásáról való döntés,
- a stratégiai foglalkoztatási kérdések kezelése,
- a vezetők kiválasztására és képzésére vonatkozó kritériumok meghatározása. (Sándor, J. 2011) (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P. 2017)

3.1 A vállalati információs rendszer stratégiai szerepe

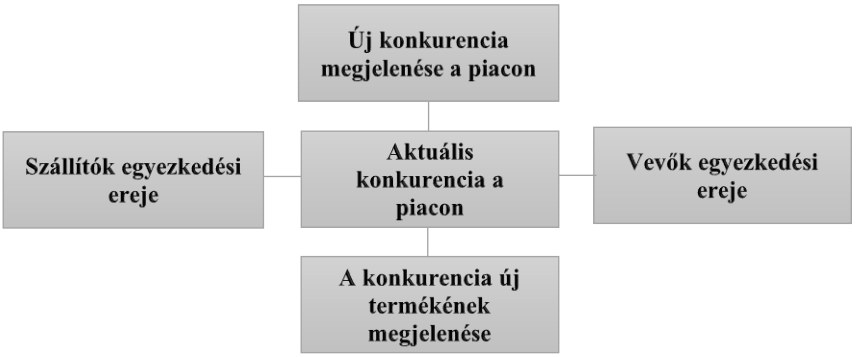
A vállalati információs rendszer (VIR) stratégiai szerepe arra utal, hogy az információs rendszer hogyan járul hozzá a szervezet hosszú távú céljainak megvalósításához, versenyképességéhez és hatékony működéséhez. Az információs rendszerek már régen túlléptek az adminisztratív és működési funkciókon, és a vezetés és a döntéstámogatás stratégiai eszközévé váltak. Az információs rendszer stratégiai szerepe lehetővé teszi a vállalkozás számára:

- **A folyamatok és erőforrások optimalizálását** - a VIR lehetővé teszi az erőforrások jobb tervezését és kezelését, valamint a belső és külső folyamatok optimalizálását.
    - Ez nagyobb hatékonyságot és alacsonyabb költségeket eredményez.
- **A döntéshozatal támogatását** - a rendszer összegyűjti, elemzi és bemutatja a releváns adatokat a vezetés számára, ami támogatja a stratégiai döntéshozatalt, és teret teremt a környezet változásaira való gyors reagálásra.
- **A versenyképesség fokozását** - a VIR lehetővé teszi az új üzleti modellek bevezetését, az innovációt és a piaci alkalmazkodást, ami a szervezet versenyképességének javulásához vezet.
- **Az innováció és az átalakulás támogatását** - a modern VIR a digitális átalakulás alapja, amely a termékek, szolgáltatások és üzleti folyamatok innovációját ösztönözheti.
- **A kockázatkezelést és a kontrollok javítását** - a VIR lehetővé teszi a hatékony nyomon követést és kockázatkezelést, így a vállalat képes előre jelezni a problémákat és kiküszöbölni a fenyegetéseket.
- **A partnerekkel és ügyfelekkel való kapcsolatok javítását** - a VIR támogathatja az ügyfélkapcsolat-kezelő (CRM) rendszereket, a beszállítókat és más partnereket, segítve a hosszú távú kapcsolatok kiépítését.

Az információs rendszerek stratégiai szerepének fogalmát gyakran tárgyalják a szervezetek és vállalkozások modern irányításával összefüggésben. **A vállalati információs rendszer stratégiai szerepe a vállalat stratégiai céljainak elérése és támogatása, mint például a teljesítmény javítása, növekedés és fejlesztés, termék- és szolgáltatásinnováció, folyamatoptimalizálás és a versenyelőny hosszú**

távú fenntartása. A vállalati információs rendszer stratégiai szerepe tehát nem korlátozódik a napi operatív tevékenységek támogatására, hanem a stratégiai támogatás, az innováció és a hosszú távú tervezés révén értéket teremt. Elengedhetetlen, hogy a vállalati információs rendszer összhangban legyen a vállalat átfogó stratégiájával, és célja, hogy segítse a vállalat jövőképének és küldetésének megvalósítását. (Sándor, J. 2011) (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P. 2017)

3.2 Porter modell



3.1. ábra: Porter öt erő modell

Michael E. Porter Versenystratégia című könyve 1979-ben jelent meg. Ez a könyv volt a Porter féle 5 tényezős modell kiindulópontja. Az öt erő elemzése alapján a vállalkozások különböző stratégiákat választhatnak, mint például a költségvezetés, a differenciálás vagy az egyes piaci szegmensekre való összpontosítás. Az egyes erőket következőképpen definiálhatjuk:

- **Új konkurencia megjelenése a piacon**
  - Az ajánlat és kereslet növelés veszélyt jelent, mivel létrehozhatja az ajánlat túlsúlyát, ami árcsökkentéshez vezet.
  - Vállalati információs rendszer nyújthat:
    - eszközöket a produkció gondos kiadásaira,
    - segítséget az akadályok kiépítéséhez az új konkurencia megjelenése ellen.
  - Racionális információ használása lehetővé teszi:
    - új termékek növekedését,
    - áru és szolgáltatás minőségi növekedést.

- **Az új konkurencia termékének megjelenése**
  - A termékek lecserélése megoldást jelenthet,
  - Megfelelő információs rendszer alkalmazása segíthet:
    - a termékek ár/érték relációcsökkentését,
    - növelni a termékek értékét,
    - megengedni az invesztíció használásának a módosítását.
- **Aktuális konkurencia a piacon**
  - Az információs rendszertől elvárjuk:
    - konkurencia előnyösséget helyes adat feldolgozásával,
    - kiadások csökkentését és minőség növelését,
    - a vállalati pozíciónk megvédését a piac területén,
    - vállalati stratégia támogatását,
    - információ feldolgozását a vevőinkről és a követelményeikről,
    - számítógéppel irányított gyártást, termelést.
- **Szállítók és vevők egyezkedési ereje**
  - Az információs rendszer (IR) feladata:
    - csökkenteni (gyengíteni) a szállítók és vevők erőtereit,
    - engedélyezni a marketing IR által a racionális árulást és vásárlást, növelni az áttekintést a szállítók és vevő aktvitásáról.

A meglévő versenytársak közötti rivalizálás az iparágban meglévő vállalatok közötti verseny intenzitására utal. A nagyfokú rivalizálás gyakran vezet az árak csökkentéséhez, a minőség javításához vagy az innovációra gyakorolt nyomáshoz. Egy vállalati információs rendszer támogathatja a versenyadatok elemzését, az árak nyomon követését, a minőség és a termékteljesítmény nyomon követését. Az analitikai eszközök és az üzleti intelligencia (BI) modulok betekintést nyújtanak a versenytársak viselkedésébe, ami jobb stratégiai döntéshozatalt tesz lehetővé.
A beszállítók erős pozícióval rendelkezhetnek, ha az iparágban kevés alternatív beszállító van, vagy ha a beszállítók egyedi inputokat biztosítanak. Egy vállalati információs rendszer támogathatja a hatékony ellátási lánc menedzsmentet, a beszállítói teljesítmény nyomon követését és az ellátási lánc menedzsmentet (SCM).
Ha a vásárlóknak nagy hatalmuk van, alacsonyabb árakat tudnak kialkudni, magasabb minőséget és szolgáltatást követelhetnek. Ez különösen akkor fordul elő, ha sok konkurens termék van, vagy ha a vevőknek alacsonyak a

váltási költségei. A vállalati információs rendszerek támogatják a CRM (Customer Relationship Management) modulokat, amelyek javítják az ügyfélkapcsolatok kezelését, jobban megértik az igényeiket és nagyobb értéket nyújtanak. Az internetszolgáltatók lehetővé teszik a szolgáltatások és termékek személyre szabását is, ami nagyobb ügyfélhűséget eredményez.

Az új versenytársak belépése az ágazatba veszélyeztetheti a meglévő szereplőket, különösen, ha alacsonyok a belépési korlátok. A vállalati információs rendszerek lehetővé teszik az automatizálást és a folyamatok hatékonyságát, ami csökkenti a költségeket és növeli a vállalkozás versenyképességét.

A termék helyettesítésének veszélye akkor áll fenn, ha más termékek vagy szolgáltatások helyettesíthetik a vállalat jelenlegi termékkínálatát, ami csökkentheti a keresletet. Az információs rendszerek támogathatják a kutatást és fejlesztést, figyelemmel kísérhetik a piaci trendeket, és segíthetnek a helyettesítés kockázatát csökkentő innovációk gyors bevezetésében.

Muszáj elmondanunk, hogy több szerző szerint ez a modell a mai modern világban már bővítést igényel, mert nem veszi figyelembe pl. a kormányzatot vagy politikai környezetet. A Porter modell egy fontos kiterjesztése található meg Adam Brandenburger és Barry Nalebuff (Yale School of Management) munkásságában, a 90-es évek közepén. A játékelméletet felhasználva, hozzáadták a modellhez a **kiegészítők koncepcióját** („a hatodik erő"), ezzel segítve a stratégiai szövetségek mögötti logika megértését. Az elképzelést, hogy a **kiegészítők**[7] adják a hatodik erőt, gyakran társították Andrew Grove nevéhez (az Intel Corporation korábbi ügyvezetője). A legtöbb forrás szerint, a hatodik erő a **kormány vagy a nyilvánosság**. Martyn Richard Jones 1993-ban Skóciában, a Groupe Bullnál tanácsadóként alkalmazva, kialakított egy továbbfejlesztett öt erő modellt 1993-ban Skóciában. Ez a Porter modellen alapul és tartalmazza a **kormányzatot** (nemzeti vagy regionális) és ún. **befolyásoló csoportokat** az elképzelt hatodik erőként. Ez a modell a Groupe Bull Tudásvagyon Menedzsment Szervezet (Knowledge Asset Management Organization) nevű kezdeményezésének körében végzett munka eredménye. Porter közvetetten elutasította más erők bevezetését, az öt erőt befolyásoló „faktoroknak" nevezve az innovációt, kormányzatot és

[7] *Pl. Coca-Cola és McDonald's. Kiegészítő vállalatok: Coca-Cola (üdítőitalgyártó) és McDonald's (gyorsétteremlánc). Együttműködés: A Coca-Cola és a McDonald's hosszú távú stratégiai partnerséget kötött. A McDonald's éttermei kizárólag Coca-Cola termékeket árulnak. Eredmény: A McDonald's stabil beszállítója a minőségi italoknak, míg a Coca-Cola kizárólagos forgalmazó partnerrel rendelkezik, ami növeli piaci részesedését és eladásait.*

kiegészítő termékeket. Tegyünk fel egy kiegészítő kérdést. *Hogyan járulhatnak hozzá a vállalati információs rendszerek a hatodik erő visszaszorításához vagy hatékony alkalmazásához?* A vállalati információs rendszerek az alábbi módon segíthetnek a vállalkozásoknak kezelni e külső tényezőkből eredő nyomást és kockázatokat:

- **A szabályozási változások nyomon követése**:
  - A vállalati információs rendszer nyomon követheti és elemezheti a jogszabályi és szabályozási környezetben bekövetkező változásokat. Integrálhatók olyan külső adatbázisokkal vagy portálokkal, amelyek nyomon követik az új szabályozásokat, és figyelmeztetik a vezetést a lehetséges kockázatokra vagy lehetőségekre. Például a megfelelőség okából az ERP-rendszereket frissíthetik az új adószabályokkal vagy környezetvédelmi előírásokkal.
- **A nyilvános kommunikáció és az átláthatóság javítása**:
  - A CRM és a BIS kommunikációs modulok segíthetnek a nyilvánossággal, az ügyfelekkel és a társadalmi csoportokkal való kommunikáció javításában. A vállalkozás átlátható tájékoztatást nyújthat gyakorlatáról és kezdeményezéseiről, ezáltal pozitív képet alakíthat ki magáról, és elkerülheti a közvélemény nyomását. Az információs rendszerek például nyomon követhetik a nyilvánosság visszajelzéseit, és javíthatják a panaszokra vagy problémákra adott válaszokat.
- **A gazdasági és politikai kockázatok kezelése**:
  - Az üzleti intelligencia (BI) modulok lehetővé teszik a makrogazdasági tényezők és a politikai kockázatok elemzését. Ez lehetővé teszi a vállalkozás számára, hogy jobban megtervezze tevékenységeit, diverzifikálja a kockázatokat és optimalizálja működését a különböző politikai és gazdasági körülmények között. Egy példa erre az analitikai eszközök használata az infláció vagy a kamatlábak változásának a vállalat költségeire gyakorolt hatásának előrejelzésére.
- **A megfelelés és a megfelelőség biztosítása**:
  - Az internetszolgáltatók beállíthatók a megfelelés automatikus nyomon követésére és kezelésére (megfelelésmenedzsment), ami segít megelőzni a bírságokat vagy más szabályozási szankciókat. Az információs rendszerek például nyomon követhetik a

gyártási folyamatok környezeti hatását, hogy biztosítsák a környezetvédelmi előírásoknak való megfelelést.

Hogyan járulhatnak hozzá a vállalati információs rendszerek lehetőségei a kiegészítők kapcsán? Például, a következőképpen:

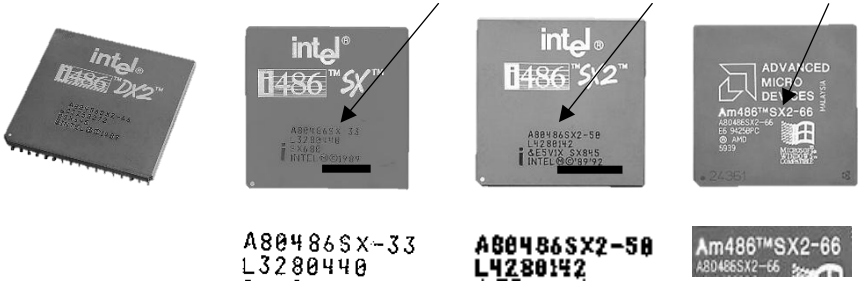
- **Kapcsolattartás a kiegészítő vállalkozásokkal:**
  - A vállalati információs rendszerek (BIS) lehetővé teszik a hatékony ellátási lánc menedzsmentet (SCM), a készletkövetést, valamint a partnerekkel való együttműködés elemzését és irányítását. A vállalkozás nyomon követheti a kiegészítő termékek teljesítményét, koordinálhatja a kiegészítő partnerekkel folytatott tevékenységeket (pl. közös marketingkampányok), és azonosíthatja az új együttműködési lehetőségeket.
- **A termékek és szolgáltatások integrációjának javítása:**
  - Az BIS-ek (pl. ERP-rendszerek) segíthetnek a kiegészítő termékek egyetlen rendszerbe történő integrálásában, egyszerűsítve a rendelést, a logisztikát és az ügyféltámogatást. Ez lehetővé teszi az ügyfelek számára, hogy a teljes termékcsomagot (pl. hardver és szoftver) egy rendszeren keresztül rendeljék meg, ami növeli a kényelmet és a lojalitást.
- **A kiegészítő termékek iránti kereslet előrejelzése:**
  - Az üzleti intelligencia (BI) modulok elemezhetik a kiegészítő termékek értékesítését, és a múltbeli adatok alapján megjósolhatják a keresletet. Ezen eszközök segítségével a vállalkozás jobban meg tudja becsülni, hogy mely kiegészítő termékek a legfontosabbak a vásárlók számára, és biztosítani tudja azok elérhetőségét.
- **A fejlesztés és az innováció támogatása:**
  - Az adatelemzés segítségével a vállalkozások azonosíthatják a kiegészítő termékek és szolgáltatások fejlesztésének új lehetőségeit.
  - Az információs rendszerek támogatják a részlegek közötti együttműködést, ami felgyorsítja az innovációt és az új termékek piaci bevezetését.
- **Marketing és CRM az ügyfelekkel való jobb kommunikáció érdekében:**
  - A CRM-rendszerek lehetővé teszik a jobb kommunikációt az ügyfelekkel és a kiegészítő termékekkel kapcsolatos visszajelzéseket. Az ügyfeleket így hatékonyabban lehet

tájékoztatni az új kiegészítő termékekről vagy szolgáltatásokról. (Sodomka, P.; Klčová, H. 2010)

Az INTEL-nek, mint számítógépes processzorgyártónak, az 1990-es évek elején szembe kellett néznie azzal a helyzettel, hogy egy versenytárs, az AMD megpróbált betörni a PC-processzorok piacára. Zászlóshajó processzoruk a 486DX2-66 volt. A 486DX-33 volt a legnagyobb példányszámban eladott processzor. Ezeknek a processzoroknak az ára még mindig túl magas volt ahhoz, hogy az egyre nagyobb keresletnek örvendő olcsó számítógépekben használhatóak legyenek. Szükség volt arra, hogy valahogyan ésszerű módon belépési korlátot teremtsenek egy olyan versenytárs számára, amely kifejezetten az olcsó PC-k piacát célozta meg, és jobban kiaknázzák azt a piacot, ahol a vásárlók nem voltak hajlandók magas árat fizetni a csúcsminőségű, nagy teljesítményű processzorokért.

Az INTEL ezért úgy döntött, hogy a 486DX-33-as chipek minőségét úgy rontja, hogy a gyártósor végére egy automatát helyez, amely a már legyártott és teljesen működőképes processzor lebegőpontos egységét (FPU) rontja.

A legtöbb ügyfélnek nem volt szüksége erre az irodai alkalmazások elvégzéséhez. Az INTEL így olcsóbb terméket tudott kínálni anélkül, hogy a teljes értékű 486DX-33 chipek alkalmazási piacát veszélyeztette volna. Az új processzor a 486SX-33 nevet kapta. A költségeket tekintve az egész eljárás olcsóbb volt, mint az eredeti chip tervezésébe való bármilyen beavatkozás. Így lehetővé vált, hogy az árhoz igazodva egy olcsó processzort kínáljanak az akkor elismert INTEL márkanév alatt. Az eredeti termék árának csökkentésével és az INTEL termék-portfóliójának bővítésével ez segített kiszolgálni egy olyan piacot, ahol az ügyfelek az alacsony árat preferálták, megtartani egy olyan piacot, ahol az ügyfelek hajlandóak voltak megfizetni a gyors, jó minőségű chipekért, és belépési akadályt képezni a versenytársak számára. Így egy új versenytárs belépésének veszélye átmenetileg elhárult.



3.2. ábra: i486 processzorok

3.3 Vállalati információs rendszer infrastruktúrája

A vállalati információs rendszer infrastruktúrája a jelenlegi és jövőbeli informatikai alkalmazások alapja, valamint az adatok tárolásának és átlátható hozzáférésének környezete. Műszaki erőforrásokból, információs technológiából, IT személyzetből (informatikai szakemberek), vállalati informatikai irányításból és IT-irányításból áll. A vállalati információs rendszerek infrastrukturális szolgáltatásai egyrészt a műszaki erőforrásokhoz és az információs technológiához (IT) kapcsolódnak, másrészt az IT-menedzsment és az IT-irányítás irányába orientálódnak. Ezek a szolgáltatások a vállalaton belül megosztottak.

Az első csoport (a technikai erőforrásokhoz és az információs technológiához kapcsolódó szolgáltatások) a következőkből áll:

- **Kommunikációs csatornaszolgáltatások**, amelyek magukban foglalják a szervezetet az ügyfelekkel és a partnerekkel összekötő valamennyi elektronikus kommunikációs csatorna kezelését és integrálását. Ide tartoznak például az internetes szolgáltatások (Web, Slack, Microsoft Teams, Zoom, e-mail), a hagyományos (szkennelt vagy címkézett) posta, a távközlés (fax, telefon), az EDI (*Electronic data Interchange*), stb.
- **A vállalati információs rendszer kockázatkezelésével és biztonságával kapcsolatos szolgáltatások**, amelyek magukba foglalják az információbiztonság meghatározott szintjének biztosítását, amelyet egy sor tulajdonság (bizalmasság, sértetlenség, rendelkezésre állás, bizonyíthatóság, megbízhatóság, megcáfolhatatlanság) képvisel, valamint a bizalmi környezet megteremtését valamennyi érdekelt fél (ügyfelek, partnerek, alkalmazottak, tulajdonosok) számára.
- **Kommunikációs szolgáltatások**, amelyek az egyes alkalmazás-feldolgozó helyek közötti átviteli útvonalak biztosítására irányulnak, számítógépes és távközlési hálózatokon, de ipari alkalmazásokban hálózatokon keresztül is.
- **Adatkezelési szolgáltatások**, amelyek olyan környezet létrehozására és kezelésére irányulnak, amely lehetővé teszi az adatok alkalmazásoktól független kezelését, beleértve az adatok átlátható elérhetőségének biztosítását mindazok számára, akiknek szükségük van rá.

- **Az informatikai menedzsment szolgáltatások**, melyek a műszaki erőforrások és az információs technológiák kezelésével és menedzselésével kapcsolatos tevékenységekre irányulnak, és magukban foglalják például a végberendezések, szerverek kezelését, de az új rendszerek (új informatikai alkalmazások) fejlesztéséhez és bevezetéséhez megfelelő környezet kialakítását is.
- **Infrastruktúra-alkalmazáskezelési szolgáltatások**, azaz a szervezeten belül használt alkalmazások. Az ilyen alkalmazásokat az infrastruktúrába foglalják, hogy egységes és megosztott szolgáltatásaikat ott nyújtsák, ahol szükség van rájuk.

A második csoport (az IT-menedzsment és IT-irányítás (IT governance) irányába mutató szolgáltatások) a következőkből áll:

- **IT-menedzsment szolgáltatások**, melyek egyrészt a vállalati információs rendszer fejlesztésével, másrészt a működésének biztosításával kapcsolatos szolgáltatások összességét jelentik.
- **Architektúra- és szabványosítási szolgáltatások**, amelyek a vállalati információs rendszer architektúrájának kialakítását és a megfelelő szabványok elfogadását szolgálják annak érdekében, hogy az információs technológia használata és a vállalati információs rendszer jövőbeli fejlesztése összhangban legyen a vállalat célkitűzéseivel és stratégiájával.
- **Az informatikai képzési szolgáltatások**, melyek az információs technológia és alkalmazásainak használatára vonatkozó képzési szolgáltatásokra összpontosítanak, de a vezetők képzésére is, hogy képesek legyenek azonosítani a megszerzett értéket, az információs technológia használata és az abba való befektetés által elért értéket.
- **Az informatikai kutatási és fejlesztési szolgáltatások**, melyek az új lehetőségek megtalálására összpontosítanak, amelyekkel az információs technológia értéket, versenyelőnyt vagy hatékonyságnövekedést nyújthat a vállalkozás számára.

A vállalati információs rendszer infrastruktúrájától elvárjuk, hogy legyen **rugalmas** (a rendszer azon képessége, amely lehetővé teszi számára, hogy reagáljon az előre jelzett vagy váratlan változásokra), **alkalmazkodó** (a rendszer azon képessége, hogy gyorsan alkalmazkodik a váratlan változásokhoz) és **hatékony** (a rendszer azon képessége, hogy a kitűzött célok és szükségletek teljesítése érdekében hatékony, a helyzetnek megfelelő eszközöket alkalmazzon).

Ezeket a tulajdonságokat az egyes infrastruktúra-elemek megfelelő képességi szintjének meghatározásával érik el. Ezek a következők:

- A nagyobb egészbe való **integrálódás képessége**. Ezt a jellemzőt befolyásolja a **más elemhez való kapcsolódás képessége** (akár a szervezeten belül, akár azon kívül) és a **kompatibilitás szintje** (kompatibilitás és összeférhetőség) az integrált elemek között.
- Az a képesség, hogy az egészre gyakorolt jelentős hatás nélkül megfelelő mértékben képes legyen változtatni és bővíteni, azaz megfelelő **modularitással** rendelkezzen. A modularitást ezután az jellemzi, hogy egy komponens által az egésznek kínált funkciók milyen részletességgel vannak kidolgozva, és hogy az adott komponensnek milyen mértékben kell átláthatónak lennie más elemek számára.
- Megfelelő szintű **ismeretek, tapasztalatok és készségek** megléte. Ezek a képességek a technikai erőforrások és az információtechnológia területén elvártak annak megértése terén, hogy hol és hogyan kell alkalmazni az információtechnológiát a szervezet céljainak hatékony támogatása érdekében, az üzlet megértése terén, az üzlethez illeszkedő megoldások megtalálása terén, valamint a vezetés terén. (Gála, L.; Pour, J.; Šedivá, Z. 2015)

3.4 Modern megoldások kihívásai és veszélyei

Ebben a fejezetben a sok lehetőség közül néhány olyan modernebb megoldást közelítünk meg, amelyek manapság nagyon népszerűek és elterjedtek a vállalati információs rendszereken belül is.

3.4.1 Web 1.0, Web 2.0 és Web 3.0

Az internetszolgáltatások használatának fejlődésével felmerült az igény, hogy a felhasználók bevonására alkalmasak legyenek. A Web 2.0 az internetes technológiák és weboldalak második generációjára utal, amelyek az interaktivitást, az együttműködést és a felhasználók közötti tartalommegosztást helyezik előtérbe. A Web 1.0 a World Wide Web első generációjának megjelölése (nagyjából az 1989-es létrehozásától a 2000-es évek elejéig). A web ezen verzióját gyakran "statikus webnek" nevezik, mivel fő jellemzői a statikus tartalom, a korlátozott interaktivitás és a passzív felhasználók voltak. A Web 1.0 lényegében egyirányú kommunikációt jelentett, ahol a felhasználóknak nem volt lehetőségük

aktív tartalmat létrehozni vagy módosítani. Míg a Web első generációja a Web 1.0 nagyrészt statikus volt, és interakcióra nem sok lehetőséget kínált, addig a **Web 2.0** lehetővé tette a felhasználók számára, hogy ne csak tartalmakat fogyasszanak, hanem maguk is létrehozzák és közvetlenül befolyásolják azokat. A statikus weboldalakkal ellentétben a Web 2.0 lehetővé teszi a felhasználók számára, hogy interakcióba lépjenek a webes alkalmazásokkal. A felhasználók megjegyzéseket, értékeléseket, hozzászólásokat adhatnak hozzá a tartalomhoz, hogy közvetlenül a weboldalakon hozzanak létre, szerkesszenek és osszanak meg tartalmakat. Pl. szociális hálózatok, blogok és fórumok, a YouTube, a Facebook, az Instagram vagy a Wikipedia. A Web 2.0 támogatja a felhasználók közötti együttműködést valós időben szerkeszthető eszközök segítségével, például közös dokumentumokkal vagy felhőalapú megoldásokkal, mint a Google dokumentumok. A felhasználók személyes információkat, fényképeket, videókat, cikkeket oszthatnak meg és valós időben kommunikálhatnak. A Web 2.0 oldalak gyakran használnak olyan technológiákat, mint az **AJAX** (aszinkron JavaScript és XML), amelyek gyorsabb és dinamikusabb interakciót tesznek lehetővé anélkül, hogy folyamatosan újra kellene tölteni az oldalakat. A Web 2.0 így interaktívabb és érzékenyebb webes alkalmazásokat biztosít. Megváltoztatta az alkalmazások használatának módját, mivel segítségével számos szolgáltatást ma már a böngészőn keresztül, webes alkalmazásként nyújtanak. Ahelyett, hogy a felhasználók egy szoftvert telepítenének a számítógépre, az interneten keresztül közvetlenül a webről használhatják az alkalmazásokat.
A vállalati információs rendszerekben szintén hasznosak ezek a modern megoldások. Például:

- Lehetővé tette a vállalkozások számára, hogy a szociális hálózatokon és interaktív webhelyeken keresztül jobban kommunikáljanak az ügyfelekkel.
- Az olyan eszközök, mint a Microsoft Teams, BigBlueButton és a Google Workspace megkönnyítik az alkalmazottak együttműködését. Számos vállalatban a mindennapi munka részét képezik.
- A közösségi média és a blogolási platformok a márkaépítés és a nyilvánossággal való kommunikáció fontos eszközeivé váltak.

A gyakran **szemantikus webnek** vagy **decentralizált webnek** nevezett **Web 3.0** abban különbözik az előbbitől, hogy intelligensebb és automatizáltabb adatfeldolgozásra törekszik. A Web 3.0 lehetővé teszi az információk jelentésének és kontextusának jobb megértését, ami intelligensebb keresést és személyre szabottabb szolgáltatásokat eredményez. Pontosabban:

- A Web 3.0 lehetővé teszi az alkalmazásoknak és rendszereknek, hogy jobban megértsék a weben található adatok jelentését. Ez kifinomultabb kereséseket tesz lehetővé, és pontosabb eredményeket biztosít nem csak a kulcsszavak, de a kontextus alapján is.
- Míg a Web 2.0 esetében a szolgáltatást nagyrészt nagy központi platformok (Google, Facebook, Amazon) irányítják, addig a Web 3.0 olyan decentralizált technológiákra törekszik, mint a **blockchain**, a kriptovaluták és a decentralizált alkalmazások (DApps).
- A Web 3.0 jobban kihasználja a mesterséges intelligenciát és a gépi tanulást nagy mennyiségű adat feldolgozására és elemzésére, így személyre szabott szolgáltatásokat és ajánlásokat tesz lehetővé magasabb szinten.
- A Web 3.0 célja, hogy személyre szabottabb élményt nyújtson a felhasználók számára, miközben a személyes adatok és a magánélet nagyobb ellenőrzését is ígéri.

Ezek a lehetőségek új perspektívákat nyithatnak a vállalat életében is, hiszen az adatok szemantikai elemzése (ami azt jelenti, hogy az alkalmazások képesek lesznek jobban megérteni az információk tartalmát és kontextusát), amely automatikus adatintegrálást fog jelenteni, alátámaszthatja nem csak a prediktív analízist, hanem még pontosabb döntéshozatalt is eredményezhet. Mesterséges intelligencia és gépi tanulás még szélesebb lehetőségeket kínál a vállalatok számára:

- A Web 3.0-ra való átállás a vállalati IT-infrastruktúrák frissítését igényli, ami kihívást jelenthet, és jelentős költségekkel járhat.
- Az olyan új Web 3.0 technológiák bevezetése, mint a blokklánc vagy a mesterséges intelligencia, jelentős fejlesztési, képzési és biztonsági befektetést igényel.
- A vállalkozások számára elengedhetetlen lesz, hogy stratégiáikat és munkafolyamataikat az új technológiákhoz és a decentralizált adatkezelési modellhez igazítsák. (Gála, L.; Pour, J.; Šedivá, Z. 2015) (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P., 2017)

3.4.2 Internet of Things (IoT)

A tárgyak internete (IoT) olyan fizikai eszközök, amelyek az internethez vagy egymáshoz kapcsolódhatnak, mint pl. járművek, háztartási gépek. Olyan objektumok hálózatára utal, amelyek érzékelőkkel,

szoftverekkel és egyéb technológiákkal vannak felszerelve, és amelyek lehetővé teszik az adatok egymás közötti cseréjét az interneten keresztül. Ezek az eszközök bármik lehetnek – az okosóráktól és az otthoni termosztátoktól az ipari gépekig vagy járművekig. Az IoT-eszközök érzékelőkkel vannak felszerelve, amelyek adatokat gyűjtenek a környezetükről (például hőmérsékletet, páratartalomat, GPS koorinátákat, a pulzusszámot figyelő okosórák, az otthoni eszközök érzékelői melyek az energiafogyasztást figyelik), majd ezeket az adatokat az interneten keresztül elemzés vagy további feldolgozás céljából továbbítják.

Az okosotthonoktól az ipari alkalmazásokig az IoT új lehetőségeket nyit meg az automatizálás, a hatékonyság javítása és az értékes adatok rögzítése terén. Az IoT-eszközök valós idejű információkat biztosítanak, lehetővé téve az azonnali válaszadást és a döntéshozatalt az aktuális feltételek alapján. Az eszközök emberi beavatkozás nélkül is képesek kommunikálni egymással, amit gyakran gépek közötti kommunikációnak (**Machine-to-Machine**, azaz M2M) neveznek. Az iparban ezek az adatok segíthetnek optimalizálni a termelést vagy figyelemmel kísérni a gépek állapotát a meghibásodások elkerülése érdekében, vagy lehetővé teszi a gépek monitorozását, a termelési folyamatok optimalizálását, a gépek előre jelző karbantartását és az általános termelési hatékonyság javítását. (Sándor, J. 2011) (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P., 2017)

3.4.3 BYOD és BYOC

A mobilitás fejlődése, amelyet a mobil hordozható számítógépek széles körű elterjedése képvisel, áll a **Bring-Your-Own-Device** (**BYOD**) – hozd magaddal a saját eszközödet trend középpontjában. A BYOD egy olyan informatikai irányelv, amely lehetővé teszi – vagy kifejezetten ösztönzi –, hogy az alkalmazottak, üzleti partnerek és egyéb felhasználók saját mobileszközeiket használják a vállalati adatokhoz, alkalmazásokhoz és rendszerekhez való hozzáféréshez. Jelenleg ennek a trendnek a vállalati környezetben való elfogadása kihívást jelent a vállalati és informatikai vezetés számára, mivel mind pozitív hatásokkal, mind pedig problémákkal és kockázatokkal jár. Ami a munkavállalók számára kihívás, az a vállalat számára kockázatot jelenthet, és fordítva. A 3.1. táblázat összefoglalja a BYOD hatásait, problémáit, kockázatait és kihívásait. Valószínűleg a legnagyobb kockázat az adat– és alkalmazásbiztonsággal kapcsolatos. A kockázatok e csoportjára a következő a válasz:

- **Mobileszköz-kezelés**. Adminisztratív és technikai megoldást mutat be a mobileszközök vállalati információs rendszerbe történő telepítésére, biztosítására, integrálására és kezelésére.
- **Mobil alkalmazások kezelése**. Ez egy olyan eszközkészlet, amely a vállalati alkalmazások telepítését és adminisztrációját biztosítja a munkavállalók tulajdonában lévő mobileszközök környezetében.

A második jelenség, amely a mai vállalati információs rendszerekben megfigyelhető, a **Bring-Your-Own-Cloud** (BYOC) – hozd a saját felhődet. A **BYOC** egy olyan koncepció/trend, amely lehetővé teszi az alkalmazottak számára, hogy bizonyos munkafeladatok elvégzéséhez harmadik féltől származó privát vagy nyilvános felhőszolgáltatásokat használjanak. A BYOC megjelenését a vállalati szférában néha **szürke informatikának** nevezik, amelyet a következő jellemzők illetnek meg:

- Kontroll elvesztése a vállalati erőforrások felett – a cégnek fogalma sincs arról, hogy melyik alkalmazott milyen információs technológiákat használ, azokra nincs befolyása, így informatikai erőforrásait sem tudja hatékonyan megtervezni.
- Inkonzisztens rendszerek használata – az alkalmazottak különböző eszközöket, alkalmazásokat és platformokat használnak, amelyek gyakran nem kompatibilisek egymással.
- Megnövekedett adatvesztés kockázata és az e feletti kontroll hiánya – a cégnek nincs rálátása arra, hogy ki, hogyan és mikor használja a céges adatokat, alkalmazásokat.
- Hibakockázat növelése – az adatbiztonságért teljes felelősséget vállal az a munkavállaló, aki egy adott alkalmazást választott. (Mivel sok esetben nem informatikus, nem ismer minden kockázatot!)

A fenti kockázatok ellenére a BYOC trend egyre nagyobb teret hódít a vállalkozásokban, mert ez a megoldás:

- kényelmes;
- intuitív és hatékony az alkalmazottak számára;
- használatra kész, és biztosítja a verziószámozást és az adatszinkronizálást;
- lehetővé teszi, hogy részben bármikor és bárhonnan dolgozzon;
- hatékonyabb, mivel a vállalati IT nem tud jobb megoldást kínálni. (Gála, L.; Pour, J.; Šedivá, Z. 2015) (Sándor, J. 2011) (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P., 2017) (Pavlíček, A.; Syrovátková, J. 2022)

3.1. táblázat: A BYOD hatásai, kihívásai és kockázatai

A BYOD hatásai és kihívásai	BYOD problémák és kockázatok
• A berendezések és szolgáltatások (vásárlás és szervizelés) költségmegtakarítása a felhasználó hozzáállása alapján. Saját eszközeit jobban kezeli és kíméli, mint a vállalati eszközöket. • Az alkalmazottak termelékenységének növekedése, amely azon alapul, hogy az alkalmazottak hozzászoktak a berendezéshez, és nagyon jól tudják használni azt, mert ők maguk választották. • A munkavállalók elégedettsége, amely abból ered, hogy a mobilitás és az utazás összefüggésében kényelmesebb, ha a munkavállalónak nem kell több eszközt (személyes és vállalati) magával vinnie és használnia. • Versenyelőny és innovációs potenciál, amelyet az a tény képvisel, hogy a vállalati információs rendszerek függetlensége a végfelhasználói eszközöktől megerősödik. • A vállalat vonzerőjének növelése az új alkalmazottak számára. Különösen a fiatalabb generáció nőtt fel a mobileszközökön, támaszkodik rájuk, és azok teljes mértékben az életük részét képezik. Az elavult eszközöket nem megfelelő eszköznek tekintik.	Biztonság: • Adatvédelem a készülék elvesztése esetén. • A kommunikáció védelme bizonytalan környezetben. • Védelem harmadik féltől származó alkalmazásokkal szemben. • Az eszközök használatból való kivonásának módja. Infrastruktúra: • Az erőforrások nagyfokú változatossága miatt az alkalmazásokat több platformon való futtatáshoz is hozzá kell igazítani. • Az infrastruktúra teljesítményének kiegyensúlyozása nehézkes, mivel nem lehet pontosan tudni, hogy mikor kerül sor a kommunikáció megvalósítására. Jogszabályok és adatvédelem: • A felhasználási feltételekhez való hozzájárulás biztosításának szükségessége a személyes eszközkörnyezetnek a vállalat általi nyomon követése miatt (az eszköz a vállalati alkalmazásadatokon kívül munkavállalói adatokat és alkalmazásokat is tartalmaz). Pénzügyi terület: • A munkavállalók képzésének és a help desk költségei. • A bevezetés és az üzemeltetés rejtett költségei.

Forrás: (Gála, L; Pour, J; Šedivá, Z., 2015)

3.4.4 Big Data és Open Data

A vállalati információs rendszerek nagy mennyiségű adattal rendelkeznek, melyeket a vállalat kielemezhet a döntéshozatalának javítására, a folyamatok optimalizálására, a piac és az ügyfelek jobb megértése érdekében.

3.4.4.1 Big Data

„Big Data analitika nélkül a vállalatok vakok és süketek, és úgy bolyonganak az interneten, mint szarvasok az autópályán.”

(Geoffrey Moore)



3.3. ábra: A nagyméretű adatok forrásai
Forrás: https://www.sap.com

A **Big Data** az adatforrások és formátumok széles skáláját öleli fel, beleértve a videókat (pl. biztonsági kamerák felvételei, promóciós videók, közösségi média videók), hangfelvételeket (pl. ügyfélhívások, hangutasítások, podcastok), képfájlokat (pl. fényképek, diagramok, képernyőképek, dokumentumszkennelések), szöveges adatokat (pl. jelentések, közösségi média bejegyzések, e-mailek, ügyfélértékelések, dokumentumok) és még sok mást.

Bármennyire is elképzelhetetlennek tűnik ma, az Apollo irányító számítógépe kevesebb, mint 80 kilobájt memóriával vitte a Holdra az első űrhajót. Azóta a számítástechnika exponenciális ütemben fejlődött - és vele együtt az adatgenerálás is. Valójában a világ technológiai adattárolási kapacitása az 1980-as évek óta körülbelül háromévente megduplázódott. A Statista (https://www.statista.com/statistics/871513/worldwide-data-

created/) becslése szerint 2020-ban 64,2 ZB adatot hoztak létre vagy sokszorosítottak (1 ZB = 1 000 000 000 000 000 000 000 bájt, azaz 10^{21} bájt). A mesterséges intelligencia (AI), a gépi tanulás, de a modern adatbázis-technológiák is lehetővé teszik a Big Data vizualizálását és elemzését, hogy valós időben használható meglátásokat nyújtsanak. A Big Data analitika segít a vállalatoknak az adataik alakításában, új lehetőségek felismerésében és üzleti modellek kialakításában. A BigData kifejezés olyan rendkívül nagy adatmennyiségekre utal, amelyek túl nagyok, változatosak vagy összetettek ahhoz, hogy a hagyományos adatbázis-eszközök hatékonyan feldolgozhassák őket. Valójában, a múltban a Big Data-t többnyire batch feldolgozással (nagy adattömegek egyidejű feldolgozása) dolgozták fel, ami nem tette lehetővé a valós idejű feldolgozást. Ma azonban a fejlett technológiáknak és az új platformoknak köszönhetően lehetővé vált a Big Data valós idejű vagy közel valós idejű feldolgozása. Ehhez különböző megoldásokat és eszközöket használnak, például elosztott feldolgozást felhasználva, mint az Apache Spark és az Apache Flink, streaming platformokat, mint az Apache Kafka vagy az Amazon Kinesis, vagy az In-Memory Computing-et.
Az adatokat generáló eszközök köre fenomenális ütemben növekszik - a műholdaktól a drónokon át a kenyérpirítókig. A kategorizálás céljából azonban az adatforrások nagyjából három típusba sorolhatók:

- A **közösségi médiaadatok** a közösségi média kommentek, posztok, képek és egyre inkább videók által generált adatok.
- **Gépi adatok**. Az IoT-eszközök és -gépek érzékelőkkel vannak felszerelve, és képesek digitális adatokat küldeni és fogadni. Az IoT-érzékelők segítségével a vállalatok összegyűjthetik és feldolgozhatják az eszközök, járművek és berendezések gépi adatait a vállalat egész területén. Az IDC becslése szerint 2025-re több mint 40 milliárd IoT-eszköz lesz a Földön, amelyek a világ összes digitális adatának közel felét generálják.
- A **tranzakciós adatok** a világ leggyorsabban mozgó és növekvő adatai közé tartoznak. Egy nagy nemzetközi kiskereskedelmi vállalat például óránként több mint egymillió vásárlói tranzakciót dolgoz fel. Ha pedig a világ összes vásárlási és banki tranzakciójára gondolunk, képet kapunk a keletkező adatok elképesztő mennyiségéről. Ráadásul a tranzakciós adatok egyre inkább félig strukturált adatokból állnak, beleértve a képeket és a megjegyzéseket is.

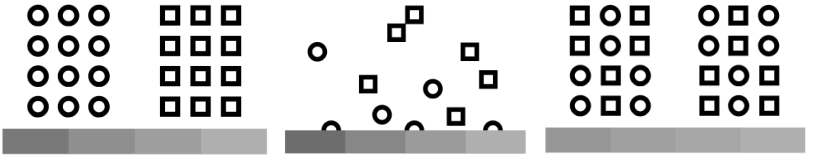
A vállalati információs rendszerekben a fent említett adatokat a vállalat elemezheti a döntéshozatal javítása, a folyamatok optimalizálása vagy a

piac és az ügyfelek igényeinek jobb megértése érdekében. A Big Data fogalmát az **5V** definiálja: Volume, Velocity, Variety, Varasity és Value.

- **Volume**: a volumen a vállalaton belül, a vállalat működésében, de a vállalat által feldolgozott külső adatok exponenciális növekedésének problémáját jelenti.
- **Velocity**: a sebesség nemcsak az adatok keletkezési oldalán jelent gyorsulási problémát, hanem a feldolgozás sebességigényének növelésében is, hogy az eredmények közel álljanak a valós idejű feldolgozáshoz.
- **Variety**: különböző adat formátumok és fájl formátumok problémáját írják le.
- **Varasity**: Ez az adatok hitelességével kapcsolatos problémát jelent, ahol azt kell feltételeznünk, hogy sok, ha nem a legtöbb adat önmagában pontatlan.
- **Value**: az adatok értékének meghatározási problémáját jelenti. Pl. egy adott érték kiválasztása egy nagy mennyiségű (akár pontatlan) adatból.

Több szerző is, (mint pl. Pokorný, 2014) további **3V** kifejezéssel szélesíti a Big Data fogalmát. A többi „V" kiegészítésként a bonyolultabb Big Data-khoz javasolt és azok speciális felhasználási eseteihez vagy adatproblémáihoz:

- **Vizualization**: az adatok megfelelő vizualizálási reprezentációjának problémái.
- **Variability**: azt a problémacsoportot képviseli, amikor a feldolgozott adatok esetében számítanunk kell azzal, hogy az adathalmaz adatai különböző kontextusokban állhatnak.
- **Volatility**: problémák az adatok érvényességével.



3.4. ábra: Strukturált, strukturáltalan és félig strukturált adatok.
Forrás: https://www.sap.com

Az adatkészleteket jellemzően az alábbi három típusba sorolják a szerkezetük, ill. az alapján, hogy mennyire könnyű indexelni őket:

- **Strukturált adatok**: az ilyen típusú adatokat a legkönnyebb rendszerezni és keresni. Ide tartozhatnak pénzügyi adatok, gépi naplók és demográfiai adatok. Az előre meghatározott oszlopokból

és sorokból álló Excel táblázatkezelő jó módszer a strukturált adatok bemutatására.

- **Strukturáltalan adatok**: az adatok ezen kategóriája magában foglalhatja például a közösségi médián közzétett bejegyzéseket, hangfájlokat, képeket és nyílt ügyfélmegjegyzéseket. Az ilyen típusú adatokat nem lehet könnyen rögzíteni a sorokból és oszlopokból álló, szabványos relációs adatbázisokban. Hagyományosan, nagy mennyiségű strukturálatlan adatot keresni, kezelni vagy elemezni kívánó vállalatoknak fáradságos kézi folyamatokat kellett alkalmazniuk. A táblázatok vagy relációs adatbázisok helyett a strukturálatlan adatokat jellemzően adattárházakban és NoSQL-adatbázisokban tárolják.
- **Félig strukturált adatok**: A félig strukturált adatok a strukturált és a strukturálatlan adatok hibridjei. Az e-mailek jó példák erre, mivel az üzenet szövegében strukturálatlan adatokat, valamint olyan szervezettebb tulajdonságokat tartalmaznak, mint a feladó, a címzett, a tárgy és a dátum.

A strukturált adatok leginkább az SQL-adatbázisok számára alkalmasak, míg a strukturálatlan adatok gyakran olyan NoSQL tárolókban kötnek ki, mint a MongoDB vagy a Cassandra.

A nagy mennyiségű adatnak nem sok gyakorlati haszna lenne a rendszerezéshez és elemzéshez szükséges mesterséges intelligencia nélkül. Ahogy Brandon Purcell, a Forrester Research elemzője fogalmaz: „*Az adat a mesterséges intelligencia éltető eleme. Ahhoz, hogy egy mesterséges intelligencia rendszer teljesíteni tudja a feladatát, adatokból kell tanulnia*”. A mesterséges intelligencia hatékony feldolgozást tesz lehetővé, különösen a komplex Big Data környezetekben, amelyekben a gépi tanulási modellek gyorsabban tanulnak a nagyobb és különfélébb adathalmazokból. A gépi tanulás a Big Data-ban virágzik, mert minél robusztusabbak az elemzett adathalmazok, annál nagyobb a rendszer lehetősége a tanulásra, a folyamatos fejlődésre és a folyamatok adaptálására. (Gála, L.; Pour, J.; Šedivá, Z. 2015) (Sándor, J. 2011) (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P., 2017) (Pavlíček, A.; Syrovátková, J. 2022) (CHLAPEK, D. et al.; 2014) (SAP Slovakia. 2024)

> *Például a videofeldolgozás során több információ is kinyerhető. Ezek lehetnek egyszerű adatok vagy metaadatok, mint például az egy adott üzletet meglátogató vásárlók száma, a látogatásuk ideje és időtartama. De a videót elemezhetjük például mozgásminták segítségével is, ahol nyomon követhetjük, hogyan mozognak a vásárlók az üzletben, az üzlet mely részeit látogatják leggyakrabban. A fejlettebb elemzések az arc- és tárgyfelismerés*

alapján azonosíthatnak tárgyakat vagy személyeket. Egyes fejlett rendszerek képesek elemezni a vásárlók arckifejezéseit, hogy meghatározzák az érzelmi reakcióikat (pl. elégedettség vagy csalódottság). Ezek az adatok betekintést nyújtanak a vásárlói elégedettségbe. Az összegyűjtött adatok más információkkal kombinálva átfogó képet adnak az ügyfelekről.

3.4.4.2 Open Data

Az **Open Data** olyan adat, amely nyilvánosan hozzáférhető, felhasználható és megosztható. Ezeket az adatokat általában kormányok, nonprofit szervezetek vagy más intézmények teszik közzé, és tartalmazhatnak például földrajzi adatokat, demográfiai adatokat, közlekedési vagy környezeti információkat. A vállalati információs rendszerekben a nyílt adatok olyan kiegészítő adatokat szolgáltathatnak, amelyeket a vállalkozások felhasználhatnak belső adataik elemzésére, előrejelzésére és gazdagítására. Egy logisztikai vállalat például a forgalomra vonatkozó nyílt adatokat felhasználhatja útvonalainak optimalizálásához. A Big Data jellemzőkre rávilágítást nyújtanak a tulajdonságaik, melyek Gála L.; Pour J.; Šedivá, Z. (2015) szerint:

- Teljes körűek - az adatokat a lehető legnagyobb mértékben közzéteszik.
- Elsődlegesek (eredetiek) - a szerző által közzétett adatok.
- Indokolatlan késedelem nélkül közzéteszik.
- Könnyen hozzáférhetőek - az adatok hozzáférhetőek az általános IKT-eszközökkel.
- Gépileg olvashatóak - az adatok olyan formátumban vannak, amelyek úgy vannak strukturálva, hogy egy szoftveralkalmazás képes a kívánt (kiválasztott) adatokat kiolvasni belőlük.
- Korlátozásmentesen hozzáférhetőek - az adatok olyan módon hozzáférhetők, amely nem jelent hátrányos megkülönböztetést egy személy vagy személyek csoportja számára.
- Szabadon hozzáférhető specifikációkkal rendelkező szabványokat használnak (nyílt szabványok) - az adatoknak olyan formátumban kell lenniük, amely szabadon hozzáférhető a felhasználásra.
- Az adatok használatára vonatkozó, világosan meghatározott feltételek (licenc) mellett, minimális korlátozásokkal hozzáférhetővé vannak téve.
- Állandóan elérhető - az adatok az adatszolgáltató által meghatározott ideig online elérhetőek.
- A felhasználók számára a lehető legalacsonyabb költséggel hozzáférhetőek.

A nyílt adatok nagyon hasznosak lehetnek a vállalkozások számára, és számos területen alkalmazhatók, mivel hozzáférést biztosítanak a nyilvánosan elérhető adatokhoz, és kiegészíthetik a belső vállalati adatokat. A nyílt adatok fontosságát talán úgy érthetjük meg a legjobban, ha megpróbáljuk megválaszolni a következő kérdést: Miért kell olyan adatok megszerzésének fáradtságába és költségeibe fektetni, amelyeket valaki már megszerezte, és csak el kell vinni? Pl. demográfiai adatok, meteorológiai adatok, makro ökonómiai mutatók (mint például kamatlábak, inflációs ráták vagy GDP) vagy statisztikai és népességi adatok, stb. (Chlapek, D. et al.; 2014) (Gála L.; Pour J.; Šedivá, Z. 2015)

3.5 Alkalmazások a kompetitív intelligencia támogatására

A kompetitív intelligencia (CI – *Competitive Intelligence*) ill. versenyképes hírszerzés – az ipari kémkedéssel ellentétben – egy etikus és legális tevékenység, amely nyilvánosan elérhető információforrásokat használ fel a vállalat versenyképességének növelésére. Fő célja a döntéshozatal támogatása, a piacelemzés, valamint a jelenlegi és jövőbeli kockázatok és lehetőségek azonosítása egy folyamatos és szisztematikus folyamatban. A kompetitív intelligencia két fő megközelítése:

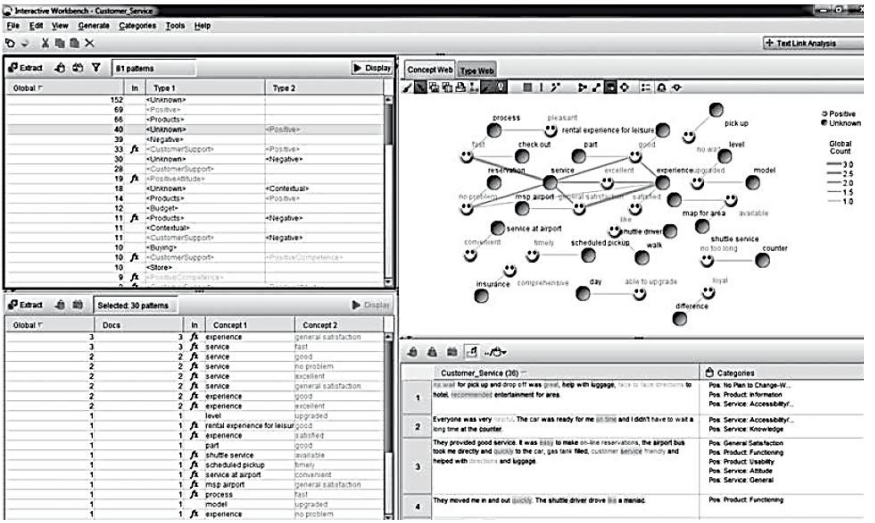
- **Aktív CI** – A kompetitív intelligenciát végző személy vagy szervezet, amely az információkat döntéshozatalának támogatására használja.
- **Passzív CI** – Az a hozzáállás, amelyben egy szervezet tudatában van annak, hogy versenytársai versenyintelligenciát alkalmaznak vele szemben, és védekezni kíván ezek ellen.

A passzív CI kulcsfontosságú eszköze az információbiztonsági menedzsment, amely segít megvédeni a vállalat érzékeny adatait. Ugyanakkor figyelembe kell venni, hogy a piaci szabályozás meghatározott közzétételi kötelezettségeket ír elő a vállalatok számára. Az aktív CI kulcsfontosságú tevékenységei olyan információk megszerzésére irányulnak, amelyek lehetővé teszik a menedzsment számára, hogy megalapozott, stratégiai döntéseket hozzon.

A vezetőség követelményei **feladatokká** alakulnak át, amelyeket egy kulcsfontosságú információs terület (key information topic) képvisel. Ezen a területen belül kulcsfontosságú kérdések (key information question) kerülnek meghatározásra, pl.: Mely piacokra kíván belépni versenytársunk?

A kérdések azonosításához különböző módszereket alkalmaznak, például döntési fákat, gondolattérképeket stb. A meghatározott feladatok alapján **információgyűjtés** történik a nyilvánosan elérhető információs térből. Az információgyűjtés (néha harvesting-nek is nevezik) során különböző keresőmotorokat használnak – nemcsak nyilvános, hanem elsősorban fizetős forrásokat is. Az így nyert adatokat letöltik és tárolják az elemzéshez. A begyűjtött adatokat **elemzésnek** vetik alá, amelynek célja a megfogalmazott kérdések megválaszolása. Az elemzés négy fő típusa Molnár, Z. (2012) szerint:

- **Kontextuális elemzés** – lehetővé teszi az adott dokumentumok közötti összefüggések azonosítását.
- **Tartalomelemzés** – statisztikai módszerek segítségével kulcskifejezések és azok kapcsolatainak feltárása.
- **Kapcsolati elemzés** – az adatok közötti összefüggések azonosítása és vizuális megjelenítése, például események, helyszínek, szervezetek vagy személyek kapcsán.
- **Időbeli elemzés** – az adatok időrendi sorrendbe állítása, amely lehetővé teszi az események közötti időbeli összefüggések és fejlődési mintázatok megértését.



3.5. ábra: Szövegelemzési eredmények megjelenítése
Forrás: (Gála, L.; Pour, J.; Šedivá, Z. 2015)

A kompetitív intelligencia (CI) elemzési eszközei közé tartoznak például: Tovek Tools, IBM eszközök (SPSS Modeler, i2), Oracle Endeca Information Discovery, SAS Institute (SAS Scoring, SAS Miner), stb.

Az elemzés következtetéseit a döntéshozók számára **terjesztik**, és azok támogatást nyújtanak számukra a döntéshozatali folyamatban. Az eredmények általában informatikai megoldásokon keresztül kerülnek megosztásra, például: e-mail (elektronikus levelezés), dokumentumkezelő és tartalomkezelő rendszerek (DMS, CMS) vagy portálmegoldások. A CI eszközök vállalati alkalmazása jelentős hatást gyakorol a stratégiai döntéshozatalra, mivel segítségükkel megalapozott elemzések készíthetők. Ezek az eszközök támogatják a különböző stratégiai elemzési módszereket, például: SWOT-analízis, SCP-keret, ADL-mátrix vagy PEST(EL)-elemzés, stb. (MOLNÁR, Z. 2012) (Koloszár, L. 2013)

Ebben a fejezetben használt néhány kifejezés magyarázata

*Az **AJAX** egy olyan technológia, amely lehetővé teszi a weboldal egy részének (például űrlap, terméklista vagy térkép) frissítését anélkül, hogy a teljes oldalt újra be kellene tölteni. Ez gyorsabb és gördülékenyebb interakciót jelent a felhasználók számára. Segít a dinamikus webalkalmazások létrehozásában, ahol például szűrhet termékeket, jelentéseket írhat vagy kereshet anélkül, hogy folyamatosan várnia kellene a teljes oldal betöltésére. Amikor keres valamit a Google-n, miközben beírja a keresőkifejezést, a Google automatikusan javaslatokat kínál, anélkül, hogy a „Keresés" gombra kellene kattintania, és meg kellene várnia az új oldal betöltődését. Az AJAX a háttérben működik, hogy gyorsan betöltse ezeket a terveket a teljes oldal frissítése nélkül.*

***Blokklánc (Blockchain)** egy decentralizált technológia, amely biztonságos adatmegőrzést és átlátható tranzakciókezelést tesz lehetővé. Ez az elosztott adatbázis olyan blokkokból áll, amelyek egymással kriptográfiai módon vannak összekötve. Minden blokk tartalmazza az előző blokkra vonatkozó hivatkozást, így láncot alkotva. Ezek a blokkok nagyszámú számítógép (csomópont) között vannak elosztva, akár világszerte. Minden csomópont rendelkezik a teljes blokklánc-adatbázis azonos másolatával. A blokkláncban tárolt adatok megváltoztathatatlanok: egy blokk módosításához az összes következő blokkot is meg kellene változtatni, ami rendkívül nehéz a csomópontok többségének beleegyezése nélkül. A blokklánc-technológiát eredetileg kriptovalutákhoz fejlesztették ki, de ma már széleskörűen alkalmazzák különböző területeken, mint például a pénzügy, logisztika és ellátási lánc menedzsment. Fő előnyei közé tartozik a biztonság, az átláthatóság és a decentralizáció.*

*A **cookie** olyan eszköz, amellyel a weboldalak információkat tárolnak a felhasználói munkamenetekről, javítják az élményt (például az e-shopokban tárolják a kosár tartalmát). Ez a Web 2.0 elengedhetetlen eleme. A cookie-kat gyakran használják a felhasználók internetes viselkedésének nyomon követésére, lehetővé téve a célzott hirdetések megjelenítését – ez az általános gyakorlat, amely a Web 2.0 korszakában vált népszerűvé. Így a Web 2.0 népszerűsítette azokat a gyakorlatokat, amelyek különböző weboldalakon alkalmaznak felhasználókövetést, aminek következtében a cookie-k szigorúbb szabályozás alá kerültek, mint például a GDPR és az elektronikus adatvédelmi irányelv az Európai Unióban. A Web 3.0 korszakában a cookie-k fokozatosan kevésbé dominánssá válhatnak, különösen a decentralizációt célzó technológiák alkalmazásával.*

*A **decentralizált alkalmazások (DApps)** olyan szoftverek, amelyek decentralizált hálózaton, általában blokklánc-technológiára építve működnek. Ellentétben a hagyományos központosított alkalmazásokkal, amelyek egy vagy több központi szerveren futnak (az adatok kereshetők, és fennáll a törlés, módosítás vagy elvesztés veszélye), a DApps-ek elosztott csomópontok hálózatán futnak, ahol a felhasználók közvetlenül kommunikálnak egymással. Az adatok többszörösen tárolódnak a hálózat különböző pontjain, ami biztosítja az adatok megbízhatóságát és ellenállását manipulációval szemben. Az **adatreplikáció** mechanizmus biztosítja, hogy minden csomópont megkapja és tárolja az adatokat. Amikor új tranzakció történik, azt minden csomópont validálja egy konszenzusmechanizmuson (pl. Proof of Work vagy Proof of Stake) keresztül. Az ellenőrzés után a tranzakció bekerül a blokkláncra, és frissül az összes csomópont adatbázisában.*

4 Információs és kommunikációs technológiák, adatgyűjtés és feldolgozás

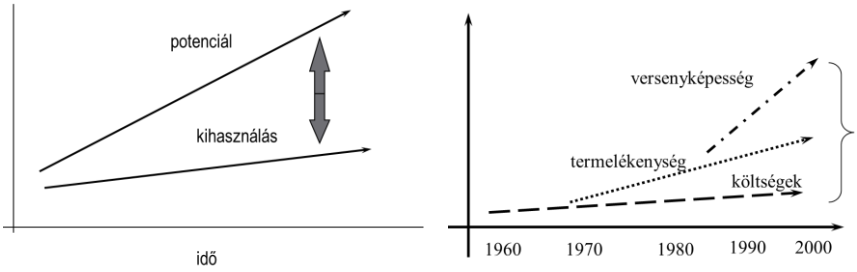
Az OECD (Gazdasági Együttműködési és Fejlesztési Szervezet) a következőképpen határozza meg az IKT-termékek fogalmát: IKT-termékek azok, amelyek fő feladata az információ elektronikus úton történő közlésének vagy feldolgozásának lehetővé tétele vagy megkönnyítése, beleértve annak továbbítását és megjelenítését (OECD, 2009).

4.1. táblázat: IT termékkategóriák – Gála L.; Pour J.; és Šedivá Z. (2011) által módosított, OECD alapján.

Kategória neve	A CPC alosztályok száma	Példák termékekre az alosztályokban
Számítógépek és perifériák	19	Számítógépek, nyomtatók, szkennerek, stb.
Kommunikációs berendezések	8	Telefonkészülékek, beleértve a mobil, tűzjelző rendszerek, épületbiztonsági rendszerek, stb.
Fogyasztói elektronikai berendezések	11	Játékkonzolok, kamerák, televíziókészülékek, stb.
IKT alkatrészek és komponensek (máshol nem említett)	14	Mágneses vagy optikai adathordozók, integrált elektronikus áramkörö,k stb.
IKT berendezések gyártása	5	Számítógépek, távközlési berendezések gyártása, stb.
Szoftver	11	Operációs rendszer, adatbázisrendszer, fejlesztőeszközök, vállalati alkalmazások, stb.
Informatikai tanácsadás és szolgáltatás	10	Informatikai tanácsadás, alkalmazási szolgáltatások nyújtása, számítógépes rendszerirányítási szolgáltatások, stb.
Távközlési szolgáltatások	12	Hangszolgáltatások, adatszolgáltatások, stb.
IKT-berendezések bérbeadása	3	Számítógépek bérbeadása, távközlési berendezések bérbeadása stb.
Egyéb IKT-szolgáltatások	6	Számítógép-telepítési szolgáltatások, számítógép-karbantartás és –javítás, stb.

A termékek lehetnek áruk vagy szolgáltatások. Az ENSZ közös termékosztályozási rendszere (CPC – Central Product Classification) szerint 99 alosztályra oszthatók. Gála L.; Pour J.; és Šedivá Z. által módosított, saját csoportosításukat a 4.1. táblázat mutatja be.

Az információs technológiával (IT) összefüggésben a **technológia** kifejezetten az információ feldolgozására használt eszközökre és módszerekre utal. A számítógépek által vezérelt információs rendszereket az információk megszerzésére, feldolgozására, tárolására, módosítására és továbbítására használjuk. Ha ezen folyamatok végrehajtására technológiát alkalmazunk (szoftvereket – módszerek, programok és a hardvereket – eszközök, berendezések), akkor **információs technológiáról (IT)** beszélünk. Az **információs és kommunikációs technológia (IKT)** kifejezés széles körben használatos, és magában foglalja mind az informatikai eszközöket, mind a kommunikációs technológiákat, például a telekommunikációt, hálózatokat, internetes infrastruktúrát, valamint a mobil és vezetékes kommunikációs rendszereket is. Az IKT jelentősége különösen abban rejlik, hogy lehetővé teszi a hatékony adatcserét, a digitális szolgáltatások fejlesztését és az információk gyors elérését globális szinten. (Gála L.; Pour J.; Šedivá, Z. 2015)



4.1. ábra: Az IT potenciálja és kihasználása

4.1 Hardver

Ebben a fejezetben azon eszközök és berendezéseket mutatjuk be, melyeket az információ feldolgozására és tárolására szolgálnak.

4.1.1 A számítógépek típusai és alkalmazási területeik

Manapság már mindenki az igényeinek (és persze az anyagi lehetőségeinek) megfelelően vásárolhat számítógépet, laptopot vagy tabletet. De nem csak személyi számítógépeket ismerünk (Personal

Computer - PC), amelyekkel a leggyakrabban találkozunk, hanem több kategóriáját is megkülönböztetjük a számítógépeknek. A következőkben a számítógépek rövid áttekintésére összpontosítunk, amelyeket méret (és teljesítmény) szerint rangsorolunk a legkisebbtől a legnagyobbig.

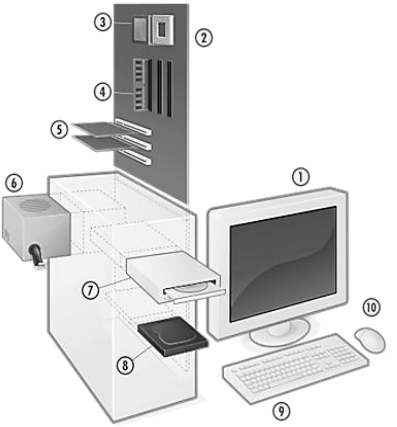
- **Zsebszámítógép (PDA) és „okos" telefon**
  - A zsebszámítógépeket elsősorban időgazdálkodásra használják, de gyakran rendelkeznek az asztali számítógépek funkcióival, és használatukat elsősorban a kis kijelző és a kis billentyűzet korlátozza. Az angol PDA - *Portable Digital Assistant* - elnevezés jól jellemzi funkciójukat. A nagyobb készülékek elsősorban zsebszámítógépként szolgálnak telefonálási lehetőséggel, az „okos" telefonok (smartphone) elsősorban mobiltelefonként működnek, alapvető PDA-funkciók lehetőségével. Muszáj elmondanunk, hogy manapság a PDA-kat már alig használják, mivel funkcióikat teljesen átvették az okostelefonok. A modern okostelefonok nemcsak PDA funkciókkal rendelkeznek, hanem számos más olyan tevékenységet is képesek elvégezni, amelyeket korábban személyi számítógépeken végeztek.
- **Multimédia táblagép**
  - Az Apple új típusú számítógépet dobott piacra az iPaddel. Billentyűzet nélküli érintőképernyővel, hordozásra alkalmas mérettel és kiváló megjelenítési, hálózati, multimédiás funkciókkal rendelkezik. Ma a piac tele van különböző gyártók (pl. Samsung, Microsoft, stb.) különböző modelljeivel, amelyek hasonló funkciókat kínálnak. A táblagépek multimédiás funkcióit ma már nem csak szórakozásra, hanem munkára és oktatásra is széles körben használják.
- **Laptop és netbook**
  - A számítógépek fejlődése során jelentősen nőtt a teljesítményük és csökkent a méretük. Napjainkban a laptopok egyre inkább A4-es noteszgép (notebook) méretűek. Teljesítményüket tekintve közel állnak asztali társaikhoz, de a korlátok elsősorban ergonómiai jellegűek (kis méret, kényelmes gépelés, egér és kijelző helyett touchpad vezérlés). Emellett nehezebben bővíthetők további alkatrészekkel. A netbook kicsi (általában 9-10 hüvelyk képernyőmértű), notebookhoz hasonló kialakítású számítógép, melynek teljesítménye internetezésre elegendő, de nem lehet rajtuk erőforrás-igényesebb

programokat pl. modern videójátékokat futtatni. A netbookok sem olyan elterjedtek már, mint egykor. Helyüket az ultrabookok és a Chromebookok vették át, amelyek még mindig kicsik és könnyűek, de jobb teljesítményt nyújtanak, mint a régebbi netbookok, de mind ebbe a kategóriába tartoznak.

- **Személyi számítógép (PC – Personal Computer)**
  - A személyi számítógépek ma már nagyon hasonlóak, nemcsak a vállalkozásokban és az iskolákban, hanem az otthonokban is. Vannak szerzők, akik úgy gondolják, hogy az otthoni számítógépek, inkább a zene és a videó felé fognak orientálódni. A személyi számítógépek ma már sokféle formátumban kaphatók, például klasszikus asztali verziók (desktop), valamint kompakt mini számítógépek, all-in-one eszközök és játék PC-k.
- **Munkaállomás**
  - Munkaállomásnak nevezik a személyi számítógépek legnagyobb teljesítményű csoportját. Nagy számítási és grafikai teljesítményük van, és ebből kiindulva magas az áruk is. Jellemzően számításigényes alkalmazásokhoz használják őket, például animációk készítéséhez, hatalmas adatok feldolgozásához, fizikai és kémiai jelenségek szimulációjához, valamint egyéb tudományos és műszaki alkalmazásokhoz. A munkaállomás-ágazatban is léteznek speciális típusok a különböző feladatokra (grafikai, tudományos, mérnöki stb.).
- **Nagyszámítógép (mainframe)**
  - A nagyvállalati hálózatok központjainak nagy teljesítményre, megbízhatóságra és hatalmas lemezkapacitásra van szükségük. Az asztali számítógépeknek és terminálóknak hálózati szerverként szolgálnak (ezeket a fogalmakat a hálózatokról szóló fejezetben közelebb ismertetjük). A nagyszámítógépek egyre fontosabb szerepet töltenek be a banki, biztosítási és egyéb iparágakban is, ahol a nagy mennyiségű tranzakció és adat feldolgozásához elengedhetetlenek.
- **Szuperszámítógép**
  - A több száz vagy több ezer processzor kombinálása óriási számítási teljesítményt ad a szuperszámítógépeknek. Ezeket katonai célokra, igényes szimulációkra (pl. a hosszú távú időjárás-előrejelzés az adatok nagyságrendje

miatt nagy számítási teljesítményt igényel) és tudományos számításokra használják. A szuperszámítógépeket ma már olyan területeken is használják, mint a mesterséges intelligencia, a gyógyszerfejlesztés és a már említett éghajlati modellezés. A legnagyobb teljesítményű számítógépek rangsorát évente frissítik a következő címen: www.top500.org.

A számítógépek osztályozása manapság nem könnyű. Végezzünk még egy osztályozást a számítógépek felhasználása szerint, ami különösen a gazdasági területeken praktikus. A **személyi számítógépek** (PC, táblagép, okostelefon, laptop, stb.) a felhasználói kéréseket dolgozzák fel, vagy más, leggyakrabban üzleti számítógépek feldolgozási eredményeit jelenítik meg. A személyi számítógépek **hordozható** és **nem hordozható** számítógépekre oszthatók. Az egyértelműség miatt nem szükséges példákat mondani. A **szolgálati számítógépek** (szerverek) a személyi vagy más szolgálati számítógépek kérései alapján végzik tevékenységüket, ami alatt azt értjük, hogy a felhasználó továbbítja számukra a feldolgozási kéréseket. Mivel nagyszámú kérést kell kezelniük, és folyamatosan kell feldolgozniuk, szükséges, hogy műszaki és technológiai kialakításuk ezt támogassa. A szükséges teljesítményt elsősorban több önálló szolgáltató számítógép nagyobb egységbe, ún. számítógép-klaszterekbe történő egyesítésével érik el. Ennek a számítógép-csoportnak egy sajátos kategóriája a szuperszámítógépek. Harmadik kategóriába tartoznak azok az **eszközök, ahol a számítógép** (vagy a megfelelő integrált áramkörök összetétele), **biztosítja a speciális funkciót**. Általában a fogyasztói elektronikába implementáljuk őket. Ilyenek például a kamerák, televíziók, a gépjárművek fedélzeti számítógépe, a hálózati routerek és egyéb hálózati berendezések eszközei, monitorok, a gyártógépek berendezései stb. Az egyik jelentős trend, amely ebbe a



4.2. ábra: Asztali számítógép általános felépítése. 1-képernyő, 2-alaplap, 3-Processzor és foglalata, 4-RAM memória és RAM foglalatok, 5-bővítőkártyák és foglalataik, pl. hangkártya, hálózati kártya (grafikus kártya AGP vagy PCIe – PCI Express foglalatban csatoljuk), 6-táp, 7-CD/DVD mechanika, 8-HDD, 9-billentyűzetet, 10-egér.
Forrás: https://es.wikipedia.org/wiki/Hardware

kategóriába tartozik, a dolgok internete (IoT – Internet of Things). Elmondhatjuk, hogy ez az eszközök / tárgyak / emberek internethez való csatlakoztatására szolgáló címke. (Szabó, B.; Márföldi, E. 2011) (Herdon, M. 2016) (Juhász, S. 2011) (Berecz; A.; Karácsony, P.; Kónya, L.; Peck T, T.; Szász, G.; Vári-Kakas, I. 2019)

4.1.2 A számítógépek legfontosabb egységei

A következő fejezetekben egy általános számítógép legfontosabb részeit (kivéve a ki és bemeneti eszközöket) mutatnánk be.

- **Processzor – CPU** (Central Procesing Unit): végrehajtja a programok által tartalmazott összes utasítást. Ez a számítógép "agya". Az utasítás-halmaz, amelyet a processzor képes végrehajtani, utasításkészletnek nevezzük. Minden processzor-típusnak saját utasításkészlete van. Ennek megfelelően a processzorokat teljes- és redukált utasításkészlettel rendelkező processzorokra osztják. A gépi kód a számítógép processzora által közvetlenül végrehajtható utasítások halmaza. A gépi kód úgy jön létre, hogy egy programozási nyelven írt programot lefordítunk a processzor által közvetlenül végrehajtható utasításokra. A processzor három részből áll: (1) aritmetikai és logikai egység, amely matematikai és logikai műveleteket hajt végre, (2) vezérlő egység, amely a program szerint vezérli a végrehajtott műveletek sorrendjét és (3) a regiszterek, amelyek nagyon gyors hozzáférésű memóriahelyek a processzoron belül – olyan helyek, ahol a processzor bizonyos alapbeállításai is tárolva vannak. Az alaplapon van egy aljzat a processzor számára – egy csatlakozó, amelybe a processzor be van helyezve. A processzor nagy mennyiségű hőt termel, ezért működés közben folyamatosan hűteni kell. A processzor végrehajtja a szoftver utasításait, és lehetővé teszi a teljes számítógép működését. Alapparaméterei között szerepel a szószélesség - jelenleg 32 vagy 64 bit, a frekvencia - jelenleg kb. 2–6 GHz és a magok száma - jelenleg átlagosan 4–8. Minden mag több szálat (threads) dolgozhat fel, ami növeli a párhuzamosságot. Meg kell említenünk, hogy a magasabb frekvencia nem mindig jelent nagyobb teljesítményt, mert a teljesítmény a processzor architektúrájától is függ. A modern processzorok alacsony frekvencián is nagy teljesítményt érhetnek el, mert az utasítások párhuzamos feldolgozására vannak optimalizálva (például a többmagos megoldásoknak, a szélesebb "pipeline"-eknek vagy a

különféle optimalizálásoknak, mint például a "out-of-order"-nek köszönhetően). A modern processzorok dinamikusan tudják változtatni a frekvenciát a terhelésnek megfelelően. Például a processzor alacsonyabb frekvencián is működhet, hogy energiát takarítson meg, és szükség esetén növelje azt (ún. "turbo boost").

- **RAM (belső) memória**: a számítógép működése közben az adatok tárolására használt modul. Ez egy kis nyomtatott áramköri lap, amely memóriachipeket tartalmaz. Az alaplapon található foglalatba (csatlakozóba) helyezik. A processzor a memóriában dolgozik. Ebben végzi a processzor az összes műveletet, benne fut az összes futó szoftver és az operációs rendszer saját maga is. Fontos paraméterei a nagysága (GB-ban) és a frekvenciája (GHz-ben). Ez manapság kb. 4–32 GB és kb. 1 GHz. A jelenlegi memóriatípusok leggyakrabban a DDR4 vagy DDR5. A késleltetés (CAS-latency) a RAM teljesítményének következő fontos paramétere jellemzően a DDR4: 15 – 20, DDR5: 30 – 40 vagy több.

> *Pl.: A szövegszerkesztő egy olyan program, amellyel szöveges fájlokat hozunk létre és szerkesztünk. A szöveg tényleges létrehozása az operatív memóriában (RAM) történik, amely azonban nem tárolja az adatokat tartósan. A szövegszerkesztő bezárása előtt a létrehozott szöveges fájlt el kell menteni a merevlemezre (HDD vagy SSD), különben az adatok elveszhetnek a memória törlésével a program bezárásakor vagy a számítógép újraindításakor.*

- **Északi és déli híd**: az alaplapon vannak elhelyezve. Az északi híd felelős a közvetlen kommunikációért a processzor (CPU), a memória (RAM), a grafikus kártya (GPU) és a gyors, nagy adatátvitelt igénylő alkatrészek között. A mai modern rendszerekben az északi híd funkciói közvetlenül a processzorba integrálhatók. A déli híd kezeli a kommunikációt a lassabb perifériák és a CPU között. Felelős olyan eszközök csatlakoztatásáért és kezeléséért, mint a merevlemezek (HDD/SSD), USB-portok, hangkártyák, hálózati kártyák és egyéb bemeneti/kimeneti eszközök. Biztosítja a kevésbé kritikus komponensek és perifériák processzorhoz való csatlakoztatását, hídként szolgálva a nagy és a kis sebességű eszközök között. A déli híd még mindig jelen van, de egyes modern platformokon feladatai egyre inkább új, több funkciót kombináló lapkakészletekre (chipset) összpontosulnak.
- **Merevlemez – HDD** az adatokat hosszú ideig tárolja. A lemez mágneses elven működik. Minden, amit elmentünk rá, akkor is

rajta marad, ha a számítógépet kikapcsoljuk. Ma a mechanikus merevlemezt lassan felváltják a teljesen elektronikus **SSD meghajtók**. Fő paraméterük a kapacitás, amely jelenleg 500 GB és 2 TB között mozog. Az SSD-k sokkal gyorsabban olvasnak és írnak adatokat, mint a HDD-k, ezért népszerűek a modern számítógépekben, bár a HDD-k nagyobb kapacitást kínálnak alacsonyabb áron. Ezeket **külső memóriának** is nevezzük.

- **Grafikus kártya - GPU** képet készít mind normál kétdimenziós (2D) kijelzőn, mind háromdimenziós (3D) megjelenítésen (játékok). Ezen kívül képes felgyorsítani a videó fájlok lejátszását vagy párhuzamos számításos feladatok elvégzésére is alkalmas. Ebből kifolyólag a GPU-kat egyre gyakrabban használják a mesterséges intelligencia és a gépi tanulás területén is, mert nagy mennyiségű adat párhuzamos feldolgozására is képesek.
- A **hangkártya** átalakítja a hangot digitális számítógépes formából, vagy hangokat dolgozz fel, illetve rögzít mikrofon vagy más közvetített forrásokból. A legtöbb modern alaplap már rendelkezik integrált hangkártyával, így a diszkrét hangkártyákat főként a professzionális audio alkalmazásokban használják.
- A **hálózati kártya** lehetővé teszi a számítógép csatlakoztatását egy (helyi) számítógépes hálózathoz. (Szabó, B.; Márföldi, E. 2011) (Herdon, M. 2016) (Juhász, S. 2011) (Berecz; A.; Karácsony, P.; Kónya, L.; Peck T, T.; Szász, G.; Vári-Kakas, I. 2019)

4.1.3 Von-Neuman- és Harvard-architektúrák

Attól függően, hogy a processzor hogyan éri el a memóriát és dolgozza fel az utasításokat, a számítógépeket **Von-Neumann-architektúrájú** számítógépekre és **Harvard-architektúrájú** számítógépekre lehet osztani. A Von-Neumann-architektúra a következő Neumann-elvek alapján működik:

- Teljesen elektronikus működés (ez Neumann idejében elektroncsöves felépítést jelentett, amit később a tranzisztoros, majd az integrált áramkörös felépítés követett). A modern korban gyakran alkalmazzák a párhuzamos utasításkezelést is. Így lehetőség nyílik több utasítás egyidejű végrehajtására, ami javítja a hatékonyságot.
- Kettes számrendszer használata (az összes művelet, pl. összeadás, szorzás, kettes számrendszerbeli logikai műveletekre redukálható)

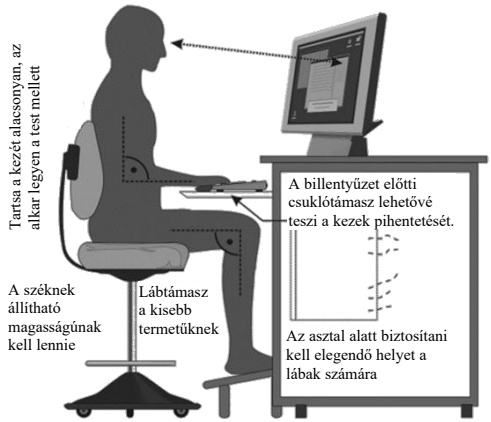
- Belső memória használata – egy memória az adatoknak és utasításoknak is.
- Tárolt program elve. A számításokhoz szükséges adatokat és programutasításokat a gép azonos módon, egyaránt a belső memóriában (operatív tár) tárolja.
- Soros utasítás-végrehajtás (az utasítások végrehajtása időben egymás után történik; ennek egy alternatívája a párhuzamos utasítás-végrehajtás, amikor több utasítás egyidejűleg is végrehajtható, – ezt a lehetőséget Neumann elvetette).
- Univerzális felhasználhatóság; Turing-gép (programozhatóság; a különböző feladatok programokkal legyenek megoldva, nem pedig erre a célra épített hardverrel). A mai számítógépek éppen ennek köszönhetően képesek univerzális problémák megoldására. A Von-Neumann-architektúra közvetlenül a Turing univerzális gép koncepciójából származik, amely megfelelő programozással bármilyen problémát meg tud oldani.
- Szerkezete öt funkcionális egységből áll (aritmetikai-logikai egység (ALU), központi vezérlőegység (CPU), memóriák, bemeneti és kimeneti egységek).

A tiszta Von-Neumann-architektúrájú processzor az utasításokat sorrendben hajtja végre. A modern processzorok azonban nemcsak a párhuzamos feldolgozást teszik lehetővé több mag vagy szál segítségével, hanem az utasításfeldolgozás több lépésre (fázisra) történő felosztását (pipelining) is, így növelve a processzor hatékonyságát és teljesítményét. A processzormag az a fizikai egység, amely tartalmazza az utasítások végrehajtásához szükséges összes **áramkört** (pl. dekóder, aritmetikai és logikai egység, regiszterek). Párhuzamosításnak egyik fajtája a szálak használása. A **logikai szál**[8] a processzor által elvégezhető feladat szoftveres reprezentációja. Egyetlen processzormag több szálat is végre tud hajtani

[8] A programban egy szál olyan, mint egy különálló feladat vagy utasításkészlet, amely más feladatokkal párhuzamosan hajtható végre. A szálak általában ugyanazon program különböző kódrészleteit hajtják végre, nem pedig ugyanazt az utasítást párhuzamosan. Egy programon belül több szál is dolgozhat. Például egy grafikus alkalmazásban az egyik szál a képernyő kirajzolásával, míg egy másik szál a felhasználói bemenet fogadásával foglalkozik. Ezek a szálak egymástól függetlenül futnak, de mindkettő ugyanannak a programnak a része. Egy másik példa lehet a böngészők világából: egyik szál lehet felelős egy weboldal megjelenítéséért, egy másik szál ezzel egyidejűleg az internetről való letöltéssel is foglalkozhat, a következő szál nyomon követheti a felhasználói interakciókat (pl. a linkekre való kattintást).

akár sorosan (egyenként), akár **multithreadingel** – párhuzamos feldolgozással. Ha például az egyik szál egy erőforrásra (például memóriára) vár, a másik szál használhatja azokat a processzor-erőforrásokat, amelyek egyébként kihasználatlanok lennének. Ismerünk hardveres threadinget (feldolgozást) is, mint pl. a Intel Hyper-Threading vagy AMD SMT (Simultaneous Multithreading).

A **pipelining** segítségével a processzor egyszerre több utasítást is feldolgozhat, minden utasítást más-más feldolgozási fázisban. Például egy utasítás végrehajtása több fázisra oszlik: (1) az utasítás lehívása a RAM-ból; (2) az utasítás dekódolása, hogy a processzor tudja, mit kell tennie; (3) a kívánt művelet végrehajtása; (4) az eredmény visszaírása a memóriába. A hagyományos feldolgozás során a processzornak egy utasítás esetében mindezen fázisokat be kellett fejeznie ahhoz, hogy a következő utasítás feldolgozását megkezdhesse. A pipelining azonban lehetővé teszi a processzor számára, hogy a második utasítást már az első dekódolása és a harmadik végrehajtása közben is lekérje. Ezáltal a processzor különböző részei folyamatosan aktívak maradnak, és hatékonyan kihasználhatók, ami az általános teljesítmény növekedéséhez vezet. Fontos tény, hogy az egyes fázisok feldolgozása a processzor különböző részein történik, így ezek a tevékenységek nem keresztezik egymást. A pipelining még egy egymagos processzoron is megvalósítható. A többmagos processzorok még nagyobb teljesítményt érnek el ilyen módon, mivel egyszerre több szálat is képesek végrehajtani. A pipelininget alkalmazó processzorok ma már alap-felszereltségnek számítanak a modern számítógépekben, okostelefonokban és szerverekben. Például a modern PC-k Intel- és AMD-processzorai fejlett pipelining-technikákat alkalmaznak, hogy növeljék a teljesítményt alkalmazások végrehajtása, többfeladatos munkavégzés és játék közben. A Von-Neumann-architektúrától eltérően a **Harvard-architektúra** különálló memóriával rendelkezik az utasítások és adatok



4.3. ábra: A számítógépes munkaállomás ergonómiai paraméterei
Forrás: (Roubal, P. 2012)

számára. Ez azt jelenti, hogy a processzor egyszerre érheti el az utasításokat és az adatokat, ami jelentősen növelheti a teljesítményt. Ezt az architektúrát leggyakrabba speciális rendszereknél használják, ahol fontos a feladatok gyors feldolgozása. A tiszta Von-Neumann-architektúrát a mai rendszerekben gyakran kombinálják modern optimalizációkkal. Mindkét architektúra elemeit számos modern processzor használja. Például a processzorok rendelkezhetnek cache memóriákkal (cache) az utasítások és adatok számára, így a Harvard architektúrához hasonló előnyöket érhetnek el, még akkor is, ha az alap architektúrában a Von-Neumann modellt követik. (Szabó, B.; Márföldi, E. 2011) (Herdon, M. 2016) (Juhász, S. 2011) (Berecz; A.; Karácsony, P.; Kónya, L.; Peck T, T.; Szász, G.; Vári-Kakas, I. 2019)

4.2 Szoftver

A szoftver a számítástechnikai eszközön használható alkalmazások (programok) összessége. A licencek elérhetősége szerint a szoftvereket a következő osztályokba sorolhatjuk:

- **Nyílt forráskódú szoftverek** (szabadon terjeszthetőek, amelyek forráskódot tartalmaznak a változtatások és fejlesztések miatt). Gyakran speciális licencek vonatkoznak rájuk (pl. GPL, Apache License), amelyek meghatározzák a terjesztés és a módosítás feltételeit.
- **Freeware** (ingyenes szoftverek). Általában nincs elérhető forráskódja, és nyílt forráskódú szoftverként nem módosítható.
- **Kereskedelmi szoftver** (fizetős) - csak akkor használható, ha a szoftver licencét megveszik, és nem sértik meg ezt. A licencelési modell különböző típusokat foglalhat magában (pl. egyszeri vásárlás, előfizetés vagy felhasználónkénti licencek).
- **Egyéb**: shareware, demo vagy trial verziók stb. (korlátozott ideig vagy más korlátozásokkal ingyenesen kipróbálható vagy használható).
- **Egyedi szoftverek**. Egy adott szervezet vagy felhasználó egyedi igényeire szabott szoftver.

Ezen kategóriákon kívül még ismerünk számos más kategóriákat is, mint pl. web (vagy internet) alkalmazások, rosszindulatú szoftverek, mint például számítógépes vírusok, malware, stb. A tárgyunk kapcsán alapvetően a szoftvereket három kategóriába oszthatjuk:

- Az **alapszoftver**, amely lehetővé teszi az alkalmazások használatát, és szolgáltatásokat biztosít számukra. Ilyen lehet pl. firmware (konkrétan BIOS (Basic Input/Output System) vagy modernebb változata UEFI (Unified Extensible Firmware Interface) vagy az operációs rendszer.
- **Alkalmazásszoftver**, amely alatt a vállalaton belüli összes alkalmazás összességét értjük. Pl. ERP, CMS, stb.
- Az alkalmazásszoftverek **fejlesztését támogató szoftver**.

Bizonyos kontextusokban ezek a kategóriák átfedhetik egymást. Például, egyes alkalmazásszoftverek beépített fejlesztőeszközökkel is rendelkezhetnek. (Szabó, B.; Márföldi, E. 2011) (Herdon, M. 2016) (Juhász, S. 2011) (Berecz; A.; Karácsony, P.; Kónya, L.; Peck T, T.; Szász, G.; Vári-Kakas, I. 2019)

4.2.1 Alapszoftver

Az alapszoftver az informatikai alkalmazások futtatásához szükséges szoftver. Ez a szoftver-kategória magába foglalja főként:

1. az **operációs rendszereket** (OS - Operating System), melyek az alkalmazások futtatását és a számítógép erőforrásainak kezelését végzik, és
2. olyan szoftvereket, melyek specifikus szolgáltatásokat nyújtanak az alkalmazások számára, vagy **kiegészítik az operációs rendszer funkcióit**, mint például a middleware, amely összekapcsolja az alkalmazásokat és az operációs rendszert, vagy az adatbázis rendszerek, amelyek az adatok kezelését biztosítják.

Az operációs rendszer olyan programok összessége, amely a számítógépen futó összes többi programot vezérli és összehangolja. Az operációs rendszer biztosítja a számítógép működéséhez szükséges vezérlési funkciók teljes körét, függetlenül attól, hogy milyen alkalmazási programok futnak rajta. Az operációs rendszer tehát nemcsak technikai alapot ad a számítógép működéséhez, hanem keretet is teremt a szoftverek biztonságos, megbízható és stabil működéséhez. Az operációs rendszer:

- Irányítja és kezeli a számítógép technikai erőforrásait, pl. kiosztja a CPU-időt az egyes programok számára, hozzáférést biztosít a beviteli és kimeneti eszközökhöz.
- Egy elkülönítő réteget hoz létre a programok számára, hogy azok különböző hardverekkel dolgozhassanak (azaz elválasztja a

hardvert az alkalmazásoktól). Hardver-absztrakciót is biztosít - így az alkalmazásoknak nem kell ismerniük a hardver pontos részleteit, amellyel dolgoznak (virtuális portok vagy virtuális eszközök).

- Speciális illesztő programok (úgyn. driver[9]) által kommunikál a technikai erőforrásokkal. Ezeket általában az eszköz gyártója fejleszti ki, és az adott eszközre „szabják". Ahhoz, hogy egy műszaki eszköz képességeit a lehető legjobban ki lehessen használni, fontos, hogy a driverek telepítve legyenek a számítógépre. (Pl. driver a nyomtatóhoz.)
- A fájlrendszeren[10] keresztül az operációs rendszer hozzáférést biztosít a lemezeken és egyéb külső számítógépes tárolókon tárolt fájlokhoz. A fájlok egyaránt tartalmazhatnak adatokat és egyéb szoftvereket.
- A felhasználók számára felhasználói felületet biztosít. A felhasználói felület elemeit az alkalmazások számára is biztosítja, azaz az alkalmazás vagy más szoftver egy adott operációs rendszer felhasználói felületének elveit használja.
- Az egyes programok feldolgozását kezeli, és egyidejű feldolgozás esetén meghatározza, hogy mely programok milyen sorrendben, mennyi időt és milyen prioritással kerülnek feldolgozásra. Ha a számítógép több processzorral rendelkezik, akkor a programok feldolgozását elosztja közöttük.
- Az operációs rendszer a hálózati interfészeken keresztül támogatja a számítógépek közötti kommunikációt.

Napjainkban a különböző számítógép-kategóriákhoz az operációs rendszerek széles skálája van forgalomban. A személyi és üzleti számítógépek területén ezek közé tartozik a Windows (Microsoft), az iOS (Apple), a macOS (Apple), az Android (Google), a Linux és annak különböző disztribúciói (fejlesztői közösségek vagy különböző vállalatok), stb.
A kommunikációs eszközök, különösen az internetes útválasztó eszközök területén például szolgál az IOS - Internetwork Operating System (Cisco), mely útválasztásra és adatkapcsolásra optimalizált.

[9] Az illesztő programok (driverek) lehetővé teszik az operációs rendszer és a hardver közötti kommunikációt, ami azt jelenti, hogy az operációs rendszer az eszközök széles skálájával képes együttműködni anélkül, hogy az alkalmazásokat minden egyes hardvertípushoz testre kellene szabni.
[10] Rendszerezik az adatokat az adattárolón. Minden operációs rendszer különböző fájlrendszereket támogathat, a tároló és az alkalmazások típusától függően. (Pl. FAT32, NTFS, ext4, stb.)

A szórakoztató elektronikában találkozhatunk a TRON (The Real-time Operating system Nucleus) operációos rendszerrel.
Az RTOS (Real-Time Operating System) operációs rendszerek fontos szerepet játszanak azokban az eszközökben, ahol azonnali reakcióra van szükség (pl. orvosi eszközök vagy autóelektronika), stb.

A **támogató programok** (utilities) kibővítik az operációs rendszerek képességeit, vagy további szükséges funkciókat biztosítanak az alkalmazások vagy a felhasználók számára. Ebben a szoftvercsoportban számos segédprogramot találunk, például különböző diagnosztikai eszközöket a technikai erőforrások működőképességének tesztelésére, sérült fájlok helyreállítására szolgáló eszközöket, külső memóriákon lévő fájlok böngészésére szolgáló eszközöket, stb.

A rendelkezésünkre álló technológiák lehetővé teszik számunkra az emuláció, a szimuláció, a virtualizáció és az alkalmazásvirtualizáció megvalósítását. Az **emuláció** során speciális szoftverekkel egy adott technikai erőforrás megjelenését hozzuk létre az alkalmazás számára. Például egy játékkonzolt emulálunk egy személyi számítógépen, azaz egy ilyen alkalmazás számára a személyi számítógép egy adott játékkonzolként jelenik meg. Az emuláció nem csak a technikai erőforrásokat emulálja, hanem olyan utasításokat is megváltoztat, amelyeket egy adott rendszer nem lenne képes végrehajtani (pl. x86 architektúra ARM-en). **Szimulációval** (szintén szoftver segítségével) egy adott operációs szoftverkörnyezet megjelenését hozzuk létre. Például egy Apple számítógépen a Microsoft Virtual PC for PowerPC segítségével képesek vagyunk egy Windows operációs rendszer környezetét létrehozni, majd ebben a környezetben más, a Windows operációs rendszerre tervezett szoftvereket futtatni. A szimuláció elsősorban a szoftver vagy rendszer viselkedésére összpontosít, nem csak a megjelenésére. A **virtualizációban** egy úgynevezett **hypervisort** használunk virtualizációs eszközként. A hipervizor egy olyan szoftverkomponens, amely „leválasztja” a hoszt operációs rendszert a technikai erőforrásokról. Attól függően, hogy a hipervizor hova kerül beillesztésre, megkülönböztetünk úgynevezett **natív virtualizálást** és **operációs rendszer szintű virtualizálást**. A natív virtualizációs erőforrások képviselői pl. a Citrix XenServer, az Oracle VM, a VMWare ESX Server, a Microsoft Hyper-V, stb. A **natív virtualizáció** biztosításához feltételezzük, hogy a technikai erőforrások (processzor) támogatják ezt a koncepciót. Példaként a HP, az IBM, a Bull, stb. kiválasztott szervercsaládjai említhetők. Itt szokunk találkozni az angol kifejezéssel „bare-metal hypervisors”, ami azt jelenti, hogy a hypervisor közvetlenül a hardveren fut, operációs rendszer szüksége nélkül. Az operációs rendszer szintű virtualizációs erőforrások képviselői közé

tartozik a VMWare (Server, Workstation), QEMU, Microsoft Virtual PC, Microsoft Virtual Server, stb. A szervezet egyedi igényeitől függően alkalmazzuk a virtualizációt a szerverekre, személyi számítógépekre, adattárolókra és hálózatokra. A **szervervirtualizációval** ezután lehetőség nyílik a rendszer erőforrásainak dinamikus hozzárendelésére az egyes alkalmazásokhoz, és ezek az erőforrások különböző fizikai számítógépek részét képezhetik. A szervervirtualizáció lehetővé teszi több szerver egyetlen fizikai eszközön történő konszolidálását, ami költségcsökkentést és hatékonyabb erőforrás-felhasználást eredményez. A vállalati rendszerek reálisan használják a virtualizációt a fizikai szerverek számának csökkentésére. Szükséges megemlítenünk még egy lehetőséget – a Live migration-t. A live migration koncepciója lehetővé teszi a futó virtuális gépek megszakítás nélküli mozgatását a fizikai hosztok között. A **személyi számítógépek virtualizálásánál** ezután lehetőség nyílik arra, hogy egy teljes PC-környezet (az összes alkalmazással együtt) egy szerveren tartózkodjon és fusson, és bármely klienseszközről (terminálról) elérhető legyen. Az **alkalmazásvirtualizáció** olyan környezet létrehozása, amely elszigeteli az alkalmazásokat az operációs rendszertől és a technikai erőforrásoktól. Ennek egyik képviselője a JVM (Java Virtual Machine)[11]. A virtuális gép futási környezetet biztosít az alkalmazások számára, és egy alkalmazás bármilyen operációs rendszeren és technikai erőforráson futtatható.

Az általunk közös néven **middleware**-nek nevezett technológia célja, hogy a sokféle (heterogén)[12] elosztott alkalmazások környezetében, amelyek gyakran önállóak (más alkalmazásoktól függetlenek), lehetővé tegye ezen alkalmazások integrálását egy nagyobb egészbe[13]. Emellett szükséges, hogy ezek a különálló alkalmazások képesek legyenek egymásnak alkalmazásszolgáltatásokat nyújtani egy számítógépes hálózaton keresztül. Nézzünk két definíciót, ami segítségével jól megérthetjük, hogy miről is van szó: (1) A middleware az a szoftver, amely az operációs rendszer és az alkalmazások között helyezkedik el, és sokoldalú szolgáltatásokat nyújt az alkalmazásoknak. (2) A szolgáltatások olyan operációs rendszerkörnyezetet hoznak létre, amely lehetővé teszi,

[11] A JVM egy példa a platformfüggetlen alkalmazásra, amely elszigeteli az alkalmazást egy adott operációs rendszertől, és lehetővé teszi, hogy különböző platformokon fusson.

[12] Ezek az alkalmazások különböző platformokon, különböző operációs rendszereken vagy különböző programozási nyelveken íródhatnak.

[13] A middleware-t az alkalmazások közötti kommunikáció és koordináció egyszerűsítésére is használják, ami különösen fontos az összetett informatikai infrastruktúrákban.

hogy a folyamatok, programok és alkalmazások egy elosztott rendszerben, hálózaton keresztül kölcsönhatásba lépjenek egymással.

- **Alapvető middleware-szolgáltatások** szükségesek az olyan alkalmazások futtatásához, amelyek feldolgozása elosztott, azaz különböző, számítógépes hálózattal összekapcsolt számítógépeken történik.
- **Integrációs middleware-szolgáltatások** szükségesek ahhoz, hogy az autonóm alkalmazások összekapcsolódjanak egy nagyobb egésszé úgy, hogy az összekapcsolt alkalmazásokat csak minimálisan zavarják.
- Az **alkalmazásintegrációs middleware-szolgáltatások** nem pusztán middleware erőforrásokból állnak, hanem a szolgáltatások itt szorosan kapcsolódnak egy adott alkalmazáshoz. Ebben a kategóriában meghatározó szerepet játszanak az úgynevezett alkalmazásadapterek. Ezek lehetővé teszik, hogy egy alkalmazás kommunikáljon a piacot uraló alkalmazással (SAP, Oracle, Salesforce stb.)[14], vagy úgy, hogy a kommunikáció megfeleljen az adott piaci szegmens szabványainak (S.W.I.F.T, ebXML, RosettaNet, stb.). Ezek az adapterek nemcsak az alkalmazások közötti kommunikációt teszik lehetővé, hanem gyakran lefordítják vagy átalakítják az adatokat a különböző rendszerek között a kompatibilitás biztosítása érdekében.

Említsünk meg néhány konkrét middleware-t. Az **Enterprise Service Bus** (ESB) egy fontos middleware platform, amely a vállalati alkalmazások központi kommunikációs pontjaként szolgál. Az ESB lehetővé teszi, hogy a különböző protokollokat vagy adatformátumokat használó különböző alkalmazások kommunikáljanak egymással. Az ESB-megoldásokra példa az Apache Camel, a MuleSoft Anypoint Platform vagy az IBM WebSphere ESB. A **Message-Oriented Middleware** (MOM) a különböző alkalmazások közötti aszinkron kommunikációra szolgál. Ez azt jelenti, hogy az alkalmazások időtől függetlenül küldhetnek és fogadhatnak üzeneteket, ami robusztusabb és rugalmasabb kommunikációt tesz lehetővé. A MOM-technológiák közé tartozik például az Apache Kafka, a RabbitMQ, az IBM MQ vagy az ActiveMQ (például, a megrendelések feldolgozása vagy értesítések küldése). A modern vállalati rendszerekben gyakran használnak API-kat (**Application Programming Interface** – alkalmazásprogramozási interfészeket) az alkalmazások közötti

[14] A middleware lehetővé teszi a különböző alkalmazások és technológiák közötti kommunikáció szabványosítását.

kommunikációra. Az API-k olyan meghatározott szabályokat és protokollokat biztosítanak, amelyek segítségével az alkalmazások kommunikálhatnak egymással. Átjáróként működik, lehetővé téve más alkalmazások számára, hogy hozzáférjenek egy adott szoftver funkcióihoz, adataihoz vagy szolgáltatásaihoz. Az **API-kezelő platformok** kezelik ezeket az API-kat, biztosítják azok elérhetőségét, biztonságát és felügyeletét. Az API-kezelési megoldásokra példa az Apigee (Google), a MuleSoft API Gateway, az AWS API Gateway vagy a Kong. Az API-menedzsment middleware lehetővé teszi a vállalatok számára, hogy szolgáltatásaikat könnyen elérhetővé tegyék más alkalmazások számára, és figyelemmel kísérjék azok teljesítményét és biztonságát. **Database Middleware** konzisztens hozzáférést biztosít a különböző típusú adatbázisokhoz, és lehetővé teszi, hogy az alkalmazások az adatbázis-rendszer típusától (MySQL, Oracle, SQL Server) függetlenül hozzáférjenek az adatokhoz. Egyszerűsíti az adathozzáférést, a skálázhatóságát biztosítja, és biztonságos adatátvitelt tesz lehetővé. Ilyen például az ODBC[15] (*Open Database Connectivity*), a JDBC (*Java Database Connectivity*), a Hibernate (ORM for Java), stb. A **Transactional Middleware** az elosztott rendszereken belüli tranzakciókat kezeli. A tranzakciós middleware biztosítja, hogy a tranzakció minden lépése helyesen kerüljön végrehajtásra, vagy hiba esetén a tranzakció teljes megszakítása (rollback). Ilyen például az IBM CICS, a Microsoft Distributed Transaction Coordinator (MSDTC) vagy az Oracle Tuxedo. A **Web Service Middleware** lehetővé teszi az alkalmazások közötti kommunikációt az interneten keresztül webszolgáltatások és szabványok, például SOAP (Simple Object Access Protocol) vagy REST (Representational State Transfer) segítségével. Ilyen például a webkiszolgáló és az alkalmazáskiszolgáló, például az Apache Tomcat, a JBoss vagy a Microsoft IIS.

Ezeken kívül még továbbiakat is ismerünk, figyelembe véve a konkrét további területeket, mint pl. Microservices Middleware vagy Object Request Brokers middleware.

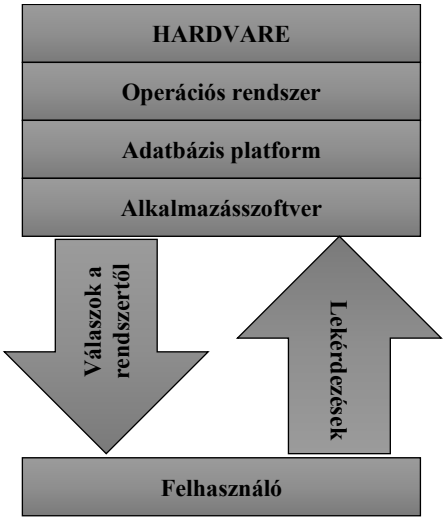
Muszáj szintén elmondanunk, hogy lehetnek különbségek a telepítéskor. Például, a teljes körű middleware megoldások (ESB, message brokers vagy

[15] ODBC segítségével képesek csatlakozni a vállalati információs rendszerekben használt legtöbb adatbázishoz. Az ODBC (Open Database Connectivity) egy olyan protokoll, amely lehetővé teszi különböző alkalmazások számára, hogy különböző adatbázis-rendszerekkel kommunikáljanak, függetlenül az adatbázis típusától vagy gyártójától. Elsősorban a relációs adatbázisokhoz készítették, de más adatbázisokhoz való hozzáférést is engedélyezheti, ha azok rendelkeznek ODBC protokollal.

API átjárók) általában független szoftverként települnek, hasonlóan egy operációs rendszerhez, és több alkalmazás számára nyújtanak szolgáltatásokat. Az ODBC és a JDBC a middleware típusai, amelyeket vagy az alkalmazás infrastruktúrájának részeként telepítenek (pl. ODBC a Windowsban), vagy az alkalmazással együtt (pl. JDBC a Java alkalmazásokban). (Szabó, B.; Márföldi, E. 2011) (Herdon, M. 2016) (Juhász, S. 2011) (Berecz; A.; Karácsony, P.; Kónya, L.; Peck T, T.; Szász, G.; Vári-Kakas, I. 2019)

4.2.2 Alkalmazásszoftver

Az alkalmazásszoftver olyan szoftver, amelyet arra terveztek, hogy konkrét feladatokat végezzen a felhasználók számára. A legtöbb vállalati rendszer alapvetően alkalmazásszoftver, mivel úgy tervezték őket, hogy konkrét üzleti feladatokat és folyamatokat kezeljenek. Ezeket a rendszereket különböző üzleti funkciók, például a pénzügyek, a gyártás, a humán erőforrás, az ügyfélszolgálat és más területek kezelésére, automatizálására és optimalizálására használják. Ahhoz azonban, hogy ezek az üzleti alkalmazások működhessenek, olyan alapszoftverekre, például operációs rendszerekre és middleware-ekre van szükség, amelyek biztosítják a támogatást és az infrastruktúrát az alkalmazások futtatásához. Alkalmazásszoftverek például az ERP (Enterprise Resource Planning), CRM (Customer Relationship Management), HRMS (Human Resource Management System), SCM (Supply Chain Management), vagy BI (Business Intelligence).



4.4. ábra: Az információs rendszer technológiai szemlélete
Forrás: (Kučera, M.; Fiľa, M.; Látečková, Z.; Stuchlý, P. 2017)

Az olyan alapszoftverek, mint az adatbázisszerverek, a middleware és az operációs rendszerek elengedhetetlenek a vállalati alkalmazások támogatásához, de önmagukban nem tekinthetők vállalati rendszernek. Például, az adatbázis-

kezelő rendszer (pl. Oracle Database, Microsoft SQL Server) olyan alapszoftver, amely a vállalati alkalmazások által használt adatok tárolásának és kezelésének infrastruktúráját biztosítja, de nem tekintjük vállalati alkalmazásnak. (Szabó, B.; Márföldi, E. 2011) (Herdon, M. 2016) (Juhász, S. 2011) (Berecz; A.; Karácsony, P.; Kónya, L.; Peck T, T.; Szász, G.; Vári-Kakas, I. 2019)

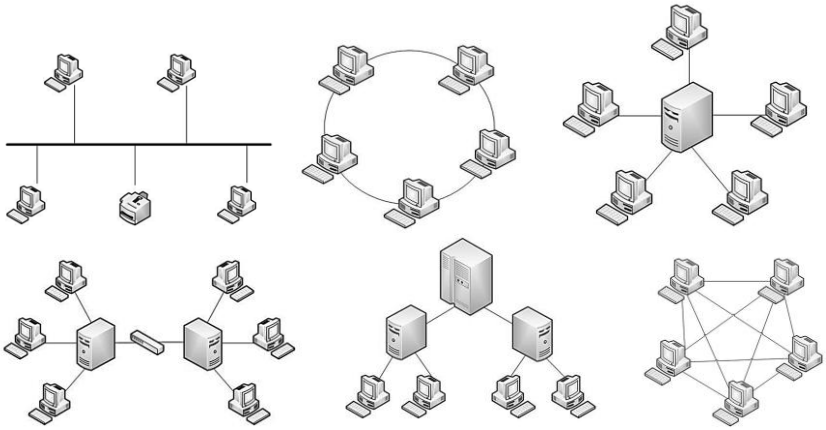
4.3 Számítógépes hálózatok

A hálózatok az önállóan is működőképes számítógépek elektronikus összekapcsolása (hardveres és szoftveres), ahol az egyes gépek külső beavatkozás nélkül képesek kommunikációra. Két számítógép akkor nevezhető összekapcsoltnak, ha azok egymás között információ cserére képesek, megfelelő protokollok[16] segítségével. A számítógépes hálózatokat kiterjedésük szerint a következő képpen kategorizálhatjuk:

- személyi hálózat – (**Personal Area Network, PAN**) okoseszközök (telefonok, laptopok, tévék, hangszórók, nyomtatók, stb.) vezeték nélküli összekapcsolását és a közöttük történő adatátvitelt biztosítja. Ezekben a hálózatokban az uralkodó technológia a Bluetooth. Azonban más technológiák is megjelennek, mint például a WiDi (Wireless Display), Zigbee vagy NFC, amelyek lehetővé teszik az adatok számítógépről történő továbbítását, majd lejátszását egy megfelelően felszerelt televíziókészüléken.
- helyi hálózat – (**Local Area Network, LAN**) a hálózati elemek összekapcsolásának alapvető eszköze vállalati szinten, és ma már gyakran használják otthon is. Az ilyen hálózatot néha a SOHO (Small Office / Home Office) rövidítéssel jelölik. Az uralkodó technológiák az Ethernet és a Wi-Fi.
- nagyvárosi hálózat – (**Metropolitan Area Network, MAN**) nagyobb kiterjedésű, mint a LAN. Gyakran egy városi agglomerációban épül ki, és általában egy területen belül különböző kampuszokat köt össze nagyobb összekapcsolási sebességgel. Ilyen például a prágai egyetemi tanszékeket összekötő PASNET (Prague Academic Computer Network) hálózat. Ilyenek szintén az önkormányzati intézmények, kórházak, könyvtárak stb. közötti hálózatok.

[16] Szabályoknak és konvencióknak a formális leírása, amely meghatározza a hálózati eszközök (csomópontok) kommunikációját (kommunikációs szabályhalmaz). Pl.: HTTP, FTP, IP, DHCP, TCP, SMTP, stb.

- nagy kiterjedésű hálózat – (**Wide Area Network, WAN**) nagyobb kiterjedésű, mint a MAN-ek, és nem korlátozódik területileg. Ezek a hálózatok kiterjedhetnek egy régióra, egy országra, egy kontinensre vagy lehetnek interkontinentálisak. Ilyen például a csehországi akadémiai hálózat (CESNET), az akadémiai intézményeket összekötő európai hálózat (GEANT), az Internet, stb. Az uralkodó technológiák a Frame Relay, az ATM (aszinkron átviteli mód), a DSL (digitális előfizetői vonal). A WAN-ok gyakran különböző technológiák (műhold, optikai szálak, mikrohullámú kapcsolatok) kombinációját használják a nagy távolságok eléréséhez. Ilyen az Internet is.
- Ritkán találkozunk a **GAN** (**Global Area Network**) kifejezéssel, amely rendkívül nagy földrajzi kiterjedésű kommunikációs hálózattokat foglalhat magában. Hatalmas földrajzi területeket fed le, és általában különböző technológiák kombinációjával működik műholdakat nem kivéve. Néhány irodalomban említésre kerül, ezért a teljesség kedvéért mi is megemlítjük.



4.5. ábra: Hálózati topológiák - olvassa soronként balról jobbra haladva: sín (busz), kör, csillag, kibővített csillag, fa és háló.
Forrás: (Szabó, B.; Márföldi, E. 2011)

Továbbiakban a LAN hálózatokkal fogunk foglalkozni, kivéve. Ha más típusra térnénk ki, arra külön felhívjuk az olvasó figyelmét. A számítógépes hálózatot a hálózati hardver és hálózati szoftver segítségével építjük ki. **Hálózati hardver** magában foglalja azokat az eszközöket, amelyek révén a tényleges hálózati összekapcsolás megvalósul (hálózati adapterek, hálózati kártyák, routerek, kábelek, stb.). A **hálózati szoftver** olyan szoftver, amely a hálózati hardverrel együttműködve

biztosítja a hálózati funkciókat. Egyes operációs rendszerek ezeket a funkciókat már tartalmazzák. **Hálózati OrgWare** magában foglalja főleg a rendszergazda feladatait, a hálózat kezelését és fenntartását is[17].

A számítógépes hálózatokban a csomópontok[18] és az azokat összekötő kommunikációs csatornák kapcsolódásának rendszerét topológiának nevezzük. Másképpen, **azt a rendszert, ami alapján a számítógépes állomások össze vannak kötve, topológiának nevezzük** (4.5. ábra). A **sín topológia** egyetlen, mindkét végén lezárt gerinckábelt használ. Minden felhasználó párhuzamosan van rákötve a közös kábelre – sínre, amit úgy is nevezünk, hogy busz. Előnyei közé tartozik a könnyű bővíthetőség, egyszerűség, állomás hibája nem zavarja a hálózat működését. Hátránya az, hogy a nagy leterhelés mellett a hálózat lelassulhat éa a közös kábel hibája az egész hálózatot megbénítja. Ezt a topológiát ritkábban használják a modern hálózatokban, mivel a sebesség és a megbízhatóság alacsonyabb a többi topológiához képest. A **gyűrű (kör) topológia** esetében minden állomás a következőhöz csatlakozik, az utolsó pedig az elsőhöz. Ezzel a kábel fizikailag gyűrűt formál. Előnyei közé tartozik a kollíziók kiküszöbölése és olcsó megvalósíthatósága. Nagy hátránya azonban az, hogy az adatok az összes állomáson áthaladnak – lelassítva ezzel a kommunikációt. Természetes, hogy egy csomópont meghibásodása a hálózat leállítását okozza szintúgy, mint egy új eszköz hozzáadása. A modern rendszerekben a token ring technológiát használják a késleltetések (lag) és az ütközések néhány problémájának megoldására. A **csillag topológia** középpontja egy központi csomópont – központi berendezés, mely lehet: Hub, Switch, Router, Modem[19], szerver, stb. Ehhez a csomóponthoz csatlakoznak a számítógépek különálló vonalakkal. Előnyei közé tartozik a könnyű változtathatóság – új számítógépek hozzáadása vagy

[17] Az OrgWare a hálózatkezelés szervezeti aspektusait fedi le, beleértve a szerepeket, felelősségi köröket és folyamatokat, amelyek a hálózatok zavartalan működését biztosítják (pl. karbantartás, biztonság, adatmentés).

[18] Saját hálózati címmel rendelkező eszköz, amely független kommunikációra képes (számítógép, nyomtató, router). A kommunikáció során a csomópont viselkedhet adóként (forrás) vagy vevőként (nyelő).

[19] *Hub:* egy egyszerű eszköz, amely adatokat küld az összes csatlakoztatott eszköznek. *Switch:* egy okosabb eszköz, amely csak a céleszközöknek küld adatokat MAC (media access control address) címeik alapján. *Router:* olyan eszköz, amely különböző hálózatokat kapcsol össze, és kezeli a köztük lévő forgalmat, biztosítja a forgalomirányítást IP (Internet Protocol address) címek alapján. *Modem:* olyan eszköz, amely jeleket továbbít és lehetővé teszi pl. az internethez való csatlakozást - a számítógép digitális jelét olyan analóg formátumra alakítja át, amely különböző típusú hálózatokon, például telefonvonalakon vagy kábelhálózatokon keresztül továbbítható, és fordítva.

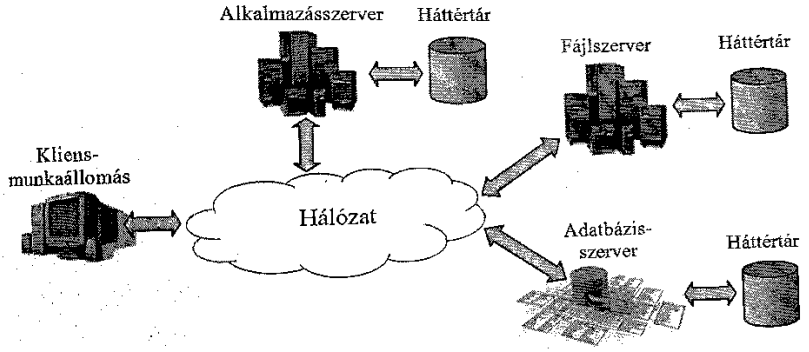
könnyű központi kezelés. A számítógép és a központi berendezés közötti kábel meghibásodása, nem befolyásolja a hálózat működését. Hátránya viszont az, hogy ha a központi berendezés meghibásodik, ez az összes hozzá csatlakozó állomás működésképtelenségét eredményezi. Ez a legelterjedtebb a modern hálózatokban (különösen a LAN-okban), és más topológiákkal összehasonlítva a legjobb megbízhatóságot és skálázhatóságot biztosítja, ezért a vállalati hálózatokban ez a legelterjedtebb modell. Ha két vagy több elemi topológia központi berendezését összekapcsoljuk, akkor **kibővített csillag topológiáról** beszélünk. Ezzel a topológiával kiterjeszthető a hálózat hatóköre és a lefedettség mértéke (például, két informatikai terem összekapcsolható így). A **fa topológia** hasonlít a kibővített csillagra, viszont, ebben az esetben nem a központi berendezések vannak összekötve, hanem a rendszer egy számítógéphez csatlakozik, amely vezéreli a topológián belül zajló forgalmat. Bármely két összekötött gép között egy és csak egy útvonal van. A **háló topológiát** akkor szokás alkalmazni, ha a lehető legnagyobb mértékű védelmet kell elérni az esetleges szolgáltatás kimaradással szemben. Például az atomerőművek hálós vezérlőrendszerében alkalmazható háló topológia. Ez a topológia magas szintű redundanciát és biztonságot nyújt, így ideális választás például a fent említett kritikus fontosságú alkalmazáshoz. Az internet ugyan minden helyet több útvonallal tud elérni, mégsem teljes háló topológia. A háló topológiában minden csomópont össze van kötve azzal a csomóponttal, amelyikkel kommunikálni kíván. A háló topológia egy speciális változata, mikor minden csomópont össze van kötve a többi csomóponttal. Ezt nevezzük **teljes topológiának**. Muszáj elmondanunk, hogy lényegében bármely elemi topológia előfordulhat egy teljes hálózat részeként, ilyenkor **hibrid topológiáról** beszélünk. A modern vállalati hálózatokban is gyakran alkalmaznak hibrid topológiákat, ahol több topológiát kombinálnak az egyes topológiák előnyeinek kihasználása érdekében. Az információs és kommunikációs technológiák folyamatos fejlődésével együtt változnak a célok és elvárások, melyeket a konkrét vállalati információs rendszerek implementációik követelnek meg. Erőforrásokhoz való hozzáférés módja szerint ismerünk **egyenrangú hálózatokat** - hálózatba kapcsolt számítógépek egymással egyenrangú partnerekként viselkednek. Az egyenrangú állomások ügyfél és kiszolgáló szerepet egyaránt felvehetnek, és ügyfél-kiszolgáló elrendezésnél a hálózati szolgáltatásokat egy kiszolgálónak (szerver) nevezett számítógép futtatja. A kiszolgáló feladata a válaszadás az ügyfelek kéréseire. A kiszolgáló olyan központi számítógép, amely folyamatosan rendelkezésre áll az ügyfelektől érkező szolgáltatásokra vonatkozó kérések fogadására. A legtöbb hálózati

operációs rendszer az ügyfél-kiszolgáló architektúrát követi. A munkaállomások általában ügyfél szerepet játszanak, a nagyobb kapacitással és speciális szoftverekkel rendelkező számítógépek pedig a kiszolgálók. További felosztásnál is ügyfél-kiszolgáló megoldásokkal fogunk foglalkozni. Leggyakrabban a következő adatfeldolgozási és adatátviteli módszereket és architektúrákat használjuk:

- **Hoszt – Terminál**: A rendszer alapegysége a szervergép (terminálszerver), ahol minden művelet végrehajtásra kerül. A terminál csak az eredményeket jeleníti meg, és lehetőséget biztosít a parancsok beadására. Előnye a központi adatbebiztosítás és a nagymennyiségű információ tárolásának lehetősége. Hátrányok közé tartozik a magas költség és a bonyolult karbantartás. Ez egy régebbi architektúra, amelyet főként a múltban használtak, amikor a terminálok „buta", saját adatfeldolgozással nem rendelkező eszközök voltak. Ma például nagyszámítógépes rendszerekben vagy terminálszerverekben használják, mint például a Citrix.
- **Fájl – Szerver**: Adatok egy megosztott fájlszerveren vannak elhelyezve, míg a feldolgozás a kliensek oldalán történik. A szerverek központi adattárolók, amelyek kliensek számára elérhetők, de a tényleges feldolgozás a kliensek oldalán történik. Ezt az architektúrát ma főként a hálózatokon történő fájlmegosztásra használják.
- **Peer to Peer**: Nem függ a szerver és kliens elválasztásától. Egyes csomópontok, egyszerre lehetnek a kliens és a szerver szerepében is. Feltételezi, hogy minden egyes felhasználó megoszt valamit a rendszeréből a többi felhasználóval (elsősorban fájlokat, tehát szoftvert, de nyomtatókat is – hardver eszközöket). Mivel számítógépek egyrangúak, gépek, mint egyenlő az egyenlővel vannak összekapcsolva. Ilyenek pl. a torrent hálózatok vagy blockchain technológia.
- **Kliens – Szerver**: Az architektúra alapja a munkamegosztás. A szerver biztosítja az adatokat és a szolgáltatásokat, míg a kliens hozzáfér és feldolgozza vagy megjeleníti azokat. A központi részt, amely az adatokat tartalmazza és szolgáltatja a felhasználónak, adatbázisszervereknek nevezzük. Ezt az architektúrát pl. a MySQL, Oracle vagy Microsoft SQL Server használja.

Vállalati hálózaton belül több szerver is működtetve lehet. Szerver szolgáltatásokat nyújt vagy megosztást a klienseknek. Alkalmazási területük szerint példaként megemlíthetünk néhányat.

A nyomtató szerver lehetővé teszi a hálózati felhasználók számára a számítógépes nyomtatók távoli elérését; a DNS-szerver lefordítja a tartományneveket (www.ujs.sk = 193.87.80.103) IP-címekre; a játékszerver lehetővé teszi a játékosok számára, hogy többjátékos játékokat játsszanak; a proxy szerver közvetett kapcsolatokat tesz lehetővé más kiszolgálókkal; a levelezőszerver biztosítja az e-mailek küldését és fogadását más szerverekkel és/vagy felhasználókkal kapcsolatban, az alkalmazásszerver egy alkalmazás futtatására szolgáló számítógép, stb.



4.6. ábra: Egy egyszerű vállalati alkalmazás működésében részt vevő rendszerek
Forrás: (Juhász, S. 2011)

Vállalati információs rendszerek kapcsán találkozunk az intranet és extranet kifejezésekkel. Az **intranet** olyan számítógép-hálózat, mely ugyanazokat a technológiákat használja, mint az internet (TCP/IP protokoll-család, HTTP átviteli protokoll, stb.) de a külvilág (az internet) felé zárt, vagy csak egy átjárón, illetve tűzfalon[20] keresztül érhető el, amely az intranet külső kapcsolatait szabályozza. Csak a belső felhasználók (alkalmazottak) számára érhető el. Információmegosztásra, belső

[20] A tűzfal olyan biztonsági rendszer (hardver vagy szoftver), amely a különböző hálózatok (például egy belső hálózat és az internet) közötti adatforgalom ellenőrzésére és kezelésére szolgál. A tűzfal előre beállított biztonsági szabályok alapján határozza meg, hogy a forgalom mely része engedélyezett vagy blokkolt. A tűzfal megvédi a hálózaton lévő számítógépeket és eszközöket a külső hálózatokból érkező jogosulatlan hozzáféréstől vagy rosszindulatú tartalomtól. Segít megvédeni az olyan fenyegetésekkel szemben, mint a hackerek, vírusok és az illetéktelen hozzáférés. A tűzfal elemzi a bejövő és kimenő forgalmat, és eldönti, hogy engedélyezze, blokkolja vagy átirányítsa azt. Lehet szoftveres tűzfal - közvetlenül az eszközre (számítógépre, szerverre) telepítve, vagy hardveres tűzfal - különálló eszköz, amely az egész hálózatot védi.

kommunikációra és belső alkalmazások használatára szolgál. Tipikus intranet egy vállalat belső hálózata, amit az internettől tűzfal választ el. Ha az intranet egy része elérhetővé válik az ügyfelek és más, a vállalaton kívüli személyek számára, akkor az intranet e része az **extranet** részévé válik. Az extranet tehát az intranet kiterjesztése, amely lehetővé teszi a külső felhasználók (pl. partnerek, ügyfelek) számára, hogy hozzáférjenek az intranet bizonyos részeihez. Az extranet az interneten keresztül is elérhető, de a hozzáférés ellenőrzött és védett (pl. jelszóval vagy VPN-nel), hogy garantálja a biztonságot.

A **TCP/IP modell** a Transmission Control Protocol/Internet Protocol (átviteli vezérlő protokoll/internetprotokoll) rövidítése. Az internetet felépítő protokollstruktúra. Kifejlesztése az ARPANET-hez köthető, 1983-ban teljesen leváltotta az addig használt NCP (Network Control Protocol) protokollt. A TCP/IP felépítése az absztrakciós rétegződési elven alapul. Minden egyes réteg egy jól definiált feladatot végez el. Minden réteg csak a vele, szomszédos réteggel képes kommunikálni. Az **alkalmazási réteg** a felhasználó által indított program és a szállítási réteg között teremt kapcsolatot. Ha egy program hálózaton keresztül adatot szeretne küldeni, az alkalmazási réteg tovább küldi azt a szállítási rétegnek. A **szállítási réteg** az alkalmazási rétegtől kapott adat elejére egy úgynevezett fejlécet (angolul: header) csatol, mely jelzi, hogy melyik szállítási réteg protokollal (leggyakrabban TCP vagy UDP)[21] küldik az adatot. **Hálózati (Internet) réteg** a szállítási rétegtől kapott header-adat pároshoz hozzáteszi a saját fejlécét, amely arról tartalmaz információt, hogy az adatot melyik végpont kapja majd meg. Az **adatkapcsolati réteg** szintén hozzárakja a kapott adathoz a saját fejlécét és **az adatot keretekre bontja**. Ha a kapott adat túl nagy ahhoz, hogy egy keretbe kerüljön, feldarabolja és az utolsó keret végére egy úgynevezett tail-t kapcsol, hogy a fogadó oldalon vissza lehessen állítani az eredeti adatot. A **fizikai réteg** továbbítja az adatkapcsolati rétegtől kapott kereteket a hálózaton. LAN hálózatokban a legelterjedtebb technológia az Ethernet. Ethernet a TCP/IP modellt használja. Ethernet hálózatokban a véletlen átvitelvezérlés az alkalmazott. Ha szabad a hálózat, akkor bármelyik állomás leadhat jelet. A véletlen átvitelvezérlés esetében minden állomás (számítógép) maga dönti el, hogy mikor kísérli meg az üzenet elküldését, éppen ezért megtörténhet, hogy két gép egy időben próbál meg adást küldeni a közösen használt csatornán. Ilyen esetben ütközésről beszélünk. Ha két üzenet ütközik, akkor mindkettő elvész, éppen ezért a véletlen átvitelvezérlések esetén az adó gépnek mindenképpen értesülnie kell az esetleges ütközésről, hogy az üzenet

[21] A TCP megbízható protokoll, hibaellenőrzéssel, míg az UDP gyorsabb, de megbízhatatlan.

küldését megismételhesse. Tipikus megvalósítása a **CSMA/CD** (Carrier Sense Multiple Access with Collision Detection)[22], azaz csatorna figyelő többszörös hozzáférés ütközés detektálással. Ez szerint, ha előfordul az ütközés (mielőtt adást indítana az állomás, előtte „belehallgat" a közegbe, hogy szabad-e), akkor az adó állomások újabb próbálkozásuk előtt véletlenszerű ideig várnak.

Ha az információközlés csak két pont (egy adó és egy vevő) között zajlik, akkor **pont-pont kapcsolatról** beszélünk. **Többpontos kapcsolatról** akkor beszélünk, ha egy adó egyszerre több vevőt lát el információval – pl. broadcast vagy multycast. (Szabó, B.; Márföldi, E. 2011) (Herdon, M. 2016) (Juhász, S. 2011) (Berecz; A.; Karácsony, P.; Kónya, L.; Peck T, T.; Szász, G.; Vári-Kakas, I. 2019)

4.3.1 Internet

Az **FNC** (Federal Networking Council) szerint az internet egy világméretű információs rendszer, amely:

- logikailag összekapcsolódik egy globális és egységes címtartományon keresztül, amely az internetprotokollon alapul, beleértve más, az alábbiakat is, és utódait;
- a hálózati elemek kommunikációját a TCP/IP protokollcsomag, vagy (a jövőben) más, az IP protokollal kompatibilis protokollok biztosítják;
- a legmagasabb szinten, nyilvánosan vagy magántulajdonban olyan technológiákon keresztül nyújt, használ vagy tesz elérhetővé szolgáltatásokat, amelyek a fenti kommunikációs elveken és a kapcsolódó infrastruktúrán alapulnak.

Ez a meghatározás azt jelenti, hogy az interneten kommunikáló hálózati elemnek egyedi címmel kell rendelkeznie (a meghatározás pontja). Ezt a címet meghatározó protokoll szerint IP-címnek (Internet Protocol) nevezik. Kétféle IP-cím létezik: IPv4 és IPv6. Míg az IPv4 még mindig a legszélesebb körben használt, az IPv6 egyre fontosabbá válik a rendelkezésre álló IPv4-címek kimerülése miatt. A cím megszerzésének

[22] A CSMA/CD volt a régebbi Ethernet-hálózatok alapja, de a kapcsolókat használó (switch) modern hálózatokban az ütközések szinte nem léteznek, mivel minden eszköznek saját dedikált csatornája van. A modern Ethernet már teljes duplex üzemmódban működik, ahol nem fordulnak elő ütközések, mivel a küldés és a vétel egyszerre történik különböző kábel szálakon. Viszont az elv továbbra is implementálva van, azért is említjük.

módja a hálózati csomópont konfigurációjától függ, és a hálózati rendszergazda határozza meg. A tényleges kommunikációt a protokollok biztosítják, amelyek a TCP/IP (Transmission Control Protocol/Internet Protocol) nevű protokollcsomagba szerveződnek (a meghatározás II. pontja). Tehát nem egy protokollról van a szó, hanem számos protokollt tartalmaz, amelyek mindegyike sajátos szerepet tölt be. Például: A TCP (Transmission Control Protocol) biztosítja az adatok megbízható továbbítását a hálózaton keresztül. Az IP (Internet Protocol) gondoskodik a csomagok címzéséről és útválasztásáról. A végpontokon (számítógépeken) ezek a protokollok az operációs rendszer részét képezik, és beállításaikat a hálózati rendszergazda határozza meg. A felső szintű szolgáltatások eléréséhez és nyújtásához a megfelelő technológiákat kell telepíteni a végcsomópontokra (a meghatározás III. pontja).

Az internet technológiája a következő elveken alapul:

- Nincs olyan központi elem, amely a rendszer szűk keresztmetszetét képezné, és amelynek meghibásodása miatt leállna a kommunikáció az interneten. Az internetet decentralizált hálózatnak tervezték, éppen a hibákkal szembeni ellenálló képessége miatt. Vannak azonban központi elemek, például a **DNS-szerverek**, amelyek kulcsszerepet játszanak a tartománynevek feloldásában. Ha ezek a kulcsfontosságú szerverek meghibásodnának, az kihatással lehetne a szélesebb internet működésére.
- A kiesésekkel és a megbízhatatlansággal számolni kell, mivel a globális hálózat különböző minőségű hálózatokból áll, amelyek különböző (technikai és politikai) környezetben működhetnek. Ezt az elvet a „**best-effort**” modelljének nevezik. Az internet nem garantálja, hogy minden csomag megérkezik a célállomásra, de az adott körülmények között a lehető legjobb eredményre törekszik.
- A maximális erőfeszítés nem garantált eredménnyel jár, ami abból adódik, hogy a hálózat különböző részein eltérő a forgalom, vagy fennakadások lehetnek, vagy a kommunikációs útvonal a kommunikációs pontok között szokatlanul hosszú.

Az internet számos szolgáltatást nyújt. Ezek a következőkre oszthatók: infrastrukturális szolgáltatások, amelyek a felhasználók és alkalmazásaik számára támogató szolgáltatásként szolgálnak; és az alkalmazási szolgáltatások, amelyek közvetlenül felhasználóknak vagy alkalmazásoknak nyújtanak szolgáltatást. Az internetes **infrastruktúrális szolgáltatások** közé tartoznak:

- **DNS** (Domain Name System), amely lehetővé teszi a felhasználók számára, hogy könnyen megjegyezhető neveket használjanak a hálózaton lévő objektumok címzéséhez (pl. www.ujs.sk, infoujs@ujs.sk, stb.), IP-címek helyett.
- **DHCP** (Dynamic Host Configuration Protocol), amely a hálózati rendszergazda által meghatározott szabályok alapján biztosítja az IP-címek és egyéb konfigurációs információk dinamikus hozzárendelését a hálózat egy végpontjához, mivel csak akkor tud kommunikálni az interneten, ha egy csomópont rendelkezik IP-címmel.
- Időszinkronizációs menedzsment szolgáltatás (**Network Time Protocol**), amely biztosítja az egyes hálózati csomópontok, és így a végszámítógépek belső óráinak helyes beállítását, hogy mindannyian azonos és pontos időt mutassanak.
- Az útválasztó eszközök frissítési szolgáltatása, amely biztosítja, hogy az útválasztó eszközök folyamatosan tájékoztassák egymást a környezetükről – **forgalomirányító protokollok**. Ezt használja például egy olyan szolgáltatás, amely így képes megtalálni az utat egy csomópontból a hálózaton keresztül, amely kommunikációt kér azzal a csomóponttal, amellyel a kommunikációt meg kell valósítani.

Az **internetes alkalmazási szolgáltatások** közé tartoznak például a következők:

- A **World Wide Web** (WWW) szolgáltatás, amely eredetileg az úgynevezett hipertext rendszerhez kapcsolódott. A hipertext olyan szöveg, amelyet nem korlátoz a linearitás, ellentétben például egy nyomtatott könyvvel. A hipertext „linkeket” tartalmaz más (ugyanabban vagy egy másik fájlban tárolt) szövegekre, és az olvasó a saját maga által választott útvonalon halad a szövegben. A hipermédia kifejezést olyan hipertextre használják, amely nem korlátozódik a szövegre, hanem például képeket, videót, hangot is tartalmaz. A hipertext leírására a HyperText Markup Language (HTML) és annak új vagy átdolgozott változatai szolgálnak. Napjainkban a szolgáltatás nemcsak a különböző alkalmazások felhasználói felületének megjelenítését biztosítja a felhasználó böngészőjében, hanem az alkalmazások egymással való kommunikációjakor is használják. Manapság a **HTML**-t olyan technológiákkal kombinálják, mint a **CSS** (a formázás) és a **JavaScript** (az interaktivitás), hogy modern weboldalakat hozzanak létre.

- Az elektronikus levelezés (**e-mail**) szolgáltatás biztosítja a felhasználók közötti üzenetek továbbítását és kézbesítését akkor is, ha a címzett éppen nincs kapcsolatban a hálózattal. Manapság olyan protokollokat használnak, mint az **SMTP** (küldésre) és az **IMAP/POP3** (fogadásra). Az e-mail szolgáltatások gyakran tartalmaznak **TLS** titkosítást.
- **FTP** (File Transfer Protocol) szolgáltatás, amely lehetővé teszi a felhasználó számára a távoli számítógép könyvtárszerkezetének manipulálását, beleértve a fájlok könyvtárakba/könyvtárakból történő fel- és letöltését anélkül, hogy a felhasználónak tudnia kellene, hogy a távoli számítógép milyen fájlrendszert használ, azaz ismernie kellene az egyes parancsokat. Az FTP-t lassan felváltják az olyan biztonságosabb protokollok, mint az SFTP (SSH File Transfer Protocol), amelyek titkosítást biztosítanak az adatátvitelhez.

Az internetnek számos olyan aspektusa van, amely kritikus a vállalati rendszerek szempontjából. Ezek a következők:

- A **proxykiszolgálók** elválasztják a belső hálózatot az internettől. (További információkért lásd a számítógépes hálózatok biztonságáról szóló fejezetet.)
- A **VPN** (Virtual Private Network) technológia kulcsfontosságú azon vállalatok számára, amelyeknek biztonságos távoli hozzáférésre van szükségük a belső hálózatokhoz. Lehetővé teszi az alkalmazottak számára, hogy biztonságosan dolgozzanak otthonról vagy más helyről, miközben a kapcsolat titkosított és biztonságos marad. A VPN a nyilvános hálózatok használata során az érzékeny vállalati adatok védelme szempontjából is fontos.
- A **tűzfalak** olyan biztonsági rendszerek, amelyek szabályozzák a bejövő és kimenő hálózati forgalmat. Segítségükkel megakadályozható az illetéktelen hozzáférés, és védelmet nyújtanak a hálózatot fenyegető támadásokkal szemben. Lehetnek szoftveralapúak (pl. operációs rendszerbe építve), vagy hardveres eszközök is. További információkért lásd a számítógépes hálózatok biztonságáról szóló fejezetet.
- A **web**es **API**-k és integrációk (alkalmazás-programozási interfészek) lehetővé teszik, hogy a különböző rendszerek az interneten keresztül kommunikáljanak egymással. A vállalatok az API-kat arra használják, hogy integrálják vállalati rendszereiket különböző külső alkalmazásokkal, például fizetési kapukkal, ERP-rendszerekkel, CRM-rendszerekkel és egyéb eszközökkel.

- Az adatátvitel során az adatok titkosítása kulcsfontosságú az érzékeny vállalati információk védelme érdekében (**SSL/TLS**, végponttól végpontig terjedő titkosítás). A **HTTPS** ma már a biztonságos webböngészés szabványa.
- A tartalomszolgáltató hálózatokat (**CDN**) globálisan elérhető webes alkalmazásokat kezelnek, a CDN-eket a teljesítmény javítására használhatják a tartalom több, földrajzilag elosztott szerveren történő tárolásával. A CDN-ek felgyorsítják a tartalomkézbesítést és csökkentik a vállalati szerverek terhelését, ami fontos a nagy adatmennyiséget feldolgozó rendszerek esetében.
- **IDS/IPS** (Intrusion Detection/Prevention Systems). Bővebb információért lásd a számítógépes hálózatok biztonságáról szóló fejezetet.

Lehet, hogy a leírtakat szakmailag kissé részletesebben fogalmaztuk, és ezért nehéz volt elsőre megérteni az összes protokollt és technológiákat, de ennek ellenére nagyon jól láthattuk a titkosításnak a fontosságát az internet szinte összes aspektusában. Manapság sok érzékeny, személyes, de fontos vállalati adatot is interneten keresztül küldünk, felhőben tároljuk és nem egy esetben a vállalati rendszerünk is online térbe van elhelyezve. Napjainkban nagy hangsúlyt fektetünk a személyi adatok védelmére, előírtuk a GDPR-t, törvénnyel büntetjük a megszegését. Viszont, egy kérdés még mindig nyitva maradt: *Ki ellenőrzi a rendszereket?* (Szabó, B.; Márföldi, E. 2011) (Herdon, M. 2016) (Juhász, S. 2011) (Berecz; A.; Karácsony, P.; Kónya, L.; Peck T, T.; Szász, G.; Vári-Kakas, I. 2019)

4.4 Adatbázisok

Az adatok tárolása két elven történhet:

- **Fájl rendszer** segítségével – egy objektum adatai vagy az azonos osztályba tartozó objektumok adatai külön fájlba kerülnek.
    - Ilyen lehet például egy fénykép, egy szöveges dokumentum (szerződés, kérelem, stb.), de speciálisan szervezett adatok is, mint például a kiállított számlák könyve, amely minden egyes kiállított számla nyilvántartásából áll.

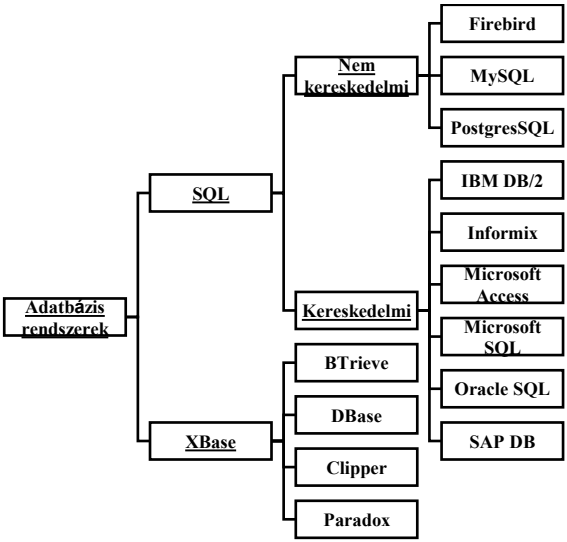
  - Az adatokhoz hozzáférő alkalmazások az operációs rendszer fájlrendszer szolgáltatásait használják, és a következő műveletek állnak rendelkezésre:
    - fájl megnyitása és bezárása;
    - új fájl létrehozása és a fájl törlése;
    - a teljes fájl olvasása, egy rekord vagy meghatározott számú bájt olvasása;
    - írás a fájlba.
  - A fájl az adatok önálló egysége, és könnyen hordozható. Ezt a tulajdonságot olyan helyzetekben használjuk, amikor az adatokat könnyen át kell vinnünk egy másik helyre.
  - A fájlrendszer **egyszerűbb adatszerkezetek tárolására alkalmas, és gyakran használják egymástól független objektumok** (pl. dokumentumok, képek) **tárolására**. Az adatbázis rendszert ezzel szemben úgy tervezték, hogy lehetővé tegye az adatok közötti kapcsolatok kezelését (pl. rendelések és ügyfelek közötti kapcsolatok).
- **Adatbázis** megvalósításssal – megoldja a fájl hozzáférés néhány problémáját, mint például a fájl megközelítésben csak nagy nehézségek árán tudjuk megragadni a különböző objektumok közötti kölcsönös kapcsolatokat, pl. a megrendelés és a vevő kapcsolata, a megrendelés és az áru, stb.; vagy a fájl megközelítésben a strukturált és a strukturálatlan adatokat csak nehezen tudjuk egymással kombinálni.
  - Az adatbázis rendszer racionalizálja és egyszerűsíti az adatkezelést, különösen a nagy és összetett adathalmazok esetében. **Az adatbázisok lehetővé teszik a könnyű manipulációt** (például az adatok lekérdezésével), **és csökkentik az adatok redundanciáját** (feleslegességét).

Az adatbázis strukturált információk vagy adatok szervezett gyűjteménye, amelyet általában elektronikusan tárolnak egy számítógépes rendszerben. Az adatbázisokat általában **adatbázis-kezelő rendszer** (DBMS – Database management System) kezeli. Az adatokat és a DBMS-t, valamint a hozzájuk kapcsolódó alkalmazásokat együttesen **adatbázis rendszernek** nevezzük[23]. Tehát, ezek alapján bátran állíthatjuk, hogy az adatbázis az információk gyűjtésének és rendszerezésének eszköze[24]. Adatbázis-kezelő rendszer olyan speciális szoftverrendszer, amellyel lehetséges az adatbázisokban tárolt adatok:

[23] Forrás: Oracle - https://www.oracle.com/sk/database/what-is-database/
[24] Forrás: Microsoft - https://support.microsoft.com/sk-sk/topic/

- definiálása,
- kezelése,
- karbantartása,
- felügyelete.

MS Office csomagon belül az MS Access egy egyszerűbb adatbázis alkalmazás. A gyakorlatban találkozhatunk például MS SQL Server[25], Oracle, vagy MySQL adatbázis-kezelő rendszerekkel is.



4.7. ábra: Néhány példa a relációs adatbázisokra – SQL, és fájlorientált adatbázisokra – Xbase.
*Leírás: **XBase, DBase, Clipper, Paradox** – az XBase formátumon alapulnak, és többnyire fájl orientált, egyszerűbb felépítésű adatbázisok. Nem relációsak, bár sokuk alapszinten támogatja a relációs funkciókat. **BTrieve** – eredetileg egy nem-relációs adatbázis rendszer. A modern változatok (Pervasive.SQL) támogatják az SQL-kiterjesztéseket, de maga a Btrieve nem hagyományos értelemben vett relációs adatbázis.*
Forrás: https://spseke.sk/tutor/prednasky/dbs/dbs.html

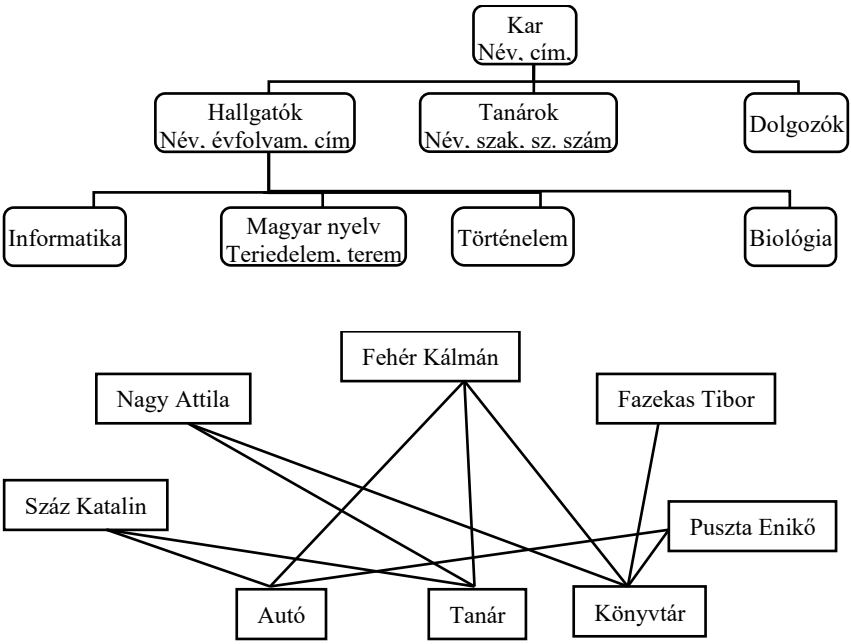
Az adatbázis-kezelő rendszerekkel szemben a következő követelményeket definiálhatjuk:

- **adatintegritás** (az adatok és kapcsolatok helyessége, érvényessége és ellentmondás mentessége = megbízhatóság),
- **rugalmasság** (az adatok egyszerű módosíthatósága),

[25] A nagyobb és összetettebb vállalati megoldásokhoz gyakran olyan robusztusabb rendszert használnak, mint az MS SQL Server.

- **hatékonyság** (gyors keresés és módosítás),
- **adatfüggetlenség** (szoftverfüggetlenség – „perzisztencia”, hardverfüggetlenség és forrásfüggetlenség),
- **adatvédelem** (illetéktelen felhasználókkal szemben – felhasználói jogok),
- **osztott hozzáférés** (több felhasználó egyidejű hozzáférésének biztosítása),
- **adatbiztonság** (védelem hardver és szoftverhibákkal szemben),
- **redundancia védelem.**

Mint fentebb említettük, a modern adatbázisok lehetővé teszik a különböző rendszerek közötti adat hordozhatóságot az adatok integritásának elvesztése nélkül!



4.8. ábra: Példák a hierarchikus (fent) és hálós (lent) adatbázis struktúrákra
Forrás: http://www.agt.bme.hu/szakm/adatb/db1.htm

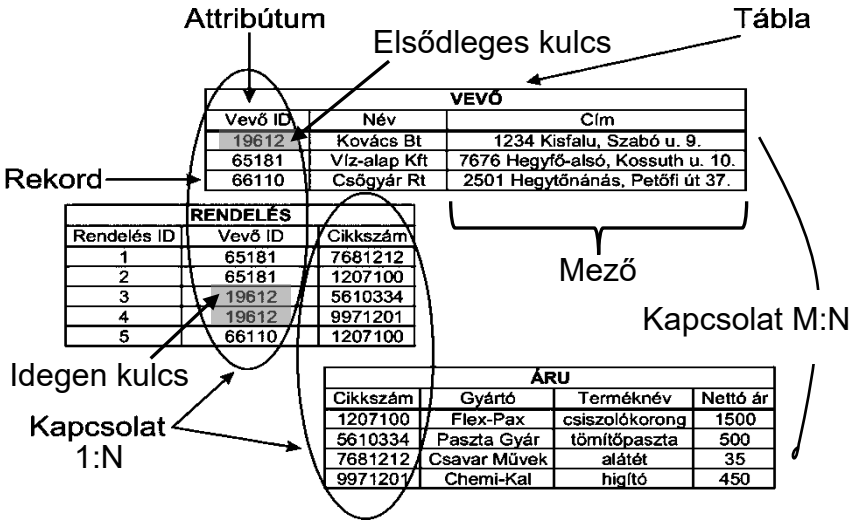
Három alap adatbázis modellt ismerünk: **hierarchikus adatbázis** modellt – struktúrája hasonlít az operációs rendszer mappa szerkezetére, **hálós adatbázis** modellt – struktúrája hasonlít a weboldalakon használt linkekre, melyek összekapcsolják az oldalakat egymás között, és **relációs adatbázis** modellt – alapja a táblák és a köztük lévő kapcsolatok kifejezése és

kezelése. (Szabó, B.; Márföldi, E. 2011) (Herdon, M. 2016) (Juhász, S. 2011) (Berecz; A.; Karácsony, P.; Kónya, L.; Peck T, T.; Szász, G.; Vári-Kakas, I. 2019)

4.4.1 Relációs adatbázisok

A relációs adatbázisok a kapcsolatok matematikai elvén alapulnak. A relációs adatbázisok alapelvei E. F. Codd fogalmazta meg 1970-ben. Ez napjaink legelterjedtebb adatbázis modellje. Az adatok tárolása táblázatos formában, a táblázat soraiban és oszlopaiban valósul meg.

A táblák oszlopokból és sorokból állnak, egyedi névvel rendelkeznek. **Rekordoknak** a tábla sorait, **mezőknek** a tábla oszlopait fogjuk nevezni. **Attribútum** alatt tulajdonságot fogunk érteni (tehát a tábla mezőit), **egyed** alatt mind azt, amit le akarunk írni, aminek az adatait tároljuk és gyűjtjük az adatbázisban. Az egyed az attribútumok összességével jellemezhető.



VEVŐ

Vevő ID	Név	Cím
19612	Kovács Bt	1234 Kisfalu, Szabó u. 9.
65181	Víz-alap Kft	7676 Hegyfő-alsó, Kossuth u. 10.
66110	Csőgyár Rt	2501 Hegytőnánás, Petőfi út 37.

RENDELÉS

Rendelés ID	Vevő ID	Cikkszám
1	65181	7681212
2	65181	1207100
3	19612	5610334
4	19612	9971201
5	66110	1207100

ÁRU

Cikkszám	Gyártó	Terméknév	Nettó ár
1207100	Flex-Pax	csiszolókorong	1500
5610334	Paszta Gyár	tömítőpaszta	500
7681212	Csavar Művek	alátét	35
9971201	Chemi-Kal	higító	450

4.9. ábra: Relációs adatbázis modell
Forrás: http://earth.geo.u-szeged.hu/~laci/ab-Geoinfo-tananyag/ch02s03.html

Ha egy adott tábla rekordjai valamilyen módon egy másik tábla rekordjaihoz társíthatók, azt mondjuk, hogy a táblák között kapcsolat áll fenn. Relációs adatbázis modelleknél három típust ismerünk: „egy-az-egyhez” (1:1), „egy-a-többhöz” (1:N), „több-a-többhöz” (N:M). Két tábla

akkor áll **egy-az-egyhez** kapcsolatban, ha az első tábla egy rekordja a második tábla egyetlen rekordjához kapcsolódik, és a második tábla egy rekordja is egyetlen rekorddal áll kapcsolatban az első táblából. Ebben a fajta kapcsolatban az egyik tábla elsődleges tábla, míg a másik másodlagos tábla lesz. Ha két tábla **egy-a-többhöz** kapcsolatban áll, akkor az első tábla egy rekordja a második tábla több rekordjához is kapcsolódhat, de a második tábla egy rekordja csak egyetlen rekorddal állhat kapcsolatban az első táblából. Két tábla akkor áll **több-a-többhöz** kapcsolatban, ha az első tábla egy rekordja a második tábla több rekordjához kapcsolódik, és a második tábla egy rekordja is több rekorddal áll kapcsolatban az első táblából. Ezt a kapcsolatot helyesen úgy hozzuk létre, hogy létrehozunk egy úgynevezett kapcsolótáblát (junction).

Az adatbázis rendszerek fontos képessége az adatok integritásának biztosítása. Megfogalmazott állítások segítségével korlátozzák a lehetséges attribútumértékeket vagy a rekordok lehetséges manipulációit, amelyek egy másik tábla rekordjaival kapcsolatban léteznek. Az adatintegritás létfontosságú az adatok konzisztenciájának fenntartásához, különösen akkor, ha több felhasználó egyidejűleg fér hozzá az adatbázishoz. Az integritás korlátozások alaptípusai közé tartozik az entitásintegritás, a hivatkozási integritás és a doménintegritás.

- Az **entitásintegritás** annak biztosítására összpontosít, hogy a táblában ne tagolódjanak duplikált rekordok. Ez általában egy olyan elsődleges kulcs[26] meglétéhez kapcsolódik, amely egyedileg azonosítja a táblázatban lévő rekordot. Például a Student tábla elsődleges kulcsára példa az ID_Student attribútum. Az idegen kulcsok valójában más táblák elsődleges kulcsai, amelyeket a vizsgált táblába átadnak.
- Az idegen kulcsok[27] létezése az úgynevezett **hivatkozási integritás** szempontjából fontos. A hivatkozási integritás elve biztosítja a következőket:
    - egy idegen kulcsként megjelölt attribútum értéke nem illeszthető be olyan attribútum értékeként, amelynek nincs megfelelő attribútumértéke abban a táblában, amelynek a kulcsa idegen kulcsként szerepel.

[26] Elsődleges kulcs: a táblázat rekordjainak egyértelmű azonosítója, értéke egyedi. Lehet egyetlen mezőből álló elsődleges kulcs vagy összetett elsődleges kulcs, melyet több mező felhasználásával képezünk.

[27] Idegen kulcs: olyan azonosító, amelynek segítségével egy másik táblázat elsődleges kulcsára hivatkozhatunk. Egy mezőben előfordulhat egyszer – 1:1 kapcsolat, vagy többször – 1:N kapcsolat.

  - egy olyan rekord, amelynek értéke idegen kulcsként szerepel egy másik táblában, nem távolítható el a táblából.
- Az integritási korlátozások utolsó típusa a **doménintegritás**, amely biztosítja, hogy a megadott adatokat attribútum értékként meghatározott adatok egy meghatározott megengedett értékkészletből kerülnek kiválasztásra. Például, egy tábla darabszám mezőjébe ne lehessen beírni karaktereket, csak számokat. Ezt úgy biztosítjuk, hogy minden mezőhöz hozzárendelünk adattípust, mint pl. szám, rövid szöveg, dátum és idő, stb.

Az adatbázis belső fizikai struktúrája rejtve marad az adatbázis-kezelő rendszer mögött álló programok és felhasználók számára. Különböző gyártók termékeiben számos adatbázis rendszerre implementálták a relációs adatbázis modellt (Oracle, MS SQL Server, MySQL, stb.). Az adatbázisban lévő adatok elérésének lehetősége miatt kialakult egy szabvány, amelyet **SQL**-nek (**Structured Query Language**) neveznek. Ezen a szabványon belül négy parancscsoportot sorolunk fel.

- **Data Manipulation Language** (DML) parancsok – az adatok adatbázisból történő lekérdezésére és módosítására szolgálnak.
  - SELECT – biztosítja az adatok kiválasztását az adatbázis tábláiból, beleértve azok rendezését és csoportosítását;
    - Pl., adatok visszaadása az „Ügyfelek” táblából: *SELECT CustomerName, City FROM Customers;*
  - INSERT – adatok beszúrása az adatbázis tábláiba;
    - Pl., a következő SQL utasítás új rekordot szúr be, de csak az „Ügyfélnév”, „Város” és „Ország” oszlopokba (az ügyfél azonosítója automatikusan frissül):
      INSERT INTO Customers (Name, City, Country)
      VALUES ('Cardinal', 'Stavanger', 'Norway');
  - UPDATE – az adatbázis táblák adatainak módosítása;
    - Pl., a következő SQL utasítás frissíti az első ügyfelet (CustomerID = 1) egy új kapcsolattartó személyel és egy új várossal:
      UPDATE Customers
      SET ContactName = 'Alfred', City = 'Frankfurt'
      WHERE CustomerID = 1;
  - DELETE – adatbázis táblák adatainak törlése.
    - Pl. a következő SQL utasítás törli az „Alfreds Futterkiste” ügyfelet az „Ügyfelek” táblából:

DELETE FROM Customers WHERE CustomerName = 'Alfreds Futterkiste';

- **Data Definition Language** (DDL) parancsok – az adatbázis sémájának meghatározására és módosítására szolgálnak.
  - CREATE - lehetővé teszi új objektum létrehozását az adatbázisban, pl. egy táblát, egy táblák közötti kötést, stb.;
    - Pl., a következő eset létrehoz egy „Személyek” nevű táblát, amely öt oszlopot tartalmaz: „SzemélyID”, „Vezetéknév”, „Keresztnév”, „Cím” és „Város”:
      CREATE TABLE Persons (
      PersonID int,
      LastName varchar(255),
      FirstName varchar(255),
      Address varchar(255),
      City varchar(255));
  - ALTER – lehetővé teszi az adatbázis objektumai, vagyis az adatbázis szerkezetének, sémájának megváltoztatását;
    - Pl. a következő SQL hozzáad egy „Email” oszlopot az „Ügyfelek” táblához:
      ALTER TABLE Customers
      ADD Email varchar(255);
  - DROP – lehetővé teszi az adatbázisban lévő objektumok törlését, azaz az adatbázis szerkezetének, sémájának megváltoztatását.
    - Pl., a következő SQL utasítás törli a meglévő „Szállítók” táblát:
      DROP TABLE Shippers;
- **Data Control Language** (DCL) parancsok – az adatbázishoz és az adatokhoz való hozzáférési jogok, valamint a tranzakció vezérlés meghatározására szolgálnak.
  - GRANT – lehetővé teszi az adatokhoz vagy funkciókhoz való hozzáférési jogok hozzárendelését egy meghatározott adatbázis felhasználóhoz;
    - Pl.:
      GRANT SELECT, INSERT ON Customers TO 'user_name';
  - REVOKE – lehetővé teszi a meghatározott felhasználó hozzáférési jogainak visszavonását;
    - Pl.:
      REVOKE SELECT ON Customers FROM 'user_name';

  - BEGIN – lehetővé teszi egy tranzakció kezdetének meghatározását;
  - COMMIT – lehetővé teszi a tranzakció megerősítésének meghatározását;
  - ROLLBACK – lehetővé teszi az adatbázis visszaállítását, amennyiben a tranzakció során hiba történt, még a tranzakció megkezdése előtt.
- **Egyéb parancsok** – a felhasználók meghatározására, szabályok megfogalmazására, pl. az adatok nemzeti ábécék szerinti rendezésére, stb.

Az SQL utasításokat a szoftverfejlesztők építik be az alkalmazásokba, de a végfelhasználók is használhatják őket. Mivel a végfelhasználók nem igénylik a nyelv szintaxisának és szemantikájának részletes ismeretét, az adatbázis rendszerek fejlesztői olyan alkalmazást adnak hozzá a rendszereikhez, amely lehetővé teszi a felhasználók számára, hogy egy grafikus felület segítségével férjenek hozzá az eredmények lekérdezéséhez, lényegében intuitív módon. Az ilyen eszközöket nevezik *Query By Example* (QBE) – lekérdezés példa alapján. QBE egy grafikus eszköz, amely lehetővé teszi a felhasználók számára, hogy SQL-lekérdezéseket hozzanak létre anélkül, hogy el kellene sajátítaniuk az SQL-szintaxist. (Szabó, B.; Márföldi, E. 2011) (Herdon, M. 2016) (Juhász, S. 2011) (Berecz; A.; Karácsony, P.; Kónya, L.; Peck T, T.; Szász, G.; Vári-Kakas, I. 2019)

4.4.1.1 XML, JSON

```
<!DOCTYPE html>
<html>
<body>
<h1>Első címem</h1>
<h2>másik címem</h2>
<p>És ez az első szövegem.</p>
</body>
</html>
```

Első címem

másik címem

És ez az első szovegem.

4.10. ábra: Egy egyszerű html kód (balra) és annak böngészőben való megjelenítése

Amikor az alkalmazás által feldolgozott adatokat továbbítani kell egy másik alkalmazásnak (például, a rendelési rendszer alkalmazásának át kell adnia az ügyfélre vonatkozó adatokat, amely értékeli az ügyfelet, és

meghatározott szintű árengedményt rendel az ügyfélhez), megfelelő átadási formátumot kell használni. Elsősorban a különböző rendszerek és alkalmazások közötti, esetleg eltérő környezetben megvalósított adattovábbítás egyszerűsége miatt kerül ismét előtérbe a közös szöveges fájl, illetve az ilyen fájlban tárolt adatok. Ilyen problémák megoldására a legtöbb alkalmazás lehetővé teszi az adatok fájlban történő jelölésének elvét, pl. <név>Jan Novák</név>, ahol <név> a kezdőjelölés, </név> pedig azt mondja, hogy az adott adatjelölés itt ér véget. A jelölés használatának kiindulópontja az **SGML** (Standard Generalized Markup Language, ISO 8879). A jelölés elvének életképességét a World Wide Web, illetve a **HTML** (HyperText Markup Language) használata bizonyította, amely képes egy adatszerkezetet úgy leírni, hogy az különböző környezetekben (különböző böngészőkkel) megtekinthető legyen, és a böngésző megérti ezt a struktúrát. A bevált elv aztán a továbbfejlesztés alapjává vált, amit a következők képviselnek **XML** (eXtensible Markup Language) és annak különböző alkalmazásai. Például, a 4.10 ábrán levő HTML kódot próbáljuk meg begépelni a jegyzettömbbe (Notepad), és mentsük el *minta.html* név alatt. A fájlt nyissuk meg a böngészőnkben. Az alkalmazások emberi felhasználó beavatkozása nélkül cserélhetnek adatokat. Az XML lehetővé teszi a különböző rendszerek és alkalmazások közötti interoperabilitást, ami létfontosságú a modern vállalati információs rendszerek számára. Az XML platformfüggetlen, és lehetővé teszi, hogy az adatokat különböző platformokon működő alkalmazások között információ vesztés nélkül lehessen kicserélni. Ez különösen fontos azokban a rendszerekben, ahol heterogén alkalmazások közötti kommunikációra van szükség. Az XML mellett a modern adatcserében gyakran használnak **JSON**-t (JavaScript Object Notation), amely könnyebb és olvashatóbb formátum, különösen a webes környezetekben és API-k használatakor. Az alkalmazások közötti, emberi beavatkozás nélküli adatátvitel kapcsán megemlíthetünk még további webszolgáltatásokat és API-kat:

- A **SOAP** (Simple Object Access Protocol) XML-t használ a rendszerek közötti strukturált információ cserére. A SOAP egy protokoll, és szigorú formázási szabályokkal rendelkezik, főleg XML használatával. Alkalmas összetett műveletekhez és vállalati alkalmazásokhoz.
- A **REST** (Representational State Transfer) gyakran használ JSON-t vagy XML-t, és népszerű a modern webes API-k esetében. A REST egy olyan architektúrális stílus, amely szabványos HTTP-módszereket és gyakran JSON-t használ. Egyszerűbb, gyorsabb és alkalmas a legtöbb webes és mobilalkalmazáshoz. (Gála, L.; Pour, J.; Šedivá, Z. 2015) (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý,

P. 2017) (Sodomka, P.; Klčová, H. 2010) (Juhász, S. 2011) (Novotný, O.; Pour, J.; Slánský, D. 2005)

Tegyük fel, hogy időjárási információkat szeretnénk lekérni a REST API-ból. Használhatjuk a HTTP GET módszert. A lekérdezés így nézne ki:

```
GET
http://api.weather.com/v1/current?city=Bratislava&country=Slovakia
```

A kiszolgáló például JSON formátumban válaszolhat:

```
{
  "city": "Bratislava",
  "country": "Slovakia",
  "temperature": "18°C",
  "condition": "Cloudy"
}
```

SOAP esetében a lekérdezés:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ws="http://www.webserviceX.NET">
   <soapenv:Header/>
   <soapenv:Body>
      <ws:GetWeather>
         <ws:CityName>Bratislava</ws:CityName>
         <ws:CountryName>Slovakia</ws:CountryName>
      </ws:GetWeather>
   </soapenv:Body>
</soapenv:Envelope>
```

A kiszolgáló hasonló XML formátummal válaszol:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
   <soapenv:Header/>
   <soapenv:Body>
      <GetWeatherResponse xmlns="http://www.webserviceX.NET">
         <GetWeatherResult>Cloudy</GetWeatherResult>
      </GetWeatherResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

4.4.2 Elosztott adatbázisok

Az elosztott adatbázis egyetlen logikai adatbázisként definiálható, amely fizikailag több, adatkommunikációs hálózattal összekapcsolt számítógépen van elosztva. Az elosztott adatbázis rendszeren olyan automatizált rendszert értünk, amely több csomópontból áll, amelyeket egy kommunikációs csatorna köt össze. A csomópontok egyenrangúak, és az egyes csomópontoknak nem kell azonos konfigurációval és teljesítménnyel rendelkezniük. A rendszer úgy viselkedik, hogy a felhasználó nem ismeri a többi csomópontot. Ezt a tulajdonságot „distribution transparency"-nek nevezik. A felhasználó nem tudja, hogy az adatokat fizikailag hol tárolják, ami az alkalmazások és a felhasználók szempontjából megkönnyíti a rendszer használatát. Ebben az összefüggésben a csomópontra úgy fogunk hivatkozni, mint egy megfelelő szoftverrel és adattárolási képességgel rendelkező számítógépre.

Azokban az alkalmazásokban, ahol elosztott adatbázis rendszert alkalmaztak, a központosított rendszerekkel szemben bizonyos előnyöket tapasztaltak:

- A rendszer megbízhatósága nagyobb, mivel, ha egy csomópont összeomlik, a többi csomópont tovább tud működni. A nagyobb megbízhatóság annak köszönhető, hogy az adatok több helyen is replikálhatók. Így, ha az egyik csomópont meghibásodik, a többi továbbra is képes adatokat szolgáltatni.
- Alacsonyabb kommunikációs költségek a kisebb átvitt adatmennyiség miatt.
- Rövidebb válaszidő, mivel a legtöbb lekérdezést helyben kezelik.
- Új csomópontot lehet bevezetni a meglévő konfiguráció megváltoztatása nélkül.
- Földrajzi közelség - Az elosztott rendszerek földrajzi elhelyezkedés alapján optimalizálhatók. Az adatokat közelebb lehet tárolni a felhasználási helyhez, ami tovább javítja a teljesítményt és csökkenti a késleltetést.

Az előnyök mellett hátrányok is felmerültek, melyek közül kiemelnénk néhányat.

- Magasabb pénzügyi költségek mind a karbantartás, mind a beszerzés tekintetében, amelyek a több hardvererőforrásból adódnak.
- Az adatok konzisztenciájának és frissítésének fenntartása, amely az elosztott feldolgozásból adódik.

- A több csomópont közötti szinkronizálás bonyolult lehet, és ismét adatkonzisztencia-problémákhoz vezethet.
- Technikai irányítási követelmények – az adatok elosztása és szinkronizálása kifinomult algoritmusokat igényel a konzisztencia fenntartásához (például a csomópontok közötti tranzakciók esetében).

Létezik egy **CAP-elmélet** (Consistency, Availability, Partition tolerance), amely az elosztott adatbázisok egyik kulcsfogalma. A CAP-elmélet szerint egy rendszer nem rendelkezhet egyszerre mindhárom tulajdonsággal (konzisztencia, elérhetőség, a hálózat szétválasztásának ellenállása), ezért kompromisszumot kell kötni. Az elosztott adatbázisok például gyakran feláldozzák a konzisztenciát a rendelkezésre állás és a hibatűrés javára, ami azt eredményezheti, hogy az adatok különböző csomópontokon elavulhatnak (esetleges konzisztencia).

Az elosztott adatbázis rendszerek két változatban épülnek fel:

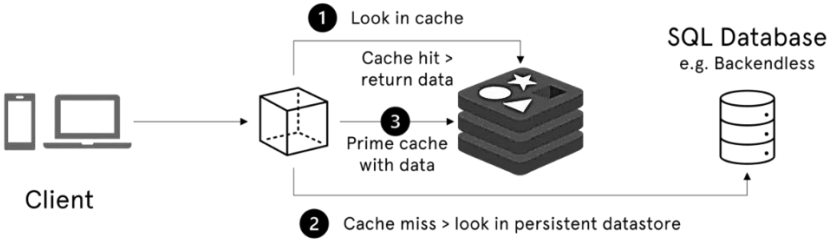
- Az elosztott adatbankok rendszere egy vagy több rendszer, különböző szintjein felépített, egymással kölcsönhatásban lévő adatbázisok rendszere.
- Egyetlen adatbank elosztott adatbankok rendszere – az egyes csomópontokban tárolt adatok vagy adatbázisrészek adatbázisai.

Megemlíthetünk néhány gyakran használt modern elosztott adatbázis rendszert: **Google Spanner** – elosztott relációs adatbázis, amely magas konzisztenciát garantál, **Amazon DynamoDB** - NoSQL adatbázis, amely a konzisztenciát feláldozza a magas rendelkezésre állásért, **Cassandra** – horizontális skálázásra optimalizált elosztott NoSQL adatbázis.

A vállalaton múlik, hogy melyik adatbázis alkalmazást választja, központosítottat vagy elosztottat. A funkcionális jellemzők a rendszerben megvalósított adatbázis-kezelő rendszertől függnek. (Szabó, B.; Márföldi, E. 2011) (Herdon, M. 2016) (Juhász, S. 2011) (Berecz; A.; Karácsony, P.; Kónya, L.; Peck T, T.; Szász, G.; Vári-Kakas, I. 2019)

4.4.3 In-memory adatbázisok – valós idejű adatfeldolgozás

A mai vállalkozások számára a valós idejű információk rendkívül fontosak, különösen azokban az iparágakban, ahol az időérzékeny döntéshozatal kulcsfontosságú. A számítógépes rendszereknek gyorsabbnak és hatékonyabbnak kell lenniük, mint a hagyományos megoldások, hogy lépést tudjanak tartani a gyorsan változó világgal. Az in-memory adatbázisok erre kézenfekvő megoldást nyújtanak, mivel az adatokat közvetlenül a RAM-ban tárolják, nem pedig merevlemezeken vagy SSD-n, ami rendkívül alacsony késleltetést biztosít. Bár a memória korábban túl drága volt, a költségek csökkenése és az új rendszerek fejlesztése lehetővé tette az in-memory technológia széleskörű alkalmazását. Az ilyen adatbázisok ideálisak valós idejű adatfeldolgozást igénylő alkalmazásokhoz, például pénzügyi szolgáltatásokhoz, valós idejű elemzésekhez és nagy teljesítményű számítástechnikához. Egy népszerű in-memory adatbázis a **Redis**, amely nemcsak gyors adatkezelést biztosít, hanem fejlett adatstruktúra támogatást is kínál különféle felhasználási esetekhez. A Redis nem relációs adatbázis. A Redis egy NoSQL adatbázis, amelyet gyakran in-memory key-value store-ként használnak. Ez azt jelenti, hogy az adatokat kulcs-érték párokban tárolja, és a hagyományos relációs adatbázisoktól eltérően nem használ relációs táblákat vagy SQL-t a működéséhez. A Redis adatbázist gyorsítótárazásra használják (*caching*) az alkalmazások teljesítményének növelésére. Ilyen adatbázist támogató rendszer a **SAP HANA** is. (Microsoft2, 2024) (SAP1, 2024) (SAP2, 2024)



4.11. ábra: Redis adatbázis tipikus működése
Forrás: https://backendless.com/redis-what-it-is-what-it-does-and-why-you-should-care/

4.4.4 NoSQL adatbázisok

A NoSQL adatbázisokat úgy tervezték, hogy elkerüljék a relációs adatbázisok fix sémájának korlátait. Ez különösen előnyös dinamikusan változó vagy strukturálatlan adatok esetén. Ezek az adatbázisok különféle adatstruktúrákat használnak, amelyek lehetővé teszik az adatok rugalmas és gyors kezelését. Például:

- **Dokumentumokat**
  - Az adatok JSON vagy XML formátumban vannak tárolva, amely strukturált és félig strukturált adatokat is képes kezelni.
- **Kulcs-értékpárokat**
  - Az adatok egyedi kulcsok alapján érhetők el, amelyekhez egy adott érték van hozzárendelve. Pl. egy online áruház termékadatai kulcsként a termékazonosítót (pl. product_765), értékként pedig a termék részletes információit (pl. név, ár, leírás) tartalmazhatják.
- **Széles oszlopokat**
  - Egy táblázathoz hasonló struktúrában az adatok oszlopokba rendezhetők, de az oszlopok dinamikusan változhatnak és különböző rekordok (sorok) esetén eltérhetnek. Ez a modell hatékony a nagy adathalmazok kezelésében. Pl. egy IoT-eszközökről származó érzékelési adatokban minden sor egy adott időbélyeghez kapcsolódik, de az oszlopok eltérő érzékelőket tartalmazhatnak.
- **Gráfokat**
  - Az adatok csomópontokként (pontokként) és élekként (kapcsolatokként) vannak ábrázolva, amelyeket különböző tulajdonságokkal ruháznak fel. Pl. egy gráfban egy "A vásárló" és egy "B termék" között az él azt jelezheti, hogy az adott vásárló megvásárolta az adott terméket.

A NoSQL adatbázisok nagy adatmennyiséget és gyors méretezhetőséget igénylő alkalmazások esetén ideálisak, például a Big Data feldolgozás vagy valós idejű alkalmazások esetében. Bár a NoSQL adatbázisok többsége nem biztosít teljes körű ACID támogatást, néhány modern megoldás már képes erre, például a MongoDB. Azonban olyan rendszerekben, ahol az adatintegritás kritikus, például banki és ERP alkalmazások, továbbra is a relációs adatbázisok az elsődleges választás.

E fejezet utolsó részében még el kell magyaráznunk, mi is az ACID. Relációs adatbázisok egyik fontos jellemzője (*Atomicity, Consistency,*

Isolation, Durability), amely biztosítja a tranzakciók magas szintű megbízhatóságát a következő tulajdonságok alapján.

- **Atomicity (atomitás):** egy tranzakció minden része sikeresen lefut, vagy egyik sem. Például, ha pénzt utalunk egyik számláról a másikra, a pénz mindkét helyen megfelelően frissül, vagy sehol sem történik változás.
- **Consistency (konzisztencia):** az adatok mindig érvényesek maradnak, a tranzakciók előtt és után is megfelelnek az adatbázis szabályainak.
- **Isolation (elszigeteltség):** a párhuzamos tranzakciók nem zavarják egymást, és nem okoznak adat-inkonzisztenciát.
- **Durability (tartósság):** a tranzakciók eredményei tartósan tárolódnak, még egy rendszerhiba esetén is. (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P. 2017) (Gála, L.; Pour, J.; Šedivá, Z. 2015) (Sodomka, P.; Klčová, H. 2010) (Juhász, S. 2011) (Novotný, O.; Pour, J.; Slánský, D. 2005)

4.4.5 Egyéb adatbázisok

Manapság több típusú adatbázist is használnak. Más, kevésbé elterjedt adatbázisok nagyon konkrét (tudományos, pénzügyi vagy egyéb) funkciókhoz vannak szabva. A legújabb adatbázisok közé tartoznak:

- Nyílt forráskódú adatbázisok.
- Felhőalapú adatbázisok.
- Adatbázis több modell támogatásával.
- és egyéb más adatbázis.

Az objektumorientált adatbázis (OODB – Object-Oriented Database) olyan adatbázis-kezelési rendszer, amely az objektum-orientált programozás (OOP – Object-Oriented Programming) koncepcióit integrálja az adatbázisok kezelésébe. Ez azt jelenti, hogy az adatokat és az azokhoz kapcsolódó viselkedéseket (műveleteket vagy metódusokat) objektumokként kezelik, hasonlóan ahhoz, ahogy az objektum-orientált programozási nyelvek (például Java, C++, Python) működnek.
Az adatbázisban az adatok objektumként vannak tárolva, amelyek magukban foglalják az adatokat (attribútumokat) és az azokra vonatkozó műveleteket (metódusokat). Pl. egy "repülő" objektum tartalmazhat attribútumokat, például gyártó, modell, évjárat, és metódusokat, mint

például *gyorsulás()*[28]. Az objektumok osztályokba szerveződnek, és az osztályok között öröklődési hierarchiát lehet kialakítani. (Szabó, B.; Márföldi, E. 2011) (Herdon, M. 2016) (Juhász, S. 2011) (Berecz; A.; Karácsony, P.; Kónya, L.; Peck T, T.; Szász, G.; Vári-Kakas, I. 2019)

[28] *gyorsulás()* a programozási nyelvben egy metódus. A metódus logikát hajt végre az attribútumokon, például növeli a sebességet a motor teljesítménye alapján.

5 Üzleti folyamatok, vállalati információs rendszerek és alrendszerek, jelentőségük a gazdasági területen

A folyamatot az ISO 9000 szerint úgy definiálhatjuk, mint „*az egymással összefüggő vagy egymással kölcsönösen ható tevékenységek összesége, mely a bemenetet kimenetté transzformálja át*". Ebből kiindulva mondhatjuk, hogy a folyamat olyan tevékenységek általános sorozata, amelyeket egy adott feladat elvégzésére terveztek. A szokásos üzleti gyakorlat szempontjából egy folyamat viszonylag határozatlan időtartamú, és célja egy bizonyos műveletsorozat folyamatos végrehajtása, amelynek hatására a bemenetek kimenetekké[29] alakulnak át, és ezek aztán más folyamatok tevékenységének tárgyává válhatnak. Gondoljunk csak bele, minden reggelt azzal kezdünk, hogy reggel munkába vagy iskolába megyünk. Általában ugyanazt a közlekedési eszközt választjuk, és minden alkalommal vásárolunk valamit a kioszkból. Munka vagy iskola után közlekedési eszközzel utazunk haza. Ezek mind ismétlődő tevékenységek, amelyek különböző résztevékenységeket vagy összefüggő tevékenységeket képviselnek, amelyeket a folyamatmenedzsmentben folyamatoknak nevezünk. A folyamat tehát a tevékenységek általános sorozata, amelyet egy adott feladat elvégzésére terveztek.

Folyamatokat több szempontból is kategorizálhatjuk. Pl. a folyamatok és szubjektum kapcsolata szempontjából ismerünk **belső** – cégen belüli és **külső** – cégen kívüli (pl. cégek közötti) folyamatokat. Leghasznosabb a kategorizálás összeállításához az ISO 9000 szabványból kiindulni, amely szerint ismerünk:

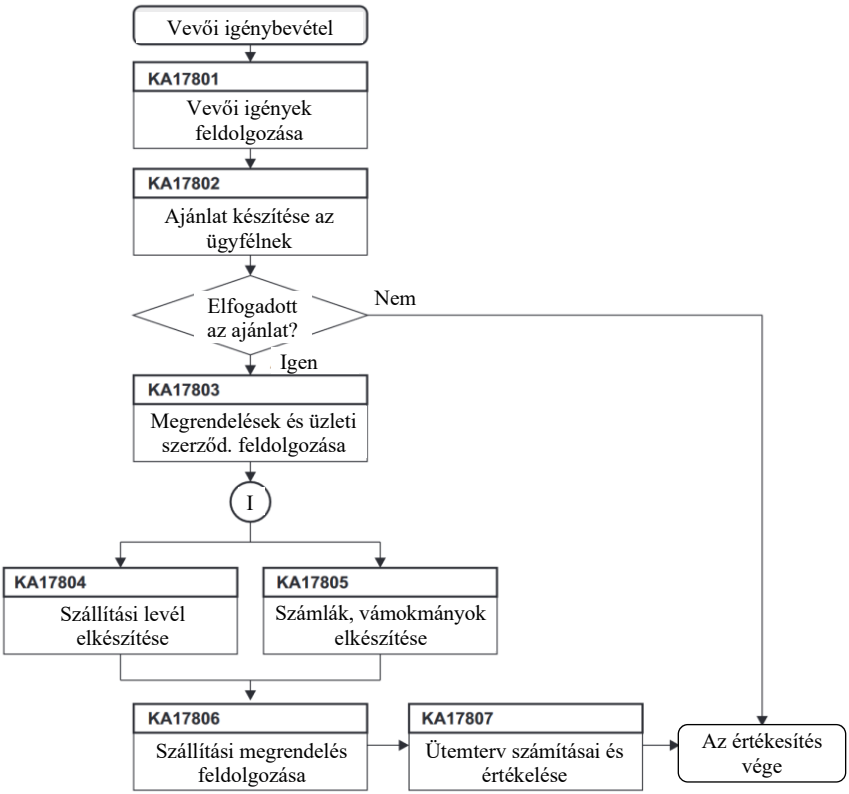
- **Irányítási (management) folyamatokat** (stratégiai tervezés, minőség- és innovációmenedzsment) – biztosítják a vállalat teljesítményének fejlesztését és irányítását, és megteremtik a belső folyamatok működésének feltételeit.

[29] Kimenet lehet termék vagy szolgáltatás. Munkaerő, anyag, technika stb. - ezeket forrásoknak nevezzük. Az ügyfél az a szubjektum, akinek a kimenetet szánjuk.

- **Alapvető (core) folyamatok** (termelés, logisztika, ügyfélkapcsolat-kezelés) – értéket teremtenek termék vagy szolgáltatás formájában az ügyfél számára.
- **Támogató folyamatok** (gazdaság, HR, IT) – biztosítják a többi folyamat működésének feltételeit azáltal, hogy tárgyi és immateriális kimenetekkel látják el őket.

A modern vállalkozás módszerei, eszközei és működése, valamint a vállalati információs rendszerek összefüggésében ezeket a folyamatokat legalább egy további folyamattal ki kell egészítenünk:

- **Innovációs folyamatok** (kutatás-fejlesztés – K+F, termékfejlesztés, piackutatás) – ezek a folyamatok az új termékek, szolgáltatások és módszerek fejlesztésére összpontosítanak, így a vállalat versenyképességét javítják.



5.1. ábra: Folyamat – példa
Forrás: (Gála, L; Pour, J; Šedivá, Z. 2015)

Szakirodalmak említenek egy ötödik, úgynevezett **kiszolgáló folyamatokat**, melyek biztosítják az alapvető és támogató folyamatokhoz szükséges anyagok, eszközök és szolgáltatások elérhetőségét.

Minden típusú folyamat fontos a vállalkozás számára, mivel mindegyiknek megvan a maga jelentősége a vállalkozás működésének biztosításában és céljainak elérésében. Még néhány rövid mondat erejéig visszatérünk a belső és külső folyamatokra. A belső folyamatok teljes mértékben a vállalkozás ellenőrzése alatt állnak, amelyekhez a vállalkozás kijelölhet egy tulajdonost, leggyakrabban egy vezetőt, aki felelős a folyamat irányításáért és folyamatos fejlesztéséért. A külső folyamatok nem állnak teljes mértékben a vállalkozás vagy a vezető ellenőrzése alatt. Ezek általában az ellátási láncokhoz vagy az ügyfélkapcsolat-kezeléshez kapcsolódó folyamatok. A fentieket figyelembe véve elmondhatjuk, hogy a folyamatoknak az alábbi jellemzői vannak (Sodomka, P.; Klčová, H. 2010):

- Megismételhető, amennyiben strukturált.
- Kimenete egy hozzáadott értékű termék vagy szolgáltatás.
- Mérhető olyan paraméterekkel, mint a minőség, a költségek, az átfutási idő, stb.
- Van tulajdonosa.
- Van egy belső vagy külső megrendelője.
- Világosan meghatározott kezdete és folytonossága van más folyamatokkal.
- A vállalat pénzügyi, anyagi és személyi erőforrásait használja fel.

A vállalkozás a környezetében bekövetkezett változások következtében a vállalkozás egyensúlyából kikerült rendszerként érzékelhető. Erre szerkezetének, állapotának és viselkedésének megváltoztatásával reagál. Azt mondjuk, hogy egy vállalkozás **életcikluson megy keresztül**. A vállalati informatika szempontjából a ciklus felfogása azért fontos, mert a különböző szakaszokban más-más igény van az információra, illetve más és más követelmények vannak annak kifejezésével és formájával, feldolgozásával és továbbításával szemben. (Koloszár, L. 2013) (Sodomka, P.; Klčová, H. 2010) (Gála, L; Pour, J; Šedivá, Z. 2015) (Gála, L; Pour, J; Šedivá, Z. 2009)

5.1 Dekompozíció / kompozíció elve

A dekompozíció / kompozíció elve azon a képességen alapul, hogy egy bizonyos „nagyobb problémát” egy sor különálló kisebb problémára

oszthatunk, majd a nagyobb problémát a megoldott kisebb problémák összeállításával oldhatjuk meg. Tegyük fel, hogy egy vállalat rendelési folyamatát szeretnénk vizsgálni. A rendelési folyamat összetett, így érdemes kisebb részfolyamatokra bontani, majd ezeket összeállítani a teljes folyamat megvalósításához. Pl:

- **Megrendelés rögzítése** – a vásárlói megrendelés rögzítése az információs rendszerben (pl. CRM rendszer).
- **Készletellenőrzés** – ellenőrizzük, hogy a rendelni kívánt termékek elérhetők-e a raktárban (pl. ERP rendszer modulja, amely a raktárkészletet és a logisztikai folyamatokat felügyeli).
- **Számlázás előkészítése** – a rendelés alapján elkészítjük a számlát (pl. pénzügyi szoftver vagy ERP rendszer pénzügyi modulja).
- **Szállítási folyamat** – a termékek kiszállítása az ügyfélhez (pl. logisztikai szoftver vagy szállítmányozási modul).

Kisebb vagy kevésbé integrált rendszerek gyakran használnak külön szoftvereket az egyes említett feladatok elvégzéséhez, ahogyan a példák is mutatják. Nagyobb, integrált rendszerek (pl. ERP) egyetlen szoftver külön moduljain belül oldják meg az összes folyamatot. Ilyen esetben minden modul egy adott feladatra specializált, de közös adatbázist használnak, így az információk könnyebben áramlanak a modulok között, és kevesebb az adatduplikáció. (Sándor, J. 2011) (Koloszár, L. 2013)

5.2 A folyamatirányítás

A sikeres vállalkozói tevékenység, a modern (tanuló) szervezetekben a dinamikusan átalakuló üzleti környezetre reagálva, minden szinten, a folyamatos fejlődés elősegítésén múlik. Nem elég a költségek csökkentése és a vezetői döntéshozatal javítása. A szisztematikus és hosszú távú innováció menedzsment az alapja a vállalat hosszú távú stabil növekedésének és piaci versenyképességének. Valószínűleg a vállalat a termékek és szolgáltatások, a partnerekkel való együttműködés és az üzleti folyamatok terén szorgalmazza az innovációt. Az innováció ezen típusai szorosan kapcsolódnak egymáshoz, és befolyásolják egymást. Az innováció minden típusa tehát hatással van az üzleti folyamatokra. Az innovációs ciklus kezelése nem igényli közvetlenül az új folyamatok bevezetését az üzlet minden területén. A változások, köztük a legjobb gyakorlatok rugalmas beépítését azonban megköveteli. A modern információs rendszerek kulcsszerepet játszanak a „Best Practices” – legjobb gyakorlatok megvalósításában, és ezáltal az üzleti folyamatok

javításában. Ez a változás a folyamatmenedzsment bevezetésének tárgya. A szervezetet ekkor üzleti folyamatok halmazának tekintik, amelyek a különböző részlegeken keresztül haladnak, és kimeneteiket a vállalat külső vagy belső ügyfeleihez juttatják el. A szervezet ezután megpróbálja a munkát összefüggő folyamatként kezelni, amelyet egyes, egymással összefüggő alfolyamatokra bontanak (folyamatfa). A folyamat-irányítású vállalat egyrészt a szervezeti struktúrából áll, amely a cég folyamat-struktúrájából (horizontális), a cégtámogató struktúrájából – részleg szerkezetből (vertikális, amely kifejezetten a folyamatstruktúrát támogatja), és a cég irányítási rendszeréből áll.

A **folyamatmenedzsment** (BPM – Business Process Management) célja a szervezetben zajló folyamatok működésének fejlesztése és optimalizálása, maximálisan figyelembe véve a vállalat szükségeit. Ennek során tiszteletben kell tartani, hogy minden fontos változást felülről kell végrehajtani. A folyamatmenedzsment ezért a szervezet stratégiai szintjén kezdődik, a stratégiai célok és az e célok elérését szolgáló eljárások (stratégiák) meghatározásával. Ennek alapján határozzák meg a fő üzleti folyamatokat. Ezek az úgynevezett folyamattevékenységek folyamatmodelleken alapulnak, és a szervezeteken belül kerülnek végrehajtásra. Az alapvető és támogató folyamatokat információs rendszereken (ERP, CRM és SCM) keresztül kezelik és integrálják.

Az üzleti folyamatokat tovább mérik és ellenőrzik. A modern információs rendszer hatékony eszköz lehet a folyamatelemzés segítésére, mivel sok információval rendelkezik. Például, a redundáns alfolyamatokat figyelhetik, hogy adnak-e ezek hozzáadott értéket a végtermékhez; a vállalat teljes termékével való elégedettséget figyelhetik; mi az a minőség, amelyért a vevő hajlandó fizetni vagy arra a kérdésre kereshetik a választ, hogy a meglévő IR/IKT biztonság elegendő-e. Ha a stratégiai szinten meghatározott kulcsfontosságú teljesítménymutatók (KPI – Key Performance Indicators) és a tényleges KPI-értékek között eltérések mutatkoznak, a folyamatok javítását kezdeményezik. Ha azonban a vállalat környezetében vagy belső feltételeiben jelentős változás következik be, a helyzetet stratégiai szinten kezelik. Ilyen például az ügyfélmegrendelés értékelésének ideje, a megrendelés ütemezési ideje, a megrendelés teljesítési ideje, stb. Ez a feladata az üzleti folyamatok menedzsmentjében használt egyik módszernek, a BAM-nak (**Business Activity Monitoring**). A BAM számos szerepkörre és alapelvre épül, amelyek a következők:

- a BAM céljainak meghatározása és a bevezetendő folyamatok kiválasztása;
- a folyamat azon pontjainak, azaz a kulcsfontosságú tevékenységek vagy csomópontok helyes azonosítása, ahol az idő- és teljesítményjellemzők mérése a leghatékonyabb;

- a mérendő jellemzők meghatározása, beleértve azok tartalmát és formális meghatározását, a mérési egységeket stb.;
- a mérés gyakoriságának meghatározása (óránkénti, napi, véletlenszerű stb.);
- az értékek méréséért felelős személyzet meghatározása;
- a kapott vagy mért értékeken végzett elemzések funkcionalitásának meghatározása;
- a mért értékek felhasználási módjának meghatározása (azaz a BPR keretében), vagy a megfigyelt folyamatok javítását célzó szervezeti és technikai intézkedések.

A BAM-alkalmazások speciális technológiákat használnak, amelyeket a middleware-hez sorolunk, és amelyek az események kezelésére és nyomon követésére szolgálnak. Ezek a technológiák a BAM-alkalmazásokat megfelelő adatokkal látják el a rendszer eseményeiről. Például, ilyen esemény lehet egy elektronikusan beérkezett megrendelés, de a gyártósor megszakadásáról szóló információ is.

Az üzleti folyamatmenedzsment számos olyan módszert és megközelítést foglal magában, amelyek célja a vállalat hatékonyságának és teljesítményének - jelen esetben elsősorban a folyamatok teljesítményének - javítása. A gyakorlatban ez egyre nagyobb figyelmet kap, mivel az ár, a minőség és az innováció mellett a versenyképesség egyik meghatározó tényezője, hogy egy vállalat milyen gyorsan képes reagálni a vevői igényekre. A folyamatok és a vállalati folyamatok teljesítménye az üzleti informatikai megoldások és azok alkalmazásainak sajátos területeként értelmezhető.

Az üzleti folyamatok menedzsmentje szempontjából Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P., (2017) szerint az úgynevezett holisztikus megközelítés alapján a vállalati információs rendszerek számunkra már jól ismert és a korábbi fejezetekben többször is tárgyalt négy alapvető kategóriáját különböztetjük meg:

- **ERP (Enterprise Resource Planning)** – ezeket a rendszereket integrált egészként használják, elsősorban a szervezet (vállalat) tulajdonában lévő belső üzleti folyamatok irányításának támogatására. Lehetővé teszik különösen az összes belső folyamat, például a termelés, a logisztika, a HR stb. irányítását.
- **SCM (Supply Chain Management)** – ezeket a rendszereket integrált egészként használják elsősorban a szervezet (vállalkozás) beszállítói és vevői által közösen birtokolt külső üzleti folyamatok menedzselésének támogatására. Lehetővé teszik a vállalkozás hatékony integrációját az ellátási láncba.

- **CRM (Customer Relationship Management)** – ezeket a rendszereket integrált egészként használják, elsősorban a szervezet (vállalat) ügyfelei által közösen birtokolt külső üzleti folyamatok irányításának támogatására. Különösen a szerződések, az értékesítés, a marketing és a szolgáltatási folyamatok hatékony irányítását teszik lehetővé.
- **MIS (Management Information System)** – ez a rendszer az előző három rendszer (ERP, SCM, CRM) fölött áll. Ezekből gyűjti a legfontosabb adatokat, amelyeket más külső forrásokból származó adatbevitellel egészít ki. Ezen adatok alapján aztán kiértékeli és információt szolgáltat az üzletviteli folyamatok számára.

A folyamatba való beavatkozás esetén manapság megkövetelt az alkalmazott megoldásról egyértelmű előnyök kimutatása. Ezek lehetnek:

- **Mérhető előnyök**
  - Elsősorban az üzleti folyamatok hatékony kezeléséből, az azonos műveletek megkettőzéséből, a hibaarányból, stb. adódó költségek teljes szerkezetének csökkenése.
- **Nem mérhető előnyök**
  - Az egyes vállalati folyamatok valós optimalizálása, a felesleges feladatok számának csökkentése, az adatinformációk és eljárások valós idejű elérhetősége és kompatibilitása anélkül, hogy azok más rendszerekbe vagy adatbázisokba való átalakítása szükséges lenne, az információs rendszer úgynevezett felhasználóbarát környezetével stb.

Fejezetünket zárhatjuk azzal a ténnyel, amit világszerte sok menedzser megértett, mégpedig, a folyamatorientáltság kulcsfontosságú előnyét, amit biztosít a vállalkozások számára. Ez elengedhetetlen a hosszú távú versenyképességük biztosításához. (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P., 2017) (Sándor, J. 2011) (Gála, L; Pour, J; Šedivá, Z. 2009) (Koloszár, L. 2013) (Sziray, J.; Gaul, G. 2006) (Sziray, J.; Gaul, G.; Égertné Molnár, É. 2007)

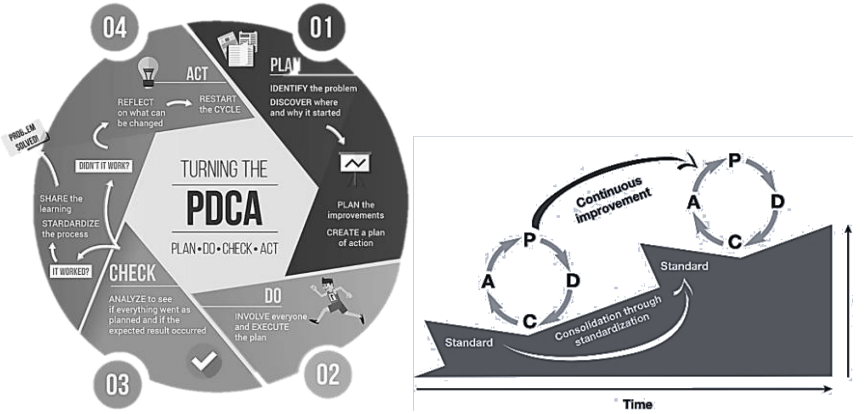
5.3 PDCA ciklus – folyamatjavító módszer

Vállalati információs rendszerektől elvárjuk, hogy teljes mértékben fedjék le és segítsenek a folyamatok irányításában, folyamatok optimalizálásában és segítséget nyújtanánk a menedzserek munkájában.

PDCA (Plan, Do, Check, Act) ciklust újra és újra le kell folytatni a folyamatos fejlődés elősegítése érdekében. A PDCA minden egyes szakasza hozzájárul ahhoz a célhoz, hogy azonosítsák, mely üzleti folyamatok működnek, és melyek azok, amelyek további fejlesztésre szorulnak. A PDCA ciklust először Dr. Walter Andrew Shewhart 1939-ben dolgozta ki, ezután Dr. William Edwards Deming az 1950-es években továbbfejlesztette. Vegyünk egy konkrét üzleti példát, – egy elektronikai alkatrészeket gyártó vállalat gyártási folyamatát, amelyen a PDCA lépéseit bemutatjuk.

- **Tervezés (Plan)**: ebben a lépésben meghatározzuk a célokat és a megoldandó problémákat. A tervezés magában foglalja a célok meghatározását, a stratégia meghatározását, az erőforrások azonosítását és a célok eléréséhez szükséges intézkedések megtervezését.
    - *A vállalat célul tűzte ki, hogy javítja termékei minőségi szintjét, és csökkenti a gyártósoron a hibás darabok számát. A terv magában foglalja a jelenlegi folyamatok elemzését, a lehetséges problémák azonosítását és a megoldásra irányuló intézkedésekre vonatkozó javaslatokat.*
- **Végrehajtás (Do)**: a második lépésben a tervezett intézkedéseket hajtják végre. Ez a lépés magában foglalja a változtatásokat és a javasolt megoldások végrehajtását.
    - *A terv szerint a vállalat változtatásokat hajt végre a gyártási folyamatokban, például korszerűsíti a gyártási technológiát, új eljárásokra oktatja az alkalmazottakat, és új ellenőrzési mechanizmusokat vezet be a hibák felderítésére*.
- **Ellenőrzés (Check)**: az intézkedések végrehajtása után a következő lépés az eredmények ellenőrzése és értékelése. Ez a lépés magában foglalja az adatok gyűjtését, az eredmények elemzését és összehasonlítását az eredeti célkitűzésekkel.
    - *A változások bevezetése után a vállalat nyomon követi az eredményeket. Adatokat gyűjt a hibás egységek számáról, elemzi a hibákhoz vezető okokat, és összehasonlítja az eredményeket a tervben meghatározott célokkal.*
- **Cselekvés (Act)**: az utolsó lépés az ellenőrzés megállapításai és eredményei alapján történő cselekvés. Ha az eredmények megfelelnek az elvárásoknak, a terv sikeresnek tekinthető. Ha nem, akkor további intézkedéseket lehet javasolni a célok elérése érdekében.

- *Az eredmények elemzése alapján a vállalatintézkedéseket tesz. Ha az eredmények pozitívak, és a hibás egységek száma csökken, a vállalat folytathatja az új eljárások alkalmazását. Ha az eredmények nem kielégítők, a vállalat további intézkedéseket hozhat (pl. eljárások finomítása, munkavállalók továbbképzésa, technológia kiigazítása).*



5.2. ábra: PDCA ciklus elve

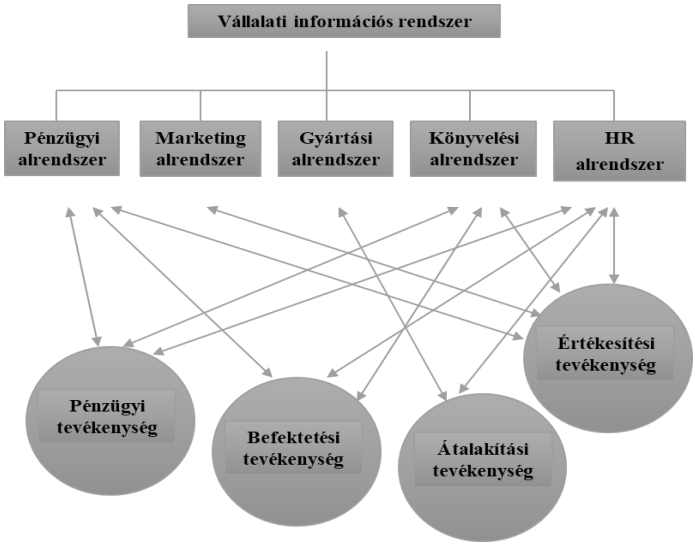
Forrás: (*Gomes da Silva, J.; Liora de Souza, I., J. 2019) (Das, T. 2024)*

E cikluson kívül még találkozhatunk több, hasonló ciklussal is, mint pl.:

- **DMAIC ciklus (Define-Measure-Analyze-Improve-Control)**, amely főként a folyamatok minőségének és hatékonyságának javítására koncentrál (Six Sigma projektekben gyakran alkalmazzák). Define (meghatározás) – a probléma vagy fejlesztési lehetőség azonosítása és a célok meghatározása. Measure (mérés) – az aktuális folyamat adatainak gyűjtése a jelenlegi teljesítmény felmérésére. Analyze (elemzés) – az adatok elemzése, a problémák és gyenge pontok azonosítása. Improve (fejlesztés) – megoldások kidolgozása a problémák megoldására és a folyamat fejlesztésére. Control (ellenőrzés) – a változtatások fenntartása és a folyamat új teljesítményszintjének biztosítása.
- **SCOR modell (Supply Chain Operations Reference)**, amely akkor hasznos, ha a vállalat nagy hangsúlyt fektet az ellátási láncra.
- **Business Process Reengineering (BPR)**, amely akkor hasznos, ha a PDCA vagy DMAIC ciklusok inkrementális javításai nem elégségesek, és alapvető változásokra van szükség. A BPR célja a folyamatok gyökeres újra tervezése annak érdekében, hogy nagyméretű javulásokat érjenek el a kulcsmutatókban.

- **Lean metodológia** – folyamatos fejlesztés, **Agilis fejlesztési ciklus**, stb. (Basu, R. 2015) (Realyvásquez, A.; García-Alcaraz, J.; Satapathy, S.; Coraza, D.; López, Y. 2024) (Pyzdek, T.; Keller, P. 2014)

5.4 Vállalati információs rendszerek és alrendszerek



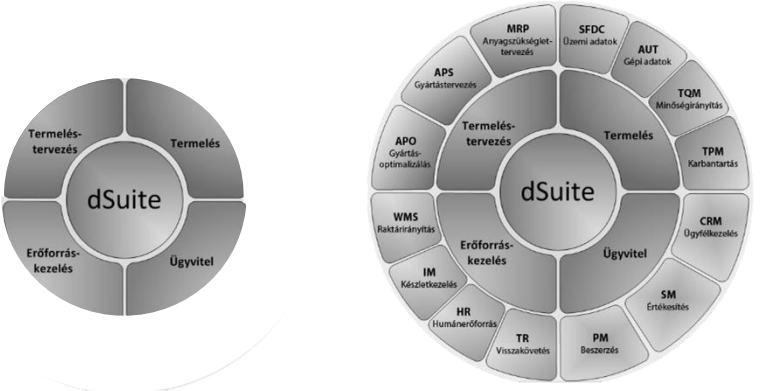
5.3. ábra: Tipikus példa a vállalati információs rendszer alrendszereire és kapcsolataira a kiválasztott tevékenységekhez.
Forrás: https://slideplayer.hu/slide/2121103/

Az 5.3. ábrán az alrendszerekre mutatunk példát. Az egyes alrendszerek legfontosabb feladatai közzé tartozhatnak a következők:

- **Marketing alrendszer**
  - Piackutatás, eladások elemzése, előrejelzése, hirdetési stratégia megtervezése, felügyelete, értékelése.
- **Gyártási alrendszer**
  - Raktárkezelés, termelés ütemezése, gyártási költségek becslése, minőségellenőrzés, CAD, CAM – Computer Aided Design, Manufacturing.
- **Könyvelési alrendszer**

- Vevő oldali számlák kezelése, szállító oldali számlák kezelése, naplófőkönyv, bérszámfejtés.

- **Pénzügyi alrendszer**
  - Pénzügyi ellenőrzés, pénzügyi előrejelzés, pénzügyi tervezés, költségvetés, készpénz és hitelgazdálkodás, portfólió menedzsment.
- **Emberi erőforrás (HR) alrendszer**
  - Munkaerő tervezés, személyes adatok nyilvántartása, továbbképzések, munkaerő ráfordítások elemzése.



5.4. ábra: dSuite mint rendszer, alrendszerei és moduljai
Forrás: https://www.ditec.sk/

Vegyünk egy konkrét példát. A **dSuite** egy olyan ERP (Enterprise Resource Planning) rendszer, amelyet Szlovákiában fejlesztettek ki különböző üzleti folyamatok integrálására. Ebben az esetben jól látható, hogy ha a dSuite-ot rendszerként értelmezzük, akkor a rendszer részei (termelés, termelés tervezése, erőforrás-kezelés, ügyvitel) alrendszereket alkotnak (5.4. ábra). Rajtunk múlik, hogy a rendszer határait meghatározzuk. Ha az egyes alrendszereket közelebbről megvizsgáljuk és rendszerként tekintjük őket – a bennük rejlő komplexitást is figyelembe véve, akkor ezeken belül is meghatározhatunk más alrendszereket, amelyeket esetünkben maguk a modulok képviselnek. (Sziray, J.; Gaul, G.; Égertné Molnár, É. 2007) (Sziray, J.; Gaul, G. 2006)

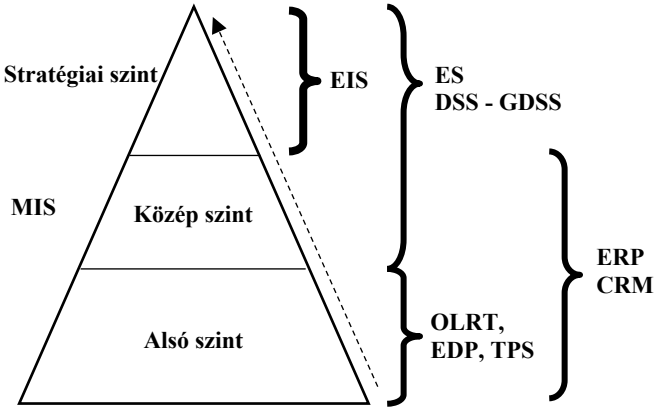
5.4.1 Az információs rendszerek fajtái

Az információs rendszereket (IR) sok szempontból felosztathatjuk. Számos cég vagy tananyag az aktuális igényekhez és területekhez igazítja

ezt a felosztást, ami jobb gyakorlati alkalmazhatóságot tesz lehetővé. Így a szakirodalomban is találkozhatunk különböző felosztásokkal. Mi aszerint fogjuk őket meghatározni, hogy mely vállalati tevékenységeket támogatnak, és azt milyen módon teszik. Szirai, J.; Gaul, G. (2006) szerint az infoormációs rendszerek két nagy csoportját különböztetjünk meg: a **vállalat mindennapi műveleteit** és a **vezetők munkáját támogató rendszereket**. Ebben a felosztásban a rendszerek és alrendszerek kereteire összpontosítunk. Fontos megjegyezni, hogy a kategóriák nem zárják ki egymást, így egy adott rendszer több kategóriába is sorolható. Például az ügyviteli rendszerek gyakran a vállalati belső együttműködést is támogatják.

A vállalat műveleteinek támogatása kulcsfontosságú a hatékony működéshez. Az alábbiakban néhány tipikus rendszert mutatunk be.

- **Adatbeviteli rendszerek** – Ide tartoznak például a vonalkód leolvasók és az RFID technológia, melyek segítik az adatok pontos és gyors rögzítését.
- **Tranzakciókat feldolgozó rendszerek (TPS)** – A tranzakciókat feldolgozó rendszerek biztosítják a vállalat tranzakcióinak kezelését. Három fő feldolgozás típust különböztetünk meg:
    - batch feldolgozás (szakaszos),
    - online feldolgozás (valós idejű azonnali feldolgozás),
    - real-time feldolgozás (amikor az adatokra pontos időben van szükség).
- **Műszaki tervezést támogató rendszerek (CAD)** – Ilyenek például a tervezői rendszerek, melyek a műszaki tervezésben, például az építészetben vagy gépészetben használt számítógépes tervezést (CAD) segítik elő.
- **Folyamatvezérlés** – A folytonos technológiák támogatása, például számítógépes gyártásvezérlés (CAM).
- **Ügyviteli rendszerek** – Ide tartoznak például a könyvelési, készletnyilvántartási és más adminisztrációs feladatokat segítő szoftverek.
- **Dokumentumkezelő rendszerek** – Ezek a rendszerek segítenek a dokumentumok digitális tárolásában, rendszerezésében és kezelésében.
- **Vállalati együttműködési rendszerek** (Groupware) – A csoportos munkavégzést segítő szoftverek (mint például a Microsoft Teams, Google Workspace vagy Slack), hatékonyan támogatják a vállalati belső kommunikációt és együttműködést.



5.5. ábra: Néhány IR elhelyezése az információs és hierarchiai piramisban

A vezetők döntéshozatali és stratégiai tevékenységének támogatása szintén nagy jelentőséggel bír. Az alábbi néhány rendszer szolgálja ezek elérését.

- **Vezetői információs rendszer (MIS – Management Information System)** – Előre specifikált jelentéseket küld a menedzsmentnek, amely az általános döntések előkészítésében és az operatív szinten történő irányításban nyújt segítséget.
- **Döntéstámogató rendszer (DSS – Decision Support System)** – Interaktív döntéstámogatás, mely a nem rutinszerű döntések meghozatalában segít. A DSS része lehet például egy szakértői rendszer, amely mesterséges intelligencia segítségével nyújt támogatást. Alapelemei az adatbázis, modellbázis és a dialóguskezelő rendszer.
- **Felsővezetői információs rendszer (EIS – Exucutive Information System)** – Ez a rendszer a felsővezetők számára biztosít információkat, amelyek a stratégiai döntések meghozatalához szükségesek. EIS rendszerek különböznek a MIS rendszerektől azért is, mert a top management jobban érdeklődik a külső információk irányt. A felső vezetés sajátos igényeinek kielégítésére szolgál. A rendszer által feldolgozott adatok közé tartozhatnak a pénzügyi összefoglalók, kritikus teljesítmény mutatók és a piaci pozíciók elemzése.

A vezetői információs rendszer részeként említhetők az alábbiak.

- **Elektronikus adatfeldolgozó rendszer (EDP – Electronic Data Processing)** – Alapvető funkciói közé tartozik az adatgyűjtés, -tárolás és -feldolgozás.
- **Döntéstámogató rendszer (DSS – Decision Support System)**.

- **Szakértői rendszer (ES – Expert System)** – Mesterséges intelligenciát használva a vezetők számára tanácsokat és javaslatokat biztosít. Például üzleti stratégiák támogatása révén javítja a teljesítményt, versenyelőnyt nyújthat, és hozzájárul a stratégiai célok eléréséhez.
- **Az On Line Real Time (OLRT)** – az „olrt" üzemmódban működő rendszerek speciális esete, amelyeket a technológiai folyamatok közvetlen vezérlésére terveztek. Jellemző rájuk, hogy minimális számú, ember által végrehajtott bemenettel és kimenettel rendelkeznek. Az alapot rendszerint numerikusan vezérelt gépek alkotják, amelyek egy számítógéphez kapcsolódnak. Ez határozza meg, hogy melyik gépen mit kell gyártani egy adott időpontban. Az egész folyamatot számítógép irányítja, emberi beavatkozás nélkül. Ezek a rendszerek jelenleg az ipar 4.0 részeként nagy fellendülést élnek át az ipari termelésben.
- **Felső vezetői támogató rendszer (EIS – Exucutive Information System).**

Az említettenek túlmenően, az utóbbi években a fejlett vállalatoknál elkezdték támogatni az információs technológia felfogását, az úgynevezett stratégiai információs rendszerek formájában, de az EIS-től eltérően. A **stratégiai információs rendszerek** (SIS) az információs rendszerek olyan alkalmazásai, amelyek célja a vállalkozás versenyképességének növelése. A vállalaton belüli – minden más típusú informatikai alkalmazás, természetesen hozzájárul ehhez a célhoz, a költségek csökkentésével és a termelékenység növelésével, illetve a vezetők döntéshozatalának javításával. Ezek mind olyan informatikai alkalmazások, amelyek többnyire a vállalaton belülre irányulnak, és annak irányítási rendszeréhez kapcsolódnak. A SIS ezzel szemben. főként a piaci területen, azaz a vállalkozás környezetében működik. Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P. (2017) szerint ezek lehetnek – például az ipari területen a szerszámok fejlesztése, a banki területen a banki gépek, stb.

Végezzünk el még egy szervezeti szint szerinti felosztást. E szerint az információs rendszereket felosztathatjuk:

- **operatív** (a napi folyamatokat támogató),
- **taktikai** (a középvezetést támogató),
- **stratégiai** (a felső vezetést és a döntéshozatalt támogató) rendszerekre.

Egyértelmű, hogy az operatív vezetés a hierarchiai piramis bázisánál, a taktikai vezetés a hierarchiai piramis közepén, és a stratégiai vezetés a

hierarchiai piramis csúcsánál fog helyet foglalni. Az egyes szintekhez hozzárendelt rendszereket az 5.5. ábra mutatja be.

Most már láthatjuk, hogy a rendszer/alrendszer megközelítés a vállalati információs rendszerek esetében nagyon fontos. Különösen az egyes részek (komplexitás és hatékonyság) kezelése szempontjából.

- Az alrendszerekre való felosztás lehetővé teszi az összetett rendszerek kisebb, jobban kezelhető részekre bontását, miközben minden résznek meghatározott szerepe és funkciója van a teljes vállalati információs rendszeren belül.
- Az alrendszerek gyakran modulárisak, lehetővé téve a vállalkozások számára, hogy könnyebben hozzáadjanak, cseréljenek vagy módosítsanak egyes modulokat, az aktuális igényeknek megfelelően. Egy új CRM (Customer Relationship Management) modul hozzáadható anélkül, hogy módosítani kellene a pénzügyi vagy termelési modulokat.
- Mindegyik alrendszer egy adott funkcióra vagy területre összpontosíthat, például humánerőforrás-menedzsmentre, pénzügyi menedzsmentre vagy logisztikára. Ez mélyebb specializációt tesz lehetővé, ami javíthatja az egyes modulok minőségét és hatékonyságát.
- Az alrendszerekre bontás lehetővé teszi a hozzáférési jogok és a rendszer egyes részei feletti vezérlés pontos meghatározását. Például, csak a kiválasztott alkalmazottak férhetnek hozzá a könyvelési vagy személyzeti modulhoz.
- A kisebb és specifikus alrendszerek optimalizálhatók a feladataikra, ami javíthatja a teljesítményüket. Például a tranzakciós modulok a sebességre, míg az analitikai modulok a nagy adatok feldolgozására optimalizálhatók.
- Az alrendszerek megkönnyítik a rendszer idővel történő bővítését, például az üzlet növekedésével. Ha az üzletág bővül, előfordulhat, hogy a teljes infrastruktúra jelentősebb beavatkozásai helyett csak egyes alrendszerek bővítésére vagy módosítására lesz szükség.
- Alrendszerek további lehetőségei.

További jelentős szempontként említhető: az interoperabilitás és integráció javítása, a döntéshozatali és elemzési eszközök támogatása, a folyamatok optimalizálása és a folyamatos fejlesztés, jobb alkalmazkodás a környezeti változásokhoz, jobb költséggazdálkodás, az agilis támogatása, a projektmenedzsment, és nem utolsósorban a felhő- és hibrid megoldások alkalmazása.

Az információs rendszerek rendszerekre és alrendszerekre való felosztása teljes mértékben támogatja a folyamatmenedzsmentet.

- Mindegyik alrendszer úgy alakítható ki, hogy egy bizonyos üzleti folyamatot vagy annak egy részét támogassa.
- Az alrendszerek lehetővé teszik a felelősség egyértelmű hozzárendelését az egyes menedzserekhez vagy csoportokhoz, amelyek bizonyos folyamatokért felelősek.
- Az információs rendszerek modularitása lehetővé teszi a rendszerek egyszerű adaptálását a folyamat igényeihez. Például, ha egy új folyamatot vezetnek be, vagy egy meglévőt módosítanak, akkor csak egy adott modul vagy alrendszer javítható anélkül, hogy az egész rendszert megzavarná.
- A moduláris alrendszerek lehetővé teszik a folyamatok jobb áttekintését, speciális elemző és jelentéskészítő eszközökön keresztül.
- A moduláris rendszerben a folyamatvezérlés támogatja a méretezhetőséget.
- Az alrendszerekre bontás támogatja az úgynevezett **szolgáltatás-orientált architektúrát** (**SOA – Service-Oriented Architecture**), amely lehetővé teszi újra felhasználható szolgáltatások létrehozását az egyes folyamatokhoz.
- Egy ilyen környezet ideális a folyamatmenedzsment számára, mivel lehetővé teszi a folyamatok összekapcsolását és szabványosítását a különböző részlegek és rendszerek között.

Az alrendszerek biztosítják a folyamatok irányításához és optimalizálásához szükséges struktúrát, rugalmasságot és eszközöket, míg a folyamatmenedzsment szisztematikus és céltudatos megközelítést biztosít az üzleti műveletek hatékonyságának és minőségének javításához.

Meg kell említenünk, hogy sok negatívummal is találkozunk. Például, az alrendszerekre bontás megköveteli az egyes modulok közötti hatékony integrációt, ami műszakilag megterhelő lehet. A független modulok eltérő technológiát vagy platformot igényelhetnek, és ez problémákat okozhat, amikor egyetlen, zökkenőmentes rendszerbe akarják csatlakoztatni őket. A komplexitás az alrendszerek számával és egymásra utaltságával nő. Az alrendszerek összekapcsolása gyakran szabványos interfészektől (pl. API-któl) függ, de ezek az interfészek gyártónkként vagy technológiánkként eltérőek lehetnek. A különböző szabványok és protokollok integrációs problémákat okozhatnak, és további erőforrásokat igényelnek a kompatibilitás biztosításához. Ha az adatokat különböző alrendszerekben tárolják és dolgozzák fel, nehéz lehet azokat konzisztensen

és szinkronban tartani. Ha egy vállalkozás úgy dönt, hogy egy adott modult új megoldásra cserél, kompatibilitási problémákba ütközhet. Egy alrendszer cseréje a rendszer több részének módosítását teheti szükségessé, ami megnövelheti a megvalósítási időt és növelheti a költségeket. Minden alrendszernek megvannak a maga karbantartási, frissítési és biztonsági követelményei, ami magasabb költségeket jelent egy monolitikus rendszerhez képest. Minden alrendszerhez saját biztonsági és hozzáférés-szabályozási beállítások szükségesek, amelyeket nehéz karbantartani és kezelni, különösen, ha nagy rendszerről van szó, sok felhasználóval és érzékeny adatokkal. Az egyik alrendszerben előforduló késések problémákat okozhatnak más folyamatokban vagy modulokban, ami kihat az általános hatékonyságra. Mivel az adatok szétszórhatók különböző alrendszerek között, az adatok összegyűjtése és konszolidálása az átfogó elemzéshez vagy jelentéskészítéshez kihívást jelenthet. Ez a probléma többletbefektetést igénylő központosított adattárházak vagy analitikai eszközök szükségességét teheti szükségessé.

Látható, hogy a minőségi vállalati információs rendszer bevezetése magas szakmai képzettséget igényel. Ezzel nem demotiválni akartuk az olvasót, inkább szerettünk volna rámutatni arra, hogy mindent meg lehet oldani, ha látjuk a célunkat. A megoldást a legjobb gyakorlatok alkalmazása jelentheti, mint például az egységes adattárház megvalósítása, szabványos API-k használata, valamint a komplexitás és a kockázatok kezelésére szolgáló központosított biztonsági szabályzat bevezetése. (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P.; 2017) (Sziray, J.; Gaul, G. 2006) (Sziray, J.; Gaul, G.; Erdős, F. 2006) (Sziray, J.; Gaul, G.; Égertné Molnár, É. 2007) (Koloszár, L. 2013)

6 Tervezés, vállalati információs rendszer építése és fejlesztése, alkalmazása

A vállalati információs rendszerek hatalmas mennyiségű adatot termelnek különböző adatbázisokban, dokumentumokban, fájlokban, weboldalakon, stb. Ezek strukturált és strukturálatlan adatok. A vállalat gyakorlatában a különböző alkalmazások egész spektruma létezik és bővül, amelyek nemcsak funkcionalitásukban, hanem számos más, többek között technológiai szempontból is különböznek egymástól. Az alkalmazások száma vállalatonként változik – kutatások szerint (Gála, L; Pour, J; Šedivá, Z. 2015) a nagyvállalatoknál több százra tehető. A SAP szerint a vállalatoknál egymilliárdos forgalomra körülbelül ötven alkalmazás jut. **Egy vállalat összes alkalmazásának összessége az alkalmazási portfóliója**. Az alkalmazásportfóliót adatbázisként vagy strukturált dokumentumként is ábrázolják, amelyet az alkalmazások életciklusuk során történő kezelésére használnak. Az alkalmazásportfólió tartalmazza az összes alkalmazás legfontosabb jellemzőit. A vállalati információs rendszerekre jellemző a nagyfokú heterogenitás és az ezzel együtt járó növekvő követelmények a kezelésükkel szemben. Ezért célszerű megfogalmazni azokat a szükséges klasszifikációs szempontokat, amelyek legalább alapvetően hozzájárulnak a jelenlegi alkalmazáspiaci kínálatban való eligazodáshoz. Ezek a szempontok a következők:

- **kinek** – milyen felhasználóknak szánják az alkalmazásokat, pl. e-kereskedelemben kereskedőknek, elemző alkalmazásokat vezetőknek, stb.;
- **adatok** – milyen adatokkal dolgoznak és milyen adatokat használnak, pl. strukturált adatok relációs adatbázisokban, strukturálatlan adatok dokumentumokban;
- **funkcionalitás** – milyen tranzakciós, analitikus vagy speciális funkciókat biztosítanak;
- **üzleti folyamatok** (vagy irányítási területek) – milyen folyamatokat támogatnak az alkalmazások, pl. termelési, üzleti, HR, stb.;

- **alkalmazott technológiák** - milyen információs és kommunikációs technológiákon fejlesztik és működtetik az alkalmazásokat (adatbázis, kommunikáció, stb.).

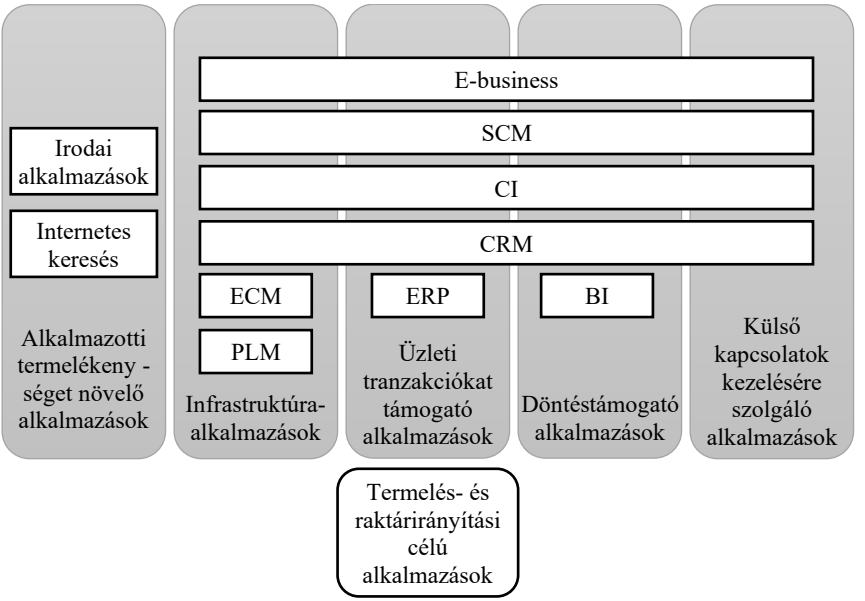
A vállalati informatika funkcionális magját – különösen a gyártó, kereskedelmi vagy szolgáltató jellegű vállalatoknál – az üzleti tranzakciók támogatására irányuló, az egész vállalatra kiterjedő alkalmazások, illetve a vállalati erőforrás-tervezés (ERP) alkalmazásai alkotják. Alapvető funkcionalitásukkal ezek az alkalmazások lehetővé teszik a felhasználók számára:

- kiterjedt adatbázisok létrehozását és frissítését – áruk, beszállítók, vevők, alkalmazottak, eszközök, számlák, stb;
- operatív jellegű folyamatok végrehajtását, azaz üzleti ügyek feldolgozását – anyagbeszerzés, áru eladás, stb. és a kapcsolódó üzleti dokumentumok (megrendelések, szerződések, számlák, vámáru-nyilatkozatok);
- a szükséges jelentések, statisztikák és alapvető elemzések elkészítését és bemutatását (jelentések a vevőkről, árukról, eladásokról, raktárkészletről, stb.).

Gyártási környezetben az ERP-alkalmazások nagyon gyakran kapcsolódnak termelésirányítási alkalmazásokhoz (gépek és berendezések), **raktárautomatizálási alkalmazásokhoz** és **termékéletciklus-kezelési** (PLM – Product Lifecycle Management) **alkalmazásokhoz**. A kommunikáció és az internetes infrastruktúra fejlődése lehetővé tette, hogy az eredetileg zárt vállalati információs rendszerek fokozatosan nyitottabbá váljanak ügyfeleik és beszállítóik felé, és kapcsolódni kezdtek más vállalatok és üzleti partnerek információs rendszereihez. Ezek az elektronikusan megvalósított kommunikációs és együttműködési kapcsolatok különböző jellemzőkkel bírnak, és az alkalmazások jellege is ennek megfelelően változik. Számos e-business alkalmazás képezi a **külső kapcsolatkezelési alkalmazások** alapját. Ezek közé tartoznak például az **e-kereskedelmi** megoldások. Az ebbe a kategóriába tartozó alkalmazások egy sajátos típusa az **ellátási lánc menedzsment** (SCM – Supply Chain Management) alkalmazások, amelyeket a sok szétszórt szervezeti egységgel rendelkező nagyvállalatok használnak belsőleg, vagy a különálló gazdasági egységek, azaz vállalkozások és szakosodott szolgáltató cégek közötti együttműködés kezelésére. Az ügyfelek, azok elégedettsége és hűsége a vállalat, és így az IT sikerének kulcsa. Ezért fejlesztettek ki a külső kapcsolatmenedzsment területén **ügyfélkapcsolatok kezelését támogató alkalmazásokat (**CRM

– Customer Relationship Management), amelyek magukban foglalják az üzleti kapcsolatok nyilvántartását és elemzését, az ügyfelekkel való kommunikáció kezelését, stb. A külső kapcsolatok területén egyre fontosabbá válnak a vállalatok, az állampolgárok, valamint a köz- és államigazgatási szervek (minisztériumok, adóhatóságok, biztosítók, stb.) közötti, illetve az e szervek közötti kapcsolatok kezelésére szolgáló alkalmazások is. Az alkalmazások ezen körére az **e-kormányzat** kifejezést használjuk. A külkapcsolat-irányítási alkalmazások kategóriájába tartoznak a piaci, illetve a piacversenyre vonatkozó információk feldolgozására irányuló alkalmazások is. Ezek a **versenyképességet támogató intelligencia alkalmazások** (CI – *Competitive Intelligence*).

A fent említett alkalmazástípusokon kívül számos más alkalmazást is bevezetnek a vállalatnál, amelyek jelentősen befolyásolják az információs rendszer általános minőségét és teljesítményét, és következésképpen az üzleti irányítás minőségét és teljesítményét. Ezeket együttesen **döntéstámogató alkalmazásoknak** nevezzük, ahol a domináns megközelítés az **üzleti intelligencia** (BI – *Bussiness Intelligence*). Ezeket az alkalmazásokat pénzügyi, üzleti és egyéb elemzésekre, üzleti tervek készítésének támogatására vagy az üzleti vezetők és szakemberek egyéni döntéshozatali tevékenységének támogatására használják.



6.1. ábra: Alkalmazásportfólió diagram
Forrás: (Gála, L; Pour, J; Šedivá, Z. 2015)

A vállalati információs rendszeralkalmazások egy speciális csoportját azok az alkalmazások alkotják, amelyek az **alkalmazottak hatékonyságának és termelékenységének növelésére** összpontosítanak.
Mindezekre az alkalmazásokra jellemző, hogy **a vállalat önálló alkalmazásszoftver-termékként vásárolhatja meg és telepíti vagy különböző gyártók terméke, esetleg integrált, nagyobb egységként kerül megtervezésre** és kiszállításra. Tekintettel arra, hogy az ERP-alkalmazások mindig a különböző típusú alkalmazásokat tartalmazó integrált megoldások középpontjában állnak, ezeket az alkalmazás komplexumokat általában **ERP II**-nek nevezik. Így magán az ERP-n kívül különböző infrastrukturális alkalmazásokat, e-business alkalmazásokat, üzleti intelligencia alkalmazásokat és egyebeket is tartalmaznak. (Gála, L.; Pour, J.; Šedivá, Z. 2015)

6.1 Cloud Computing

A Cloud Computing az informatikai szolgáltatások nyújtásának olyan modellje, amelyben a számítási erőforrások (szerverek, tárolók, adatbázisok, hálózatok, szoftverek, elemző eszközök és egyebek) elérhetők az interneten keresztül. Ezeket az erőforrásokat igény szerint biztosítják, ami azt jelenti, hogy a felhasználók anélkül használhatják őket, hogy birtokolniuk és kezelniük kellene a fizikai infrastruktúrát. Ebből kiindulva, a vállalkozásoknak nem kell drága hardverbe vagy infrastruktúrába fektetniük, csak azért kell fizetniük, amit használnak. Engedélyezik a távmunka elvégzését vagy az adatok elérését bárhonnan. Emellett a felhőszolgáltatók gondoskodnak az infrastruktúra karbantartásáról, frissítéseiről és biztonságáról, ami szintén nem terheli a vállalatot.

A Cloud Computing a nyújtott szolgáltatások típusa szerint különböző modellekre osztható:

- **Software as a Service (SaaS)**: kész alkalmazások, amelyek az interneten keresztül érhetők el. A felhasználóknak nem kell a szoftver telepítésével vagy kezelésével foglalkozniuk. Példák: Google Workspace vagy Microsoft 365.
- **Platform as a Service (PaaS)**: olyan komplett platformot biztosít a felhasználóknak, amely lehetővé teszi alkalmazások fejlesztését, bevezetését és használatát anélkül, hogy infrastruktúrával kellene rendelkeznie. Példák: Google App Engine, Oracle PaaS vagy Microsoft Azure App Services.
- **Infrastructure as a Service (IaaS)**: alapvető infrastruktúra, mint például számítási szerverek, tárhely és hálózat. Ebben a modellben

az ügyfél közvetlenül kap számítási erőforrásokat, amelyeken bármilyen szoftvert telepíthet, beleértve az operációs rendszert és az alkalmazásokat is. Az IaaS szolgáltatók közé tartozik pl. az IBM Smart Cloud, Oracle IaaS vagy Amazon Elastic Compute Cloud.

- **Communication as a Service (CaaS)**: rugalmas és költséghatékony kommunikációs megoldásokat kínál vállalatok számára. Olyan szolgáltatásokat foglal magában, mint a VoIP, videokonferencia, üzenetküldő rendszerek, call centerek vagy chatbotok. Az SaaS egy általánosabb felhőalapú modell, amely szoftvereket kínál előfizetéses alapon, míg a CaaS kifejezetten kommunikációs szolgáltatásokat biztosít.
- **Desktop as a Service (DaaS)**: felhőalapú virtuális asztali környezet, amely lehetővé teszi a felhasználók számára, hogy bármilyen eszközről, internetkapcsolaton keresztül elérjék a saját asztali rendszerüket.
- **További szolgáltatások**:
    - **BPaaS** (Business Process as a Service) – komplett folyamat lefedése,
    - **BaaS** (Backend as a Service) – felhőalapú háttérrendszer-szolgáltatás,
    - **HaaS** (Hardware as a Service),
    - **MaaS** (Monitoring as a Service),
    - **NaaS** (Network as a Service),
    - **UCaaS** (Unified Communications as a Service), stb.

A felhő telepítése szempontjából, az alábbi modelleket ismerjük:

- A **nyilvános felhő** olyan modell, amelyben egy szervezet megfelelő környezetet épített ki, majd szolgáltatásokat nyújt más szervezeteknek vagy személyeknek.
- A **privát felhő** olyan helyzetet jelent, amikor egy szervezet úgy építette fel információs rendszerét, hogy az megfeleljen a Cloud Computing jellemzőinek. Ezt követően, igény szerint a fent említett modellek valamelyikében (SaaS, PaaS, IaaS) szolgáltatást nyújt saját magának, illetve szervezeti egységeinek.
- A **közösségi felhő** egy olyan modell, ahol egy vállalatcsoport vagy egy közösség, amelynek közös érdekei, céljai, szabályai vannak, megfelelő informatikai környezetet építettek ki, amely megfelel a Cloud Computing jellemzőinek, majd ezek a szervezetek igény szerint használják ezt a környezetet.
- A **hibrid felhő** olyan modell, amelyben több felhő kapcsolódik egymáshoz oly módon, hogy a felhők közötti hordozhatóság

(például alkalmazások) biztosított, és a terheléselosztás révén az erőforrások hatékony felhasználása érhető el. Kombinálhatja a nyilvános és privát felhőket.

A felhőalapú számítástechnika ma a modern vállalkozások alapvető pillére, mivel rugalmas, hatékony és méretezhető megoldásokat tesz lehetővé az adatfeldolgozás, az alkalmazások és a számítási erőforrások területén. Szükséges megemlítenünk, hogy minden felhőszolgáltatáshoz nélkülözhetetlen a stabil internetkapcsolat. Bár a felhőszolgáltatók sokat fektetnek a biztonságba, az adatokat mégis a vállalatokon kívül tárolják, és ez számos kérdést vethet fel az érzékeny információk védelmével kapcsolatban. (Pavlíček, A.; Syrovátková, J. 2022)

6.2 Alkalmazások lehetséges megoldásai

„Az információs rendszerek komplexitása növekszik, ez ismert tény. Azok az idők, amikor az egész rendszert egy ember programozta, visszavonhatatlanul elmúltak. A rendszert először meg kell tervezni, már nem lehet csak úgy leülni és kódot írni. Egy programozói csapatban kell dolgozni a megvalósításhoz, mert egy ember biztosan nem elég. Nagyon jó kommunikációnak kell lennie a csapaton belül."

(Hartinger, D., 2024)

Jelenleg többféle megközelítés létezik az IT-alkalmazások kezelésére, amelyek végső soron különböző módon kombinálhatók egy vállalati alkalmazásportfólión belül. Az egyes alkalmazások fejlesztése is különböző szempontok szerint történhet. Az alábbiakban néhány szempontot említünk meg, amelyek alapján az alkalmazások fejlesztését jellemezhetjük.

Az első megközelítés az alkalmazás megoldásait a megoldók szempontjából vizsgálja. E megközelítés szerint az alkalmazásokat felosztjuk:

- A **vállalat alkalmazottai által fejlesztett alkalmazások** (úgynevezett „házon belüli", néha „*in – house"* megoldások). Ezek általában olyan alkalmazások, amelyek egyedi, speciális funkciókat biztosítanak, amelyek versenyelőnyt jelentenek a vállalat számára, vagy biztonsági okokból szükségesek (mivel ezek esetében nagyobb kontrollra is lehet szükség). Ilyen például a banki rendszerek. Sok esetben azonban sokkal egyszerűbb alkalmazásokról is szó van, pl. műszerfalak készítése egy üzleti

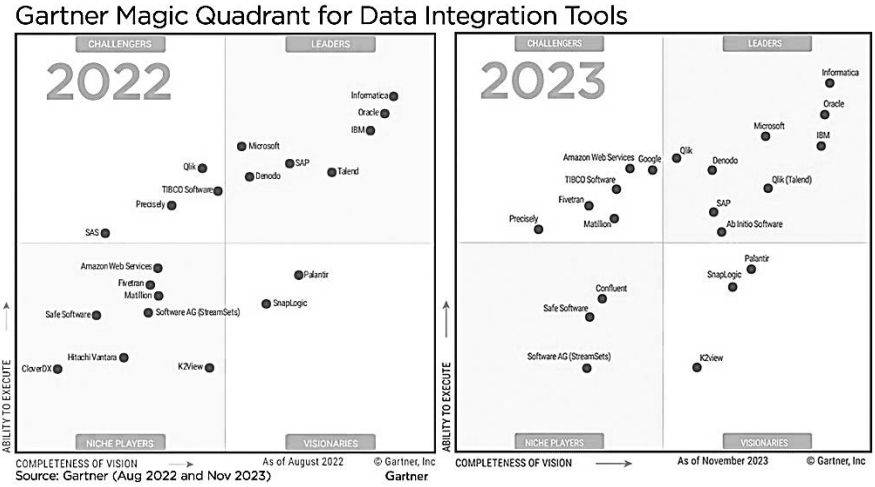
intelligencia környezetben, ahol a vállalat alkalmazottai a rendelkezésre álló információs technológiák és eszközök segítségével saját maguk képesek a szükséges alkalmazást felépíteni. A vállalat mérete is befolyásolhatja, hogy melyik lehetőséget választják. Kisebb vállalatoknál ez gyakran költséghatékonyabb megoldás, különösen, ha a vállalatnak nincs szüksége teljesen automatizált megoldásokra.

- **Külső cég** (vagy magánszemély) **által kifejlesztett alkalmazások** (**outsourcing**)[30]. Ez általában akkor fordul elő, ha a vállalat nem rendelkezik elegendő munkatárssal, és kiszervezi az alkalmazás fejlesztését. Az ilyen alkalmazások tipikus esetei a vállalati szintű tranzakciós ERP-alkalmazások, ügyfélkapcsolat-kezelő (CRM) alkalmazások, stb. Muszáj elmondanunk, hogy manapság a külső fejlesztés trendje figyelhető meg. Ugyanis, néha a vállalat szempontjából sokkal optimálisabb a vállalati információs rendszert megvásárolni vagy bérelni, mintsem több fizetést biztosítania a fejlesztők számára. Ezen kívül, saját fejlesztésű megoldások esetében, mivel kevesebb fejlesztő dolgozik rajtuk, előfordulhat, hogy a frissítések (például jogszabályi változások beépítése) késedelmet szenvednek. A mai modern vállalati információs rendszerek fejlesztői átfogó támogatási szolgáltatásokat kínálnak (frissítések, oktatások, testre szabás, stb.) és természetesen az előre megegyezett anyagi háttér mellett a fejlesztési kapacitásukkal is rendelkezésre készen állnak. Számos előnye van a külső fejlesztésnek, azonban bizonyos biztonsági és specifikus igények esetén a saját alkalmazásfejlesztés lehet hatékonyabb megoldás.

A második megközelítés az alkalmazás megoldását az alapján vizsgálja, hogy az alkalmazás lényegében változatlan formában használható-e egy másik vállalkozás által. Itt gyakorlatilag a 2.2. fejezetből indulunk ki, ahol ezt a bontást egyszerűsítettük és e fejezet céljaira adaptáltuk. E megközelítés szerint az alkalmazások a következőképpen oszthatók fel:

[30] Az outsourcing magában foglalhatja a szoftver teljes fejlesztését, karbantartását és frissítését, valamint olyan kiegészítő szolgáltatásokat is, mint az oktatás vagy a testre szabás. Fontos azonban megjegyezni, hogy a külső beszerzés nemcsak outsourcing formájában jelenhet meg, hanem olyan modellekben is, mint például a SaaS (*Software as a Service*), ahol a vállalat nem fejlesztést vásárol, hanem egy szolgáltatást bérel.

- **Típusalkalmazások** (*Type Application Software* – TASW, *Commercial Off-The-Shelf* – COTS[31]) olyan alkalmazások, amelyeket várhatóan különböző vállalatoknál használnak majd, akár egy gazdasági ágazaton belül, akár ágazatok között. Ma már elvileg a vállalatirányítás minden területére léteznek szabványos alkalmazások. Az alkalmazások a vállalkozás hatékonyságának és termelékenységének növelésére irányulnak.
- **Egyedi alkalmazások**, amelyeket a vállalat sajátos igényei szerint hoznak létre (úgynevezett egyedi alkalmazási szoftverek, *Individual Application Software* – IASW). Ezek általában olyan alkalmazások, amelyekre a vállalatnak vagy versenyelőny elérése érdekében van szüksége, vagy azért, mert a vállalat sajátos körülményeihez nem lehet szabványos alkalmazás megoldást használni. Bizonyos esetekben az IASW az ASW alkalmazás szoftver (*Application Software*) általánosításának alapjává válhat, és így a TASW létrehozásának alapjává.



6.2. ábra: TASW piaci helyzete 2022 és 2023-ban
Forrás: https://www.climberbi.co.uk/qlik-and-talend-leaders-in-the-2023-gartner-magic-quadrant-for-data-integration-tools/

[31] A COTS olyan szoftverekre utal, amelyek a kereskedelmi piacon, általában késztermékként állnak rendelkezésre. A COTS szoftverek általánosan alkalmazhatóak, ezért az alkalmazások típusalkalmazásokhoz sorolhatók. A COTS kifejezést elsősorban az angol környezetben használják, és azt hangsúlyozza, hogy kereskedelmi forgalomban lévő és elérhető termékről van szó, amely nem egyetlen ügyfélre szabott.

A TASW területének piaci helyzetével olyan szakosodott tanácsadó cégek foglalkoznak, mint a Gartner (Magic Quadrant), a Forrester Research (Forrester Wave) stb. A 6.2. ábra a „Magic Quadrant”-ra mutat példát.

Harmadik megközelítésünk az alkalmazásmegoldást az alapján vizsgálja, hogy az alkalmazások összekapcsolódnak-e egymással, és ha igen, milyen szorosan. E megközelítés szerint két szélsőséges határeset figyelhető meg.

- Alkalmazások, amelyek **szorosan összekapcsolódnak** egy „csomag” alkalmazássá. Egy ilyen típusú alkalmazást más alkalmazásokkal együtt kell használni, amelyek együttesen egy alkalmazáscsomagot alkotnak. Az összetétel ekkor nem teszi lehetővé, hogy az alkalmazás önmagában használható legyen, ugyanakkor a csomag funkcionalitása nem nélkülözheti az alkalmazás funkcionalitását. Erre példa a vállalati szintű ERP-alkalmazások, ahol a beszerzési modulok elvileg nem használhatók a főkönyvi modul nélkül.
- Olyan alkalmazások, amelyek **gyenge, szabványokon alapuló kapcsolattal rendelkeznek** más alkalmazásokkal, és ez a kapcsolat az üzenetcsere elvén valósul meg. Egy ilyen típusú alkalmazás önállóan is használható, és ha egy alkalmazáscsomagban egy kompozíció tárgya, akkor szükség esetén a csomagban lecserélhető, mivel az alkalmazás a kompozícióban csak üzenetváltás révén kötődik. Erre példa lehet egy e-kereskedelmi megoldás és annak kapcsolata egy vállalati szintű tranzakciós ERP-alkalmazással.

A negyedik megközelítés a szerint tekinti az alkalmazás megoldását, hogy milyen mértékben lehet beleavatkozni. E megközelítés szerint az alkalmazások az alábbiak lehetnek:

- **Fekete dobozos alkalmazások**. Ezek olyan alkalmazások, amelyek programozási tervébe az alkalmazás készítőjén kívül senki más nem avatkozhat bele, és az alkalmazásnak van egy pontosan meghatározott interfésze a más alkalmazásokkal való kommunikációra, amelyet azonban csak a készítő módosíthat.
- **Szürke dobozos alkalmazások**. Ezek olyan alkalmazások, amelyek programozási tervébe az alkalmazás készítőjén kívül senki más nem avatkozhat bele, de az alkalmazás többé-kevésbé szabványosított interfészeket biztosít, amelyek segítségével más alkalmazásokkal való kommunikációt lehet biztosítani.

- **A fehér dobozos alkalmazások**. Ezek olyan alkalmazások, amelyek esetében mind a programtervezésbe, mind az alkalmazás által biztosított interfészekbe bele lehet avatkozni.

Jelenleg az alkalmazások túlnyomó többsége szürke dobozos vagy fehér dobozos. A fekete dobozos alkalmazásokat főként ipari alkalmazásoknál használják. (Bruckner, T.; Voříšek, J.; Buchalcevová, A. 2012)

Az ötödik megközelítés az alkalmazásmegoldást az alapján vizsgálja, hogy a felhasználó számára az alkalmazás hogyan kerül átadásra és működtetésre. E megközelítés szerint az alkalmazásokat az alábbiak szerint osztjuk fel:

- **On Premise** alkalmazások, ahol az alkalmazás a vállalat technikai erőforrásaira telepített szoftver formájában jelenik meg.
- **On Demand** alkalmazások, ahol az alkalmazás SaaS felhőalapú számítástechnikai szolgáltatás formájában valósul meg, akár nyilvános, magán (privát), közösségi vagy hibrid felhőből biztosítják.

Az utolsó **hatodik megközelítés** az, amely figyelembe veszi az alkalmazáskód nyitottságának szempontját. A változatok egész sorából itt csak a következő kettőt említjük meg.

- **Nyílt forráskódú szoftverek** (OSS – *Open Source Software*) megfelelnek az OSI (*Open Source Initiative*) által meghatározott irányelveknek és licenceknek, amelyek biztosítják a forráskód szabad hozzáférését és módosítását. (Lásd a 2.2 fejezetet.)
- A **védett szoftver** olyan szoftver, amelynek forráskódja zárt, tehát nem nyilvános, és a felhasználók nem módosíthatják vagy terjeszthetik szabadon. Az ilyen szoftverek használatát a szerző vagy a kiadó licencfeltételekkel szabályozza, és ezeket általában egy EULA (*End-User License Agreement*) formájában teszi közzé. Az EULA meghatározza a végfelhasználók jogait és kötelességeit, és korlátozza a szoftver módosítását, másolását vagy újraelosztását. (Bruckner, T.; Voříšek, J.; Buchalcevová, A. 2012)

6.3 Az alkalmazások fontos jellemzői

Több száz alkalmazásszoftver van a piacon. A következőkben feltérképezzük, amit hasznos megfigyelni. A könnyebb eligazodás

érdekében, ezeket a jellemzőket: az üzleti jellemzők, a funkcionalitást leíró jellemzők, a nyújtott szolgáltatásokkal kapcsolatos jellemzők és egyéb jellemzők szerint kategorizáltuk. A következő alfejezetekben sorra vesszük őket.

6.3.1 Üzleti jellemzők

A jellemzők e csoportja meghatározza a termék piaci helyzetét, az alkalmazás biztosításának módját, a szállító vállalatok jellemzőit és egyéb gazdasági és üzleti jellemzőket.

Az alkalmazásszoftver alapvető azonosítója **a kereskedelmi neve** és a **verziója**.

- A kereskedelmi nevek a piac fejlődésével változhatnak; például az eredeti Axapta nevű, a dán Damgaard cég által kifejlesztett alkalmazásszoftvert a Microsoft által történt felvásárlását követően Microsoft Dynamics AX néven emlegetik, ami egyben azt is tükrözi, hogy egy adott termékcsaládba tartozik.
- Egy termék verziója általában a piacon lévő fejlesztési változatok sorrendjét tükrözi, pl. 9.2-es verzió.

Az **alkalmazás szállítási módját** az alkalmazás megoldási lehetőségei jellemzik. Az alapvető változatokat az előző alfejezetben soroltuk fel. Az alkalmazásszoftver szállításában részt vevő szervezetek tekintetében különbséget kell tenni a:

- **szoftver fejlesztője**,
- közvetlen **szállítója** (pl. egy külföldi vállalat helyi fióktelepe),
- **termék ügyfélnél történő bevezetését biztosító végrehajtó** között.

Nyilvánvaló, hogy mindezen szerepek hordozója lehet egy és ugyanaz a vállalat vagy három különböző vállalat, illetve más lehetséges kombinációk. Fontos ezeknek a hírneve. Nézzünk meg néhány fontos jellemzőt, amik közösek az összes említett pozícióhoz.

- **A vállalat és termékeinek szakmai irányultsága:** azaz, hogy a vállalat milyen gazdasági ágazatokra specializálódik a megoldásaival és szoftvereivel – például ipari vállalatok, kereskedelmi cégek, telekommunikáció, bankok, közigazgatási szervezetek, stb. Gyakorlatban a gyártók és alkalmazásszoftvereik

jellemzően több ágazatra összpontosítanak, különböző kombinációkban.

- **A vállalat létszáma:** az alkalmazottak – projektmenedzserek, tanácsadók, elemzők, fejlesztők, technikai támogató személyzet és más szakemberek – számát és struktúráját figyelembe véve. A termékek vagy teljes megoldások kiválasztásakor különösen fontos az egyes alkalmazottak tapasztalata is (szakmai önéletrajzuk alapján), különösen azoké, akik a termék bevezetésében és az azt követő szolgáltatások biztosításában közvetlenül részt vesznek az ügyfélnél.
- **A vállalat gazdasági ereje:** különösen fontos paraméter a szoftverfejlesztést végző vállalatoknál. Mivel a nem megfelelő beruházási lehetőségek jelentős akadályt jelenthetnek a jövőben a további szoftverfejlesztésnél, ami befolyásolhatja a vállalat erejét és hatékony működését.

Az **ár jellemzők** az egyes alkalmazási szoftvertermékekhez és a nyújtott szolgáltatásokhoz kapcsolódnak.

- **Az ASW (alkalmazásszoftver) szállítás**ára vonatkozó árak általában a termék beszerzési árának a terméklicencek árában kifejezett összetétele, a termék új verzióinak (frissítések) ára, a karbantartás ára és az egyéb kísérő szolgáltatások ára. Az árak e tekintetben jelentősen eltérhetnek az egyes szállítók között, ahogyan a termék vételárának meghatározási módja is, amely általában a megvásárolt licencek számához kapcsolódik.
- **A felhőalapú** számítástechnika megjelenésével összefüggésben az alkalmazások **szolgáltatás**ként (SaaS) is nyújthatók. Ekkor az árképzés jellemzők eltérnek, és az árakat a szolgáltatás igénybevételéhez, illetve a megrendelt funkciók meghatározott időn belüli fogyasztásához számítják.

Az alkalmazásszállító által nyújtott szoftvertermékekre és szolgáltatásokra vonatkozó **garanciák** alapvető fontosságúak, különösen a garanciák terjedelme (azaz az általuk lefedett területek) és a garanciális feltételek[32] szempontjából.

- A garanciák általában a funkcionalitás esetleges hibáinak javítására, valamint a jogszabályi és felhasználói követelmények változásai miatt szükséges módosításokra terjednek ki.

[32] Azok a kikötések és szabályok, amelyek meghatározzák, milyen körülmények között érvényesíthető a garancia.

- A garanciális feltételek közé tartozik a hibaelhárítás gyorsasága garanciális és garancián túli esetekben, valamint a garanciális szolgáltatás teljes időtartama.
- SaaS alkalmazásmegoldások esetén jelentős garanciális feltétel, hogy a szolgáltató a szolgáltatás megszűnésekor átadja a vállalatnak a szolgáltatás használata során keletkezett összes adatot.

Az **alkalmazásszoftverek referencia telepítései** olyan konkrét ügyfeleket (vállalkozásokat) képviselnek, akik már működtetik az alkalmazás-szoftvert, és ahol annak funkcionalitása és egyéb működési jellemzői és szolgáltatásai reálisan ellenőrizhetők.

- Az alkalmazásszoftver funkcionalitásának, szolgáltatásainak és tényleges működésének referenciatelepítésen történő ellenőrzése általában az egyik legfontosabb tényező a rossz termék- és szállítóválasztás kockázatának csökkentésében.

Az üzleti jellemzők alapvető fontosságúak az alkalmazás megfelelő kiválasztásához. **Még a nagyon jó alkalmazásszoftverek is leértékelődhetnek a gyakorlatban a rossz megvalósítás, a rossz felhasználói képzés vagy az ügyfél gyenge együttműködése miatt**. (Gála, L.; Pour, J.; Šedivá, Z. 2015) (Sodomka, P.; Klčová, H. 2010) (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P. 2017)

6.3.2 Funkcionalitást leíró jellemzők

Nyilvánvalóan, nagyon fontos jellemzőről van szó a felhasználó szempontjából. A probléma általában a funkcionalitás jelentős mértkével, a funkciók eltérő strukturálásával és az eltérő terminológiával függ össze.
Az átfogó koncepcióját az **alkalmazásszoftver architektúrája** fejezi ki. Ez tehát az egyes szoftvermodulok (pl. értékesítési modulok, beszerzési modulok, operatív modulok és mások) elrendezése, amelyekből az alkalmazási szoftver áll, – mivel a vállalaton belüli ASW bevezetése általában modulonként valósítható meg. Az alkalmazási szoftver architektúrája meghatározza továbbá azt, hogy:

- az ASW a vállalatirányítás mely folyamatait vagy területeit fedi le funkcióival;
- az egyes modulok hogyan kapcsolódnak egymáshoz, milyen adatok és eszközök felhasználásával;

- milyen lehetőségek vannak a kiválasztott ASW-modulokból a végső megoldás felépítésére;
- hogyan oldják meg az alkalmazási szoftver nyitottságát (azaz, milyen lehetőségek vannak más alkalmazási modulok, események egymás utáni összekapcsolására), más beszállítóktól származó alkalmazásszoftverek;
- hogyan kezelik az ASW működésének irányítását és felügyeletét (az egyes modulok használatának gyakorisága, az adatbázisok állapota, a szoftver hibái, módosítási kérelmek), beleértve a távoli irányítási lehetőségeket. (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P. 2017) (Sodomka, P.; Klčová, H. 2010)

6.3.3 Szolgáltatásokkal kapcsolatos jellemzők

Természetesen, a szolgáltatás szintje, terjedelme és minősége erősen beszállítófüggő, és ugyanazon ASW esetében jelentősen eltérhet. Másrészt a nyújtott szolgáltatások szintje függ az ügyfél igényeitől és a személyzet kapacitásától, az informatikai csapat erejétől is.

- Az alkalmazásszoftver **karbantartása** elsősorban a jogszabályi változásoknak megfelelő módosításokat és az ügyfél igényeinek megfelelő, folyamatos kiigazításokat foglalja magában. Az új termékverziók esetében a periodicitás lényeges (általában hat hónap és egy év közötti). Az ASW új verzióinak telepítésével és a folyamatos módosítások végrehajtásával kapcsolatban alapvető fontosságú e tevékenységek szervezeti irányítása, különösen a felhasználók tájékoztatása a végrehajtott módosításokról és telepítésekről.
- A **forródrót-szolgáltatás** ma már gyakorlatilag minden IT-szállító szolgáltatásának általános része. Az egyetlen kérdés a rendelkezésre állás, vagyis az, hogy a nap mely szakában működik, elérhető-e szombaton, vasárnap és ünnepnapokon is.
- A **tervezési szolgáltatások** szintje szempontjából alapvető fontosságú, hogy:
    - ki nyújtja ezeket a szolgáltatásokat, azaz az ASW fejlesztője, a ASW forgalmazója, a tanácsadó cég, stb.;
    - hogyan vonják be a tervezésbe az ügyfél személyzetét - felhasználókat, elemzőket, programozókat, stb.;
    - milyen tervezési módszertanokat és szoftvereszközöket használ a szállító a tervezéshez, és hogyan bocsátja azokat a megrendelő rendelkezésére;
    - milyen idő- és költségintenzitású a tervezési munka.

A **képzési szolgáltatások** az alkalmazásszoftver-szállító szolgáltatásainak szokásos részét képezik, és elengedhetetlen számukra, hogy:

- milyen a képzési szolgáltatások struktúrája, (pl. módszertani képzés a megvalósítási csoportok számára, felhasználói képzés az egyes modulokhoz);
- mi az egyes képzési típusok várható időbeosztása;
- milyen a képzés formátuma, a képzési anyagok színvonala, a képzés nyelve, a képzési létesítmények technikai felszereltsége.

Meg kell jegyezni, hogy különösen az on-demand alkalmazások, azaz a SaaS megjelenésével a hagyományos képzési szolgáltatásokat felváltják az elektronikus oktatóanyagok és a képzés egyéb elektronikus formái (e-learning). A **tanácsadási szolgáltatások** általában elsősorban az ASW moduljaihoz és különböző üzemeltetési szempontokhoz kapcsolódnak (pl. hozzáférési jogok, archiválás, válaszidővel kapcsolatos kérdések, stb.). Az ASW és az információs rendszerek szállítói azzal a feladattal szembesülnek, hogy részt vesznek az ügyfél üzleti irányításának átfogó fejlesztésében, és nem korlátozzák hatáskörüket a projektek tisztán informatikai szempontjaira.
Az ASW-hez egyéb szolgáltatások is kapcsolódnak, pl. a működés **távfelügyelete**, programok **távjavítása** és módosítása, a rendszer és az adatbázis-konfiguráció **folyamatos optimalizálása**. (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P. 2017) (Sodomka, P.; Klčová, H. 2010) (Gála, L.; Pour, J.; Šedivá, Z. 2009)

6.3.4 Egyéb jellemzők

A **működési környezet jellemzői** az alkalmazások megoldásainak fontos tulajdonságai. Meghatározzák, hogy milyen információs technológiák szükségesek az alkalmazás futtatásához, és hogyan kell telepíteni és működtetni magukat az információs technológiákat. Ez jellemzően a következőket foglalja magában:

- az operációs rendszerek típusait és verzióit;
- adatbázis-rendszerek típusait és verzióit;
- a köztes szoftverek kategóriáit és funkcióit;
- egyéb információs technológiákat (lásd a 4. fejezetet).

A **nyelvi és nemzeti lokalizáció** határozza meg, hogy az alkalmazásszoftver mely nyelveken működtethető és mely állam törvényei szerint működnek az egyes modulok.

- A nyelvek száma különösen fontos a multinacionális vállalatok és a külföldi vállalatok képviselete szempontjából. Fontos jellemző az ASW normál működése során a nyelvek közötti váltás lehetősége. A nyelvi környezet szempontjából fel kell mérni a különböző nyelvek használatának lehetőségét a képernyőkön (menük, adatmezők leírása, hibaüzenetek), a súgóban, a szöveges adatbevitel és feldolgozás során, valamint a nyomtatott dokumentációban történő normál kommunikációban. Jelenleg a világ legelterjedtebb alkalmazásszoftverei mindig több nyelvet támogatnak (némelyikük több tucatot).
- A nemzeti lokalizáció biztosítja, hogy az alkalmazás megfeleljen az adott ország jogszabályainak, amelyben az alkalmazást működtetik vagy használják (pl. cseh lokalizáció esetében az alkalmazás pénzügyi modulja megfelel a Cseh Köztársaságban érvényes számviteli törvénynek).

Az **alkalmazási dokumentáció** magába foglalja a következő dokumentumokat:

- felhasználói dokumentumok – tartalmazzák az ASW általános felépítését és logikáját, az egyes modulok funkcionális leírását és a működésükre vonatkozó utasításokat;
- üzemeltetési dokumentumok – tartalmazzák az adatbázisok, az archiválási műveletek és a biztonsági mentések kezeléséhez szükséges utasításokat;
- tervezési és fejlesztési dokumentumok;
- az ASW üzembe helyezéséhez használt végrehajtási módszerek, technikák és eszközök dokumentációja.

Az ASW típusú megoldásokkal összefüggésben nagyon gyakran felmerül a kérdés, hogy ezek az alkalmazások az ügyfél igényei szerint testreszabhatók-e. A testreszabás egy speciális típusa, amely nem jár az alkalmazás kódjába való beavatkozással, a **customization** (**testre szabás**). A testreszabás során egy szoftvert vagy szolgáltatást az adott felhasználó vagy vállalat egyedi igényeihez igazítanak, hogy jobban megfeleljen a specifikus követelményeknek és munkafolyamatoknak. Ez a szoftvermegoldás paraméterértékeinek beállításának folyamata a kívánt viselkedésének biztosítása érdekében. A testreszabás főleg a következőkre összpontosít:

- az adatfeldolgozás módjának beállítása – pl. lehetőség van a bérek kiszámításánál alkalmazandó megfelelő módszer beállítására;

- a képernyők szerkezetének és az egyes adatmezők elrendezésének beállítására, ugyanúgy a kimeneti jelentések beállítására;
- a különböző bemeneti adatmezők alapértelmezett értékeinek beállítására, hogy mely adatokat kell és melyeket nem szabad megadni;
- a navigációs séma (preferált, kedvenc funkciók) beállítására és a felhasználói csoportokhoz való hozzárendelésre;
- a vizuális elemek megjelenítési módjának beállítására (színek, betűtípusok, stb.).

A testreszabás az alkalmazás vállalati környezetben történő bevezetése során, azaz a rutinüzem megkezdése előtt történhet, amikor egy készletet hoznak létre, amely az egész alkalmazás viselkedését állítja be. Az egyes felhasználók az alkalmazás használata során is végezhetnek testreszabást, pl. a felhasználói felület (vizuális elemek) beállítása esetében. Ezen a szinten minden egyes felhasználó számára egyéni testreszabási készlet jön létre.

Az **akadálymentesítés** esetében olyan környezet és állapot megteremtéséről van szó, amelyben az alkalmazás használata nem jelent semmilyen akadályt a felhasználó számára. Ebben az összefüggésben gyakran szóba kerülnek a fogyatékkal élő (látás-, hallás- és mozgássérült, tanulási és koncentrációs zavarokkal küzdő) vagy speciális eszközökkel (pl. kevésbé elterjedt webböngészők) történő hozzáférhetőség biztosításával kapcsolatos kérdések. (Számos ajánlás létezik ezen a területen, pl. a Web Accessibility Initiative (WAI, http://www.w3.org/WAI/), US SECTION 508 (http://www.section508.gov), stb.).

Az alkalmazások **használhatósága** azt méri, hogy a felhasználók mennyire jól tájékozódnak (navigálnak) bennük, mennyire könnyen értik az elrendezést és a kezelőszerveket, és milyen benyomást hagynak maguk után. Azt is megmutatja, hogy mennyire találják egyértelműnek, könnyen érthetőnek és intuitívnak a használatukat, vagy ellenkezőleg, zavarosnak és nehezen elérhetőnek találják a célt. Jakob Nielsen szerint (Gála, L; Pour, J; Šedivá, Z. 2015) a webes alkalmazások használhatósága öt alapvető célt követ. Ez alapján a felhasználó:

- az újonnan meglátogatott honlapot megérti és használni tudja;
- ismerős weboldalon el tudja érni a céljait;
- megjegyzi az oldal elrendezését, és a következő látogatáskor gyorsan felidézi azt;
- a lehető legkevesebb súlyos hibát követi el, vagy gyorsan kijavítja azokat;

- kellemes élményt nyújt a webhely használata. (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P. 2017) (Sodomka, P.; Klčová, H. 2010) (Gála, L.; Pour, J.; Šedivá, Z. 2009)

6.4 Információs rendszerfejlesztést támogató szoftverek

Egy vállalati információs rendszer fejlesztésének irányítása, alkalmazásainak fejlesztése és működésének irányítása nem lehetséges megfelelő szoftverek nélkül.

6.4.1 Computer Aided Systems Engineering

Ebben a kategóriában a legismertebbek azok az eszközök, amelyeket együttesen számítógéppel támogatott rendszerfejlesztésnek **(CASE** – Computer Aided Systems Engineering) nevezünk. Az „S" betű azonban jelenthet szoftvert is, és akkor számítógéppel támogatott szoftverfejlesztő eszközökről beszélünk (CASE – Computer Aided Software Engineering). Ezek az eszközök a vállalati információs rendszerre vagy szoftverrendszerre vonatkozó információk közös tárházaival dolgoznak, hogy biztosítsák az információk konzisztenciáját a rendszer életciklusának minden szakaszában. A CASE eszközök célja, hogy segítsék és automatizálják a szoftverfejlesztés különböző lépéseit, így növelve a hatékonyságot és a minőséget. A CASE rendszerek olyan szoftvereket, eszközöket és módszertanokat foglalnak magukba, amelyek támogatják a szoftver teljes életciklusát a tervezéstől a fejlesztésen át a karbantartásig.
A CASE eszközök általánosan két alapkategóriába sorolhatók, attól függően, hogy a fejlesztési folyamat melyik fázisát támogatják:

- **Upper CASE** (felső CASE): ezek az eszközök a fejlesztési folyamat elején hasznosak, és az elemzés[33], tervezés, követelmények meghatározása és modellezés fázisait támogatják. Céljuk a követelmények, folyamatok és modellek rögzítése a szoftver tervezésének kezdeti szakaszában. Például,

[33] Business CASE eszközök az üzleti elemzés és tervezés területére specializálódnak. Segítik a vállalati folyamatok modellezését, a követelmények feltárását, valamint a gazdasági döntéshozatalt. Az ilyen eszközöket például üzleti elemzők használják az üzleti igények és követelmények meghatározására.

diagramkészítők, mint UML (*Unified Modeling Language*) vagy adatmodellező eszközök, pl. ERD (*Entity relationship Diagram*).

- **Lower CASE** (alsó CASE): ezek az eszközök a fejlesztési folyamat későbbi szakaszaiban, a kódolásban, tesztelésben[34], hibakeresésben és karbantartásban[35] segítenek. Itt tehát a konkrét szoftverkód előállítása, hibajavítása és tesztelése történik. Példák lehetnek a kódszerkesztők, verziókezelő rendszerek, valamint a különböző tesztelő és hibajavító eszközök.

Tekintettel a vállalati információs rendszerek üzemeltetésére és karbantartására, ezt a két kategóriát szélesíthetjük a következővel:

- **Integrált CASE** eszközök (I-CASE): ezek a rendszerek egy teljesen integrált környezetet biztosítanak, amely átfogóan támogatja a fejlesztés minden szakaszát, a tervezéstől a kódoláson át a tesztelésig és a karbantartásig. Az integrált CASE eszközök gyakran egy platformon belül kínálják az Upper és Lower CASE eszközök funkcionalitását, így a fejlesztők az összes szükséges eszközt egy környezetben érhetik el, tehát a fejlesztési folyamat összes szakasza egy egységes keretrendszerben van kezelve.

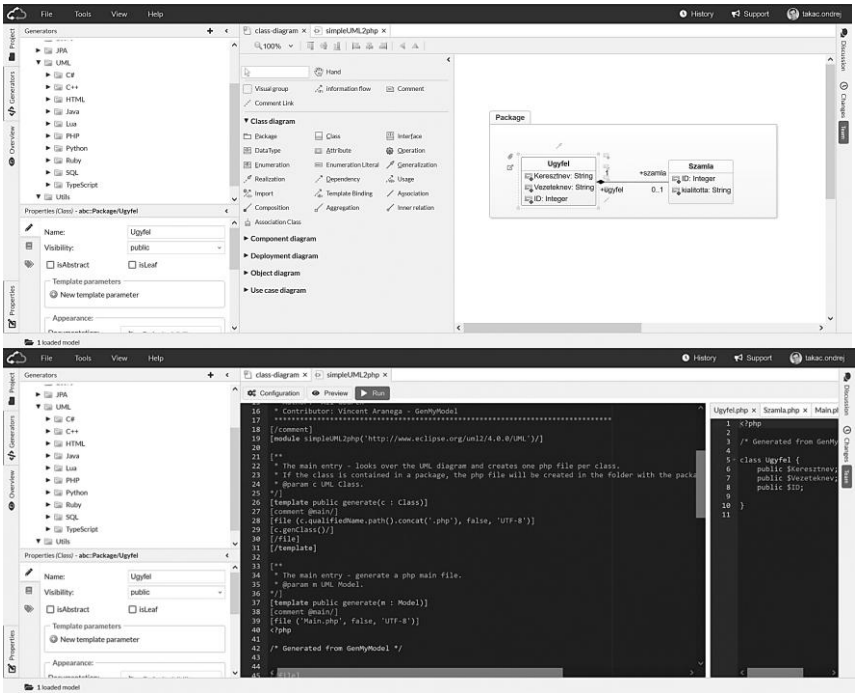
A CASE tehát egy nagyon széleskörű fogalom, amely rengeteg specifikus eszközt és technológiát foglal magába a szoftverfejlesztés különböző szakterületein. Megemlíthetünk, pl. a Special CASE eszközöket, melyek egy adott technológia vagy módszertan köré épülnek, mint például az adatbázis-tervező eszközök, adatmodellező eszközök, vagy akár AI által támogatott tervezési eszközök. Ezeket olyan speciális feladatokhoz használják, amelyeknél az általános CASE eszközök nem elég hatékonyak. A CASE eszközök tehát szűk kapcsolatban vannak a különböző feladatok ellátásával. Ezen eszközök közül csak néhányat említsünk:

- analitikai eszközök;
- modellező és folyamatmodellező eszközök;
- diagramkészítő eszközök, amelyek lehetővé teszik különböző diagramok létrehozását az információfeldolgozás vizualizálására;

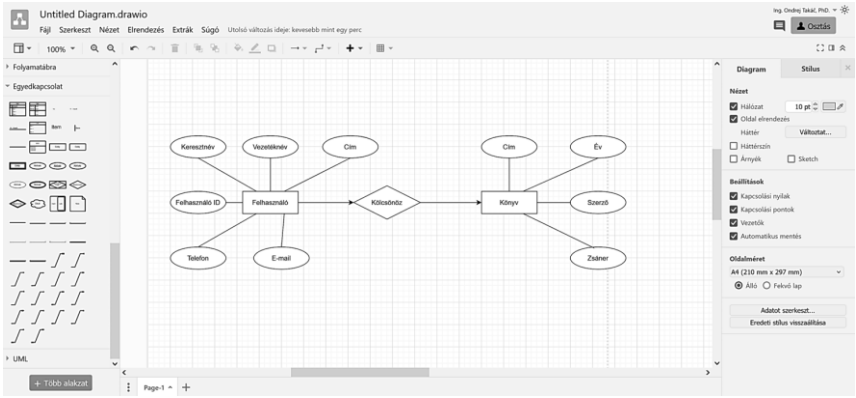
[34] Testing CASE eszközök kifejezetten a szoftvertesztelést és hibajavítást támogatják. Automatikus tesztelő eszközök, hibakövető rendszerek és regressziós tesztelési eszközök tartoznak ide. Ezek az eszközök különösen a minőségbiztosítási csapatok számára hasznosak, segítik a szoftver hibáinak azonosítását és kijavítását.

[35] Maintenance (karbantartó) CASE eszközök a szoftver életciklusának fenntartásához szükséges feladatok támogatására szolgálnak. Ilyenek például a hibajegyek kezelése, a verziókövetés és a konfigurációkezelés. Ezek az eszközök fontosak, mivel segítik a szoftver folyamatos frissítését és fejlesztését.

- projektmenedzsment eszközök;
- dokumentációs eszközök, amelyek valamennyi fázis tevékenységeinek dokumentációját létrehozzák;



6.3. ábra: Példa online UML környezetre (fent) és kód generálásra (lent)
Online környezet: https://app.genmymodel.com/



6.4. ábra: Példa online ERD (egyed-kapcsolt diagram) környezetre
Online környezet: https://app.diagrams.net/

- prototípuskészítő eszközök, amelyek a kész szoftverrendszert szimulálják oly módon, hogy az érdekelt felek (pl. felhasználók) elegendő információt kapjanak a szoftver funkcionalitásáról és felhasználói felületéről;
- programozási eszközök, amelyeket általában egy integrált fejlesztőkörnyezet (IDE) képvisel, amelyet számos kész megoldási könyvtár és a megoldás szimulálására szolgáló eszköz egészít ki (az eszközök lehetővé teszik az algoritmus felírását – azaz a programkód generálását – egy meghatározott programozási nyelvre pl. Java, Visual Basic, C#, PHP, JavaScript stb.);
- webfejlesztő eszközök;
- konfigurációkezelő eszközök, beleértve a változáskezelést is, amelyek a rendszer / szoftver verzióinak és revízióinak fejlesztését irányítják;
- minőségirányítási eszközök;
- a rendszer/szoftver karbantartását támogató eszközök, amelyek a szoftverhibák keresésére, naplózására és a hibajelentések automatikus generálására összpontosítanak;
- és további más eszközök.

Nagyon gyakran ezek az eszközök egy bizonyos módszertanhoz kapcsolódnak, amely megfogalmazza a vállalati információs rendszer fejlesztésének alapelveit, eljárásait, és kialakítja annak kereteit. (Sodomka, P.; Klčová, H. 2010) (Gála, L.; Pour, J.; Šedivá, Z. 2009)

6.4.2 A rendszerüzemeltetés irányítása

A második fő csoportot azok a szoftvermegoldások alkotják, amelyek a vállalati információs rendszer működésének irányításához kapcsolódó támogató tevékenységekre összpontosítanak. A szoftvermegoldások széles skálájából ebben a csoportban az alábbiakat említhetjük.

- Informatikai és technológiai felügyeleti eszközök, amelyek képesek valós idejű információkat szolgáltatni a vállalati információs rendszer egyes összetevőinek eléréséről, használásáról és rendelkezésre állásáról.
- A külső számítógépes tárolókon található adatok biztonsági mentésére szolgáló eszközök és a kiesés utáni helyreállítás eszközei.
- Incidens- és problémakezelő eszközök (nem csak a rendszerbiztonsággal kapcsolatosak), amelyek támogatják a help

desk tevékenységeket, és kezelik az incidens vagy probléma megoldásának eszkalálási folyamatát, azaz továbbítják azt a megfelelő személyzetnek.

- Olyan eszközök, amelyekkel a felhasználók (ügyfelek) képesek dinamikusan kialakítani az üzleti információs rendszer vagy annak összetevőinek jellemzőit, úgynevezett önkiszolgáló eszközök. Ezeknek nem kell kizárólag a megjelenés vagy a tartalom testreszabására irányulniuk, hanem lehetővé teszik a felhasználók (és a vállalati részlegek) számára az erőforrások szükség szerinti dinamikus kiosztását. (Sodomka, P.; Klčová, H. 2010) (Gála, L.; Pour, J.; Šedivá, Z. 2009)

6.5 DevOps (Development Operations)

A **DevOps** egy módszertan vagy megközelítés a szoftverfejlesztés területén, amely a „fejlesztés" (Dev – *Development*) és az „üzemeltetés" (Ops – *Operations*) kombinációját jelenti. Célja, hogy javítsa az együttműködést azon csapatok között, melyek felelősek a szoftver fejlesztéséért, üzemeltetéséért és karbantartásáért. Ez a módszertan nagy hangsúlyt fektet az automatizálásra, a folyamatos integrációra (CI – *Continuous Integration*) és a folyamatos telepítésre (CD – *Continuous Deployment*), hogy javuljon a szoftverek szállítási sebessége és minősége.
A DevOps kulcsfontosságú elvei a következők:

- együttműködés a fejlesztői és üzemeltetői csapatok között;
- a folyamatok, például a szoftvertesztelés és telepítés automatizálása;
- folyamatos szállítás (*Continuous Delivery*) – az új szoftververziók gyakori és stabil telepítése;
- monitorozás és visszacsatolás – az alkalmazások és az infrastruktúra teljesítményének nyomon követése a gyors reagálás érdekében.

A Continuous Integration (CI), azaz a folyamatos integráció egy olyan gyakorlat a szoftverfejlesztésben, amely során a fejlesztők rendszeresen egyesítik (integrálják) a kódban történt változtatásaikat a projekt fő ágába (main branch). A CI fő célja, a kódban lévő problémák gyors felismerése és javítása, valamint annak biztosítása, hogy az új funkciók vagy hibajavítások gyorsan integrálódjanak és automatikusan tesztelve legyenek.

- A fejlesztők gyakran, néha naponta többször is hozzáadják a változtatásaikat a fő tárhoz (*repository*), pl. ha a fejlesztő új

funkciót ad hozzá a programhoz vagy kijavít egy hibát. Ez csökkenti a kódban lévő nagyobb konfliktusok kockázatát.

- Minden új kódváltozás, amelyet a fejlesztő feltölt a tárba, automatikusan elindít egy tesztkészletet.
- Ha a tesztek sikeresen lefutnak, a változások stabilnak minősülnek, és kisebb az esélyük, hogy hibát tartalmaznak.
- Ha a tesztek sikertelenek, a fejlesztő azonnal visszajelzést kap.

Continuous Deployment (CD) és Continuous Delivery (szintén CD) nagyon hasonló fogalmak, de lényeges különbség van a folyamat utolsó lépésében. Mindkettő a szoftverek automatikus telepítéséhez kapcsolódik, de abban különböznek, hogy milyen mértékig terjed ki az automatizáció.

- **Continuous Delivery** azt jelenti, hogy a szoftver mindig olyan állapotban van, hogy bármikor telepíthető legyen a produkciós környezetbe. A fejlesztési, tesztelési és telepítési előkészítés folyamata automatizált, de a tényleges telepítéshez manuális beavatkozás vagy jóváhagyás szükséges – a csapat eldöntheti, hogy pontosan mikor telepítik a szoftvert. Ezzel a módszerrel a fejlesztői csapat biztosítja, hogy a kód mindig kész legyen a telepítésre, de a telepítést szükség vagy ütemezés alapján végzik el.
- **Continuous Deployment** esetében minden, ami a tesztelésen sikeresen átment és telepítésre készen áll, automatikusan telepítésre kerül a produkciós környezetbe, manuális beavatkozás nélkül. A Continuous Deployment nagyon magas szintű automatizációt és megbízható tesztelést igényel, mivel bármilyen hiba, ami bekerül a kódba, azonnal bekerülhet a produkciós környezetbe is. A Continuous Deployment hasonlóan működik, mint az automatikus frissítések az ismert szoftverek esetében.

A **produkciós környezet** az a környezet, ahol az alkalmazás „élesben” fut, és ahol a nyilvánosság vagy a végfelhasználók használják. A produkciós környezet a legfontosabb környezet, mivel itt működik az alkalmazás úgy, ahogy kell, valós körülmények között, és hozzáférhető a felhasználók számára. A produkciós környezet tehát az a hely, ahol az alkalmazás „elérhető”. A produkciós környezet **nem egy operációs rendszer**, bár az operációs rendszer a produkciós környezet része. A produkciós környezet[36] egy tágabb fogalom, amely mindent magában

[36] A produkciós környezet tehát egy fiktív kifejezés, amely magában foglal minden platformot, hardvert, hálózatot, egyéb szoftvert és adatbázist, amennyiben azokat valamilyen szoftver, például egy vállalati információs rendszer használja.

foglal, ami a szoftverrendszer vagy alkalmazás működtetéséhez szükséges a valós felhasználók számára. Tehát, operációs rendszereket, hardver és felhő infrastruktúráját vagy cloud-ját, adatbázisokat, biztonsági elemeket, és egyéb elemeket, amelyek biztosítják, hogy a szoftver biztonságosan és megbízhatóan működjön a valós felhasználók számára[37].

A **deploy** (vagy telepítés) egy kifejezés a szoftverfejlesztés területén, amely azt a folyamatot jelenti, amikor egy szoftvert vagy alkalmazást üzembe helyeznek, azaz telepítik egy olyan környezetbe, ahol azt valós felhasználók használhatják. Ez a folyamat magában foglalja a szoftver állományainak átvitelét, telepítését és konfigurálását a szervereken vagy más eszközökön, ahol futni fog, és szolgáltatásokat nyújt majd. A telepítés különböző környezettekben történhet, például:

- fejlesztői környezet (IDE – *Integrated Development Environment*): itt dolgoznak a fejlesztők, ahol a kódot írják és tesztelik;
- tesztkörnyezet (*Testing environment*): itt történnek a tesztelések az alkalmazás működésének ellenőrzésére;
- staging környezet (SE – *Staging Enviroment*) a lehető legjobban hasonlít a produkciós környezetre, ebben végzik az utolsó ellenőrzést, hogy megbizonyosodjanak arról, hogy az alkalmazás helyesen fog működni, valós körülmények között;
- produkciós környezet (*Production Environment*): a végleges környezet, ahol az alkalmazás valós felhasználók számára fut.

Kompletizálást tekintve ezt a fejezetet a telepítések felosztásával zárjuk. A telepítés három módon oldható meg:

- **manuális telepítés**;
- **automatizált telepítés** (Continuous Deployment): az egész telepítési folyamat automatizált, így a szoftver automatikusan települ a sikeres tesztelés után (például, olyan eszközökkel, mint a Jenkins vagy a GitLab CI/CD);
- **Blue-Green telepítés**[38]: egy stratégia, ahol két verzió létezik a produkciós környezetben (kék és zöld). Az új szoftververzió a zöld környezetbe települ, miközben a kéken még fut az alkalmazás. A kék környezet az aktuális, éles verzió, amelyet a felhasználók

[37] Például egy webes alkalmazás a produkciós környezetben futhat Linux operációs rendszeren, használhat MySQL adatbázist, telepítve lehet felhőszervereken (például az AWS-en), és biztosítva lehet titkosítással és tűzfalakkal.

[38] A Blue-Green telepítés lényege, hogy két különálló, párhuzamos környezetet hozunk létre (például „kék” és „zöld”), amelyek közül az egyik élesben fut, a másik pedig készen áll az új verzió telepítésére.

használnak. Miután a zöld környezetbe telepített verzió készen van, egy gyors váltással a zöld környezet válik élessé, és ez lesz a felhasználók által elért produkciós környezet. A következő frissítéskor a környezetek szerepei megfordulnak. Előnye a minimális kiesés, mert a felhasználók számára alig észrevehető a váltás és gyors visszaállítás, ha probléma van az új verzióval.

A DevOps a vállalati rendszer teljes életciklusára összpontosít, és automatizálja a telepítési, tesztelési és felügyeleti folyamatokat. A CASE inkább a tervezési fázist automatizálja, és a tervezési diagramok alapján kódot generál. (Kim, G.; Humble, J.; Debois, P.; Willis, J. 2016) (Humble, J.; Farley, D. 2010)

6.6 Az emberek és a siker

A gyakorlat azt mutatja, hogy sok esetben – még a kisebb vállalatoknál is – jelentős különbségek figyelhetők meg a vezetőség nézeteiben. Ahhoz, hogy konkrét intézkedéseket lehessen hozni a munkavállalók motivációjának növelésére, először is ismerni kell a vállalkozás helyzetét. Az emberek, vagy akár a menedzsment negatív hozzáállását számos tényező befolyásolhatja. Ezek az alábbiak lehetnek.

- **Orientációs korlátozások:** például az információs rendszerek piacának alapos ismeretének hiánya, a rendelkezésre álló megoldások és beszállítók áttekintésének nehézségei.
    - Pl. egy tanácsadó cég bevonása segíthet a piaci áttekintésben és a megfelelő megoldás kiválasztásában.
- **Technikai korlátozások:** az infrastruktúra hiánya, amely szükséges lenne az információs rendszer, vagy annak egy kritikus részének megvalósításához.
    - Pl. a helyi rendszerek telepítése helyett a felhőalapú megoldásokra való átállás lehet hatékony és költségkímélő alternatíva.
- **Lélektani korlátozások:** az emberek természetes vonakodása és ellenállása a változásokkal szemben, amit le kell küzdeni.
    - Pl. a vállalat tréningeket szervezhet, és támogató csoportot hozhat létre, hogy megkönnyítse az átállást és növelje a munkavállalói elégedettséget.
- **Nyelvi korlátozások:** különböző nyelvi akadályok az alkalmazásokban vagy a segédprogramokban.

- Pl. érdemes fontolóra venni egy olyan megoldást, amely többnyelvű támogatást biztosít, vagy helyi fordítást kell készíteni az alkalmazottak számára.

Az emberek természetes ellenállásának leküzdése a változásokkal szemben, és így az új információs rendszer bevezetésére vagy egy meglévő információs rendszer korszerűsítésére irányuló erőfeszítésekkel szemben is megfelelő irányítási megközelítést igényel.

Lewin változásmodellje (Kurt Lewin's Change Model) a változásmenedzsment klasszikus megközelítése, amelyet számos iparágban alkalmaznak. Kurt Lewin által tervezett modell célja, hogy megmagyarázza, hogyan tudnak a szervezetek hatékonyan átmenni egyik állapotból a másikba. A modell három alapvető fázist tartalmaz.

- ***Kiolvasztás**: ebben a fázisban a jelenlegi állapotot meg kell változtatni, és az embereket meg kell győzni, hogy hagyjanak fel a régi szokásokkal és gyakorlatokkal. Ez egy előkészítő fázis, amelyben a vezetők a változás szükségességének tudatosítására összpontosítanak. Gyakran magában foglalja a változás okainak kommunikálását és a változás elfogadására irányuló motiváció kialakítását.*
- ***Változás**: amikor új folyamatokat, rendszereket vagy szokásokat vezetnek be. Ebben a fázisban gyakorlati lépésekre van szükség a változás megvalósításához és a munkavállalók támogatására az új módszerek átvételéhez. A kommunikáció, a támogatás és a képzés itt kulcsfontosságú az ellenállás csökkentése és a változás elfogadásának növelése érdekében.*
- ***Újrafagyasztás**: az utolsó fázisban az új folyamatok és magatartásformák standardként beágyazódnak. A cél annak biztosítása, hogy a változás stabil és a szervezeti kultúra részévé váljon. Ez magában foglalhatja a struktúra kiigazítását, a rendszeres visszajelzést és az új munkamódszer elfogadásáért járó jutalmakat.*

Lewin modelljét egyszerű (természetesen létezik több modell is, pl. Úszó gleccser modellje), de hatékony változáskezelési eszköznek tartják, mert világos lépéseket és fázisokat biztosít ahhoz, hogy a szervezetek sikeresen alkalmazkodjanak az új változásokhoz. A modell opponensei gyakran azzal érvelnek, hogy a modell nem nagyon akceptálja a vállalat evolúcióját és valószínűleg figyelmen kívül hagyja a szociális aspektusok egyensúlyhiányát.

- **Értelemi korlátozások:** az információs rendszer különböző funkciói megértésének hiánya.

    - Pl., itt is a vezetőség ismertető tréningeket szervezhet, hogy megmagyarázza, hogyan növelheti a rendszer a vállalati hatékonyságot és a pontosságot.
- **Politikai korlátozások:** az ország jogi környezetében bekövetkező változásokból eredő kényszerű módosítások, például a számvitelt érintő törvénymódosítások.
    - Pl. a könyvelési rendszer frissítése, vagy akár egy teljesen új, szabályozáshoz igazított szoftver beszerzése válhat szükségessé.
- **Pénzügyi korlátozások:** nem megfelelő a pénzügyi háttér az információs rendszer beszerzéséhez.
    - Pl. a cégek megfontolhatják a bérleti konstrukciókat vagy előfizetés-alapú szolgáltatások igénybevételét.
- **Egyéb tényezők:** bármely további akadály, amely hátráltathatja az információs rendszerek sikeres bevezetését.
    - Pl., időhiány miatt egy vállalatnál nehezen találják meg az optimális időpontot az új rendszer bevezetésére (a napi működést nem akarják megzavarni). Ezért olyan időszakot választanak, amikor kevesebb zavar keletkezik a működésben.

Ezen pontok közelebbi vizsgálata megmutatja, hogyha komolyan akarunk bevezetni egy új információs rendszert, akkor csak a megfelelő vezetői döntéseken múlik, hogy milyen megoldásokat találnak. Egy információs rendszer konkrét projektként történő megvalósításakor előnyösebb, ha az alkalmazottak csapatban dolgoznak. A csoportban és a csapatban végzett munka irányítása eltérő. Csoportban végzett munka esetén fontos a feladatok megfelelő elosztása a tagok között, és a szükséges eredmények összehangolása. A csapatmunka olyan feladatok megoldására szolgál, amelyekben az összes tag kreatív potenciálját összekapcsolja. (Sziray, J.; Gaul, G. 2006) (Sziray, J.; Gaul, G.; Erdős, F. 2006) (Sziray, J.; Gaul, G.; Égertné Molnár, É. 2007)

6.7 A vállalati információs rendszer bevezetésének hatékonysága

A vállalati információs rendszer bevezetésének hatékonysága nem mindig számszerűsíthető egyértelműen. Ezért az információs rendszer bevezetésének hatékonyságát és hasznosságát két módon fejezhetjük ki:

- kvantifikálható (mennyiségileg mérhető) előnyök meghatározásával;
- nem egyértelműen számszerűsíthető – nem kvantifikálható előnyök meghatározásával.

A **kvantifikálható előnyök esetében** általában azt vizsgáljuk, hogy milyen különbség mutatkozik a vállalati információs rendszer alkalmazásával elért haszon és az implementációval és üzemeltetéssel járó költségek között. Fontos, hogy a szoftverbe történő beruházás és a bevezetés költségei gondosan megtervezettek, számszerűsítettek és a rendszer működése során folyamatosan nyomon követettek legyenek. A pénzügyi elmélet számos módszert kínál a gazdasági hatékonyság kiszámítására. Általában az alábbi, beruházási projektek értékelésére szolgáló mutatók alkalmazhatók.

- **Befektetésarányos megtérülés** (*ROI – Return on Investment*) – az a százalékos arány, amely a nettó haszon és a teljes költség hányadosaként számítható ki. Ha a ROI értéke 100%, akkor minden egyes befektetett euró 1 euró nyereséget és 1 euró költségvisszatérítést hoz:

$$ROI = \frac{\sum kumulált\ nettó\ haszon}{teljes\ költség} \cdot 100$$

- Az **éves átlagos költség** kifejezése:

$$R = O + i \cdot J + V$$

ahol:

R – éves átlagos költségek,
O – éves értékcsökkenési leírás (amortizáció),
i – kamatláb (százalékban kifejezve, osztva 100-zal), amely a minimális elvárt megtérülést jelenti,
J – beruházási költség,
V – egyéb éves üzemeltetési költségek.

- **Diszkontált költségek** módszere – ez a módszer akkor alkalmazható, ha a különböző időszakokban felmerülő költségeket is figyelembe kívánjuk venni, vagyis azokat diszkontálni kell:

$$D = J + V_r - L$$

ahol:

D – diszkontált költségek,
J – beruházási költség (ár),
V_r – diszkontált egyéb üzemeltetési költségek,
L – a beruházás diszkontált maradványértéke.

- **Átlagos hozam** kifejezése:

$$I = \frac{\sum_{i=1}^{n}(Z_i + O_i)}{n}$$

ahol:

I – az éves átlagos hozam,
Z – az éves adózott nyereség az egyes években,
O – éves értékcsökkenési leírás az egyes években,
i – az adott év,
n – az élettartam éveinek száma.

- **Nettó jelenérték** (*NPV – Net Present Value*) – a nettó jelenérték a beruházásból származó diszkontált pénzáramlások és a tőkekiadások különbségeként számítható ki. Ha az értéke nagyobb, mint 0, akkor a beruházási projekt elfogadható, és biztosítja az elvárt hozamot, illetve növeli a vállalat piaci értékét:

$$NPV = \sum_{n=1}^{N} \left(\frac{P_n}{(1+i)^n} \right) - K$$

ahol:
P – az adott években elért pénzáramlás (adózás utáni nyereség + amortizáció),
i – kamatláb (százalékban kifejezve, osztva 100-zal),
N – az élettartam éveinek száma,
K – a kezdeti beruházás összege.

- **Belső megtérülési ráta** (*IRR – Internal Rate of Return*) egyenlete, melyben *i* az ismeretlen:

$$K = \sum_{n=1}^{N} \frac{P_n}{(1+i)^n}$$

ahol:
P – az adott években elért pénzáramlás (adózás utáni nyereség + amortizáció),
i – kamatláb (belső megtérülési ráta, IRR; százalékban kifejezve, osztva 100-zal),
N – az élettartam éveinek száma,
K – a kezdeti beruházás összege.

A fent említett mennyiségileg mérhető előnyök mellett léteznek **nem kvantifikálható előnyök is**, amelyek közé tartozik például:

- az adatok bázisának racionalizálása,
- a vállalati folyamatok hatékonyabb átalakítása,
- a termelési tényezők jobb kihasználása,
- hatékonyabb készletgazdálkodás,
- az ügyfélszolgálat javulása,
- a vállalat könnyebb bővíthetősége és növekedése,
- valamint az üzleti döntéshozatal hatékonyságának növelése.

Ezek az előnyök vagy azonnal megjelennek **a rendszer bevezetése után** (pl., a vezetői munka gyorsítása és hatékonyabbá tétele), vagy **hosszabb távon**, például jobb minőségű kimutatások és jelentések formájában, amelyek támogatják a vezetői döntéshozatalt. (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P., 2017)

7 Vállalati információs rendszerek életciklusa, karbantartása

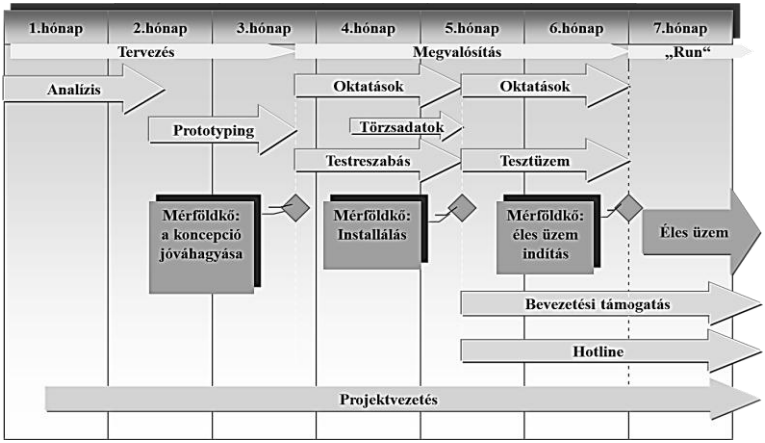
Egy vállalati információs rendszer bevezetése egy vállalaton belül általában komoly beruházási költséget jelenthet. Egy ilyen méretű beruházás nyilván csak akkor éri meg a vállalatnak, ha az alkalmazásával olyan költségmegtakarítást, ill. bevételnövekedést érnek el, ami kellően megnövekedett hasznot eredményez.

A vállalati információs rendszerek életciklusa több szakaszból áll, melyeket gyakran különböző nézőpontok alapján határoznak meg. Az életciklus-modellezés célja, hogy segítsen megtervezni és hatékonyan kezelni az információs rendszerek fejlesztését, bevezetését, működtetését és fenntartását. A leggyakrabban alkalmazott szakaszok a következők:

- stratégiai tervezés,
- elemzés,
- tervezés és fejlesztés,
- tesztelés,
- bevezetés,
- üzemeltetés és karbantartás,
- lecsengés és kivonás.

Az első szakasz a **stratégiai tervezés**, ahol meghatározzák az információs rendszer követelményeit, céljait, és az elérni kívánt eredményeket. Fontos a szükséges erőforrások felmérése, és a projekt kereteinek kidolgozása. Az információs rendszerek tervezése, fejlesztése és bevezetése a gyakorlatban jellemzően **projekt formában történik**. Az ilyen projektek jól meghatározott szakaszokra vannak bontva, amelyekben a célkitűzések, feladatok és felelősségi körök egyértelműen le vannak osztva. A projekt egyes fázisainak mérföldkövek mentén történő nyomon követését alkalmazzák, pl. a költségek és az időkeretek hatékony kihasználására.
Az **elemzés** fázisában történik az igények részletes vizsgálata, a folyamatok és a jelenlegi rendszer alapos elemzése. A cél, hogy feltárják azokat a hiányosságokat, amelyeket az új rendszernek kell kezelnie, és meghatározzák azokat a funkciókat, amelyek az üzleti folyamatokat legjobban támogatják.

A **tervezés és fejlesztés** (*Design & Development*) fázisában kialakítják a rendszer architektúráját és megtervezik az alkalmazás funkcióit. A tervezést követően kezdődik a fejlesztés, ahol a konkrét technológiai megoldásokat implementálják.
A **tesztelési szakaszban** az elkészült rendszer különböző próbákon megy keresztül annak biztosítása érdekében, hogy a rendszer a meghatározott követelmények szerint működik, és mentes legyen a hibáktól.
A **rendszer bevezetésének** (*Deployment*) **szakaszában** a vállalat élesben kezd el dolgozni az új rendszerrel. Ez a szakasz magában foglalhatja a felhasználók oktatását és támogatását, az adatátvitelt és az esetleges finomhangolásokat is. Az információs rendszerek bevezetésekor, különösen, ha egy régi rendszert cserélnek le, gyakran használnak olyan megközelítést, ahol a régi és az új rendszer egy ideig párhuzamosan működik. Ez lehetővé teszi a zökkenőmentes átállást és csökkenti az adatvesztés, a hibák vagy a működés megszakadásának kockázatát.
Az **üzemeltetés és karbantartás** (*Operation & Maintenance*) során a rendszer élesben működik, és folyamatos támogatást, karbantartást igényel, hogy mindig optimális állapotban legyen. Ennek része a hibajavítás, a kisebb módosítások, valamint az esetleges teljesítményoptimalizálások.
A **lecsengés és kivonás** (*Decline & Retirement*) a rendszer életciklusának utolsó fázisa. Az adott információs rendszert már elavultnak tekintik, vagy más rendszerek váltják fel. Ilyenkor a régi rendszer kivonása történik, hogy átadja helyét az új megoldásnak.



7.1. ábra: Példa az információs rendszer bevezetésének lépéseire időbontásban. Forrás: https://www.slideserve.com/salma/infor-com-fitt-tessz-k

Több megoldás is létezhet egy vállalati információs rendszer bevezetésére. Sziray, J.; Gaul, G.; Molnár, É. (2007) szerint, egy vállalati

információs rendszernek természetes életciklusa 5–8 évre tehető. Az egyes fázisai és azok időtartama a következő:

- a prototípus és a kezdeti kiépítés előállítása (6–8 hónap),
- alkalmazás és továbbfejlesztés (18–24 hónap),
- felhasználók számának növelése és az alkalmazás megszilárdítása (18–24 hónap),
- komolyabb átalakítás az újabban felmerülő igények alapján (18 hónap),
- újra üzemeltetés (12 hónap).

7.1 Vízesés-, V- és agilis módszerek

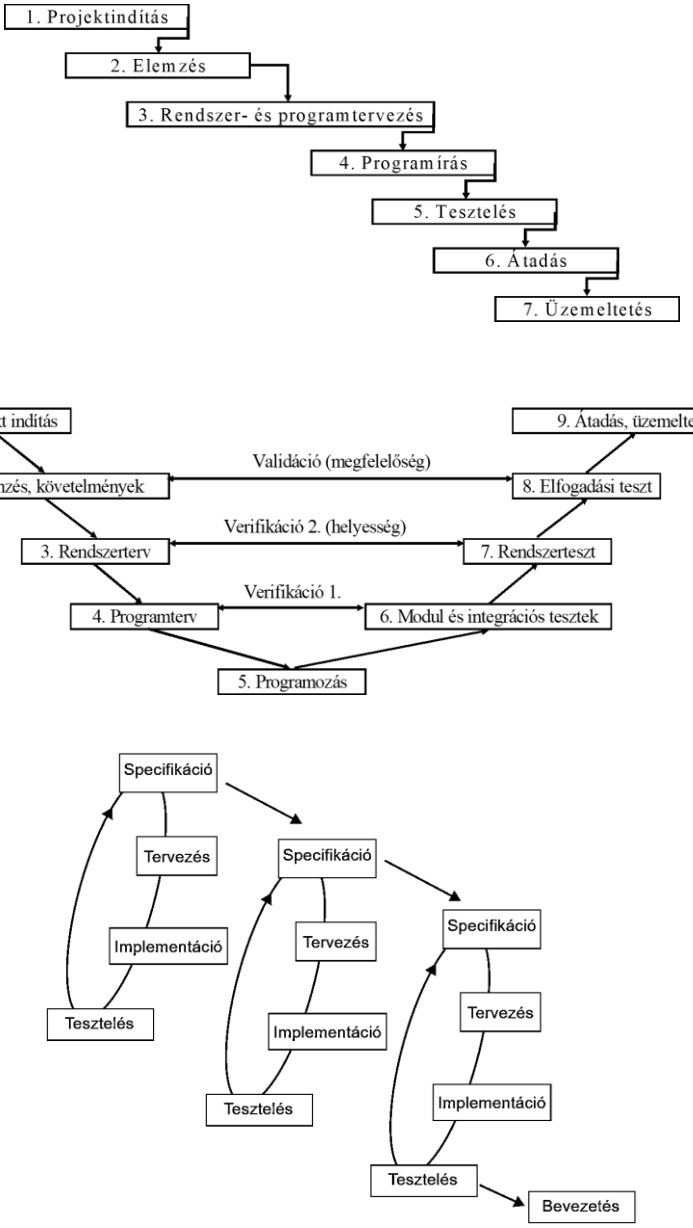
A V-modell és a vízesés modell is egy klasszikus fejlesztési megközelítések, amelyek gyakran kapcsolódnak az információs rendszerek életciklusához. Ezeket a modelleket főként a tervezési és fejlesztési szakaszban alkalmazzák. A **vízesés modell** a szoftverfejlesztés egy lineáris megközelítése, amelynél az egyes szakaszok egymás után következnek, és csak az egyik szakasz lezárása után lehet továbblépni a következőre. A vízesés modell lépései megfelelnek az életciklus szakaszainak (tervezés, elemzés, implementálás, tesztelés, bevezetés és karbantartás). Egyértelmű szakaszokkal és jól dokumentált lépésekkel rendelkezik, ami különösen jól működik jól meghatározott, stabil követelmények mellett, melyek már a kezdetektől fogva egyértelműek és kevés változás várható. A vízesésmodellt elsősorban nagy és bonyolult informatikai rendszerek esetén alkalmazzák, amikor a projektben részt vevők száma nagy (általában 20 feletti). Hátránya viszont, hogy kevésbé rugalmas a változásokra, mivel nehéz visszalépni egy korábbi szakaszba, ha új követelmények merülnek fel. Az ilyen projektek átfutási ideje hosszú, gyakran akár több évet is igénybe vehet, és az eredmény csak a projekt végén áll rendelkezésre. A **V-modell** fokozott hangsúlyt fektet a tesztelési tevékenységekre. A V-modell egyértelműen megmutatja a fejlesztési és tesztelési lépéseket, és minden fejlesztési szakaszhoz párosít egy tesztelési szakaszt, ezzel biztosítva, hogy minden tervezett elem ellenőrizve legyen, mielőtt a rendszer élesben használatba kerülne. A V modell integrálja a tesztelést minden szakaszban, ami korán feltárhatja a hibákat, és így időt takaríthat meg. Ez a modell jól működik stabil projektek esetében, a szigorú minőségi követelmények betartására és a hibák minimalizálása érdekében használjuk. El kell mondanunk, hogy kevésbé rugalmas, ha a követelmények időközben

változnak. Az **agilis[39] módszerek** eltérnek a hagyományos módszerektől, és az információs rendszer teljes életciklusában szerepet vállalnak, sőt gyakran az éles működtetésben és karbantartásban is alkalmazhatók. Az agilis megközelítés központi eleme a rugalmasság, a gyors reagálás a változó igényekre és a folyamatos javítás, ami jól illeszkedik az információs rendszerek dinamikus környezetéhez. Előnye, hogy lehetővé teszi a fejlesztési folyamat folyamatos ellenőrzését és javítását, ami fontos a gyorsan változó piaci környezetben. Az agilis megközelítés tipikus jellemzői a következők.

- **Iterációs folyamatok** – a fejlesztést rövid, ismétlődő ciklusokban (iterációkban vagy sprintekben) végzik, amelyeken belül a csapat egy-egy részfeladatot vagy működőképes verziót hoz létre.
- **Folyamatos visszajelzés és tesztelés** – a rendszeres visszajelzések alapján a csapat gyorsan javít és finomít, így a kész rendszer jobban megfelel a felhasználói igényeknek.
- **Rugalmasság a változó követelményekhez** – az agilis módszertan alkalmazkodik a változó igényekhez, legyenek azok akár üzleti követelmények, akár technológiai fejlesztések.
- **Keresztfunkcionális csapatok** – az agilis projektekben interdiszciplináris csapatok dolgoznak együtt, amelyekben minden szükséges szakértelem megvan, így gyorsabb döntéshozatal és hatékonyabb fejlesztés valósulhat meg.
- **Minimális bürokrácia és önállóság** – a csapatoknak nagyobb önállóságuk van a döntéshozatalban, így gyorsabban tudnak reagálni a felmerülő kihívásokra.

Manapság modern, agilis módszerek pl. a Scrum és a Kanban, de vannak más érdekes irányzatok is, pl. az eXtreme Programming vagy a DSDM. Emellett régebbi megközelítések is ismertek, például a spirális fejlesztési modell, az evolúciós modell, a gyors alkalmazásfejlesztés (RAD) vagy a Rational Unified Process (RUP), stb. Nézzük meg részletesebben a Scrum módszert. A Scrum-ban a projekt munkafolyamata sprint ciklusokra van bontva, amelyek általában 2-4 hétig tartanak. Minden sprint végén a csapat egy működő, tesztelt rendszerverziót ad át.

[39] Az agilis megközelítést 2001-ben dolgozták ki, és négy alappillérre épít: az agilis módszertanok az emberek közötti együttműködésre építenek; a fókusz azon van, hogy minél hamarabb működő rendszerverziókat hozzanak létre, amelyeket a felhasználók azonnal kipróbálhatnak; a fejlesztés során folyamatos kapcsolatot tartanak fenn az ügyféllel; jellemző a változásra való reagálás a tervek követése helyett.



7.2. ábra: Fentről lefelé haladva: Vízesés modell, V modell és agilis modell - példák
Forrás: Komló, 2013,
https://nkerepo.uni-nke.hu/xmlui/bitstream/handle/123456789/14049/Web_PDF_Mestersegesintelligencia-eszkozok_szakertoi_rendszerek_alkalmazasa_a_kozigazgatasban.pdf

A Scrum fő lépései:

- **Sprint** – rögzített hosszúságú ciklusok, amelyekben meghatározott célokat érnek el.
- **Scrum csapat** – interdiszciplináris csapat, amelyben minden szerepkör (fejlesztők, tesztelők stb.) együtt dolgozik.
- **Scrum rutinok** – napi stand-upok, sprint tervezés, sprint demók és retrospektív találkozók, amelyek célja a folyamatos kommunikáció és javítás.
- **Product Backlog és Sprint Backlog** – az összes feladatot és követelményt tartalmazó listák, amelyeket a csapat priorizál a sprint céljainak megfelelően. (Komló, 2013)

7.2 Karbantartás

Az információs rendszerek karbantartásának tervezése kulcsfontosságú a rendszerek hosszú távú megbízhatósága és hatékonysága érdekében. Gondoljunk csak bele. Ha a rendszert le kell állítani, pl. elhanyagolt hiba miatt, akkor az pénz veszteséget jelent. Ilyen esetekben nem tudjuk előre megmondani, menyi időt fognak a javítások igényelni. Sőt, ilyen kiesések esetében előfordulhat az adatok elvesztése, aminek következményei is lehetnek a rendszer egyes hierarchiai szintjein. A karbantartásokat különböző szempontokból közelíthetjük meg és különböző csoportokra oszthatjuk. Az alábbiakban **főleg** a karbantartások **mérete és tervezése szempontjából** állítottuk össze a kategorizálást.

- **Rutin karbantartások**.
    - Ezek a mindennapi vagy heti rendszerességű, kisebb feladatok, amelyek célja a rendszer zökkenőmentes működésének fenntartása. Ilyen lehet pl. a szükséges biztonsági frissítések elvégzése, a számítógép megfelelő leállítása vagy energiaellátás figyelése.
- **Tervezett karbantartások**.
    - Ezeket előre meghatározott időpontokban hajtják végre, pl. negyedévente vagy évente. A tervezett karbantartás során gyakran komolyabb rendszerellenőrzéseket, szoftver-frissítéseket és hardveres ellenőrzéseket végeznek. Ide tartoznak olyan feladatok is, amelyeknél a gyártó vagy a rendszergazda előre meghatározza, hogy a komponensek, mint pl. egy merevlemez, egy adott időszak vagy használati határ után cserére szorulhatnak.

- **Nagyobb méretű karbantartások**.
    - Ezek a nagyobb, ritkábban végzett, de komolyabb beavatkozást igénylő feladatok, amelyek lehetnek infrastruktúra-frissítések, adatbázis-reorganizációk vagy nagyobb szoftververziók telepítése. Ezekhez gyakran hosszabb rendszerleállás is társulhat, amelyet alapos tervezés előz meg. Célja az, hogy az egész rendszert és infrastruktúrát átfogó módon ellenőrizzék, optimalizálják vagy frissítsék. Ha egy vállalat úgy dönt, hogy a teljes információs rendszerét lecseréli vagy egy új platformra áll át, azt is gyakran a nagy karbantartás részeként kezelik. A nagy karbantartás valóban stratégiai fontosságú, és gondos tervezést igényel, hogy minimalizálják a leállásból adódó költségeket és zavarokat. Vészhelyzeti terveket (*Contingency Plans*) is készítenek, amelyek lehetővé teszik, hogy gyorsan visszaálljanak egy korábbi állapotra, ha valami váratlan hiba történne.
- **Eseményalapú (ad-hoc) karbantartások**.
    - Ezeket a karbantartásokat váratlan hibák, problémák esetén végzik. Gyakran nem előre tervezett, inkább a felmerülő problémák gyors megoldására irányul.
- **Prediktív karbantartások**.
    - Ezt a módszert az adatvezérelt megközelítés alapján végzik, amikor különböző rendszermutatók figyelése alapján előre jelzik a meghibásodásokat, és ezek előtt próbálják elvégezni a szükséges beavatkozásokat. A prediktív karbantartás során figyelve vannak különböző rendszer- vagy teljesítménymutatók, mint pl.: hőmérséklet, vibrációk, áramfogyasztás, teljesítménycsökkenés, stb.

> *Például, az adatelemző algoritmusok azonosítják, ha egy eszköz hőmérséklete meghaladja az átlagos értékeket, és ha azzal együtt növekvő vibráció figyelhető meg, az nagy valószínűséggel azt jelezheti, hogy az eszköz alkatrészei kopnak, és közelgő meghibásodás várható. Ilyenkor a prediktív karbantartási rendszer figyelmeztetést küld, hogy időben végezzék el a szükséges cseréket. (Preston, W. C. 2021)*

Elmondhatjuk, hogy a karbantartáshoz kapcsolódó változtatásoknál célszerű formális változáskezelési eljárást (jóváhagyás, tesztkörnyezet, visszagörgetési terv) alkalmazni. (Preston, W. C. 2021)

7.2.1 Vészhelyzeti tervek

A **vészhelyzeti tervek** azokat az előre meghatározott lépéseket tartalmazzák, amelyeket akkor alkalmaznak, ha a karbantartás vagy frissítés során váratlan problémák lépnek fel. Nézzünk néhány példát, amelyeket gyakran szoktak alkalmazni a gyakorlatban:

- adatmentési és visszaállítási terv (*Backup and Restore*),
- Failover vagy redundáns rendszer bevezetése,
- Roll-back terv,
- kommunikációs terv,
- alternatív hozzáférési módok,
- tesztkörnyezet.

Mielőtt bármilyen nagyobb változtatást végrehajtanának, általában **teljes adatmentést** (*Full Backup*) készítenek az aktuális rendszerről. Ha a karbantartás során váratlan hiba lép fel, és vissza kell állítani a korábbi állapotot, a mentett adatokat és beállításokat gyorsan vissza lehet tölteni. Ez a gyakorlat különösen fontos adatbázis-frissítések esetén. Viszont itt is létezhetnek más megoldások is. Például attól függően, hogy mennyi adatot kell menteni, milyen gyors visszaállításra van szükség, és mennyi tároló kapacitásunk van, létezik a már említett teljes adatmentés mellett néhány gyakorlatba bevált megoldás. Ezek az alábbiak.

- **Részleges mentés** (*Differential Backup*)
    - Csak azokat az adatok vannak mentve, amelyek a legutóbbi teljes mentés óta módosultak. A visszaállításnál szükség van az utolsó teljes mentésre, és a legutolsó részleges mentésre.
- **Növekményes mentés** (*Incremental Backup*)
    - Csak azok az adatok vannak mentve, amelyek a legutóbbi mentés óta változtak, akár teljes, akár részleges volt a legutóbbi mentés. Ez a leggyorsabb módja a mentésnek, és kevesebb tárhelyet igényel, de a visszaállításnál minden egyes növekményes mentést alkalmazni kell az utolsó teljes mentésre.
- **Log mentések** (*Transaction Log Backups*)
    - Pl. egy adatbázis tranzakciós naplóiról készít biztonsági mentést. Ez lehetővé teszi, hogy a legapróbb változások is visszaállíthatók legyenek, akár egy bizonyos időpontra pontosan. Az adatbázisban történt módosítások logját

követi, és visszaállítható bármely időpontra, amíg a naplóadatok rendelkezésre állnak.

Stratégia szempontból elgondolkozhatunk azon, hogy ezek nem kombinálhatók-e valamilyen módon egymás között, kiemelve azt az előnyt, amit együtt kínálhatnának. Válaszunk az, hogy igen. A vállalatok kombinálják ezeket a módszereket. Például hetente egyszer készíthetnek egy teljes mentést, majd naponta növekményes mentést vagy részleges mentést. A kritikus rendszerek esetén gyakoriak lehetnek a log mentések is, hogy minimalizálják az adatvesztés esélyét. A teljes mentés kombinálható a roll-back tervvel és akár a failover rendszerrel is, hogy növelje a rendszer biztonságát és rendelkezésre állását. A kombinációk nagyon hatékonyak lehetnek a gyors visszaállítás és a folyamatos működés biztosítása érdekében. Az információs rendszert tekintve mondhatjuk, hogy ugyanezen elveket használják, ahol nemcsak az adatokat, hanem a rendszer konfigurációit, szoftverfrissítéseket és beállításokat is figyelembe veszik.

A kritikus rendszerek esetében használhatnak **redundáns** rendszereket vagy **failover megoldásokat**. Ez azt jelenti, hogyha az elsődleges rendszer meghibásodik, a forgalmat automatikusan egy másik, ugyanolyan rendszerre irányítják. Például, a banki vagy egészségügyi rendszerek gyakran használnak ilyen megoldásokat, hogy biztosítsák a folyamatos elérhetőséget. A **redundáns rendszer** esetében a redundancia azt jelenti, hogy van egy vagy több tartalék komponens (pl. szerver, hálózati eszköz, merevlemez stb.), amely készen áll a működésre, ha az elsődleges rendszer meghibásodik. Ez azonban nem feltétlenül jelenti azt, hogy a tartalék komponensek aktívan működnek és **nem minden esetben szinkronizáltak**. Például lehet, hogy csak készenlétben állnak. Ebből kiindulva tehát megkülönböztetünk:

- **meleg redundanciát** (*Hot Redundancy*):
  - a tartalék rendszer folyamatosan működik, és azonnal át tudja venni a feladatot hiba esetén.
- **hideg redundanciát** (*Cold Redundancy*):
  - a tartalék rendszer nincs bekapcsolva, és szükség esetén manuálisan vagy automatikusan aktiválódik.

A **failover rendszer** már egy aktív, automatizált folyamat, amely a szolgáltatást vagy feladatot átváltja egy másik, működőképes komponensre, ha a fő rendszer meghibásodik. A failover rendszer általában hot redundanciával működik (pl. egy failover klaszter esetében, ha az egyik szerver meghibásodik, a klaszter másik tagja azonnal átveszi a feladatot

automatikusan)[40], de létezhetnek Warm vagy Cold failover konfigurációk is, attól függően, hogy milyen gyorsaságú átállást igényel az adott alkalmazás vagy üzleti környezet, vagyis:

- **Hot Failover** – a tartalék rendszer teljes mértékben működőképes és folyamatosan szinkronizálódik az elsődleges rendszerrel, így hiba esetén az átállás azonnali.
- **Warm Failover** – a tartalék rendszer nem feltétlenül működik teljes kapacitással vagy azonnal, de részben készen áll. Ilyenkor az átállás valamivel több időt vehet igénybe, mert például csak néhány lépés szükséges ahhoz, hogy teljes kapacitással működhessen.
- **Cold Failover** – a tartalék rendszer nincs bekapcsolva vagy aktívan készenlétben, így meghibásodás esetén késleltetéssel lép működésbe mivel a teljes rendszer aktiválásához és szinkronizálásához időre van szükség.

Roll-back terv egy olyan terv, amely lehetővé teszi, hogy ha a frissítés vagy a karbantartás sikertelen, akkor gyorsan vissza tudjanak térni a korábbi, jól működő verzióra. Például, ha egy új szoftverfrissítés hibákat okoz, a rendszergazda gyorsan visszaállíthatja a régi verziót. A **kék-zöld telepítés** (*Blue-Green Deployment*) egy olyan megközelítés, amely kifejezetten a roll-back terv hatékony végrehajtására szolgál, és nagyon hasznos nagy karbantartásoknál vagy frissítéseknél.

Ha a karbantartás vagy frissítés váratlan problémák miatt elhúzódik, fontos, hogy az érintett felhasználókat, partnereket és ügyfeleket értesítsék. A **kommunikációs terv** pontosan meghatározza, hogy ki, mikor és milyen csatornán értesíti a felhasználókat. Érintettek csoportok a belső felhasználók, partnerek (olyan cégek, amelyek közvetlen kapcsolatban állnak az információs rendszerrel) vagy ügyfelek (a végfelhasználók, akik szolgáltatásokat vesznek igénybe a rendszeren keresztül). Kommunikációs csatornák lehetnek pl.: email, SMS vagy push értesítések, weboldal vagy alkalmazáson belüli értesítés esetleg közösségi médiák, melyek széles körű tájékoztatást tudnak nyújtani a nyilvánosság számára.

Szintén kritikus rendszerek esetén egyes cégek **alternatív hozzáférési módokat** biztosíthatnak, például egy egyszerűsített,

[40] Tehát, redundáns rendszer átállását a rendszergazdának kell inicializálni (pl. manuálisan bekapcsolni), adatok nincsenek minden esteben szinkronizálva. Failover rendszereknél az átállás teljesen automatikus, ha kell, akkor a rendszer kapcsolja be a Cold vagy Warm rendszereket is. Így a failover rendszerek biztosítják a legmagasabb fokú automatizálást és gyors reagálást egy meghibásodás esetén, míg a redundáns rendszerek inkább egy tartalék kapacitásként szolgálnak, ahol a beavatkozás gyakran manuális.

alapfunkciókat biztosító verzió segítségével. Így, ha a fő rendszer nem érhető el, a felhasználók még mindig el tudják végezni a legfontosabb feladatokat. Például, az e-kereskedelmi platformoknál a vásárlók böngészhetik a termékeket, de a részletes termékleírások vagy a felhasználói profil funkciói esetleg ideiglenesen nem elérhetők; vagy ha az internetbank karbantartás alatt áll, a mobilalkalmazás egyszerűsített verziója még működőképes lehet. Az alternatív hozzáférési módok tehát lehetővé teszik, hogy a felhasználók a legfontosabb feladatokat akkor is el tudják végezni, amikor a teljes rendszer nem elérhető. Ezek a megoldások biztosítják, hogy a kritikus szolgáltatások folyamatosan hozzáférhetők legyenek

A nagyobb karbantartásokat gyakran egy **tesztkörnyezetben** végzik el először. Itt kipróbálják az összes frissítést és ellenőrzik, hogy minden jól működik-e. Ha a tesztkörnyezetben minden rendben van, akkor kisebb a kockázata annak, hogy a valódi környezetben problémák adódjanak. (Preston, W. C. 2021)

7.3 Veszélyek feltárása

A fejlesztés során számos probléma merülhet fel. Sziray, Gaul és Molnár (2007) szerint a leggyakrabban előforduló problémák közé tartoznak az alábbiak.

- A szervezeti változásokat nehezen követi a vállalati információs rendszer.
- Nem megfelelő projektvezetés gátolhatja a hatékony fejlesztési folyamatokat. Egy átgondolt projektvezetési módszertan (pl. agilis módszertanok, mint az említett Scrum vagy Kanban) azonban segíthet ebben.
- Gyenge kommunikáció alakul ki a felhasználók és az információszolgáltatók között, ami félreértésekhez vezethet.
- A vezetők közül sokan elzárkóznak az információk megosztásától, mivel úgy érzik, ezzel „védik birodalmukat”.
- Sokan csodát várnak a vállalati információs rendszertől, és ha az nem felel meg az elvárásaiknak, csalódottan elfordulnak tőle. Fontos, hogy a rendszer bevezetése előtt reális elvárásokat fogalmazzanak meg a vezetők, és világos kommunikációval tegyék érthetővé a felhasználók számára.
- Kulcsfontosságú, hogy a legbefolyásosabb vezetők támogassák a projektet, mert ennek hiányában az egész fejlesztés kudarcba fulladhat. A vezetők támogatása biztosítja, hogy a projekt

fontosságot kapjon a szervezeten belül, és a megfelelő erőforrások rendelkezésre álljanak a sikeres megvalósításhoz.

- Fontos, hogy a felhasználók csoportja érdekeltté váljon a rendszer használatában, ezzel elősegítve a vállalaton belüli együttműködést. Ha a felhasználók megértik a rendszer előnyeit, és részesülnek annak pozitív hatásaiból, nagyobb lesz a motivációjuk a rendszer hatékony használatára. (Preston, W. C. 2021)



7.3. ábra: Projektmunkán belül nagyon fontos a kommunikáció
Forrás: https://www.slideserve.com/salma/infor-com-fitt-tessz-k

Nézzünk egy példát arra, hogy két ember mennyire másképpen értheti ugyanazt a mérföldkövet. Például, Te azt mondod otthon, hogy háromkor indulunk, össze kell pakolni. Szerinted a mérföldkő az, hogy háromkor beindítod a kocsit. De a feleséged számára a mérföldkő nemcsak az, hogy már az autóban fog ülni, hanem az is, hogy rendben lesz a lakás, a háziállatok meg lesznek etetve, az edények el lesznek mosogatva, stb. Itt látható, hogy két ember mennyire másképp latja ugyanazt a mérföldkövet.
Forrás: (Procházka, P. 2024)

Az olvasó figyelmét szeretnénk szintén felhívni arra is, hogy a veszélyek fennállnak akkor is, mikor már az információs rendszert bevezettük. Ezek leküzdése nagyon fontos a vállalati információs rendszer maximális kihasználása szempontjából.

Ebben a fejezetben használt néhány kifejezés magyarázata

Az 1990-es években a Chrysler egy új bérügyi rendszer, a Chrysler Comprehensive Compensation System (C3) fejlesztésébe kezdett, amelynek célja az addig használt, több különálló szoftverből álló rendszer leváltása volt. Kent Beck 1996-ban csatlakozott a projekthez, hogy javítsa a fejlesztési folyamatokat, és ennek során dolgozta ki az ***eXtreme Programming (XP)*** *módszertant. A projektben Ron Jeffries is kulcsszerepet játszott. Az XP egy inkrementális fejlesztési megközelítés, amelyben a fejlesztés és a tesztelés szorosan összekapcsolódik, akár néhány órás ciklusokra rövidítve a visszacsatolási időt. Egyik legfontosabb sajátossága a tesztvezérelt fejlesztés (TDD), amelyben először a teszteket írják meg, majd ezekhez igazítják a kódot, biztosítva a folyamatos minőségellenőrzést. Az XP további jellegzetessége a páros programozás (pair programming), amely során két fejlesztő közösen dolgozik egy kódrészen: egyikük írja a kódot, míg a másik folyamatosan ellenőrzi. Bár az XP módszertana sikeresnek bizonyult, a C3 projektet végül 2000-ben leállították, amikor a Chrysler egyesült a Daimlerrel, és a rendszer fejlesztését nem folytatták tovább. Ennek ellenére az XP azóta is népszerű fejlesztési módszer maradt, különösen agilis környezetekben. (Komló, 2013)*

A ***spirális fejlesztési modellt*** *Barry Boehm amerikai matematikus publikálta az 1980-as évek végén az „A Spiral Model of Software Development and Enhancement" című cikkében. Ez a modell több szempontból is eltér a korábbi fejlesztési megközelítésektől. Egyrészt a szoftverfejlesztést nem egy lineáris folyamatként kezeli, hanem egy spirál alakú ciklusként ábrázolja, amelyben a fejlesztés ismétlődő fázisokon megy keresztül. Másrészt a modell kiemelt figyelmet fordít a fejlesztési kockázatok azonosítására és kezelésére. A spirál minden egyes ciklusa négy szektorra osztható:*

- *Célok meghatározása – az adott fejlesztési szakasz céljainak meghatározása és a követelmények tisztázása.*
- *Kockázatelemzés és kezelés – a potenciális kockázati tényezők azonosítása, valamint stratégiák kidolgozása ezek minimalizálására vagy kezelésére.*
- *Fejlesztés és tesztelés – a kockázatok kiértékelése után a legmegfelelőbb fejlesztési modell kiválasztása. A spirál minden ciklusában eltérő fejlesztési modell is alkalmazható a kockázatelemzés eredményei alapján.*
- *Döntés a folytatásról – a fejlesztési folyamat következő lépéseinek meghatározása. A ciklus végén eldől, hogy a fejlesztés folytatódik-e egy újabb iterációval, vagy lezárásra kerül a projekt.*

Ez a megközelítés lehetővé teszi a rugalmas fejlesztést, a folyamatos kockázatkezelést, valamint a projekt céljainak és követelményeinek iteratív finomhangolását. (Komló, 2013)

Gyors alkalmazásfejlesztés *– a Rapid Application Development (RAD) egy iteratív és folyamatalapú szoftverfejlesztési modell, amely gyors fejlesztési ciklusokat és rugalmas alkalmazkodást tesz lehetővé a változó követelményekhez. A módszert James Martin népszerűsítette az 1990-ben megjelent Rapid Application Development című könyvében, azonban a RAD koncepciója már a 1980-as években elkezdett formálódni, elsősorban a hagyományos vízesés modell lassúsága és merevsége miatt. Az IBM és más nagyvállalatok kezdték el alkalmazni a RAD-hoz hasonló fejlesztési módszereket a gyorsabb szoftverfejlesztés érdekében. A RAD fő sajátosságai közé tartozik az iteratív és inkrementális fejlesztési folyamat, amely rövid fejlesztési ciklusokkal és működő prototípusok gyors létrehozásával segíti a szoftverfejlesztést. A módszer kulcseleme a folyamatos felhasználói visszacsatolás, amely lehetővé teszi, hogy az alkalmazás funkcionalitása a fejlesztés során fokozatosan formálódjon a valós igények alapján. Emellett a RAD erőteljesen támaszkodik*

az integrált fejlesztőkörnyezetek (IDE), vizuális fejlesztői eszközök és adatbázis-kezelő rendszerek használatára, amelyek jelentősen felgyorsítják a fejlesztési folyamatot. Bár a RAD módszertana számos előnyt kínál, bizonyos kompromisszumokkal is jár. A gyors fejlesztés miatt előfordulhat, hogy a végleges alkalmazás nem teljesen optimalizált, és kompromisszumokat kell kötni a teljesítmény, a skálázhatóság vagy a funkcionalitás terén. Ennek ellenére az előnyök, például a jelentősen rövidebb fejlesztési idő, a felhasználói igények gyors kielégítése és a grafikus felhasználói felület (GUI) előtérbe helyezése, gyakran ellensúlyozzák ezeket a hátrányokat. A RAD különösen jól alkalmazható olyan projektek esetén, ahol a követelmények gyorsan változnak, és a gyors piacra lépés kiemelt szempont. Ugyanakkor nagy és összetett vállalati rendszerek fejlesztésére kevésbé alkalmas, mivel ezek esetében a részletes tervezés és dokumentáció elengedhetetlen. Napjainkban a RAD elveit gyakran kombinálják más agilis fejlesztési módszerekkel, például a Scrum vagy az eXtreme Programming (XP) megközelítéseivel. (Komló, 2013)

*A **Rational Unified Process (RUP**) egy iteratív és inkrementális szoftverfejlesztési módszertan, amelyet eredetileg a Rational Software Corporation fejlesztett ki az 1990-es években. Az IBM 2003-ban felvásárolta a Rational Software-t, és azóta továbbfejlesztette és integrálta saját fejlesztési megoldásaiba. A RUP célja, hogy strukturált, mégis rugalmas keretet biztosítson a szoftverfejlesztés számára, különösen nagy és összetett rendszerek esetében. A RUP egyik kulcsfontosságú sajátossága az iteratív és inkrementális fejlesztés, amely lehetővé teszi a folyamatos visszacsatolást és a kockázatok kezelését a projekt során. A módszertan hangsúlyozza a működőképes prototípusok létrehozását, valamint a fejlesztést támogató számítógépes eszközök, például integrált fejlesztői környezetek (IDE) és modellező eszközök (pl. UML) használatát. A RUP négy egymást követő fázisra bontja a fejlesztési folyamatot:*

- *Előkészítés (Inception) – a projekt céljainak, üzleti igényeinek és alapvető követelményeinek meghatározása.*
- *Kidolgozás (Elaboration) – az architektúra, a rendszer alapvető struktúrájának megtervezése és az első prototípusok elkészítése.*
- *Megvalósítás (Construction) – a szoftver fejlesztése, tesztelése és finomhangolása.*
- *Átadás (Transition) – a rendszer telepítése, éles üzembe helyezése és a végfelhasználók oktatása.*

A RUP három dimenzió mentén írja le a fejlesztési folyamatot:

- *Dinamikus perspektíva – a fejlesztési fázisok és iterációk bemutatása.*
- *Statikus perspektíva – a végrehajtandó folyamatok és azok összetevőinek meghatározása.*
- *Gyakorlati perspektíva – a fejlesztési folyamat során alkalmazott legjobb gyakorlatok, például az iteratív fejlesztés, a folyamatos visszacsatolás és a kockázatkezelés.*

A RUP tehát egy jól strukturált és skálázható módszertan, amely a nagy, összetett fejlesztési projektek számára biztosít átlátható keretet, miközben iteratív jellegének köszönhetően lehetőséget ad a rugalmas alkalmazkodásra és a kockázatok folyamatos kezelésére. (Komló, 2013)

*Az **evolúciós rendszerfejlesztési modell** lényege, hogy a fejlesztés nem egy előre meghatározott, teljes rendszer kialakításával kezdődik, hanem egy kezdeti verzióval, amelyet a felhasználói visszajelzések alapján fokozatosan továbbfejlesztenek. A fejlesztők egy alapinformációs rendszert hoznak létre, amelyet a felhasználók véleményeznek, majd az így kapott visszacsatolások alapján több iteráción keresztül finomítanak, amíg el nem érik a végleges, minden igényt kielégítő verziót. Az evolúciós fejlesztésnek két fő típusa ismert:*

Feltáró fejlesztés:

- *az első lépésben a fejlesztők egy alapkövetelményeknek megfelelő, működőképes rendszert hoznak létre, amelyet a felhasználók már használhatnak;*
- *a további fejlesztések a valós használat során szerzett tapasztalatokra és felhasználói visszajelzésekre épülnek;*
- *a rendszer iteratív módon fejlődik, a felhasználók új követelményeket fogalmaznak meg, amelyeket a fejlesztők beépítenek, így a végleges rendszer közösen formálódik.*

Eldobható prototípus készítése:

- *ebben az esetben a cél a felhasználói igények és elvárások pontosabb megértése;*
- *azok a követelmények, amelyek nem egyértelműek, egy gyorsan elkészített prototípusban kerülnek megvalósításra;*
- *a felhasználók visszajelzései alapján a prototípus folyamatosan módosul, amíg világossá nem válnak a végleges rendszer funkcionalitásai;*
- *a végleges rendszer nem a prototípus továbbfejlesztésével jön létre, hanem annak tapasztalatai alapján új fejlesztés indul.*

Előnyei közé tartozik a felhasználóközpontú fejlesztés, amely biztosítja, hogy a rendszer az igényekhez igazodjon és az, hogy rugalmas és iteratív folyamat, amely gyorsan képes reagálni a változó követelményekre. A működő verziók folyamatosan használhatók, így a felhasználók korán kipróbálhatják a rendszert.

Hátránya, hogy a fejlesztési folyamat nehezen átlátható, mivel a verziók száma gyorsan növekedhet. A gyakori módosítások miatt nehéz pontos és teljes dokumentációt készíteni, ami megnehezítheti az üzemeltetést és a karbantartást. Magas erőforrásigény is hátránya, mivel a folyamatos visszacsatolás és finomhangolás sok fejlesztői kapacitást igényelhet. (Komló, 2013)

8 Integrált vállalatirányítási rendszer

Az üzleti tranzakciók támogatására szolgálnak az ún. **tranzakciós alkalmazások**. Ezek az alkalmazások különféle vállalati irányítási feladatokat **integrálnak egyetlen alkalmazásba** (vagy egy szorosan integrált alkalmazáskészletbe), amelyek **közös adatbázison osztoznak**. Ezáltal minimalizálják az adatok inkonzisztenciáját és hibáit, valamint megszüntetik az adatok kezelésében rejlő hatékonysági hiányosságokat. Az adatok például csak egyszer kerülnek rögzítésre, de többször is felhasználhatók. **A vállalati információs rendszer egyik kulcsfontosságú tranzakciós alkalmazása az ERP rendszer** (*Enterprise Resource Planning* – vállalati erőforrás-tervezés, ill. **integrált vállalatirányitási rendszer**). Az ERP egy olyan alkalmazás vagy alkalmazási szoftver, amely lehetővé teszi az összes rendelkezésre álló üzleti erőforrás és tevékenység kezelését és koordinálását. Bár az ERP kifejezés széles körben elterjedt a gyakorlatban, az ERP rendszerek funkcionalitása jelentősen túlmutat azon, amit a név közvetlenül sugall. Napjainkban ezek az alkalmazások nemcsak az erőforrások tervezésére összpontosítanak, hanem a vállalat erőforrásainak teljes életciklusát képesek kezelni.

8.1 ERP rendszerek

Az ERP információs rendszer **adatforrásként szolgál** más típusú alkalmazások számára is. Az ERP által kezelt adatok továbbításra kerülnek a vállalat egyéb alkalmazásaiba, például az ügyfélkapcsolat-kezelésbe (CRM), az ellátási lánc menedzsmentbe (SCM), az e-kereskedelmi rendszerekbe, stb. Emellett az ERP-adatok gyakran az üzleti intelligencia (BI) rendszerek elsődleges adatforrásai, amelyeken keresztül különféle üzleti, marketing-, HR-, kapacitás- és egyéb elemzések készülnek. Ebből következik, hogy az ERP rendszer adatminősége és az adatbázisok megbízhatósága kulcsfontosságú. Az ERP rendszer hibái vagy gyenge adatminősége rontja azoknak az alkalmazásoknak a hatékonyságát is, amelyek az információs rendszer és a vállalati irányítás szintjének növelésére szolgálnak.

Az ERP megoldások általában a belső alapján, azaz a **szoftverarchitektúra** szempontjából kerülnek definiálásra. Ez dokumentálja, hogy az ERP rendszert milyen **programmodulok** (pl. pénzügy, gyártás, készletkezelés, HR modulok, stb.) **és eszközök alkotják**, valamint ezek hogyan működnek együtt. Mivel a különböző vállalatok eltérő informatikai támogatási igényekkel és prioritásokkal rendelkeznek, az ERP moduláris szerkezete lehetővé teszi a modulok fokozatos bevezetését a vállalat specifikus szükségleteinek és prioritásainak megfelelő sorrendben. Napjaink ERP architektúrája általában nemcsak a klasszikus alkalmazásmodulokat foglalja magában (például pénzügy, értékesítés, gyártás, stb.), hanem számos egyéb, működési és támogató funkciót is biztosít. Példák az ERP rendszerek modern megközelítésére:

- **QASIDA ERP rendszer** (https://asseco-qasida.com/).
    - A QASIDA ERP rendszer az ügyfél szerverein (On-Premise) telepítve működik, és modulokat tartalmaz, például: gyártás, raktárkezelés, pénzügy, kontrolling és egyéb területek számára.
    - Az QASIDA ALKALMAZÁSOK a felhőben működnek, és API-k segítségével teljes mértékben integrálva vannak az ERP rendszerrel. Ezek pl. a Service, Mobile, AI platform, Rental, iCep vagy OCR.
- **Odoo** (https://www.odoo.com/sk_SK). Az Odoo több mint 50 modult kínál, amelyek a vállalat különböző igényeinek megfelelően testre szabhatók.

8.1. táblázat: Vastag kliens és vékony kliens összehasonlítása

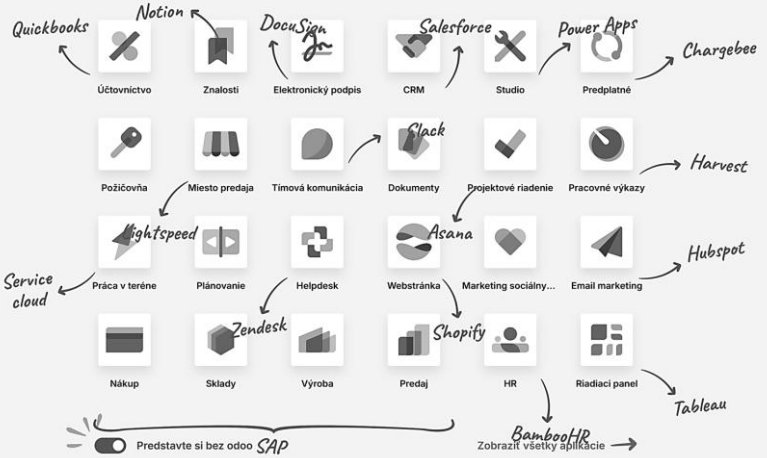
	Vastag kliens	**Vékony kliens**
Feldolgozás helye	Helyi gép	Felhő, szerver, böngészőn keresztüli elérhetőség
Erőforrásigény	Nagyobb helyi számítási kapacitás szükséges	Alacsonyabb, kevésbé erőforrás igényes
Helyi működés lehetősége	Offline módban is működhet	Általában nem működik offline
Telepítés	Az alkalmazásokat a helyi gépre kell telepíteni	Nem igényel telepítést, böngésző alapú

Forrás: Saját szerkesztés

A **vastag kliens** egy olyan alkalmazás, amelyet a helyi számítógépre telepítenek, és az adatfeldolgozás nagy részét helyben végzi, például egy helyi ERP rendszer akkor is működhet, ha nincs internetkapcsolat. A **vékony kliens** egy olyan alkalmazás, amely minimális helyi erőforrást igényel, és a feldolgozást, valamint az adatokat **távoli szerverek végzik**

(8.1. táblázat). Példa erre egy böngészőből elérhető felhőalapú alkalmazás. A QASIDA ERP rendszer egy vastag kliens alkalmazás, amelyet a vállalat szerverein (On Premise) telepítenek. A rendszer fő funkciói és adatfeldolgozása helyben, a vállalat saját infrastruktúráján történik. A QASIDA ALKALMAZÁSOK vékony kliensek és a felhőben futnak.

Mikor érdemes ERP rendszert bevezetni?
Feltehető a kérdés, hogy mikor van szüksége egy vállalatnak ERP rendszerre. A válasz egyszerű: akkor, amikor az ERP használata hasznos a vállalat számára, és jelentősen segíti a tevékenységeket.
Ahogy egy vállalat növekszik, egyre több és több informatikai eszközre van szüksége, beleértve a különböző szoftvereket is. Például: saját népszerűsítésük érdekében sokan készítenek weboldalt WordPress-ben; az ügyfelekről szóló adatokat Excel-ben tárolják; e-mail kampányokhoz MailChimp-et használnak; az ügyfélkapcsolatokat CRM rendszerben, például HubSpot-ban kezelik; a számlázáshoz SuperFaktúra szoftvert választanak; az interneten elérhető adatok tárolásához Google Drive-ot alkalmaznak; a gyors kommunikációhoz Slack-et, a projektek támogatásához Asana-t, a munkaidő nyilvántartásához Toggl-t, a szolgálati utak kezeléséhez pedig Fleetio-t. Ezek a megoldások számos előnnyel bírnak: általában felhasználóbarátok, jól működnek mobiltelefonon és tableten, és az alapfunkciók gyakran ingyenesen elérhetők. Ugyanakkor, ezek egy idő után túlterheltté válhatnak, főleg, ha a különféle szoftverek között manuálisan kell adatokat exportálni vagy importálni. Az egyedi rendszerek nem mindig kapcsolhatók össze (még API-k segítségével sem), és nincs egységes adatbázis.



Odoo ERP néhány modulja, amely helyettesíthet számos szoftvert
Forrás: (https://www.odoo.com/sk SK)

Az ERP rendszerek ezekre a problémákra kínálnak megoldást: minden egy helyen található, egységes adatbázist használ, a kezelőfelületek konzisztens megjelenésűek, és minden modul integráltan működik. Ez teljesen új perspektívát ad az adatok kezelésére. Például észrevehetővé válnak az ismétlődő tevékenységek, vagy optimalizálhatóvá válnak a nem hatékony folyamatok. Az API-k integrációja egyes modern ERP rendszerekben már alapvető, és nem mindig szükséges manuális beavatkozás. Az ERP rendszerek modularitása lehetővé teszi, hogy a bevezetés fokozatos legyen: kezdetben csak néhány modullal indul, majd a vállalat igényei szerint továbbiakkal bővíthető.

Fontos ugyanakkor kiemelni, hogy az ERP rendszer vállalatok számára drága lehet (Nemcsak a kezdeti bevezetés, hanem az éves licencdíjak is komoly kiadást jelenthetnek. Ez különösen felhőalapú ERP rendszerek esetén releváns. Ne felejtkezzünk el a Microsoft és más licencekről se.), és nem megfelelő implementálás esetén nem biztos, hogy minden folyamatot úgy támogat, ahogyan azt elvárnák. Ezért kiemelten fontos, hogy már a bevezetést megelőzően alaposan kommunikáljunk a szállítóval. Két alapvető kérdésre kell választ kapni a tervezés korai fázisaiban. Ezek az alábbiak.

- *Milyen infrastruktúrával rendelkezünk? – Fel kell mérni a vállalat jelenlegi helyzetét, beleértve az adatokat, a hálózatot és az adatbázisokat.*
- *Milyen további funkciókra van szükség? – Például: automatikus számlázás banki fizetéssel és termék kiszállításával.*

A szállító ezek alapján tud javaslatokat tenni, meghatározhatja, hogy a termék mit képes megvalósítani, és mire van szükség további fejlesztésekhez. A rendszert az adott vállalat igényeihez igazítják, és szükség esetén további programozási órákat biztosítanak. Az ERP személyre szabásához néha fejlesztői csapat szükséges, ami szintén idő- és költségigényes lehet.
Forrás: (Vychodil, T. 2024)

A szakirodalomban létezik az ERP és az ERP II fogalma. Az ERP rendszerek célja a vállalati erőforrások integrált kezelése. Az ERP II egy kiterjesztett fogalom, amely az ERP rendszerek újabb generációját írja le. Külső kapcsolatokat is kezelnek, webalapú technológiákra támaszkodnak, támogatják a valós idejű adatintegrációt, az e-business-t és a kollaborációs folyamatokat. Tartalmazhatnak például CRM és Supply Chain Management (SCM) modulokat. A legtöbb modern ERP rendszer már tartalmazza az ERP II funkcióit, tehát **a fogalmak közötti különbség elmosódott**. Az ERP II fogalom leginkább a 2000-es évek elején volt népszerű, amikor **az internet elterjedésével** az ERP rendszereket elkezdték

kibővíteni külső integrációs lehetőségekkel. **Ma már az ERP II kifejezést ritkábban használják, mivel az integrált és webalapú ERP megoldások váltak szabvánnyá**. A kisebb és közepes vállalatoknál előfordulhat, hogy az ERP rendszer az egyetlen információs rendszer, főleg ott, ahol egyszerűbbek az üzleti folyamatok. Az ERP rendszer bevezetése után nincs szükség további speciális szoftverekre, amikor az ERP moduljai elégségesek az összes szükséges funkció lefedéséhez. Nagyobb vállalatoknál azonban az ERP rendszer nem az egyetlen használt rendszer, hanem az egyik központi elem, amelyet más rendszerekkel integrálnak – lásd a II. kötet: *Üzleti intelligencia, OLAP rendszerek*. (Gála, L; Pour, J; Šedivá, Z. 2015)

8.1.1 ERP rendszerek testre szabása

Az ERP rendszerek, amelyeket általános alkalmazásként szállítanak, általában testre szabást (customization) igényelnek a konkrét vállalat vagy felhasználó igényeinek megfelelően. A testreszabás jellemzően a felhasználói igények elemzése alapján történik, és az ERP rendszer bevezetésének és implementációjának egyik legfontosabb lépése. A testreszabás az alábbiakat foglalhatja magában (Gála, L; Pour, J; Šedivá, Z. 2015).

- A funkciók és kommunikációs struktúrák módosítása. Ide tartozik pl. a menüstruktúrák módosítása, egyes opciók eltávolítása vagy új funkciók hozzáadása.
- Információs struktúrák módosítása. Ide tartozik pl. az űrlapok, riportok, jelentések és áttekintések testre szabása.
- Alapértelmezett értékek beállítása. Ide tartozik pl. a nyelv (magyar), a pénznem (HUF) alapértelmezettként való megadása.
- Adatkategóriák és adatok terjedelmének beállítása. Ide tartozik pl. a szervezeti struktúra meghatározása, számlatükör beállítása, költséghelyek struktúrájának definiálása.
- Szabványos számítások és ellenőrzések módosítása. Például árkalkulációk, hitelkeret-ellenőrzések testre szabása.
- Technológiai beállítások módosítása. Szín- és keretstílusok, valamint más vizuális elemek szabványos beállítása.

A testre szabást két szinten végezhetjük.

- **Felhasználói szintű testre szabás.**
    - Ezeket a módosításokat a felhasználó saját maga is elvégezheti az ERP rendszer beállításain keresztül.

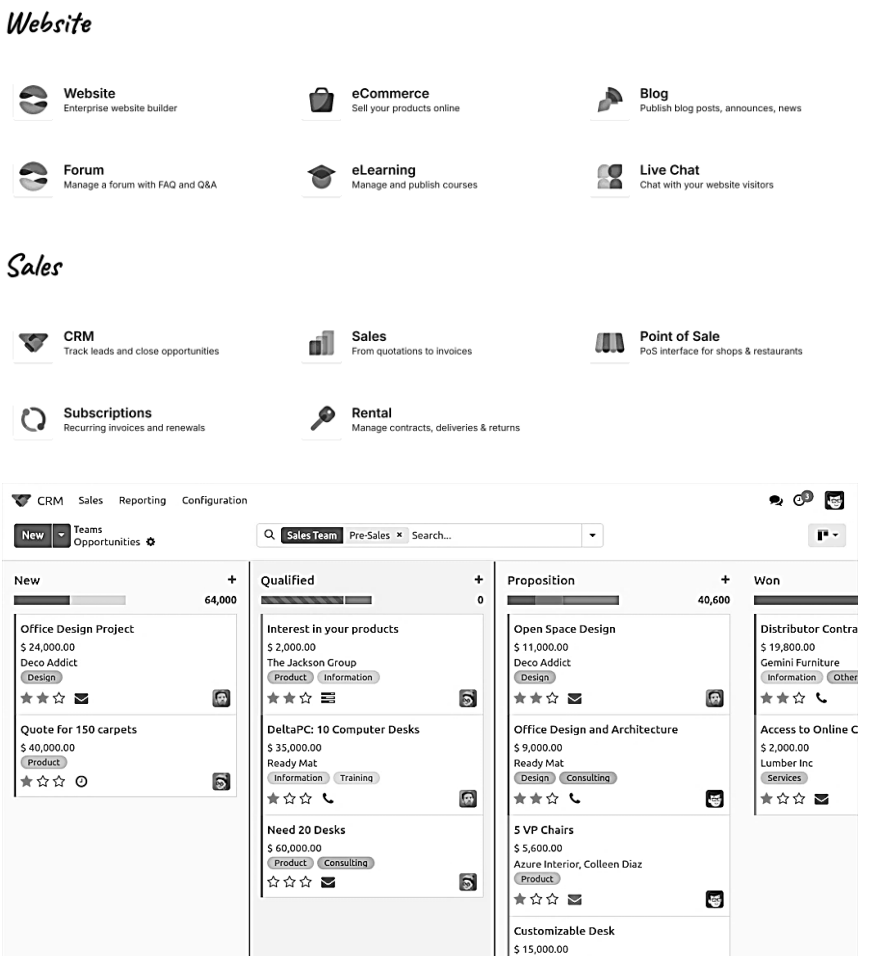
Például, az alapértelmezett nyelv vagy pénznem megadása, valamint egyszerű riportok és képernyők személyre szabása.

- **Szállítói szintű testre szabás.**
  - Az összetettebb módosításokat, például az adatbázis struktúrák vagy speciális számítások és funkciók módosítását általában az ERP rendszer szállítójának kell elvégeznie az implementáció során.

Egy modern ERP rendszer az Odoo, melyik alapvetően a következő modulokat tartalmazza (https://www.odoo.com):

- **Odoo Website**:
  - Website
  - eCommerce
  - Blog
  - Forum
  - Learning
  - Live Chat
- **Odoo Sales**:
  - CRM
  - Sales
  - Point of Sale
  - Subscriptions
  - Rental
- **Odoo Finance**:
  - Accounting
  - Invoicing
  - Expenses
  - Documents
  - Spreadsheets
  - Sign
- **Odoo Inventory & Manufacturing**:
  - Inventory
  - Manufacturing
  - PLM
  - Purchase
  - Maintenance
  - Quality
- **Human Resources**:
  - Employees
  - Recruitment
  - Time Off
  - Appraisals
  - Referral
  - Fleet
- **Odoo Marketing**:
  - Marketing Automation
  - Email marketing
  - SMS Marketing
  - Social Marketing
  - Events
  - Survey
- **Odoo Services**:
  - Project
  - Timesheet
  - Field Service
  - Helpdesk
  - Planning
  - Appointments
- **Odoo Productivity**:
  - Discuss
  - Approvals
  - IoT: Internet of Things
  - VOIP
  - Knowledge
- **Odoo Customization**:
  - Studio

Az ERP rendszerek esetében gyakran találkozunk a modul és alkalmazás kifejezéssel. Az alkalmazások és modulok közötti különbség gyakran **inkább terminológiai, mint funkcionális**. Feljebb bemutatott alkalmazások gyakorlatilag moduloknak tekinthetők. Az "alkalmazás" kifejezés inkább a modern, felhasználóbarát megközelítést tükrözi. A "modul" kifejezés technikai értelemben az ERP rendszer szerves részét jelenti.



8.1. ábra: Példa az Odoo rendszer moduljaira (fent). CRM modul ablaka (lent). Forrás: https://www.odoo.com/page/all-apps

Testreszabhatóságnak lényeges lehetősége a modulok kiválasztása. Ez azt jelenti, hogy a felhasználó szabadon választhat a különféle modulok gyűjteményéből, amelyek együttesen egy teljes, robusztus vállalati

információs rendszert alkotnak. Ez a moduláris felépítés lehetővé teszi, hogy a vállalatok az információs rendszert pontosan a saját igényeikre szabják, figyelembe véve a működésük során legfontosabb folyamatokat és prioritásokat. Az egyes modulok kiválasztásával és testre szabásával a vállalat biztosítani tudja, hogy csak azokat a funkciókat valósítsa meg, amelyek valóban relevánsak a számára, ezzel minimalizálva a felesleges bonyolultságot és költségeket. A rugalmasság, amelyet ez a megközelítés biztosít, nemcsak a rendszer implementálásakor, hanem annak bővítésekor vagy módosításakor is jelentős előny. Például egy növekvő vállalat először csak a pénzügyi modulokat használhatja, majd később, a működése komplexitásának növekedésével, integrálhatja a készletgazdálkodási, projektmenedzsment vagy emberi erőforrás modulokat. Ez nemcsak költséghatékony, hanem stratégiailag is előnyös, mivel a vállalat lépésről lépésre építheti ki információs rendszerét, a fejlődése ütemében. Ezen túlmenően a moduláris felépítés nagy előnye, hogy a modern rendszerek API-kon keresztül más szoftverekkel is könnyen integrálhatók, így a vállalat meglévő infrastruktúrájához zökkenőmentesen csatlakoztathatók. (Gála, L; Pour, J; Šedivá, Z. 2015) (https://www.odoo.com)

8.1.2 Analitikai alkalmazások és üzleti intelligencia integrációja

Az üzleti intelligencia (BI – *Business Intelligence*) eszközeinek integrálása az ERP rendszerekbe egyre fontosabb szerepet játszik az ERP funkcionalitásának bővítésében. Ez lehetőséget ad arra, hogy az ERP adatbázisaiban tárolt adatokon közvetlenül analitikai eredményeket hozzunk létre.

Az ERP rendszerek képesek analitikákat végezni, azonban ezek nem érik el az adattárház-alapú elemzések szintjét. Az ERP analitikák jellemzője, hogy közvetlenül az ERP adatbázisán alapulnak, amelyben az adatok szigorúan konzisztensek, pontosak, mert ezek közvetlen hatással vannak a vállalat napi működésére. Például, ha egy mezőbe dátumot kell menteni, oda csak dátum formátumú adat kerülhet, vagy nem tárolódnak hiányos adatok (például, ha a cím mezőből hiányzik a házszám, az adatot nem rögzíti a rendszer). Ezzel szemben az adattárházak (*Data Warehouses*) elemzései kevésbé érzékenyek a hiányos adatokra. Például, ha 50 000 tranzakció közül egy helyen hiányzik a házszám, az nem jelent lényeges problémát az üzleti elemzés szempontjából, de az ERP adatbázisában viszont a házszám hiánya komoly működési problémát okozhat, például a raktár nem tudja, hová szállítsa a csomagot. (Gála, L; Pour, J; Šedivá, Z. 2015)

8.1.3 ERP rendszerek kategorizálása

Az ERP rendszerek különböző szempontok alapján kategorizálhatók. Az alábbiakban két leggyakrabban használt megközelítést mutatunk be.

- A vállalat mérete alapján ismerünk:
  - Nagyvállalatokat: több, mint 500 alkalmazottal rendelkező cégek. **Nagyvállalati ERP** pl. a SAP S/4HANA, Microsoft Dynamics 365 vagy Oracle ERP.
  - Középvállalatokat: 50 és 500 közötti alkalmazottal rendelkező cégek. **Középvállalati ERP** rendszerek pl. Microsoft Dynamics 365 vagy Odoo.
  - Kisvállalatokat: legfeljebb 50 alkalmazottal rendelkező cégek. **Kisvállalati ERP** pl. a Zoho ERP vagy Odoo.
- Az ERP rendszer funkcionalitása és integrációs szintje alapján az ERP alkalmazások kategorizálhatók a szerint is, hogy mennyire fedik le a vállalati irányítás kulcsfontosságú területeit, illetve milyen mértékben támogatják a vállalati folyamatok integrációját. E megközelítés alapján az alábbi csoportokat különböztetjük meg:
  - **All-in-One ERP rendszerek**: nagyszabású alkalmazások, amelyek széleskörű funkcionalitást kínálnak, és komplexen lefedik a vállalati irányítás egészét. Előnyei a teljes körű funkcionalitás és magas szintű folyamatintegráció. Hátrányuk a komplex megoldás, amely magasabb költségekkel és nagyobb testreszabási igényekkel jár.
  - **Best-of-Breed ERP rendszerek**: olyan megoldások, amelyek egy-egy specifikus vállalati területre vagy iparágra specializálódnak (pl. autóipar, vegyipar, kiskereskedelem). Előnyeik a magas szintű funkcionalitás a kiválasztott területen. Hátrányuk, hogy nem fedik le a vállalati irányítás teljes spektrumát, ezért gyakran kiegészítő megoldásokra és integrációs projektekre van szükség.
  - **Lite ERP rendszerek**: könnyített verziók, amelyeket főként kis- és középvállalatok számára terveztek. Előnyeik az alacsonyabb ár, az egyszerűbb és gyorsabb implementáció. Hátrányaik a korlátozott funkcionalitás és a rendszer bővítési lehetőségeinek szűkesítése.

Ez a két szempont a leggyakrabban alkalmazott az ERP rendszereknél.

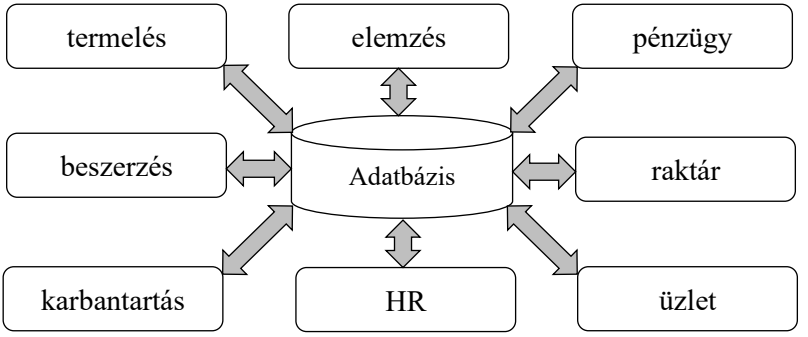
Az ERP rendszerek kategorizálásánál még megemlíthetjük a technológiai architektúra szerinti szempontot is. A már említett **moduláris ERP**-k mellett ismerünk **monolitikus ERP**-ket is, melyek egyetlen egységes rendszerként működnek, továbbá **nyílt forráskódú ERP**-ket, melyeknek forráskódja nyilvános, így az szabadon módosítható és testre szabható. (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P., 2017)

8.1.4 ERP rendszerek jellemzői

Az ERP rendszereket az alábbi jellemzők definiálják.

- **Moduláris és integrált működés**.
  - ERP rendszerek a teljes vállalati működést lefedik.
- **Hagyományos, felhőalapú és hibrid megoldások**.
  - Léteznek vékony vagy vastag kliens megoldásaik.
- **Mobilitás és távoli hozzáférés**.
  - Felhőalapú, mobilbarát megoldások, amelyek könnyen elérhetők bárhonnan. A távmunka és globális csapatok együttműködése szempontjából vált alapvető követelménnyé.
- **Redundanciák kiküszöbölése**.
  - ERP rendszerek egységes adatbázisa csökkenti az adatok depilációját és az ezzel járó hibák lehetőségét.
- **Mesterséges intelligencia és gépi tanulás integráció**.
  - Intelligens funkciókkal rendelkeznek, melyek nemcsak az adatfeldolgozást javítják, hanem például prediktív analízist, automatizálást és személyre szabott ajánlásokat is lehetővé tesznek.
- **Egységes konzisztens vállalati/szervezeti adatbázis**.
  - ERP rendszerek egyik kulcsfontosságú előnye.
- **Törekvés a szolgáltatások teljességére**.
  - Az ERP rendszerek célja, hogy a vállalat összes főbb funkcióját lefedjék.
- **Grafikus felhasználói felület**.
  - Egyszerűen kezelhető és testre szabható felhasználói felületek támogatják a hatékonyságot.
- **Valós idejű adatok és analitika**.
  - Beépített analitikai modulok segítik a riportok készítését.
- **Széles körű információ hozzáférési lehetőség**.
  - A hozzáférési szintek szerepkörökhöz és jogosultságokhoz igazíthatók.

- **Nyitott integrációs képességek**.
  - API-k támogatásával érhetők el.
- **Automatizálás és IoT eszközökkel való összekapcsolhatóság**.
  - ERP rendszerek támogatják az IoT eszközökkel való kommunikációt.
- **Egyértelmű felelősségek a rendszerben lévő adatokért**.
  - Az ERP rendszerek támogatják a szerepköralapú hozzáférés-kezelést, ami biztosítja az adatok védelmét és a felelősség egyértelmű elosztását.
- **Skálázhatóság**.
  - A vállalati növekedés támogatása.
- **Többnyelvű és multikulturális támogatás**.
  - Nemzetközi vállalatok esetében.
- **Testreszabhatóság**.
- **Valós idejű együttműködés**.
  - Hatékonyságnövelés.
- **Adatbiztonság és titkosítás**.
  - ERP rendszerek fejlett biztonsági funkciókkal rendelkeznek, beleértve az adatátvitel titkosítását.
- **Prediktív karbantartás**.
  - A gyártásban és az eszközgazdálkodásban az IoT és AI technológiák integrációjával támogatja a berendezések előre jelzett karbantartását, minimalizálva a leállási időt.
- **Költségcsökkentés**.
  - ERP rendszerek integrált működése révén csökkenthetők a különálló rendszerek fenntartásával járó költségek.



8.2. ábra: ERP rendszer modell
Forrás: (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P., 2017)

Az ERP-rendszerek hatása napjaink üzleti világára egyszerűen nem hagyható figyelmen kívül. Mivel a vállalati adatok és folyamatok egyetlen rendszerben kerülnek összegyűjtésre, a vállalkozások képesek összehangolni a különböző részlegek működését, valamint optimalizálni a munkafolyamatokat, ami jelentős költségcsökkentéssel járhat. Az Oracle vállalat (2024) a következő előnyöket emeli ki:

- tisztább üzleti átláthatóság, amelyet a kimutatások és valós idejű információk biztosítanak;
- csökkentett működési költségek a hatékonyabb üzleti folyamatok és bevált módszerek alkalmazása révén;
- fokozott együttműködés a felhasználók által megosztott adatokon keresztül, például szerződéseknél, igényléseknél és rendeléseknél;
- nagyobb hatékonyság egy közös, átfogó felhasználói élmény segítségével, amely számos üzleti funkciót lefed;
- egységes infrastruktúra, amely a back office-tól a front office-ig minden üzleti tevékenységet konzisztensen kezel;
- magasabb felhasználói elégedettség a felhasználói élmény és az intuitív kialakítás miatt;
- kisebb kockázat, köszönhetően a jobb adatintegritásnak és pénzügyi ellenőrzéseknek;
- kevesebb adminisztratív és működési költség, az integrált rendszereknek köszönhetően. (Gála, L.; Pour, J.; Šedivá, Z. 2015) (Gála, L.; Pour, J.; Šedivá, Z. 2009) (Juhász, S. 2011) (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P. 2017) (Sodomka, P.; Klčová, H. 2010) (Sziray, J.; Gaul, G.; Erdős, F. 2006) (Sziray, J.; Gaul, G.; Égertné Molnár, É. 2007)

8.2 SCM rendszerek

A végtermék előállításában és annak végső vásárlóhoz történő eljuttatásában általában több vállalat is részt vesz. Ezek alkotják az úgynevezett ellátási láncot (S*upply Chain*). A klasszikus, lineáris ellátási lánc a következő elemekből áll: beszállító → gyártó → disztribútor → kiskereskedő → vásárló. Napjainkban azonban az ellátási láncok inkább hálózati szerveződésűek, amelyek összetettebbek és dinamikusabbak. Az ellátási lánc folyamatai az alábbi három fő irányú áramlást foglalják magukban: információáramlás, pénzügyi áramlás, anyag- és termékáramlás. Ezek az áramlások kétirányúak, azaz mindkét irányban zajlik a kommunikáció és az adatok cseréje. Az **ellátási lánc menedzsment**

(SCM – *Supply Chain Management*) az ellátási lánc irányítására és működésének optimalizálására szolgáló eszközök és folyamatok összessége. Célja az, hogy az ellátási lánc minden elemét a lehető legnagyobb hatékonysággal működtesse, figyelembe véve a végső vásárlói igényeket. Az ellátási lánc folyamatait gyakran a **SCOR modell** (*Supply Chain Operation Reference Model*) alapján határozzák meg, amely az alábbi öt szinten elemzi a folyamatokat.

- **Első szint**: az ellátási lánc öt kulcsfolyamatát definiálja, a tervezést, beszerzést, gyártást, szállítást és reklamációkezelést.
- **Második szint**: az első szint folyamatait stratégiai szempontból alfolyamatokra bontja. Például a „megrendelésre történő gyártás" stratégiájához tartozó alfolyamat lehet a „termék összeszerelése".
- **Harmadik szint**: az alfolyamatok részletes tevékenységeket tartalmaznak. Például a „termék összeszerelése" alfolyamat egyik tevékenysége a darabjegyzék (BOM) összeállítása.
- **Negyedik szint**: a tevékenységekhez tartozó munkafolyamatokat, azok bemeneteit, kimeneteit és konkrét műveleteit határozza meg.
- **Ötödik szint**: a munkafolyamatok automatizálási lehetőségeit specifikálja, amelyhez információs technológiákat használ.

Az ellátási lánc egyes elemei közötti információáramlás általában elektronikus üzenetváltás útján történik. Napjainkban leggyakrabban az interneten keresztül. Az üzenetek struktúrája és tartalma a **strukturált adatok cseréjére vonatkozó szabványokon** alapul. Az **EDI** (*Electronic Data Interchange*) olyan szabványosított módszer, amely lehetővé teszi strukturált adatok (például megrendelések, számlák, jóváírások) cseréjét a kereskedelmi partnerek információs rendszerei között, elektronikus úton. Az egyik legelterjedtebb szabvány az **UN/EDIFACT** (*United Nations / Electronic Data Interchange for Administration, Commerce, and Transport*). Ez a szabvány globálisan elfogadott, és magán- valamint közszektorban egyaránt használatos.

Az ügyfélkövetelmények hatékony kiszolgálásának érdekében az SCM alkalmazások mellett gyakran használnak haladó tervezési és ütemezési (APS – *Advanced Planning and Scheduling*) megoldásokat is. Az APS-t az összes erőforrás egyidejű, szinkronizált tervezése jellemzi, az összes ismert korlátozás figyelembevételével, és alapvetően az MRP II (*Manufacturing Resource Planning*) módszert váltja fel a vállalati szintű tranzakciós ERP-alkalmazásokban.

A piacon különféle SCM megoldások érhetők el, melyeket összegezve két kategóriába sorolhatjuk:

- ERP modulokként bővített megoldások, mint pl. SAP SCM, vagy
- az önálló SCM alkalmazások, mint pl. Oracle Value Chain Planning, JDA Software, stb. (Gála, L; Pour, J; Šedivá, Z. 2015)

8.2.1 Az SCM gyakorlati alkalmazása

Az SCM információs rendszerek szinte minden iparágban megtalálják helyüket. Napjainkban a vásárlók egyre nagyobb nyomást gyakorolnak a vállalatokra, a termékek minél gyorsabb szállítása érdekében, miközben a gyártók törekednek a minimális raktárkészletek fenntartására és az időben történő szállítás (just-in-time) elvének követésére. Ezeknek az igényeknek az összehangolása információs technológiák nélkül úgy, hogy mindkét fél elégedett legyen, gyakorlatilag lehetetlen lenne. Éppen ezért az SCM rendszerek ma már elengedhetetlen részei minden nagyobb gyártási és disztribúciós megoldásnak.

Az SCM rendszereket ma intenzíven használják különböző logisztikai vállalatok, például a UPS, a FedEx vagy a Szlovák Posta. Ezek a megoldások lehetővé teszik a logisztikai vállalatok számára, hogy hatékonyan irányítsák járműveik logisztikai útvonalait, valamint valós idejű online nyomon követést biztosítsanak a szállított áruk számára. Egy gyakorlati példa erre: a megrendelések ügyfelekhez történő kiszállítása futárszolgálatok segítségével. Ilyen esetekben vonalkódokat használnak, amelyek segítségével az árukat az elküldéstől kezdve a végső kézbesítésig megjelölik és szkennelik. Az SCM rendszer segítségével a feladó, a címzett és maga a szállítmányozó cég is valós időben követheti a csomagok mozgását. A GPS modulok és az SCM rendszerek összekapcsolásával nemcsak a járművek aktuális pozíciója figyelhető meg, hanem a szállítási útvonalak is optimalizálhatók, hogy költséghatékonyabb és gyorsabb szállítást érjenek el. (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P., 2017)

8.3 CRM rendszerek

Az **ügyfélkapcsolat-kezelés** (CRM – *Customer Relationship Management*) egy komplex megoldás, amely információs technológiákból, technikai eszközökből, vállalati folyamatokból és emberi erőforrásokból áll. Célja az ügyfélkapcsolatok irányítása és javítása, különösen az értékesítés, marketing és ügyfélszolgálat területeken. A CRM alkalmazások szoftvermegoldásként egy célzott kombinációját jelentik a tranzakciós,

analitikai és infrastrukturális alkalmazásoknak. Ez a kombináció a CRM rendszerek három alapvető funkcionális részében tükröződik: operatív rész, kooperatív rész és analitikai rész. Vegyük őket sorra.

- Az **operatív CRM** rész a mindennapi ügyfélkezelési tevékenységeket támogatja, különösen az ügyfelek felkutatásában, megszerzésében és hosszú távú megtartásában. Funkciói az alábbi területekre összpontosulnak:
  - Értékesítési tevékenységek automatizálása (SFA – *Sales Force Automation*), mint pl. az értékesítési képviselők munkájának támogatása, kapcsolatkezelés, értékesítési lehetőségek nyomon követése, potenciális ügyfelek kezelése vagy megrendelési ciklusok átfogó irányítása.
  - Marketingtevékenységek automatizálása (EMA – *Enterprise Marketing Automation*), ahol az alkalmazások a klasszikus marketingfunkciók támogatására fókuszálnak, pl. marketingkampányok elemzése, tervezése és megvalósítása, vagy a vállalati termék innováció szükségességének felmérése.
  - Ügyfélszolgálat és szerviz (CSS – *Customer Service and Support*). Ez magában foglalja a részletes információk biztosítását a vállalat termékeiről, reklamációk kezelését és jótállási és jótálláson túli szervizfolyamatok irányítását.

  Az operatív CRM jellemzően **tranzakciós adatfeldolgozásra támaszkodik**, és kiterjeszti a vállalati szintű tranzakciós alkalmazások funkcionalitását. Ennek célja, hogy a mindennapi operatív tevékenységek gördülékenyebbek legyenek, miközben hatékonyabb ügyfélkapcsolat-kezelést biztosítanak.
- A **kooperatív CRM** célja a vállalat által kiépített összes kommunikációs csatorna irányított kezelése. Ezek a csatornák nemcsak a meglévő, lojális ügyfelek, hanem a potenciális ügyfelek kiszolgálására is irányulnak. A kooperatív CRM kiterjeszti a korábban használt ügyfélkapcsolati módokat (pl. postai levelezés, faxok, telefonos kapcsolatok, személyes találkozók) olyan modern csatornákra, mint az internet és a mobilkommunikáció. Ezeket a kommunikációs csatornákat általában kapcsolati központok (*Contact Center* vagy *Call Centerek*) koordinálják. Ezek a CRM alkalmazások és technológiák központi hozzáférést biztosítanak az ügyfelek számára a vállalathoz. A kooperatív CRM-ben a tranzakciós feldolgozás funkciói keverednek az infrastrukturális alkalmazások funkcióival, így biztosítva az ügyfélkapcsolatok hatékonyabb kezelését.
- Az **analitikai CRM** rész az ügyfelekről szóló tudás feldolgozására és elemzésére összpontosít. Fő funkciói közé tartoznak:

- ügyfél szegmentáció,
- marketingkampányok elemzése,
- ügyfélviselkedés előrejelzése (predikció),
- ügyfélérték (CV – *Customer Value*) számítása és elemzése.

Az analitikai CRM általában az operatív és kooperatív CRM-ből, valamint más rendszerekből (pl. ERP, e-business) származó ügyféladatokat használ. Az adatelemzésekhez üzleti intelligencia (BI) technológiákat és alkalmazásokat vesz igénybe. Az analitikai CRM és a BI technológiák kombinációját (CI – ***Customer Intelligence***) nevezzük. A CI célja az ügyfelek mélyebb megismerése, beleértve az ügyfél értékének meghatározását, preferenciák feltárását, rizikófaktorok elemzését (pl. a versenytárshoz való átállás valószínűsége).

A CI elsődlegesen az ügyfél-interakciókból származó adatok gyűjtésére és elemzésére összpontosít. Az alábbi forrásokból nyer adatokat:

- kereskedelmi kapcsolatok dokumentációja,
- kapcsolati központok feljegyzései,
- elektronikus kereskedelmi alkalmazások adatai,
- vállalati portálok használati adatai,
- ún. „*Clickstream*” elemzések.

A CI analitikai képességei lehetővé teszik a vállalati elemzők számára, hogy az ügyfél értékére összpontosítsanak, és ezen alapulva optimalizálják az ügyfélszolgálati minőséget az egyes szegmensek számára. Az ügyfélérték a múltbeli és jövőbeli haszon és költségek számszerűsítésén alapul.

A közösségi hálózatok, mint például a Facebook vagy a Twitter, új dimenziót hoztak az ügyfélkapcsolat-kezelésbe, miközben jelentős változásokat idéztek elő a vállalatok ügyfélkezelési megközelítésében. Az ügyfélkapcsolat-kezelésben a közösségi hálózatok alkalmazásával foglalkozó területet **social CRM**-nek (sCRM) nevezik. A social CRM egy olyan filozófia és vállalati stratégia, amelyet technológiai platformok, vállalati szabályok, folyamatok és közösségi jellemzők támogatnak. A social CRM alkalmazások funkciói a közösségek és azok adott kontextusában zajló beszélgetéseire fókuszálnak. A klasszikus CRM rendszerektől eltérően, itt a beszélgetéseket nem a vállalat kezdeményezi, hanem az ügyfél.

Jelenleg számos különböző CRM alkalmazás érhető el a piacon. Ezek lehetnek ERP-t kiegészítő modulok, amelyek az ERP rendszerek

funkcionalitását bővítik ki, vagy önálló CRM megoldások, amelyek kizárólag az ügyfélkapcsolatok kezelésére specializálódtak. Példák az ismert CRM alkalmazásokra: Salesforce, Microsoft Dynamics CRM, Oracle Siebel Customer Relationship Management. (Gála, L; Pour, J; Šedivá, Z. 2015) (Rosický, A. 2010)

8.3.1 A CRM gyakorlati alkalmazása

Bár a CRM rendszerek (ügyfélkapcsolat-kezelési rendszerek) használata elsőre kissé absztraktnak tűnhet, gyakorlati alkalmazásuk rendkívül széleskörű, és az emberek sokkal gyakrabban találkoznak velük, mint gondolnák – akár ügyfélként, akár egy partner (például beszállító) munkatársaként.

Egy gyakorlati példa erre, a különböző törzsvásárlói klubkártyák rendszere, amelyeket kereskedelmi láncok, üzlethálózatok vagy partnerek használnak. Már a kártyára való regisztrációval az anonim vásárlóból konkrét ügyfél válik. A rendszer a kártya használatával nyomon követi az ügyfél különféle vásárlási adatait, például a vásárlások mennyiségét, típusát, gyakoriságát, időpontjait, fizetési módját, stb. Ezek az adatok lehetővé teszik, hogy a vállalat személyre szabott, automatizált ajánlatokat készítsen, amelyeket e-mailben vagy hagyományos postai úton juttatnak el a vásárlókhoz. Emellett a kedvezményrendszer ösztönzi a vásárlókat arra, hogy rendszeresen visszatérjenek az adott üzlethálózathoz. Ilyen rendszereket használnak például a jól ismert kereskedelmi láncok, mint a TESCO, a SLOVNAFT, a SHELL, a DM drogéria, valamint számos étterem- és kávézólánc, illetve más vállalkozás. (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P., 2017)

8.3.2 CRM rendszerek kategorizálása

Kučera, M.; Fiľa, M.; Látečková, A. és Stuchlý, P. (2017) a CRM rendszereket az ügyfél által igényelt funkcionális megoldás alapján kategorizálják:

- **CRM All-in-One**: az ügyfél által igényelt funkcionális megoldás tekintetében.
- **CRM Best-of-Breed**: kizárólag az ügyfélkapcsolat-kezelés egy adott területére tervezték, nagy pontosság és részletes funkcionalitás jellemzi egy adott területen.

- **Lite CRM**: a szabványos CRM-megoldások egyszerű változata, amelyet nagyon korlátozott funkcionalitás jellemez. (Gála, L; Pour, J; Šedivá, Z. 2015)

8.4 Egyéb tranzakciós rendszerek

Tranzakciós rendszerek közül megemlíthetjük még az alábbiakat:

- **TMS** (*Transportation Management System*)
  - A TMS rendszerek tranzakcióként kezelik a szállítmányozási eseményeket. Ezek gyakran kapcsolódnak az SCM és ERP rendszerekhez.
- **HCM** (*Human Capital Management*)
  - A HCM rendszerek tranzakciókat kezelnek az alkalmazottak adataival kapcsolatban, például a munkabérek feldolgozása vagy munkaidő-nyilvántartása. Ezek a tranzakciók gyakran integrálódnak az ERP rendszerek HR moduljához.
- **SRM** (*Supplier Relationship Management*)
  - Az SRM rendszerek tranzakcióként kezelik a beszállítókkal kapcsolatos eseményeket, például a beszerzési rendelések leadása és követése, a szerződések létrehozása vagy a szállítói teljesítmény értékelése. Ezek a tranzakciók szorosan kapcsolódnak az SCM és ERP rendszerek adataihoz, és ERP rendszerek moduljaként is szerepelhetnek.
- **PLM** (*Product Lifecycle Management*)
  - A PLM rendszerek fő fókusza a termékek életciklusának menedzselése és a tervezési adatok kezelése, ami általában nem tranzakciós jellegű. Kivételek lehetnek a tervezési változások jóváhagyásai vagy gyártási dokumentációk verziókezelései. Ezek inkább adminisztratív tranzakciók, mintsem operatívak. (Gála, L; Pour, J; Šedivá, Z. 2009) (Sodomka, P.; Klčová, H. 2010)

9 Információs rendszerek támogatása a döntéshozatalban

"*Egy jó döntés mindig egy tapasztalaton alapul. A tapasztalat pedig gyakran egy rossz döntés eredménye*."

(Rita Mae Brown)

Frederickson 1984-ben a problémákat három kategóriára osztotta fel. Ennek a felosztásnak manapság is jelentősége van, főleg a vállalati információs rendszerek kapcsán. A **jól strukturált** (*Well-structured*), a **részben strukturált** (*Structured*) vagy csak strukturált és a **rosszul strukturált** (*Nonstructured*) vagy strukturálatlan problémák kategóriáit adja meg. A jól strukturált problémák megoldása automatizálható, hiszen ismertek a helyes megoldás lépései. A részben strukturált problémáknál általában a megoldás legalább egy része a megoldók vállat terheli (felelősség). Ebben az értelemben a produktív gondolkodás elvárt. Nem egyértelműek viszont a rosszul strukturált problémák (Sziray, J.; Gaul, G.; Égertné Molnár, É. 2007). A rosszul strukturált problémákat célszerű három szinten megközelítenünk:

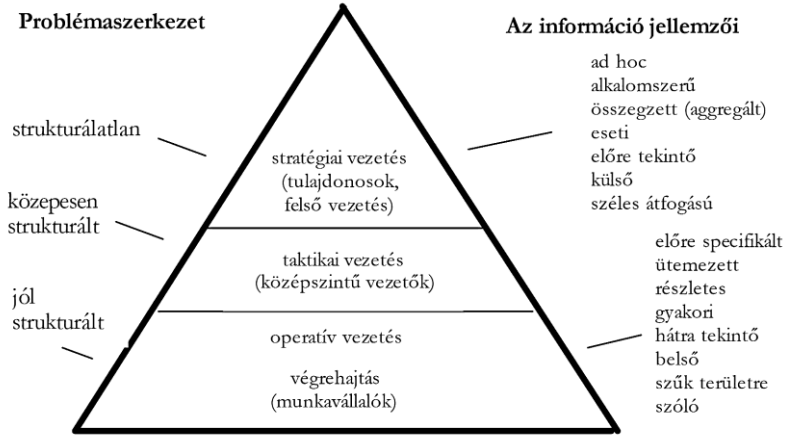
- **Információk szintjén** – hogyan állítsuk elő a szükséges információkat?
- **Tevékenységek szintjén** – hogyan használjuk fel az információkat tevékenységek elindítására?
- **Értékek szintjén** – mire és hogyan használjuk fel az információkat?

A vezetői, illetve a hozzájuk kapcsolódó döntési szinteknek az elméletét Robert N. Anthony dolgozta ki az 1960-as években. A vállalati hierarchia három szintjét dolgozta ki. Az alsó szint az operatív vezetés, a középszint a taktikai vezetés és a felső szint a stratégiai vezetés. Nem nehéz belátni, hogy milyen problémák szerepelnek az egyes szinteken. Az irodalom említ még egy szintet is, az operatív szintet, ami az adatfeldolgozási szint.

A gyakorlatban használjuk az ún. **programozható** és **nem programozható döntések** kifejezéseket is. Ezek a kifejezések Herbert A. Simon, Nobel-díjas közgazdász nevéhez fűződnek, aki a jól strukturált

problémák esetében a programozható, és a nem strukturált problémák esetében a nem programozható döntés kifejezéseket használta (az irodalomban még találkozhatunk a **strukturált** és **nem strukturáld döntés** kifejezéssel is, melyek szintén ez előbbiekre utalnak). Egyértelmű, hogy a jól strukturált problémák megoldhatók ismert algoritmusok alkalmazásával. Ilyen döntések meghozatalára tehát az információs rendszer is képes, aminek használata emberi erőforrásokat szabadíthat fel. Nem strukturált döntés esetén maga a probléma sem mindig jól definiált, a döntéshozó nem látja azt át teljesen, nincs tisztában minden összefüggéssel. A megoldás lépései ilyenkor nem határozhatók meg előre, sokszor ötletszerűek, változó feltételektől függnek. Ne felejtkezzünk meg arról sem, hogy a gyakorlatban előforduló döntések nagy része a jól strukturált és nem strukturált problémák jegyeit hordozzák. (Kacsukné Bruckner, L.; Kiss, T. 2007).

Strukturáltság foka szubjektív kérdés. Ha pl. az autó elektronikus vezérlőegységének egyik részében meghibásodik egy program, akkor ezt a strukturáltság szempontjából másképen fogja látni a közgazdász, autószerelő vagy programozó. **Probléma strukturáltsága a problémamegoldó személy tudása és tapasztalata alapján minősíthető**. Például, egy kereskedelmi igazgató számára a készletgazdálkodási probléma félig strukturált problémát jelent. (Sziray, J.; Gaul, G.; Égertné Molnár, É. 2007) (Kacsukné Bruckner, L.; Kiss, T. 2007) (Antal, P. 2021) (Rosický, A. 2010)



9.1. ábra: A döntéshozatali hierarchia és információ igénye
Forrás: (Sziray, J.; Gaul, G.; Égertné Molnár, É. 2007)

9.1 Döntéshozatalok az egyes szinteken

Az üzleti folyamatok során a közreműködő személyek, szerepüktől függően, kisebb vagy nagyobb gyakorisággal hoznak különböző jellegű döntéseket, amihez információra van szükségük. A döntéshozatalhoz minőséges információra van szükség. Kacsukné Bruckner, L.; Kiss, T. (2007) szerint a minőséges információnak következő tulajdonságai vannak.

- **Releváns** – hasznos és fontos az adott döntés meghozatalához, és javítja annak minőségét.
- **Időszerű** – az információ, amit a döntéshozatalhoz felhasználunk, ne legyen elavult.
- **Pontos** – az információ legyen hibamentes. Ezt a gyakorlatban nem mindig tudjuk betartani, ilyenkor fontos, hogy definiált és elfogadott hibahatáron belül legyen.
- **Ellenőrizhető** – ez azt jelenti, hogy az információ pontosságáról meg tudjunk győződni, ellenőrizni tudjuk azt. Ezt egyrészt megtehetjük úgy, hogy olyan információval hasonlítjuk össze, melyről tudjuk, hogy pontos.
- **Teljes, de minimális** – szerezzünk be minden olyan információt, melyre a döntéshez szükségünk lesz. Viszont figyeljünk oda, hogy a feleslegeseket szűrjük ki. A menedzser számára is a túl sok információ zavaró lehet és hibához vezethet.
- **Könnyen érthető** – az információ világos, jól érthető, könnyen áttekinthető és magyarázó legyen a felhasználó számára.

9.1. táblázat: A döntéshozatal három szintjének jellemzői

Jellemzők	**A döntéshozatal szintje**		
	Operatív	**Taktikai**	**Stratégiai**
Problémák változatossága	alacsony	közepes	nagy
Strukturáltság foka	magas	közepes	alacsony
Bizonytalanság foka	alacsony	közepes	magas
Szükséges ítélőképesség foka	alacsony	közepes	magas
A döntés hatásának időtartama	napok	hónapok	évek
Programozható döntések	legtöbb	néhány	semmi
Tervezési döntések	kevés	kb. fele	legtöbb
Ellenőrzési, felügyeleti döntések	legtöbb	kb. fele	kevés

Forrás: (Kacsukné Bruckner, L.; Kiss, T. 2007)

9.2. táblázat: A különböző döntési szinteken szükséges információ jellemzői

Információ jellemzői	A döntéshozatal szintje		
	Operatív	Taktikai	Stratégiai
Függés belső információtól	nagyon magas	magas	közepes
Függés külső információtól	alacsony	közepes	nagyon magas
Információösszegzés foka	nagyon alacsony	közepes	magas
Real-Time információ igénye	nagyon magas	magas	közepes
Előre jelző információ igénye	alacsony	magas	nagyon magas

Forrás: (Kacsukné Bruckner, L.; Kiss, T. 2007)

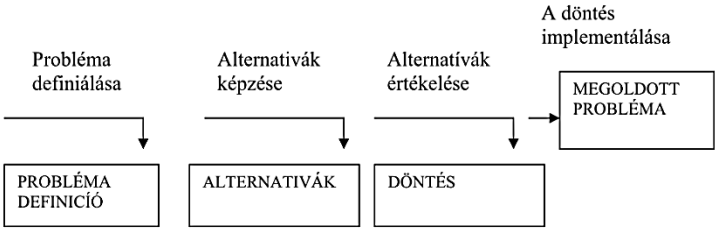
A stratégiai szinten a vállalat hosszú távú céljait és politikáját határozzák meg. Itt tűzik ki a jövőbeni célokat, és megtervezik az elérésükhöz vezető lépéseket. Az ilyen döntésekhez összefoglalt, könnyen átlátható, grafikus vagy táblázatos formátumú információ szükséges. Ezen a szinten a menedzserek gyakran kevéssé strukturált, bizonytalan és változatos problémákkal találkoznak, ahol jelentős szerepet kapnak a külső forrásokból, például a versenytársakról, piaci helyzetről és gazdasági környezetről szóló információk. A felső vezetőknek nem szabad elmerülniük a részletekben; az összefüggéseket és kapcsolatokat kell látniuk, hogy következtetéseket vonhassanak le. Stratégiai döntés lehet például egy új gyáregység helyszínének kiválasztása vagy annak eldöntése, hogy milyen termékek gyártásával érdemes foglalkozni.

A taktikai döntések középvezetői szinten, azaz az osztály- és részlegvezetők szintjén születnek. Ezek a döntések a vállalat alegységeinek rövid és középtávú terveit, céljait és költségvetéseit határozzák meg, illetve a stratégiai döntések végrehajtását is ezen a szinten végzik. A döntéshozatal itt főként a belső, vállalaton belüli információkon alapul, bár a külső, környezetből érkező adatok is szerepet kaphatnak. A részletek ebben az esetben már fontosabbá válhatnak, de gyakran szükség van az információk összefoglalására is. Példaként említhetők a költségkeretek felhasználásával és elosztásával kapcsolatos döntések.

Az operatív döntéseket az alsó vezetői szinten hozzák meg. Itt rövid távú, a konkrét feladatok végrehajtására irányuló problémákkal foglalkoznak. Az alsó szintű döntéshozatalban különösen fontos a részletes, pontos és naprakész, „Real-Time" információ. A legtöbb probléma jól strukturált, és megoldásuk gyakran programozható. Mivel a feladatok jelentős része már automatizált, az alsó vezetők nagymértékben támaszkodnak a számítógépes információs rendszerekre. A döntések itt szinte kizárólag a vállalaton belüli információkra épülnek, a külső tényezők szerepe pedig minimális. (Kacsukné Bruckner, L.; Kiss, T. 2007)

9.2 Problémamegoldás szisztematikus megközelítése

A vállalati információs rendszerek területén a problémamegoldás szisztematikus megközelítése azt jelenti, hogy a problémákat tervezetten, strukturált módon oldják meg. Itt általában lépésről lépésre átgondolt folyamatot alkalmaznak, hogy biztosítsák a hatékony és eredményes problémamegoldást.



9.2. ábra: A döntési folyamat lépései (William E. Leigh és Michael E. Doherty szerint). Forrás: (Herdon, M. 2006)

Ez a megközelítés magába foglalhat különféle módszereket és eszközöket. Különböző irodalomban különböző lépésekkel találkozhatunk, de mindegyiküknek van egy közös része, amiben megegyeznek. William E. Leigh és Michael E. Doherty szerint, amit a 9.2. ábra is szemléltet, a következő képpen írja le e folyamat egyes lépéseit:

- **Probléma definiálása és azonosítása**.
    - Meg kell érteni a probléma gyökerét, hogy pontosan mi okozza a zavarokat vagy hibákat. Ez magában foglalja a probléma okainak feltárását, az érintett területek és folyamatok azonosítását, valamint a lehetséges következmények megértését. A problémaállapotot (észlelt, jelen idejű helyzet) esetében, definiálnunk kell a célállapotot is, amelyt el akarunk érni. Amennyiben lehetséges, a problémát részproblémákra bontjuk, mivel gyakran a részprobléma önállóan elemezve könnyebben megoldható.
    - A döntési folyamatban felhasznált adatok begyűjtése szintén ehhez a ponthoz sorolható. Az adatok egyrészt a vállalat belső adatbázisából, a szervezet saját tranzakciót feldolgozó rendszeréből származnak, másrészt külső adatbázisokból érkeznek, és végül vannak a menedzser saját kezelésében lévő, személyes adatok is. A külső

adatforrások alatt érthetjük pl. az újságokat, TV, internetet, stb. A külső adatok bevonása növeli a döntéstámogató rendszer (DSS) pontosságát és relevanciáját, mivel így nemcsak a belső, történeti adatok alapján hoz döntéseket, hanem figyelembe veszi a jelenlegi gazdasági vagy piaci változásokat.

- A döntéstámogató rendszerek különféle analitikai eszközökkel (pl. statisztikai elemzések, trendek, kockázatelemzés) segíthetnek a probléma pontos megértésében, támogatva a helyes probléma-meghatározást, ami kulcsfontosságú a későbbi lépések sikeréhez. Szintén az adatbázis-kezelő lekérdezések adatai jó kiinduló pontot biztosíthatnak.
- Itt megemlíthetjük a minőségi és mennyiségi adatanalízist is. A külső adatok bevonásakor mindig fennáll a kockázata annak, hogy az adatok hibásak, hiányosak vagy sérültek lehetnek. A **minőségi analízis** esetében a döntéstámogató rendszerek ellenőrzik, hogy a külső adatok pontosak, teljesek és konzisztensek legyenek, és kiszűrik a hibás vagy hiányos adatokat, hogy megbízható döntések alapjául szolgáljanak. A **mennyiségi analízisnél** a döntéstámogató rendszerek ellenőrzik, hogy a szükséges mintanagyság és frissesség rendelkezésre áll-e. **Adattisztítás** (részletesebben: 11. fejezet) alatt azt értjük, hogy a döntéstámogató rendszerek eltávolítják a hibás és irreleváns adatokat, majd feldolgozzák azokat, hogy a döntési modellek számára alkalmasak legyenek, így csökkentve a hibalehetőségeket. A döntéstámogató rendszerek különböző **minőségi mutatókat** alkalmaznak, például pontosságot és megbízhatóságot, hogy mérhetővé és nyomon követhetővé tegyék az adatminőséget, növelve a rendszer hatékonyságát. A döntéstámogató rendszerek ellenőrzik, hogy a külső adatok pontosak és relevánsak legyenek, különösen stratégiai szinten, ahol az adatminőség kulcsfontosságú.

- A probléma definiálása után következő lépésként több irodalom is az **elemzést és adatgyűjtést** tárgyalja. A

probléma részleteinek alapos feltárása tartozik ide, például adatok elemzése abból a szempontból, hogy kiderüljön, milyen tényezők állnak a háttérben. Ez a lépés az egyes lépéseken belüli feladatok hasonlósága miatt a probléma definiálása alá sorolható, és mi is ezt így tesszük, ami a vállalati információs rendszerek szempontjából teljesen logikus[41].

- **Lehetséges megoldások kidolgozása**.
    - Ide tartozik a több alternatíva – több lehetséges megoldási javaslat kidolgozása a probléma kezelésére. Ezek lehetnek különböző stratégiai lépések, beavatkozások, új folyamatok vagy technikai megoldások. Használva van a tapasztalat, tudás, de a döntéshozó intuíciója is. Nem kizárt a Brainstorming vagy Benchmarking.
        - A döntéstámogató rendszerek támogatást nyújtanak a döntéshozók számára abban is, hogy objektív és mérhető kritériumokat állítsanak fel a lehetséges megoldások kiértékelésére. Ilyenek lehetnek a költségek, az időráfordítás, a minőség, az erőforrásigény vagy a szervezet stratégiai céljainak való megfelelés.
        - Másik fontos rendszertulajdonság, amit itt megemlíthetünk, a modellek használata. A döntéstámogató rendszerek modellezési és szimulációs képességei lehetővé teszik, hogy a döntéshozók előre láthassák a különböző megoldások lehetséges hatásait. Ez különösen hasznos, mert segíthet az alternatívák kockázatainak, költségeinek és várható előnyeinek objektív összehasonlításában.

> *Ha nekünk kellene dönteni egy fontos probléma esetében, hogy pl. milyen változásokat szükséges végbevinni annak érdekében, hogy javuljon a termékünk pozíciója a piacon, akkor eszünkbe juthat a jobb marketing bevezetése, a termék innoválása, a termék árának csökkentése stb. – mindezek nem mások, mint alternatívák.*

[41] Az elemzést és az adatgyűjtést általában nem külön lépésként, hanem a probléma azonosítása és megoldás kidolgozása részeként kezelik. Mi csak felhívtuk a figyelmet néhány irodalomban előforduló magyarázatra.

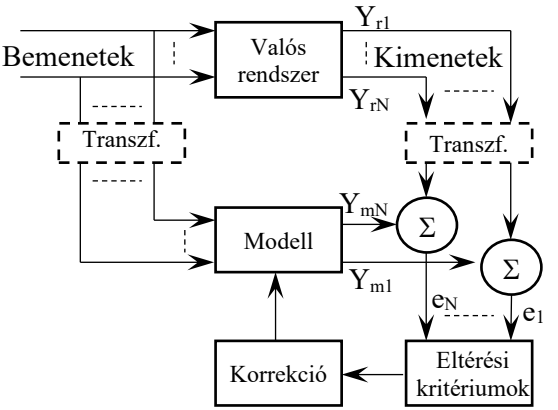
- **Megoldás kiválasztása - döntés**.
    - Ezek a javaslatok a vállalat vezetői vagy döntéshozói számára különféle lehetőségeket kínálnak egy adott problémára. A kiválasztási folyamatban a döntéshozók különféle szempontokat vesznek figyelembe: a költségeket, az időtényezőt, a várható eredményeket, a kockázati szinteket, valamint az egyéb, a szervezet stratégiájával és céljaival összhangban lévő kritériumokat.
        - A döntéstámogató rendszerek általában több megoldási javaslatot is képesek generálni az elemzett adatok alapján, amivel segítik a döntést.
- **Implementálás, implementálás értékelése és finomítás**.
    - Miután kiválasztották a legmegfelelőbb megoldást, következik az implementáció, ami azt jelenti, hogy a kiválasztott döntést beültetik a gyakorlatba. Az implementálás során a döntéstámogató rendszer figyeli a megvalósítás hatékonyságát, és szükség esetén módosítási javaslatokat tesz.
        - A döntéstámogató rendszerek figyelhetik a megoldás hatékonyságát és teljesítményét is, így, ha szükséges, a vezetés gyorsan be tud avatkozni vagy módosítani tud a folyamatokon.

Az ilyen szisztematikus megközelítést például olyan keretrendszerek is támogatják, mint a PDCA-ciklus (Plan-Do-Check-Act), illetve olyan problémamegoldási modellek, mint az *Ishikawa-diagram* vagy az *5 Miért technika*, amelyek magyarázata túlmutat e tankönyv keretein, ezért részletesebben nem fogunk velük foglalkozni. (Antal, P. 2021)

9.3 Modellek használata és tanulás a döntéstámogató rendszereken belül

A **modellezés** egy olyan kísérleti folyamat, amelynél **a vizsgált rendszerhez** (az eredetihez), **bizonyos kritériumok mellett, hozzárendelünk egy másik rendszert – modellt**, **amely a modellezési folyamat eredménye**. A modellezés alatt **bizonyos tevékenységek halmazát is érthetjük, melyek a modell kifejlesztéséhez vezetnek**. Az ilyen modell reprezentálja a valós rendszer struktúráját, viselkedését. Azt is állíthatjuk, hogy minden modell minden esetben valamilyen valóságnak az egyszerűsítése, viszont engedi megérteni a valóságot is. George Box szavait

idézve: „*Minden modell rossz, de némelyik hasznos lehet*”. A politikai térképre tekintve láthatjuk, hogy miért. Bár a térkép nem tartalmazza az összes részletet, az államhatárok nem pontosak, de mégis nagyon jó elképzelést nyújt számunkra. A modellnek viselnie kell a rendszer összes fontos és jelentős tulajdonságát!



9.3. ábra: Modellezési folyamat

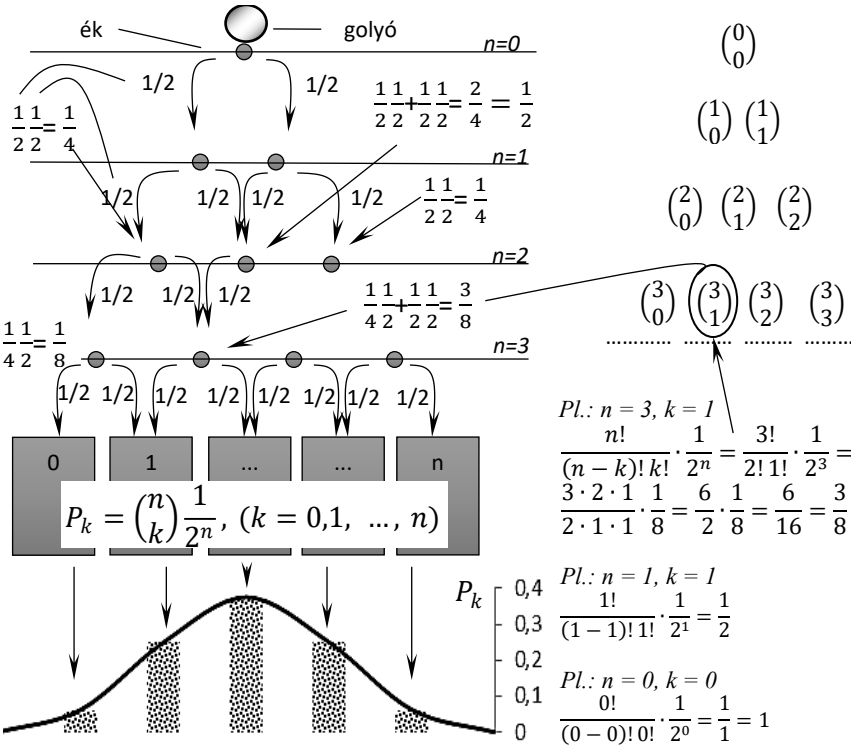
*Leírás: Modellezés során validálást és más vizsgálatokat is használnak. Ha a modell kimenetei (Y_{m1}, Y_{m2}, …, Y_{mn}) nem azonosak a valós rendszer viselkedésének mérhető kimeneteivel (Y_{r1}, Y_{r2}, …, Y_{rn}) az azonos bemeneti paraméterek mellett, akkor különböző eltérési kritériumok alapján korrekcióra van szükség (($e_1=Y_{r1}-Y_{m1}$); ($e_2=Y_{r2}-Y_{m2}$); …; ($e_n=Y_{rn}-Y_{mn}$) $\Rightarrow$ **If** [e_1, e_2, …, $e_n \neq 0$] **Then** …), amíg a modell kimenetei nem megfelelők. Σ - jellel az e_1, e_2, …, e_n eltéréseket előállító egységet jelöljük. Transzf. alatt a fizikai vagy üzleti paraméterek átalakítását fogjuk érteni olyan matematikai értékekre, melyek használhatók a számítógépes modelleknél.*

Forrás: (Takáč, O. 2017)

A továbbiakban fontos modellekre mutatunk néhány példát.

- **Matematikai modell –** egy absztrakt modell, amely matematikai jelöléseket használ a rendszer leírásához. Az ilyen modelleket, úgy a természettudományokon belül (fizika, biológia, elektrotechnika, stb.), mint a társadalomtudományoknál (gazdaságtudomány, szociológia, stb.), ill. a mérnöki tudományágak esetében is egyaránt használjuk.
- **Fizikai modell –** szigorúan csak fizikai törvények alapján van összeállítva, ahol a vizsgált „anyag” kisebb, olcsóbb vagy jobban kezelhető tömegben van használva.
- **Számítógépes modell –** bizonyos részben a matematikai modellből indul ki, majd ezekre épülnek rá az algoritmusok, a változók, a paraméterek, végül ezek egy programban vannak összefoglalva. Az

ilyen modell számítógépek segítségével szimulációnak van alávetve, és a megkapott eredmény könnyen interpretálható.



9.4. ábra: Példa egy matematikai modellre – Galton-deszka

*Leírás: Eresszünk egy golyót az első ékre. A golyó ½ valószínűséggel, a gravitáció hatására jobbra, és ½ valószínűséggel balra gurul – n = 0 sor. Ha pl. balra gurulna, akkor egy másik ékkel találkozna, ahol újra ½ valószínűséggel jobbra, ill. balra folytatná az útját. Csakhogy, az n = 2 soron az a valószínűség, hogy a sor első ékére esne a golyó már ½*½ = ¼. Annak a valószínűsége, hogy a sor középékére essen, ½*½ + ½*½ = ½, és annak, hogy a másik szélre, ¼. Feltehetünk most egy kérdést. Ha több golyót indítanánk, mennyi golyó esne bele a 0-dik dobozba, mennyi az 1-be és mennyi az n-edik dobozba? Erre szintén a valószínűségek (P_k) adnak választ.*

Forrás: (Takáč, O. 2017)

Ha tehát egy bizonyos modell szimulációnak van alávetve, **szimulációs modellnek** nevezzük. A szimulációs modellel végrehajtott kísérletet pedig **szimulációs kísérletnek**. Ezek sorozatát ugyanazon célból **szimulációs vizsgálatnak** nevezzük. **A szimulációt olyan módszerként definiálhatjuk, mely a modellezett rendszerről egy modell segítségével** (változtatva a bemeneti paramétereket) **új tudáshoz és megismeréshez vezet.** Erre példa a természet, melyek matematikai összefüggésekkel leírhatók, aminek eredménye egy matematikai modell lenne. Az ilyen modellből számítógépes modellt is készíthetnénk, melyen elvégezhetnénk

a szimulációkat. Egyértelmű, hogy a szimulációs kísérletet nem egyszer, hanem statisztikai kiértékelések segítségével többször is meg kell ismételni mindaddig, amíg a keresett információ elegendő pontosságú nem lesz.
Vegyük példaként a már említett Galton-deszkát. A szimuláció során minden egyes alkalommal, pl. 100 golyót indítunk el, és a golyók véletlenszerűen haladnak végig a pályán az akadályokon keresztül. Minden egyes szimuláció más eredményt hoz, mivel a golyók mozgása a véletlenszerűségen múlik, és a véletlen által generált eltérések miatt az egyes rekeszekbe eső golyók száma mindig különböző lesz. Fontos megérteni, hogy bár az eredmények szimulációnként változnak, a **modell maga nem változik**. Ez azt jelenti, hogy a Galton-deszka szabályai és struktúrája állandó, és a rendszer működése mindig ugyanazokat a statisztikai elveket követi. Az egyedi szimulációk során megfigyelt eltérések tehát nem a modell változásából fakadnak, hanem a **sztochasztikus folyamat** természetéből (sztochasztikus modell). Ez a példa jól szemlélteti a valószínűség és a statisztika alapelveit, amelyeket más területeken is alkalmaznak. Pl. a pénzügyi modellek esetében az árfolyam-ingadozások előrejelzésénél megfigyelhető, hogy a szimulációk eltérő eredményt adhatnak, de a modell állandó marad. Szintén egy jó példa lehet a közlekedés modellezése. A forgalom szimulációja során a járműáramlás változhat, de a modell paraméterei nem változnak.

A döntéstámogató rendszerek nagyon fontos tulajdonsága és szolgáltatása, hogy modellezési és szimulációs képességekkel rendelkezik. A modellezés és szimuláció előnyeit következőképpen foglalhatnánk össze.

- Kevésbé költséges, mint az esetleges rossz döntés következményeinek a viselése.
- Gyors, nem időigényes. Modellek segítségével akár hónapokat, éveket is tudunk néhány perc alatt szimulálni.
- A jó modell képes a jövőbe látni, bizonyos dolgokat előre jelezni.
- A modell kezelése könnyű (intuitív).
- Több lehetőséget is megengednek kipróbálni, több különálló modellek egy idejű használatával vagy szimuláció során különböző bemenetek beállításával. Így szerzett kimeneteket elemezhetjük.
- Ha használunk mesterséges intelligenciát, akkor lehetőségünk van a tanulásra és betanítására.

A modellezés és szimuláció hátrányai közül legalább a következőket említsük meg:

- Az üzleti rendszer modellezése meglehetősen nehéz, bonyolult feladat, melyben könnyű hibázni. Egy ilyen hiba következménye a

hibás modell, amely rossz, könnyen félrevezető eredményeket szolgáltat.

- A modellezéshez magas szintű matematikai ismeretek szükségesek, mellyel a menedzserek általában nem rendelkeznek.

A döntéstámogató rendszereknek fontos szerepe van a releváns adatok vagy jellemzők kiválasztásában és előkészítésében. Ez az úgynevezett „**Feature Engineering**" folyamat elősegíti, hogy a modellek jobb döntési eredményeket adjanak. A döntéstámogató rendszerek gyakran több különböző modellt használnak a döntéshozatal támogatására. Ezek a modellek különböző szempontokat és adatokat vesznek figyelembe, amelyeket különböző súlyokkal értékelnek. Bemeneti adatok lehetnek például a termék ára, annak gyártási költségei, a piaci kereslet, stb. Ezeket a bemeneteket **súlyozhatják** a szerint, hogy mennyire relevánsak az adott döntés szempontjából. A súlyozás azt biztosítja, hogy az igazán fontos tényezők (pl. a terméknek a piaci ára nagyobb fontosságú, mint a javasolt ára, stb.) nagyobb hatással legyenek a döntési eredményre. Az AI gyakran beépül ezekbe a modellekbe, hogy „tanuljon" és folyamatosan javítsa a súlyozást és a modell működését. Fejlettebb rendszerek képesek **hibajavításra és önkorrekcióra**. Azok a döntéstámogató rendszerek, amelyek a tanulás során képesek önmagukat javítani vagy frissíteni (önjavító algoritmusok), képesek felismerni, ha bizonyos döntési minták elavultak. Így automatikusan módosíthatják döntéseiket. A döntéstámogató rendszerek modelljei gyakran igényelnek folyamatos finomhangolást és validálást, hogy pontosak maradjanak. A **validáció** során külön adatcsoportokkal tesztelik a modellt, így biztosítva, hogy a tanulási folyamat jól generalizálható legyen, és ne csak a tanulóadatokra adjon pontos válaszokat. **A döntéstámogató rendszerek tanulása és a modellek használata erősen összefügg**. **A modellek és a tanulás elválaszthatatlanok a döntéstámogató rendszerek esetében**, hiszen a modell képezi a rendszer tanulási folyamatának alapját. Ahhoz, hogy a döntéstámogató rendszer képes legyen tanulni és hasznos előrejelzéseket vagy javaslatokat adni, megfelelő matematikai vagy statisztikai modellre van szüksége, amelyen alapulhat. A modell a tanulás alapját képezi, mivel az adatokat ezen a modellen keresztül elemzi és értelmezi. A modell választása (például regressziós modell, döntési fa, neurális hálózat, stb.) meghatározza, hogy a rendszer milyen módon és milyen pontossággal tudja az adatokat értékelni és a tanulási folyamatot végrehajtani. A döntéstámogató rendszerek tanulása az adatfeldolgozási és elemzési képességeikre épül, amelyek mesterséges intelligenciát és gépi tanulási algoritmusokat is használhatnak. A döntéstámogató rendszerek több módon és különböző adatból tanulhatnak. A tanulásuk történhet az alábbi módon:

- a múlttal kapcsolatos adatelemzés alapján (a múltbeli adatok alapján mintázattokat és trendeket azonosítanak, amelyek segítik a jövőbeni döntéseket),
- mélységi tanulás és neurális hálózatok alapján (fejlettebb döntéstámogató rendszerek mélységi tanulási modelleket, például neurális hálózattokat alkalmaznak, melyek nagy mennyiségű adatot dolgoznak fel és komplex döntési szabályokat tanulnak meg),
- adaptív tanulás alapján (például egy pénzügyi döntéstámogató rendszer folyamatosan figyeli a tőzsdei mozgásokat, a rendszer idővel tanulhat, és módosíthatja előrejelzéseit vagy figyelmeztetéseit a változó trendek alapján),
- interaktív tanulás alapján (döntéstámogató rendszerek gyakran kapnak visszajelzéseket a felhasználóktól, például döntések vagy eredmények után, és ezeket a visszajelzéseket felhasználhatják további finomhangolásra),
- és további módon.

Ezek a módszerek (pl. a múlttal kapcsolatos adatelemzés, mélytanulás) egymással is kombinálhatók, ami a döntéstámogató rendszerek rugalmasságát növeli. Nézzünk példát egy gépi-tanulás alapú döntéstámogató rendszerre, mely felügyelt tanulást alkalmaz:

> *Legyen egy konkrét probléma, amelyet a menedzser az adatbázisban levő adatok alapján megold. Ezt a megoldást közli a döntéstámogató rendszerrel is. Ezzel a rendszer megkapja azt a mintát, amelyen a jövőbeli hasonló döntéseknél dolgozhat. Egy idő után a menedzser visszatér a döntéséhez és értékeli, mennyire volt hatékony vagy optimális. Később megint hasonló problémát old, újra és újra. Ezt az értékelést a rendszer „megtanulja", ami azt jelenti, hogy a rendszer felismeri, mely döntések váltak be, és melyek nem. Egy idő után a rendszer megtanul hasonló problémákat oldani, a menedzser döntései alapján (őt szimulálva).*

Muszáj azonban elmondani, **hogy a rendszer nem válik önálló döntéshozóvá,** hanem „javaslatokat kínál", amelyek segíthetik a menedzser döntéseit.

Ez a fajta rendszer egyre gyakoribb, különösen ott, ahol nagy mennyiségű strukturált adat és ismétlődő döntési helyzetek vannak. Például, egy készletgazdálkodási döntéstámogató rendszer javaslatokat tehet, hogy mikor és milyen mennyiségben rendeljen új készletet. Az így betanított rendszer, a menedzser hibáit is megtanulhatja. Egy nagy vállalaton belül,

több menedzser hasonló típusú döntéseket hoz. A döntéstámogató rendszer nemcsak egyetlen menedzser döntéseire támaszkodik, hanem **több menedzser döntéseit** és azok eredményeit elemzi. Így a rendszer egyfajta „átlagos" vagy leggyakoribb döntési mintát tanulhat meg, ami statisztikailag nagyobb eséllyel lesz helyes. Ha például egy menedzser döntése többször is hibásnak bizonyul, a döntéstámogató rendszer nem veszi figyelembe azokat a mintákat, amelyek ehhez a döntéshez vezettek. A döntéstámogató rendszer képes lehet súlyozni a menedzseri döntéseket – például ebben az esettben előnyben részesíti azokat a menedzsereket, akik hosszú távon jobb eredményeket értek el. Ha egy menedzser döntései következetesen eredményesek, a rendszer ezekre a döntésekre nagyobb „súlyt" helyezhet. A döntéstámogató rendszerek tanulási képességeit **külső adatok** bevonása is gazdagíthatja, pl. külső gazdasági vagy piaci adatok, melyek lehetővé teszik a döntések szempontjából szélesebb kontextus megismerését. Ne feledkezzünk el az etikai és jogi szempontokról sem a döntéshozatalban, melyeket be kell építeni a döntéstámogató rendszerekbe is. Például, az egészségügyi rendszerek esetében figyelembe kell venni a beteg jogait és az adatvédelmi előírásokat, míg a pénzügyi szektorban a döntéseket a vonatkozó szabályozási előírások szerint kell alakítani. (Kacsukné Bruckner, L.; Kiss, T. 2007) (Takáč, O. 2017) (Antal, P. 2021)

9.3.1 Minták és tudás tárolása

A döntéstámogató rendszerek által „megtanult" mintákat és tudást többféle helyen tárolhatjuk, attól függően, hogy milyen típusú adattal és modellel dolgozunk.

- A legtöbb döntéstámogató rendszer rendelkezik egy vagy több **adatbázissal**, ahol az alapadatok, mint például az ügyféladatok, pénzügyi adatok vagy operatív adatok kerülnek tárolásra. A megtanult minták, súlyozott értékek, döntési eredmények, valamint a visszajelzések szintén elhelyezhetők ebben az adatbázisban.
- Az AI által használt gépi tanulási modellek gyakran speciális **modelltárakban**[42] (MR – *Model Repository*) vannak tárolva. Ezek

[42] A modelltár olyan adatstruktúra vagy tároló, amely különböző típusú adatmodelleket, elemzési modelleket, prediktív modelleket vagy gépi tanulási modelleket tartalmaz, amelyeket a döntéstámogató rendszer használhat a döntési folyamat során. A modelltárat gyakran helyi szervereken vagy a vállalati adatközpontban tárolják. Sok vállalat a modelltárakat felhőalapú infrastruktúrákban tárolja (pl. AWS, Azure, Google Cloud), mert így a modellek elosztott környezetben, bárhonnan elérhetők.

a tárak tartalmazzák az egyes modellek beállításait, súlyait, tanulási paramétereit és korábbi döntési mintákat. Az AI itt tudja frissíteni a modelleket és hozzáadni az új tanulási eredményeket.

- Léteznek olyan döntéstámogató rendszerek, amelyek rendelkeznek különálló **tudásbázissal**[43], amely strukturált információkat és szabályokat tárol a korábbi döntések alapján. A tudásbázisok használata lehetővé teszi, hogy az AI a megtanult információkat jól strukturált formában tárolja és később újra felhasználja.
- Egyre gyakoribb, hogy a döntéstámogató rendszerek és az AI-alapú modellek **felhőalapú környezetekben** működnek. A felhőben tárolt modellek és adatbázisok rugalmas hozzáférést biztosítanak a különböző típusú adatokhoz és a frissítésekhez. Emellett az adatok és modellek bármikor skálázhatók, így egy nagyobb vállalat több részlege is hozzáférhet ugyanahhoz a döntéstámogató rendszerhez.
- Egyes rendszerek az AI döntési folyamatait és a tanulási eredményeket **eseménynaplókban** (**logokban**) tárolják, amelyek későbbi elemzésre is felhasználhatók. Ezekből a logokból a rendszer visszakeresheti, hogy korábban milyen döntéseket hozott és milyen tényezők befolyásolták a döntést. (Kacsukné Bruckner, L.; Kiss, T. 2007) (Takáč, O. 2017) (Antal, P. 2021)

9.4 Döntéstámogató rendszerek

A vezetői információs rendszereket a rutin, strukturált döntések támogatására fejlesztettek ki. Az 1970-es évek elején jelentek meg először olyan rendszerek, melyek már egy adott problémára tudtak koncentrálni, és interaktív módon járultak hozzá a döntési folyamathoz. A döntéstámogató

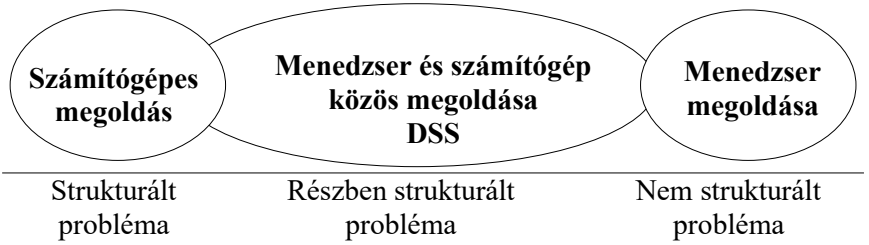
[43] Hol található fizikailag a tudásbázis és mi az? A tudásbázist tárolhatjuk merevlemezen, tehát egy speciális **adatbázisként vagy egyszerű fájlstruktúrában** (lásd az adatbázisokkal foglalkozó fejezetet), ahol specifikus, döntéstámogatáshoz szükséges adatok vannak mentve. Nagyobb rendszerek esetében, különösen, ha a döntéstámogató rendszer több forrásból integrál adatokat, a tudásbázis **elosztott környezetben** is működhet. Néhány tudásbázis például relációs adatmodellt használ. Ha komplexebb kapcsolatokra van szükség (például összetett szabályrendszerre), objektum-orientált adatmodelleket használhatnak. Néhány döntéstámogató rendszer speciális ontológiákat vagy szemantikus hálókat használhat. Az **ontológia** olyan fogalmi modell, amely egy adott szakterületen lévő fogalmakat és ezek kapcsolatait írja le. A **szemantikus** háló egy gráf alapú tudásreprezentáció, ahol csomópontok (pontok) és élek (kapcsolatok) képviselik a fogalmakat és azok közötti kapcsolatokat.

rendszereket két nagy fő csoportra osztjuk, azaz **modell vezérelt**, és **adat vezérelt** döntéstámogató rendszerekre. Történelmileg először a **modell vezérelt rendszerek** alakultak ki, eredetileg ezeket nevezték döntéstámogató rendszereknek (*DSS – Decision Support System*). Ezek a hagyományos döntéstámogató rendszerek a kis- és közepes vállalatok igényeit még ma is tökéletesen kielégítik. **Az adatvezérelt döntéstámogató rendszerek nagyméretű** adatbázisok elemzésén alapulnak, amire a nagy vállalatoknak van szükségük.

Az információs rendszereknek alkalmazkodniuk kell a döntés típusokhoz, és a különböző döntésekhez különböző információt kell szolgáltatniuk. Egyértelmű, hogy egy felső vezető döntéseinek nagy része nem strukturált problémákból áll, míg az alsóbb szinteken a nagyobb strukturáltság figyelhető meg. Azért a nem strukturált döntésekhez átfogóbb jellegű, előre nem mindig definiálható információra van szükség, amihez rugalmas információs rendszert, interaktivitást és ad hoc lekérdezéseket kell biztosítani. A túl sok ismeretlen tényező és kölcsönhatás miatt ezek a problémák emberi döntést igényelnek, a számítógép nem képes helyettesíteni a döntéshozót. A túl sok ismeretlen tényező és kölcsönhatás miatt ezek a problémák emberi döntést is igényelnek, az IT megoldások nem mindig képesek helyettesíteni a döntéshozót. A valós problémák nagy része egyaránt tartalmaz programozható és nem programozható elemeket, vagyis a döntési problémák nagy része félig strukturált. Az ilyen esetekben lehetnek a döntéstámogató rendszerek a menedzser segítségére, amikor a döntés az ember és a számítógép közös munkájának az eredménye.

A döntéstámogató rendszer fogalmat olyan rendszerek leírására használjuk, amelyek támogatják, de nem helyettesítik a vezetőket döntéshozatali tevékenységükben. Tehát a döntéstámogató rendszer (DSS – *Decision Support System*) a számítógépre készített szoftver[44] eszközök egy készlete, amelyet egy menedzser használ a probléma megoldására a döntéshozói tevékenysége során. A döntéstámogató rendszert Kacsukné Bruckner, L. és Kiss, T. (2009) szerint úgy definiálhatjuk, mint egy **olyan integrált számítógépes eszközök összessége, amely döntési modellek, adatbázisok és a döntéshozó saját ítélőképességének segítségével interaktív módon nyújt segítséget nem programozható vagy részben programozható döntések meghozatalában** egyedi, speciális problémákban, jobb döntés meghozatal céljából.

[44] Ezek a programok nagy számításigényű algoritmusokat használnak, melyek paraméterezhetők, így viszonylag magas a problémamegoldó képességük.



9.5. ábra: A döntéstámogató rendszerek és problémák struktúrája.
Forrás: (Kacsukné Bruckner, L.; Kiss, T. 2007)

Napjainkban a döntéstámogató rendszerek kibővültek és átalakultak az új technológiák, például a Big Data, a mesterséges intelligencia (AI – *Artificial Intelligence*) és a gépi tanulás (ML – *Machine Learning*) segítségével. Ezek a technológiák lehetővé teszik, hogy a döntéstámogató rendszerek még hatékonyabbak legyenek a nagy mennyiségű adat elemzésében, a trendek felismerésében és a prediktív modellek készítésében. Tehát, hagyományos értelemben nem feltétel az AI és ML technológia használata. A döntéstámogató rendszerek létrehozásához és működtetéséhez minimálisan a következő öt erőforrásra van szükség: **hardver**, **szoftver**, **adatbázisok**, **modellbázisok** és **emberi erőforrások**. Felépítésüket tekintve a döntéstámogató rendszerek minimálisan három részből állnak: **kommunikációs alrendszer** (a kommunikációs alrendszeren értjük mindazokat a hardver- és szoftvereszközöket, melyek a rendszer és a felhasználó közötti kommunikációt megvalósítják), **modellező alrendszer** (lásd tovább) és **adatbázis-kezelő alrendszer**.

A mesterséges intelligenciát és a gépi tanulást (ML – *Machine Learning*) használó technológiák a modern vállalati információs rendszerekben különféle formákban jelenhetnek meg (a döntéstámogató rendszereket nem kivéve). Általában háromféle megközelítést ismerünk.

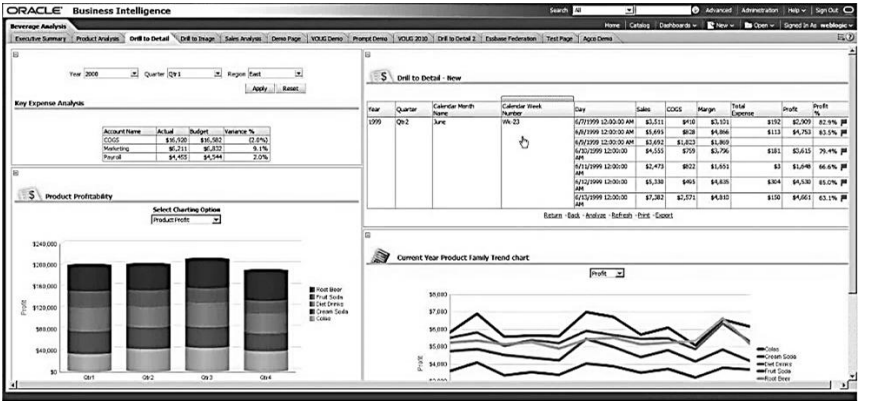
- **Önálló szoftverk –** például, önálló prediktív elemzési eszközök vagy chatbotok, amelyeket a vállalat beépíthet.
- **Integrált modulok –** nagyobb, komplex vállalati információs rendszereken belül fordulnak elő, mint például az ERP-rendszereken (*Enterprise Resource Planning*) vagy a CRM-rendszereken (*Customer Relationship Management*) belül.
- **Felhasználható API-k és szolgáltatások –** főleg felhőalapú szolgáltatások esetében. Ezek a modulok gyakran API-kon keresztül érhetők el.

A döntéstámogató rendszerek manapság arra specializálódnak, hogy analíziseket, vizualizációkat, predikciókat és jelentéseket készítsenek a vezetők számára. A modern döntéstámogató rendszerek egyrészt információt szolgáltatnak a menedzsernek, rendszeres vagy speciális jelentések formájában, nagy mennyiségű adatot kezelve és feldolgozva, másrészt modellezési képességekkel rendelkeznek, különböző matematikai, analitikai modellek segítségével előrejelzéseket, elemzéseket készítenek, javaslatokat tesznek. A döntéstámogató rendszerek is alkalmazkodtak a modern vállalati információs rendszerekhez, melyekben különféle formákban jelenhetnek meg. Ez esetben is három megközelítést ismerünk.

- **Önálló döntéstámogató rendszerek –** olyan dedikált rendszerek, amelyeket kifejezetten döntéstámogatásra fejlesztettek ki. Általában stratégiai döntésekhez használják, gyakran specifikus üzleti problémák megoldására. Nézzünk két példát:
    - **Oracle Hyperion –** egy pénzügyi döntéstámogató eszköz.
        - Önálló pénzügyi elemző szoftver, amely az Oracle ERP rendszerekkel integrálható, de önállóan is használható.
        - Az Oracle Hyperion segítségével részletes pénzügyi elemzések végezhetők. Könnyen integrálható a vállalati adatokkal.
    - **Power BI –** adatvizualizációs döntéstámogató eszköz.
        - Önálló üzleti intelligencia szoftver, amely vizualizálja az adatokat.
        - Alkalmas nagy adatmennyiségek gyors feldolgozására és vizualizációjára. Integrálhatóak más adatforrásokkal is, pl. Excellel, felhőalapú adatbázisokkal, stb.
- **Integrált döntéstámogató modulok –** nagyobb vállalati információs rendszerek, például ERP- vagy CRM-rendszerek, gyakran tartalmaznak döntéstámogató funkciókat integrált modulokként. Integrált döntéstámogató modul esetén az elemzés közvetlenül a meglévő adatokon történik, amelyeket már a vállalat tárol és használ a napi működés során. Mutatunk két példát:
    - **IBM Cognos Analytics –** integrált vállalati intelligencia (BI – *Business Intelligence*) és döntéstámogató rendszer.
        - Részben önálló, de könnyen integrálható más vállalati információs rendszerekkel, például ERP vagy CRM rendszerekkel.

  - Mesterséges intelligencia alapú funkciókkal is rendelkezik, például adatok automatizált elemzésével és előrejelzésekkel.
- **SAP BusinessObjects –** integrált döntéstámogató modul a SAP ERP rendszerben.
  - A SAP rendszeren belül elérhető döntéstámogató modul.
  - A SAP BusinessObjects segítségével az ERP rendszer adatait gyors elemzésekre és jelentések készítésére lehet használni.

- **Hibrid megoldások –** bizonyos esetekben a DSS rendszerek felhőalapú szolgáltatások formájában működnek, és API-kon keresztül kapcsolódnak a vállalat rendszereihez. Így ezek részben önállóak, de közvetlenül integrálódnak is a vállalati adatfolyamokba. Nézzünk egy példát:
  - **Qlik Sense –** felhőalapú döntéstámogató és BI platform.
    - Önálló, felhőalapú döntéstámogató rendszer, amelyet más vállalati rendszerekkel is összekapcsolhatnak API-kon keresztül.
    - Nagyon jó vizualizációs képességekkel rendelkezik.



9.6. ábra: Példa az Oracle BI Hyperion rendszerre.
Forrás: https://images.app.goo.gl/9WPbv5ugRBj7hvec9

A gyakorlati problémák során nagyon sokszor előfordul, hogy a döntéseket nem egyetlen személynek kell meghoznia, hanem több ember együttes munkájának és konszenzusának eredményeként jönnek létre. A döntéstámogató rendszereket eredetileg az egyéni döntéshozatal

támogatására fejlesztették ki. Csak később, a 80-as évek vége felé jelentek meg olyan rendszerek, melyek már a csoportos döntések meghozatalában is segítséget nyújtanak. Ezen eszközök a **csoportos döntéstámogató** rendszerek (GDSS – *Group Decision Support System*). A GDSS-ek különböznek az egyéni DSS-ektől abban, hogy eszközöket és funkciókat kínálnak a közös munkához és a csoportos döntéshozatal támogatásához. A GDSS-ket olyan helyzetekben alkalmazzák, ahol több résztvevő különböző szaktudással és perspektívával járul hozzá a döntéshozatali folyamathoz, és fontos, hogy a végső döntés kollektív megegyezésen alapuljon. A következőkre gondolunk:

- stratégiai üzleti döntések esetén, ahol több menedzsernek kell konszenzusra jutnia;
- projektkiválasztási folyamatokban, ahol a döntéshozatal során sok szempontot és érdeket kell figyelembe venni;
- válságkezelésben vagy krízis helyzetekben, ahol gyors és együttműködő döntéshozatal szükséges.

Tehát a csoportos DSS (GDSS) nem csak a több menedzsertől való tanulásra épül, hanem egy olyan rendszer, amely a csoportos döntéshozatalt aktívan támogatja az együttműködési eszközök és struktúrák révén. A csoportos döntéshozatal egyik nagyon fontos eleme a résztvevők közötti **kommunikáció**. A GDSS-nek tehát, a hagyományos döntéstámogató rendszerek már megismert képességei mellett rendelkeznie kell a kommunikáció kialakítását és megkönnyítését elősegítő eszközökkel is. A csoportos döntéstámogató rendszerek környezeti kialakítása a döntéshozó csoport térbeli és időbeli elhelyezkedésétől függ. Ennek megfelelően négy különböző típusú GDSS-t különböztetünk meg. Ezek az alábbiak.

- A döntéshozók **azonos időben azonos helyen tartózkodnak**. Ennek jellegzetes megvalósítása a **döntési szoba**, ahol lehetőség nyílik arra, hogy a döntéshozók személyesen is találkozzanak. A szoba felszereltségéhez tartoznak a helyi hálózatba kapcsolt számítógépek, melyeken keresztül mindenki beviheti az ötleteit. A brainstorming itt hatékonyan alkalmazható. Szintén elterjedtek a szavazóprogramok. Egy ilyen szoba lehet a tárgyalóterem.
- A döntéshozók **azonos időben hozzák a döntést, de különböző helyen** tartózkodnak. Ennek jellegzetes megvalósítása a **távkonferencia.**
- A döntéshozók **különböző időpontokban, de azonos helyen** tartózkodnak. Ez például úgy fordulhat elő, ha különböző műszakokban, különböző időbeosztással folyik a munka. Kommunikációra az **e-mail** is megfelelő, de fejlettebb

eszközökkel, például **helyi döntési hálózattal**, még hatékonyabb az adatmegosztás, ahol lehetőség van közös programok futtatására, egymás adatainak elérésére.

- A döntéshozók **különböző időben hozzák a döntést, és különböző helyen** is tartózkodnak. Nagy multinacionális vállalatoknál fordul ez gyakran elő, ahol a föld külömböző részein dolgozó leányvállalatok döntéshozói együttműködnek. A döntéshozási helyszíneket **védett hálózattal** összekötve hasonló szintű együttműködés érhető el, mint az azonos helyszínen dolgozók esetében.

A GDSS-ek képesek a döntési folyamat strukturálására, például lépésről lépésre vezetni a csoportot a probléma definíciójától kezdve az alternatívák kidolgozásán át egészen a végső döntésig. Ahogyan a döntési szobáknál is megemlítettük, a GDSS-ek valóban szavazási és értékelési mechanizmusokat is tartalmaznak a konszenzus elősegítésére, és a GDSS-ek gyakran tartalmaznak olyan szavazási vagy értékelési mechanizmusokat, amelyek lehetővé teszik a csoport tagjainak, hogy ha kell, akkor anonim módon is kifejezhessék a preferenciáikat, és könnyebben konszenzusra jussanak. A GDSS hátrányát a magas költségek jelentik, amik hosszú ideig limitálták a felhasználását. A technológiai fejlődés és a költségek csökkenése miatt a csoportos döntéstámogató rendszerek mára széles körben elérhetővé váltak.
A modern GDSS-ek különféle interaktív eszközöket is kínálnak, mint például a valós idejű dokumentumszerkesztés és vizuális elemzési eszközök. Itt is manapság jelentősen jelen van az AI és felhőalapú megoldások, melyek lehetővé teszik a GDSS-ek számára a globális hozzáférést. A mesterséges intelligencia és a gépi tanulás javíthatják a döntéstámogató rendszerek képességeit, például az adatfeldolgozás és az előrejelzések terén. A mobil eszközök integrálása lehetővé teszi a döntéshozók számára, hogy bárhol és bármikor részt vegyenek a folyamatban. A GDSS kihívásai közé tartozhatnak a kulturális különbségek, a felhasználók hajlandósága, képessége, hogyan tudják alkalmazni az új technológiákat. Nemzetközi csoportok esetén a kulturális eltérések befolyásolhatják a kommunikációt és a döntéshozatalt. Például, az IBM GDSS-ket használnak a globális csapatok közötti együttműködésre. A jövőbeli fejlődés valószínűleg az integráció felé fog haladni, más rendszerek (ERP vagy CRM), virtuális és kiterjesztett valóság, valamint szociális hálózatok irányába.
A GDSS rendszerek **elsődlegesen a közös munkára és a csoportos döntéshozatalra összpontosítanak**, nem pedig a hagyományos értelemben vett üzleti intelligencia (lásd a következő fejezeteket) eszközeire. A GDSS

rendszerek tartalmaznak elemző és vizualizációs eszközöket is, amelyek segítik a döntési folyamatot, de a **fő fókuszuk a csoport tagjai közötti kommunikáció és együttműködés támogatása**, mint pl. a Google Workspace (Meet, Drive, Docs), az IBM Planning Analytics vagy Decision Lens. A **Decision Lens** egy kifejezetten csoportos döntéshozatalra és priorizálásra fejlesztett GDSS, amely különböző döntési kritériumok és prioritások alapján segíti a döntéshozókat. A szoftver lehetővé teszi a döntések mérését, szavazást és több szempont figyelembevételét, amelyek mind hozzájárulnak az objektív és átlátható csoportos döntéshozatalhoz.
A GDSS rendszerek tehát támogatják a csoportos döntéshozatalt, növelik az együttműködési hatékonyságot és megkönnyítik a konszenzusra jutást fontos döntésekben. (Antal, P. 2021) (Kacsukné Bruckner, L.; Kiss, T. 2007)

9.5 Alapvető különbség a vezetői információs rendszer és döntéstámogató rendszer között

A vezetői és döntéstámogató rendszerek közötti különbségeket a következő 9.3. táblázat foglalja össze.

9.3. táblázat: Különbségek a vezetői és döntéstámogató rendszerek között

Vezetői információs rendszerek (MIS)	Döntéstámogató rendszerek (DSS)
Főleg operatív, részben taktikai szintű döntések esetén használják.	Főleg taktikai, részben stratégiai szintű döntések esetén használják.
Strukturált, programozható döntéseknél hasznos	Főleg félig strukturált problémák esetén hasznos.
A döntéshozatali folyamatban a feladat-meghatározás és adatgyűjtés, valamint a megvalósítás fázisát támogatja.	A döntéshozatali folyamat mind a négy lépését – feladat meghatározása és adatgyűjtés, tervezés, választás, megvalósítás – támogatja.
Általános, gyakran előforduló problémákhoz nyújt segítséget, személytől függetlenül.	Egyedi problémák megoldásában segít, az adott döntéshozó stílusához és igényeihez igazodva.
Előre definiált formátumú jelentéseket készít periodikusan, kivételes események bekövetkezésekor vagy igény szerint.	Interaktív lekérdezések és válaszok rugalmasan módosítható formátumban, a döntéshozó igényei szerint.
A szolgáltatott információt a vállalat belső adatbázisában meglévő adatok	Az információt belső és külső adatbázisok, valamint matematikai,

átalakításával, feldolgozásával, csoportosításával állítja elő.	analitikai modellek és szimuláció segítségével állítja elő.
Az egész vállalat vagy egy funkcionális terület általános teljesítményéről szolgáltat információt.	Egy-egy speciális, egyedi problémával kapcsolatos információt szolgáltat.
Kimenetük főleg nyomtatott jelentések.	Jellemzően képernyőre szolgáltatott eredményeket, igény szerinti nyomtatást biztosít.
A jelentések nehezen áttekinthető formátumúak.	A jelentések könnyen áttekinthető, felhasználóbarát, grafikus, táblázatos formátumúak.
Merev és nehezen módosítható rendszerről beszélünk.	Rugalmasan, a felhasználó által módosítható és fejleszthető rendszerről beszélünk.

Forrás: (Kacsukné Bruckner, L.; Kiss, T. 2007)

A DSS a MIS rendszerekkel szemben összetettebb elemzési és modellezési eszközökkel segíti a stratégiai és nem rutin jellegű döntések meghozatalát, lehetővé téve a „mi lenne, ha” típusú forgatókönyvek elemzését. Az MIS célja a szervezet napi működésének támogatása, ellenben a DSS a döntéshozók számára nyújt mélyebb elemzéseket és alternatív javaslatokat a döntések meghozatalában. (Antal, P. 2021)

Befejezés

Évekkel ezelőtt Baťa, Ford vagy akár Philips is, és sok más nagycégtulajdonos felismerte, hogy a sikeres döntéshozatal alapja a megfelelő információk birtoklása és azok hatékony felhasználásában rejlik. Napjainkban a vállalatok nagy mennyiségű, akár strukturált vagy strukturálatlan adatokat kezelnek, melyek elemzéséhez, feldolgozásához elengedhetetlenek, és ebben kulcsszerepet is játszanak az információs rendszerek. Ezek a rendszerek már nem csak támogató eszközök, hanem a vállalkozás elengedhetetlen elemei is egyben. A mai üzleti világban használatuk néélkül már nem lehet versenyképesnek maradni .

Nem telt el száz év azóta, amióta a Howard Aiken az IBM-mel együtt egy közös fejlesztéssel megalkották az első elektromechanikus számítógépek képviselőjét, a Mark I-et (1944). Azóta ez a technológiai fejlődés nem lassult le, sőt, egyre gyorsabb fordulatszámot vett. A hagyományos üzleti modellek alapoktól átalakultak, és a vállalati információs rendszerek pótolhatatlan részévé váltak a modern vállalkozások működésének. A globalizáció gyorsuló üteme a vállalati szférában is egyre erősebben érezteti hatását. A jövőben kulcsfontosságú lesz az a tudás, amely összhangban az információs és kommunikációs technológiák lehetőségeivel alapot biztosít a vállalkozáshoz.

A 2000-es évek elején egy vállalatvezető számára elegendő volt, hogy egy Excel-táblázatban vezesse az üzleti adatokat. Ma már egy ilyen elképzelés teljesen idejét múlt. Ebben a tankönyvben arra törekedtünk, hogy ne csak az alapfogalmakat, a működési elveket és a terminológiát mutassuk be, hanem mélyebben is foglalkozzunk, többek között olyan fontos rendszerekkel, mint az integrált vállalatirányítási rendszerek, vállalati intelligenciát támogató rendszerek, vezetői információs rendszerek, döntéstámogató rendszerek vagy a szakértői rendszerek. Bemutattuk az új technológiai trendek és irányvonalakat is, mint például az e-business, felhő alapú megoldások vagy a mesterséges intelligencia és prediktív analitika alkalmazásait, melyek nemcsak a nagyvállalatok kiváltsága, hanem egyre inkább a kis- és középvállalkozások számára is elérhető eszközök. Mindemellett nem feledkeztünk el az információs rendszerek biztonságáról sem, a különböző információs rendszerek közötti integrációról, metaadatok és vállalati dokumentumok kezeléséről, és

bemutattuk az ilyen rendszerek fejlesztésének és karbantartásának hátterét is.

A digitális korszak kihívásai és lehetőségei előtt állunk. Talán most érthetjük meg igazán a technológia, valamint az információs rendszerek alkalmazásainak valódi értékét és fontosságát. Bízunk benne, hogy ezzel a tankönyvvel nemcsak elméleti tudást adtunk át, hanem inspiráltuk és motiváltuk az olvasót, és felkeltettük az érdeklődését a vállalati információs rendszerek irán. Bízunk benne, hogy az itt megszerzett tudást majd a gyakorlatban is sikerrel tudja kamatoztatni.

Ezúton ajánljuk a kedves olvasó figyelmébe a tankönyv 2. kötetét: Vállalati információs rendszerek II címen, amely ennek a tankönyvnek a folytatása és a Vállalati információs rendszerek tantárgy tartalmi töltetének szerves részét alkotja.

Végezetül, Peter Drucker idézetével búcsúzunk az olvasótól. Drucker pragmatikus hozzáállásával már a tankönyv elején is találkozhattunk, akinek a nézetei rendkívül jól illeszkednek az információs rendszerekhez.

„A legnagyobb veszély a változó időkben nem maga a változás, hanem az, ha a tegnapi logikával próbálunk cselekedni.”

Szerzők

Felhasznált irodalom

1. ACKOFF, R. L. 1999. Re-Creating the Corporation: A Design of Organizations for the 21st Century. Oxford: Oxford University Press. ISBN-10: 9780195123876, ISBN-13: 978-0195123876.
2. Adobe. (update: 4.6.2023). Data warehouse vs. database — key differences explained. Elérhető interneten: Adobe.com, *https:// business.adobe.com/blog/basics/data-warehouse-vs-database.*
3. ANTAL, P. 2014. Valószínűségi döntéstámogató rendszerek. Budapest: Typotex Kiadó. ISBN: 978-963-279-184-5.
4. BAJOR, T. 2017. Szoftverfejlesztés lépései. Elérhető interneten: Programozz Te Is!, *https://programozzteis.hu/ blog/szoftverfejlesztes-lepesei/*.
5. BASKARADA, S.; KORONIOS, A. 2013. Data, Information, Knowledge, Wisdom (DIKW): A Semiotic Theoretical and Empirical Exploration of the Hierarchy and its Quality Dimension. Australasian Journal of Information Systems. 18. DOI: 10.3127/ajis.v18i1.748.
6. BASL, J.; BLAŽÍČEK, R. 2012. Podnikové informační systémy: podnik v informační společnosti. 3., aktualiz. a dopl. vyd. Praha: Grada, 323 s. Management v informační společnosti. ISBN 978-80-247-4307-3.
7. BASU, R. 2015. Managing Quality in Projects. Routledge. DOI: 10.4324/9781315249872.
8. BEDNÁRIK, É.; PAKAINÉ KOVÁTS, J. Strukturált problémamegoldó technikák. Sopron: Nyugat-magyarországi Egyetem Kiadó. ISBN 978-963-334-269-5. Elérhető interneten: *https://lignogroup.hu/wp-content/uploads/2020/02/Bednarik-Pakaine_strukturalt_problemamegoldo_technikak_165x235_email.pdf*.
9. BENCSÍK, A.; TÓBIÁS KOSÁR, S. 2016. Tudásmenedszment. Komárno. Selye János Egyetem. ISBN 978-80-8122-179-8.
10. BERECZ; A.; KARÁCSONY, P.; KÓNYA, L.; PECK T, T.; SZÁSZ, G.; VÁRI-KAKAS, I. 2019. Informatikai alapok. Budapest: Gábor Dénes Főiskola.
11. BONTIS, N. (2002). Assessing Knowledge Assets: Knowledge-Based Value Creation. Routledge.

12. BÖDŐK, ZS. 2005. Magyar feltalálók a távközlés történetében. Dunaszerdahely: NAP kiadó. ISBN 80-89032-64-8. Elérhető interneten: magyar elektronikus könyvtár, https://mek.oszk.hu/04500/04500/04500.pdf.
13. BRUCKNER, T.; VOŘÍŠEK, J.; BUCHALCEVOVÁ, A. 2012. Tvorba informačních systémů: principy, metodiky, architektury. 1. vyd. Praha: Grada, 357 s. Management v informační společnosti. ISBN 978-80-247-4153-6.
14. BRYNJOLFSSON, E.; MCELHERAN, K. (2016). The Rapid Adoption of Data-Driven Decision-Making. In: American Economic Review 106 (5): 133–39. ISSN 0002-8282. DOI: 10.1257/aer.p20161016.
15. BUJDOSÓ, G. Bevezetés az SQL-be. Elérhető interneten: Debreceni Egyetem, *https://arato.inf.unideb.hu/bujdoso.gyongyi/kurzusok/sql/bevsql.pdf*.
16. CARDON, D. (20.11.2024). Database vs Data Warehouse: A Comparative Review. Elérhető interneten: Healthcatalyst.com, *https://www.healthcatalyst.com/learn/insights/database-vs-data-warehouse-a-comparative-review*.
17. CHLAPEK, Dušan et al. 2014. Otevřená a propojitelná data – metodiky, postupy, nástroje a praxe. In: DATAKON. Ostrava: Vysoká škola báňská Technická univerzita. ISBN 978-80-248-3545-7.
18. COURSERA. (update: 16.11.2023). Data Warehouse vs. Database: What's the Difference? Elérhető interneten: Cousera.org, *https://www.coursera.org/articles/data-warehouse-vs-database*.
19. Csontos, R.; Szabó, R. 2019. Hálózati tanulás – tanuló hálózatok. Vezetéstudomány - Budapest Management Review. 50. 10.14267/VEZTUD.2019.01.01.
20. DAS, T. 2024. SMED Techniques for Rapid Setup Time Reduction in Electronics Industry. In: Journal of Scientific and Engineering Research. 11 (4): 257-269. ISSN 2394-2630. DOI: 10.5281/zenodo.11311488.
21. DECISION L. (16. 11. 2024). Modernizing How Government Prioritizes, Plans, and Funds. Elérhető interneten: Decisionlens.com, *https://www.decisionlens.com/*.
22. DOMSCHITZ M., RUZSA É. 2013. A szervezetfejlesztés és a tanuló szervezetek. Elérhető interneten: Széchenyi István Egyetem, *https://kti.uni-nke.hu/document/vtkk-uni-nke-hu/a-szervezetfejlesztes-es-tanulo-szervezetek.original.pdf*.
23. DOUCEK, P.; NOVOTNÝ, J.; PECÁKOVÁ, I.; VOŘÍŠEK, J. 2007. Lidské zdroje v ICT – Analýza nabídky a poptávky po IT

odborníc ích v ČR. 1. vyd. Praha: PROFESSIONAL PUBLISHING, 2007. 202 s. ISBN 978-80-86946-51-1.
24. DRUCKER, P. F. 1998. The Coming of the New Organization. Harvard Business Review on Knowledge Management, Harvard Business School Press, 1998.
25. Erwig, M. 2020. Disaster Recovery Planning: Ensuring Business Continuity & System Availability. McGraw-Hill.
26. GÁLA, L.; POUR, J.; ŠEDIVÁ, Z. 2009. Podniková informatika. Praha: Grada Publishing. ISBN 978-80-247-2615-1.
27. GÁLA, L.; POUR, J.; ŠEDIVÁ, Z. 2015. Podniková informatika. Praha: Grada Publishing. ISBN 978-80-247-5457-7.
28. GOMES da SILVA, J.; LIRA de SOUZA, I., J. 2019. How to compare energy efficiency inspection tests on TVs? In: International Journal for Innovation Education and Research. 7 (11): 1368-1393. ISSN 1368-1393. DOI: 10.31686/ijier.Vol7.Iss11.1946.
29. GOSH, P. (20.11.2024). Data Warehouse vs. Database. Elérhető interneten: Dataversity.net, *https://www.dataversity.net/data-warehouse-vs-database/*.
30. HANÁČEK, P., STAUDEK, J. 2000. Bezpečnost informačních systémů. Metodická příručka zabezpečování produktů a systémů budovaných na bázi informačních technologií, Úřad pro státní informační systém.
31. HARTINGER, D. (04.11.2024). Úvod do UML. Elérhető interneten: ITnetwork.sk, *https://www.itnetwork.sk/navrh/uml/uml-uvod-historia-vyznam-a-diagramy/uml/uml-uvod-historie-vyznam-a-diagramy*.
32. HERDON, M. 2016. Informatika. Debrecen: Debreceni Egyetem. Elérhető interneten: *http://users.atw.hu/akiram77/info1/j2.pdf*.
33. HOSŤOVECKÝ, M. 2013. Operačné systémy a počítačová architektúra. Trnava: Pedagogická fakulta Trnavská univerzita v Trnave. ISBN 978-80-8082-683-3. Elérhető interneten: *https:// pdfweb.truni.sk/e-ucebnice/ospa/*.
34. HUMBLE, J.; FARLEY, D. 2010. Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation. Addison-Wesley. ISBN 978-0321601919.
35. JAISWAR, A. 2024. Data Warehouse: Architecture Overview. In: TechRxiv. DOI:10.36227/techrxiv.171779354.46288271/v2. Elérhető interneten: TechRvix.org, *https://www.techrxiv.org/users/791577/articles/1072655-data-warehouse-architecture-overview*.
36. JUHÁSZ, S. 2011. Vállalati információs rendszerek műszaki alapjai. Bicske: SZAK kiadó kft. ISBN: 978-963-9863-22-4.

37. KACSUKNÉ BRUCKNER, L.; KISS, T. 2007. Bevezetés az üzleti informatikába. Budapest: Akadémiai Kiadó Zrt. ISBN: 978-963-05-8460-9.
38. KACSUKNÉ BRUCKNER, L.; KISS, T. 2009. Bevezetés az üzleti informatikába – Amit a sikeres üzletembernek tudni kell. Budapest: Akadémiai Kiadó Zrt. ISBN 978-963-05-8460-9.
39. KIM, G.; HUMBLE, J.; DEBOIS, P.; WILLIS, J. 2016. The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations. IT Revolution. ISBN 978-1942788003.
40. KOLOSZÁR, L. 2013. Vállalati információs rendszerek. Sopron: Nyugat-magyarországi Egyetem kiadó. ISBN 978-963-334-121-6.
41. KOMLÓ, Cs. 2013. Információs rendszerek tervezésének módszertana. Eger: Eszterházy Károly Főiskola nyomda. Elérhető interneten: *https://www.webler.hu/tananyag/informaciosrendszerek.pdf*.
42. KUČERA, M.; FIĽA, M.; LÁTEČKOVÁ, A.; STUCHLÝ, P. 2017. Podnikové informačné systémy. Nitra: Slovenská poľnohospodárska univerzita v Nitre. ISBN: 978-80-552-1723-9.
43. KUNSTOVÁ, R. 2007. Návrh modelu řízení ECM v kontextu řízení informatiky. Disszertációs munka. Praha: Vysoká škola ekonomická v Praze.
44. Microsoft. (19. 11. 2024). Az online elemzési feldolgozás (OLAP) áttekintése. Elérhető interneten: Microsoft.com, *https://support.microsoft.com/hu-hu/office/az-online-elemz%C3%A9si-feldolgoz%C3%A1s-olap-%C3%A1ttekint%C3%A9se-15d2cdde-f70b-4277-b009-ed732b75fdd6*.
45. Microsoft1. (update: 28.9.2024). Metaadat-szabványok. Elérhető interneten: Microsoft.com, *https://learn.microsoft.com/hu-hu/azure/cloud-adoption-framework/scenarios/cloud-scale-analytics/govern-metadata-standards.*
46. Microsoft2. (update: 4.10.2023). Memóriabeli adatbázisok. Elérhető interneten: Microsoft.com, *https://learn.microsoft.com/hu-hu/dotnet/standard/data/sqlite/in-memory-databases*.
47. Microsoft3. (23.11.2024). Számítógépvírusok: ismertetés, megelőzés és helyreállítás. Elérhető interneten: Microsoft.com, *https://support.microsoft.com/hu-hu/topic/sz%C3%A1m%C3%ADt%C3%B3g%C3%A9pv%C3%ADrusok-ismertet%C3%A9s-megel%C5%91z%C3%A9s-%C3%A9s-helyre%C3%A1ll%C3%ADt%C3%A1s-53dc9904-0baf-5150-6e9a-e6a8d6fa0cb5*.

48. Microsoft4. (23.11.2024). Mik azok a kártevő szoftverek? Elérhető interneten: Microsoft.com, *https://www.microsoft.com/hu-hu/security/business/security-101/what-is-malware*.
49. Microsoft5. (23.11.2024). ActiveX-vezérlők használata az Internet Explorer 11-ben. Elérhető interneten: Microsoft.com, *https://support.microsoft.com/hu-hu/windows/activex-vez%C3%A9rl%C5%91k-haszn%C3%A1lata-az-internet-explorer-11-ben-25738d05-d357-39b4-eb2f-fdd074bbf347*.
50. MIŠÚT, M. 2013. IKT vo vzdelávaní. Trnava: Trnavská univerzita v Trnave. ISBN 978-80-8082-695-6. Elérhető interneten: *https://pdfweb.truni.sk/e-ucebnice/iktv/.*
51. MIŠÚT, M.; JUHÁSOVÁ, L. 2014. Databázové systémy 2. Trnava: Trnavská univerzita v Trnave. ISBN 978-80-8082-775-5. Elérhető interneten: *https://pdfweb.truni.sk/e-ucebnice/databazove-systemy2/.*
52. MOLNÁR, Z. 2012. Competitive intelligence, aneb, jak získat konkurenční výhodu. Vyd. 1. Praha: Oeconomica, 254 s. Odborná kniha s vědeckou redakcí. ISBN 978-80-245-1908-1.
53. MongoDB. (20.11.2024). Databases vs. Data Warehouses vs. Data Lakes. Elérhető interneten: Mongodb.com, *https://www.mongodb.com/resources/basics/databases/data-lake-vs-data-warehouse-vs-database.*
54. NOVOTNÝ, O.; POUR, J.; SLÁNSKÝ, D. 2005. Business Intelligence. Praha: Grada Publishing, a. s. ISBN 80-247-1094-3.
55. ORACLE. (27.11.2024). Mire szolgál az ERP? Elérhető interneten: Oracle.com, *https://www.oracle.com/hu/erp/what-is-erp/*.
56. PAVLÍČEK, A.; SYROVÁTKOVÁ, J. 2022. Základy moderní informatiky. Průhonice: Professional Publishing s.r.o. ISBN 978-80-88260-59-2.
57. POKORNÝ, J. 2014. Big Data: jejich ukládání, zpracování a použití. In: DATAKON. Ostrava: Vysoká škola báňská – Technická univerzita. ISBN 978-80-248-3545-7.
58. PRESTON, W. C. 2021. Modern Data Protection: Ensuring Recoverability of All Modern Workloads. O'Reilly Media. ISBMN: 978-1492094050.
59. PRIBILOVÁ, K. 2013. Databázové systémy 1. Trnava: Trnavská univerzita v Trnave. ISBN 978-80-8082-680-2. Elérhető interneten: *https://pdfweb.truni.sk/e-ucebnice/databazove-systemy1/.*
60. PROCHÁZKA, P. (17. 11. 2024). PPP - PODCAST PAVEL PROCHÁZKA. Elérhető interneten: YouTube,

https://www.youtube.com/@podcastpavelprochazka/featured; https://www.youtube.com/shorts/p66FbZJycxU.

61. PYZDEK, T.; KELLER, P. 2014. The Six Sigma Handbook (5th ed.). McGraw-Hill. ISBN: 978-007-184-0538.
62. REALYVÁSQUEZ, A.; GARCÍA-ALCARAZ, J.; SATAPATHY, S.; CORAZA, D.; LÓPEZ, Y. 2024. The PDCA (Plan-Do-Check-Act) Cycle. DOI: 10.1007/978-3-031-70984-5.
63. ROSICKÝ, A. 2010. Člověk: informace a znalost. Informační management. Praha: Professional Publishing, s. 53-80. ISBN 978-80-7431-010-2.
64. ROUBAL, P. 2012. Informatika a výpočetní technika pro střední školy. Brno. Computer Press. ISBn 978-80-251-3228-9.
65. SAP. (20.11.2024). Mi az adatbányászat? Elérhető interneten: Sap.com, *https://www.sap.com/hungary/products/technology-platform/hana/what-is-data-mining.html.*
66. SAP1. (21.11.2024). Mi az SAP HANA? Elérhető interneten: SAP, *https://www.sap.com/hungary/products/technology-platform/hana/what-is-sap-hana.html*.
67. SAP2. (21.11.2024). Real benefits of an in-memory database. Elérhető interneten: SAP, *https://www.sap.com/westbalkans/insights/in-memory-database.html*.
68. SAP Slovakia. (25. 10. 2024). Čo sú veľké dáta? Elérhető interneten: SAP, *https://www.sap.com/sk/products/technology-platform/what-is-big-data.html*.
69. SIDLÓ, Cs. 2004. Data Warehouse – Adattárház. Elérhető interneten: ELTE.hu, *https://scs.web.elte.hu/Work/DW/adattarhazak.htm*.
70. Simplilearn. (20.11.2024). Difference Between Data Warehouse and Database. Elérhető interneten: Simlilearn.com, *https://www.simplilearn.com/data-warehouse-vs-database-article*.
71. SODOMKA, P.; KLČOVÁ, H. 2010. Informační systémy v podnikové praxi. Brno: Computer Press, a.s. ISBN: 978-80-251-2878-7.
72. Solix. (20.11.2024). Működési adattár. Elérhető interneten: Solix.com, *https://www.solix.com/hu/kb/operational-data-store*.
73. SQUEDA, R. 2018. MicroStrategy Quick Start Guide: Data analytics and visualizations for Business Intelligence. 1st edition. Birmingham: Packt. ISBN-13: 9781789136241.
74. SZABÓ, B.; MÁRFÖLDI, E. 2011. Számítógépes hálózatok. Eger: Eszterházy Károly Főiskola.
75. SZABÓ, M.; SINGER, P.; VARGA, A. 2011. Tanulás hálózatban. Budapest: Oktatáskutató és fejlesztő intézet. ISBN 978-963-682-

690-1. Elérhető interneten: magyar elektronikus könyvtár, *https://mek.oszk.hu/13000/13015/13015.pdf.*
76. SZEGHEGYI, Á. 2018. Döntéstámogató rendszerek. Elérhető interneten: Óbudai Egyetem, *https://old2.kgk.uni-obuda.hu/sites/default/files/DONTESTAMOGATO-RENDSZEREK-BSC-180827.pdf.*
77. SZIRAY, J.; GAUL, G. 2006. Vállalati információs rendszerek I. Győr: Szécsényi István Egyetem.
78. SZIRAY, J.; GAUL, G.; ERDŐS, F. 2006. Vállalati információs rendszerek II. Győr: Szécsényi István Egyetem.
79. SZIRAY, J.; GAUL, G.; ÉGERTNÉ MOLNÁR, É. 2007. Vezetői információs rendszerek. Győr: Szécsényi István Egyetem.
80. ŠTRBO, M. 2020. Modelovanie databázových systémov. Trnava: Trnavská univerzita v Trnave. ISBN 978-80-568-0340-0.
81. TAKÁČ, O. 2017. Modellezés és szimuláció. Komárno: J. Selye University. ISBN: 978-80-8122-203-0.
82. TestIt. (20.11.2024). Adatbányászat / Data mining: mi mindent rejthet az adat? Elérhető interneten: Testit.com, *https://test-it.com/hu/aidata/adatbanyaszat/.*
83. TOBIN, D. (20.11.2024). Data Warehouse vs. Database: 7 Key Differences. Elérhető interneten: Integrate.io, *https://www.integrate.io/blog/data-warehouse-vs-database-what-are-the-key-differences/.*
84. TÖRÖKNÉ KARAKAS, M. 2019. Adatbázis alapok. Elérhető interneten: slideshare.net, *https://www.slideshare.net/slideshow/adatbazis-138714750/138714750#5*.
85. TYRYCHTR, J.; VASILENKO, A. 2015. Business Intelligence in Agribusiness – Fundamental Concepts and Research. Brno: KONVOJ, spol. s r. o. ISBN 978-80-7302-170-2.
86. UNGAR, D.; KIMELMAN, D; ADAMS, S. 2015. Inconsistency Robustness for Scalability in Interactive Concurrent-Update In-Memory MOLAP Cubes. Elérhető interneten: Researchgate, *https://www.researchgate.net/publication/266442745_Inconsistency_Robustness_for_Scalability_in_Interactive_Concurrent-Update_In-Memory_MOLAP_Cubes*.
87. VYCHODIL, T. (25.11.2024). Kedy je vhodné zaviesť do firmy ERP systém l ERP / CRM Odoo. Elérhető interneten: 26House, https://www.youtube.com/watch?v=Pmrw_SmziIA
88. W3Schools. (25. 10. 2024). SQL Tutorial. Elérhető interneten: SQL, *https://www.w3schools.com/sql/default.asp*.

Tárgymutató

A

A döntési funkció 40
A prediktív funkció 40
Adat 25, 34
Adat vezérelt döntéstámogató rendszerekr 227
adatbázis 114
Adatbázis-kezelő rendszer 114
Adatbázisok 113
Adatbiztonság 53
Adatreplikáció 82
Ad-hoc karbantartás 184
Agilis módszerek 181
Aktív CI 79
Alapszoftver 95
Alkalmazásportfólió 150
Alkalmazásszoftver 101
Alkalmazásvirtualizáció 98
All-in-One ERP rendszerek 202
alrendszerek 139
Általános rendszerek 44
Analitikai CRM 208
API 99, 112
Átlagos hozam 175
Automatizált telepítés 171

B

BaaS 152
BAM 134
Bat'a 27
Befektetésarányos megtérülés 175
Belső megtérülési ráta 176
Best-Effort 110
Best-of-Breed ERP rendszerek 202
Big Data 74
Big Data 3V 76
Big Data 5V 76
Bit 25
Blockchain 82
Blokklánc 82
Blue-Green telepítés 171
Bölcsesség 35
BPaaS 152
BPM 134
BPR 138
Business Activity Monitoring 134
BYOC 72
BYOD 71

C

CaaS 152
CASE 165
CDN 113
Cloud Computing 151
Cold Failover 187
Communication as a Service 152
Computer Aided Systems Engineering 165
Continuous Delivery 170
Continuous Deployment 170
Continuous Integration 169
Cookie 82
CPU 89
CRM 45, 101, 136, 149, 207
CRM All-in-One 210
CRM Best-of-Breed 210
Csillag topológia 104

D

DaaS 152
DApps 82
Data Control Language 120
Data Definition Language 120
Data Manipulation Language 119
DBMS 114

DCL....120
Dekompozíció....132
Déli híd....90
Deploy....171
Desktop as a Service....152
determinisztikus rendszereket....29
DevOps....169
DHCP....111
Diagnosztikai funkció....40
DIKW....33
dinamikus rendszer....29
Diszkontált költségek....175
DML....119
DNS-szerver....110
Döntéstámogató rendszer....226
Drucker....27

E

EDI....206
Egyed kapcsolt diagram....167
Egyedi alkalmazás....155
Egyenrangú hálózat....105
EIS....143
Elosztott adatbázisok....124
Emuláció....97
End-User License Agreement....157
Entitásintegritás....118
entrópia....25
ERP 44, 101, 135, 151, 194, 198, 201
ERP jellemzők....203
Eseményalapú karbantartás....184
Északi híd....90
EULA....157
Éves átlagos költség....175
Evolúciós rendszerfejlesztési modell 191
Extranet....108
eXtreme Programming....190

F

Fa topológia....105
Failover megoldás....186
Fájl – Szerver....106
Fájl rendszer....113
Feature Engineering....223
Fehér doboz....157
Fekete doboz....156
felső vezetés....38
feltételes törzsadatok....36
Folyamat....130
Folyamatirányítás....133
Folyamatmenedzsment....134
forródrót-szolgáltatás....161
Freeware....94
FTP....112

G

Garanciák....159
Global Area Network....103
GPU....90, 91
Grafikus kártya....91
Gyors Alkalmazásfejlesztés....190
Gyűrű topológia....104

H

HaaS....152
Háló topológia....105
hálós adatbázismodell....116
Hálózati hardver....103
Hálózati kártya....91
Hálózati OrgWare....104
Hálózati szoftver....103
Hálózati topológia....103
Hangkártya....91
Hardver....85
Harvard architektúra....93
HCM....211
HDD....90
Hibrid felhő....152
Hibrid rendszerek....46
Hibrid topológia....105
Hideg redundancia....186
hierarchikus adatbázis....116
Hivatkozási integritás....118
Holisztikus megközelítés....135
Hoszt – Terminál....106
Hot Failover....187
HTML....111
HTTPS....113

I

IaaS....151
I-CASE....166
IDE....171
IDS/IPS....113
információ....24
Információ....24, 34
információs hierarchiának....34

Infrastructure as a Service 151
In-memory adatbázisok 126
Integrált CASE 166
Integrált döntéstámogató modulok ...229
integrált vállalatirányitási rendszer...194
Internet 109
Intranet 107
Ipar 1.0 56
Ipar 2.0 56
Ipar 3.0 56
Ipar 4.0 55
Ipari automatizálás.....55

J

jelentések és elemzések 37
JSON 121

K

Karbantartás 183
Kisvállalati ERP 202
Kliens – Szerver 106
kompetitív intelligencia 79
Kompozíció 132
Kooperatív CRM 208
Korlátozások.....172
Középvállalati ERP 202
középvezetés.....38
Közösségi felhő 152
KPI 134

L

Laptop 86
Lean metodológia 139
Lite CRM.....211
Lite ERP rendszerek 202
Local Area Network 102
Log mentés 185
Logikai szál 92
Lower CASE 166

M

MaaS 152
Magyarázó funkció.....40
Machine-to-Machine 71
mainframe.....87
Management information System.....136
Meleg redundancia 186
Merevlemez.....90
Metropolitan Area Network.....102
Middleware.....98
MIS.....136, 233
Modell vezérelt döntéstámogató rendszer 227
Modellezés 219
Modelltár 225
moduláris ERP.....203
Moduláris felépítés 43
modularitás 68
monolitikus ERP.....203
Multimédia táblagép.....86
Multithread 93
Munkaállomás 87

N

NaaS 152
Nagyobb méretű karbantartás.....184
Nagyszámítógépek 87
Nagyvállalati ERP 202
negatív visszacsatolás.....266
Nem programozható döntések 212
netbook.....86
Nettó jelenérték 176
NoSQL adatbázisok.....127
Növekményes mentés.....185
Nyílt forráskódú ERP 203
Nyílt forráskódú rendszerek 46
Nyilvános felhő 152

O

ODBC.....100
Odoo.....196, 199
OLRT 143
On Demand 157
On Line Real Time 143
On Premise 157

Ö

Önálló döntéstámogató rendszer.....229

O

Open Data.....78
Open Source Software 157

operációs rendszer 95
Operatív CRM 208
operatív irányítás 38
Oracle Hyperion 229

P

PaaS....151
Passzív CI....79
PDA....86
PDCA 136
Peer to Peer....106
Personal Area Network....102
Pipelining 93
Platform as a Service 151
PLM....149, 211
Porter modell 60
Pourbaix 27
Power BI....229
pozitív visszacsatolás....266
Pragmatikai nézet 27
Prediktív karbantartás....184
Privát felhő 152
Processzor....89
Product Lifecycle Management 211
Produkciós környezet 170
Programozható döntések....212

Q

QASIDA ERP....195, 196

R

RAM....90
Rational Unified Process....191
Redis....126
Redundáns rendszer....186
referenciaadatok 36
Relációs adatbázis 117
rendszer infrastruktúra....66
REST....122
Részleges mentés....185
Roll-back terv 187
RUP....191
Rutin karbantartás....183

S

SaaS....151
SaaS rendszerek....46
SCM 45, 101, 135, 149, 205
SCOR 138
SCOR modell 206
Service-Oriented Architecture....145
Shannon....24
Sín topológia....104
SMTP 112
SOA....145
SOAP....122
Social CRM 209
Software as a Service....151
Specializált rendszerek 45
Specifikus rendszerek....45
Spirális fejlesztési modell 190
SQL 119
SQL Server....115
SRM 211
statikus rendszer 29
Számítógép hálózatok....102
Számítógépes modell....220
Szemantikai nézet....27
Személyi számítógépek 87
Szervervirtualizáció....98
Szimuláció....221
Szimulációs modell 221
Szintaktikai nézet....27
Szoftver 94
sztochasztikus rendszereket....29
Szuperszámítógép....87
Szürke doboz....156

T

TCP/IP modell....108
Teljes adatmentés 185
Teljes topológia 105
Tervezett karbantartás....183
Tesztkörnyezet....188
Típusalkalmazás 155
TMS....211
törzsadatok 36
tranzakciós adatok 35
tranzakciós alkalmazások 194
Tudás....34
Tudás tárolása....225
Tudásbázis....226

U

UML 167

UN/EDIFACT 206
Upper CASE.....165

V

vállalati információs rendszer.....28, 30
Vállalati információs rendszerek
életciklusa.....178
vállalati stratégia.....58
Vastag kliens 195
Vékony kliens.....195
Virtualizáció 97
Vízesés modell 180
V-modell.....180
Von-Neumann architektúra 91
VPN.....112

W

Warm Failover.....187
Web 1.0 68
Web 2.0 68
Web 3.0 69
Wide Area Network.....103
Wiener.....24
World-Wide Web 111

X

XML.....121, 122

Z

Zsebszámítógépek 86

Mellékletek

Mérési egységek és az információs technológia

A mértékegységek többszöröseinek meghatározásához a tízes alapú (10^n) SI-rendszert[45] (SI – *International System of Units*) alkalmazzuk, míg a kettőhatvány-alapú (2^n) mértékegységeket az IEC (IEC – *International Electrotechnical Commission*[46]) által meghatározott bináris előtagokkal jelöljük.

M.1. táblázat: Az SI szerinti mértékegységek többszörösei

Előtag	Jelölés	Többszörös (hatvány)	Többszörös (számérték)	Szám neve
yotta	Y	10^{24}	1 000 000 000 000 000 000 000 000	kvadrillió
zetta	Z	10^{21}	1 000 000 000 000 000 000 000	trilliónta
exa	E	10^{18}	1 000 000 000 000 000 000	trillió
peta	P	10^{15}	1 000 000 000 000 000	billiárd
tera	T	10^{12}	1 000 000 000 000	billió
giga	**G**	**10^{9}**	**1 000 000 000**	**milliárd**
mega	**M**	**10^{6}**	**1 000 000**	**millió**
kilo	**k**	**10^{3}**	**1 000**	**ezer**
hekto	h	10^{2}	100	száz
deka	da	10^{1}	10	tíz
-	alapegység	10^{0}	1	egy
deci	d	10^{-1}	0,1	tized
centi	c	10^{-2}	0,01	század
milli	**m**	**10^{-3}**	**0,001**	**ezred**
mikro	**µ**	**10^{-6}**	**0,000 001**	**milliomod**
nano	**n**	**10^{-9}**	**0,000 000 001**	**milliárdod**
piko	p	10^{-12}	0,000 000 000 001	billiomod
femto	f	10^{-15}	0,000 000 000 000 001	billiárdod
atto	a	10^{-18}	0,000 000 000 000 000 001	trilliomod
zepto	z	10^{-21}	0,000 000 000 000 000 000 001	trilliárdod
yokto	y	10^{-24}	0,000 000 000 000 000 000 000 001	kvadrilliomod

Forrás: https://www.prevodyjednotiek.sk/nasobky_diely/

Az SI egy általánosan elfogadott szabványrendszer, amely fizikai mértékegységeket határoz meg, például a métert, a kilogrammot és a másodpercet. Az SI előtagjai (kilo, mega, giga, tera, stb.) az informatika területén is használatosak, de ebben az esetben az IEC által meghatározott formátumban jelennek meg. Mivel az SI a tízes alapú számrendszert alkalmazza, és a számítástechnikában kettőhatvány-alapú mértékegységeket használnak, ez gyakran félreértésekhez vezet. Ennek megoldására az IEC 1998-ban bevezetett egy külön előtagrendszert, amely kifejezetten az informatikai adattárolási egységek pontos megkülönböztetésére szolgál.

[45] Az SI rövidítés a francia Système International d'Unités kifejezésből származik.
[46] Nemzetközi Elektrotechnikai Bizottság.

M.2. táblázat: Bináris többszörösök

SI előtag (tízes alapú)	IEC bináris előtag (kettőhatvány-alapú)	Érték
1 kilobit (kb) = 10^3 b = **1 000 b**	1 kibibit (Kib) = 2^{10} b = **1 024 b**	1 024 bit
1 megabit (Mb) = 10^6 b = **1 000 000 b**	1 mebibit (Mib) = 2^{20} b = **1 048 576 b**	1 048 576 bit
1 gigabit (Gb) = 10^9 b = **1 000 000 000 b**	1 gibibit (Gib) = 2^{30} b = **1 073 741 824 b**	1 073 741 824 bit
1 terabit (Tb) = 10^{12} b = **1 000 000 000 000 b**	1 tebibit (Tib) = 2^{40} b = **1 099 511 627 776 b**	1 099 511 627 776 bit
1 kilobájt (kB) = 10^3 B = **1 000 B**	1 kibibájt (KiB) = 2^{10} B = **1 024 B**	1 024 bájt
1 megabájt (MB) = 10^6 B = **1 000 000 B**	1 mebibájt (MiB) = 2^{20} B = **1 048 576 B**	1 048 576 bájt
1 gigabájt (GB) = 10^9 B = **1 000 000 000 B**	1 gibibájt (GiB) = 2^{30} B = **1 073 741 824 B**	1 073 741 824 bájt
1 terabájt (TB) = 10^{12} B = **1 000 000 000 000 B**	1 tebibájt (TiB) = 2^{40} B = **1 099 511 627 776 B**	1 099 511 627 776 bájt

Forrás: (Gála, L.; pour, J.; Šedivá, Z. 2009)

A mindennapi életben az SI előtagokat (kilo, mega, giga, tera stb.) széles körben használják az informatikában is, még akkor is, ha technikailag nem mindig pontosak. Ez különösen igaz az adattároló eszközök, például merevlemezek (HDD, SSD), USB-k és memóriakártyák esetében, ahol a gyártók az SI előtagokat alkalmazzák. Például egy 256 GB-os USB-meghajtó a gyártó szerint 256 000 000 000 B kapacitással rendelkezik. Ezt az értéket viszont soha a számítógépünk operációs rendszere nem fogja így látni, mivel bináris számrendszert használ, így az eszköz körülbelül 238 GiB-ként (gibibájtként) jelenik meg. Miért? Tegyük fel, hogy van 255 550 554 112 B (bájt) nagyságú memóriánk, amit a gyártó legyártott. Számítsunk:

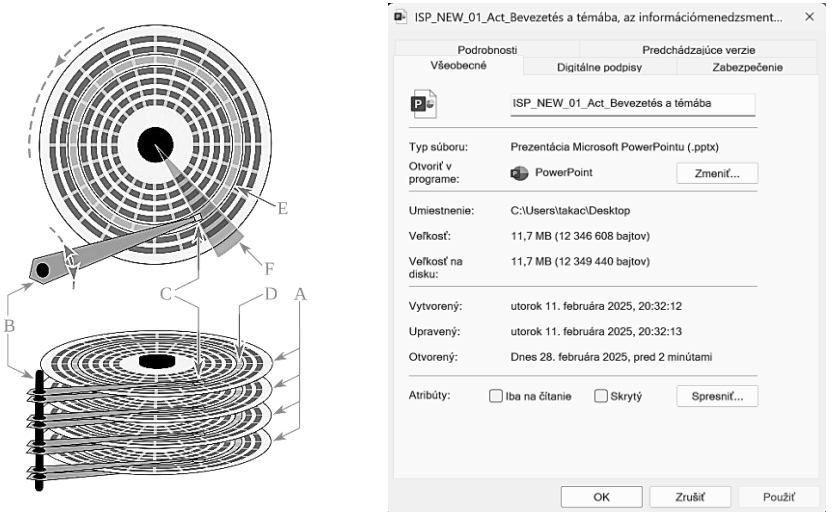
- SI szerinti számítás:
  - 255 550 554 112 B/1000 = 255 550 554, 112 KB/1000
    ≈ 255 550, 554 MB/1000
    ≈ **255, 551 GB ≈ 256 GB**
- IEC szerinti számítás:
  - 255 550 554 112 B/1024 = 249 561 088 KiB/1024
    = 243 712 MiB/1024
    = **238 GiB**

Felmerül egy teljesen logikus kérdés: *De akkor a gyártó miért kerekítette 256-ra és nem 250-re vagy 260-ra?* A válasz a számítógépek bináris működésében és a memóriakezelésben rejlik. A gyártók az SI (tízes alapú)

rendszert használják, de a kapacitást bináris szempontok szerint is optimalizálják. Ezért választják a 128, 256, 512, 1024, … GB-os értékeket, mert ezek közel esnek a 2^n bináris sorozathoz:

$$2^0; 2^1; 2^2; 2^3 \ldots \Rightarrow 1, 2, 4, 8, 16, 32, 64, 128, 256, 512; 1024; 2048 \ldots$$

A gyártók az SI előtagokat marketing és iparági konvenciók miatt alkalmazzák, mivel ezek a nagyobb számok kedvezőbb benyomást keltenek a vásárlók számára (500 GB nagyobbnak tűnik, mint 465 GiB). Bár az IEC 1998-ban bevezette a bináris előtagokat (KiB, MiB, GiB stb.), ezek kevésbé terjedtek el a kereskedelmi termékeknél. A felhasználók többsége nem ismeri a bináris előtagokat, ezért az SI használata egyszerűbb és ismertebb.



M.1. ábra: Lemez szerkezete (balra) és egy fájl tulajdonságai (jobbra).
Forrás (baloldali kép): https://cs.wikipedia.org/wiki/Cylindr-Hlava-Sektor

Leírás: A – Lemeztányérok (mágneses réteggel bevont korongok), B – Forgó kar (amely az összes olvasó/író fejet tartja), C – Olvasó és író fej (head), D, E – Cilinder (sávok összessége, amely minden lemeztányéron áthalad, azaz egy henger), F – Szektor (szögletes szelet, amely sávokat tartalmaz)

A merevlemezen az adatok szektorokba vannak mentve. Ha egy szektorba akár csak egy bitet is beírunk, az egész szektor lefoglalásra kerül. Ha egy fájl nem fér el egyetlen szektorban, akkor több szektorra van szükség, amelyeket klaszternek nevezünk. Ilyen lemezfelosztást láthatunk az M.1. ábrán (balra). Ha megnézünk egy fájl méretét a Windows rendszerben, két értéket láthatunk. Ez a különbség nemcsak az SI és IEC szabványok eltéréséből fakad, hanem a lemez formázási sajátosságaiból is. Az M.1. ábrán (jobbra) például a következő két érték szerepel:

- fájlméret: 12 346 608 B,
- lemezen foglalt méret: 12 349 440 B.

Ez a különbség a lemez formázásakor, meghatározott szektorméret miatt alakul ki. Windows 11 Pro operációs rendszerben az NTFS fájlrendszer van alkalmazva, ahol az alapértelmezett szektorméret 4096 B (4 KiB). Ha elosztjuk a két értéket a szektormérettel:

- 12 346 608 / 4096 = 3 014,31 → Ez azt jelenti, hogy a fájl teljes egészében 3 014 teljes szektort lefoglal, és az utolsó szektorból még 2 832 B-ot használ.
- 12 349 440 / 4096 = 3 015 → A fájl ténylegesen 3 015 szektort foglal el, **mivel az NTFS fájlrendszer[47] teljes szektorokban kezeli a fájlokat**, így a maradék adat miatt a már említett egy további szektor (3 015-ik) is lefoglalásra kerül.

Végezetül, még egy példát mutatunk. Tegyük fel, hogy az adatletöltési sebesség 40 Mbps. *Mekkora adatmennyiséget tudunk letölteni egy másodperc alatt?* Ez azt jelenti, hogy egy másodperc alatt 40 megabit adatot továbbítunk. Mivel 1 byte = 8 bit, ez a sebesség átszámítva körülbelül 5 MB/s (megabájt per másodperc). A számítógépes rendszerek azonban a bináris mértékegységet (1 MiB = 1 048 576 B) használják, ezért a valóságban a felhasználó által érzékelt sebesség körülbelül 4,77 MiB/s lesz. Ennek okai a következők.

- **Tízes számrendszer**: az internetszolgáltatók mindig megabit/másodpercben (Mbps) adják meg a sebességet a tizedes (SI) rendszer szerint: 1 Mb = 1 000 000 bit. Tegyük fel tehát, hogy a letöltési sebesség 40 Mbps. Az Mbps jelentése megabit per

[47] A fájlrendszer (file system) egy adatkezelési struktúra, amely meghatározza, hogyan kerülnek a fájlok és adatok tárolásra, elnevezésre, rendszerezésre és elérésre egy háttértárolón (például merevlemezen, SSD-n, pendrive-on). Windows rendszereken pl.:

- NTFS (New Technology File System) – Alapértelmezett fájlrendszer Windows 10/11 esetén, támogatja a jogosultságkezelést, nagy fájlméreteket.
- FAT32 (File Allocation Table 32) – régebbi, kompatibilis szinte minden eszközzel, de max. 4 GB-os fájlméretkorláttal rendelkezik – tehát a fájlok max. mérete 4 GB lehet.
- exFAT (Extended File Allocation Table) – FAT32 utódja, nagyobb fájlokat is kezel.

A fájlrendszer a lemezen foglalt helyet szektorokra osztja, majd ezeket nagyobb egységekbe, ún. klaszterekbe szervezi. Amikor egy fájlt elmentésre kerül, a fájlrendszer kijelöli a szükséges szektorokat és hozzárendeli a fájlhoz.

second, azaz egymillió bit másodpercenként. Mivel 1 Bájt (B) = 8 bit (b), ezért 40 megabit/s ÷ 8 = 5 megabájt/s (MB/s). Az egy másodperc alatt letöltött adatmennyiség, tehát 5 MB.

- **Bináris számrendszer**: A bináris rendszerben (ahogyan például a Windows is mutatja) 1 MiB = 1 048 576 B. Így, az 5 000 000 ÷ 1 048 576 ≈ 4,77 MiB/s.

A tényleges letöltési sebesség 40 Mbps esetén tehát 4,77 MiB/s, marketing célból ezt 5 MB/s értékként tüntetik fel. (Gála, L.; Pour, J.; Šedivá, Z. 2015) (Sándor, J. 2011) (Kučera, M.; Fiľa, M.; Látečková, A.; Stuchlý, P., 2017) (Pavlíček, A.; Syrovátková, J. 2022)

Hálózattípusok, hálózati modell és hálózati eszközök

A számítógépeket használó rendszerekben a kommunikációt három alapvető hálózattípus biztosítja:

- A **telekommunikációs hálózat** (például a telefonhálózat) leegyszerűsítve végponti eszközökből (telefon, fax), telefonközpontokból és átviteli utakból áll. Amikor egy kommunikációs igény merül fel, a rendszer biztosítja az útvonal létrehozását az egyik végponttól a telefonközponton keresztül egy másik központhoz, végül pedig a másik végponti eszközig. A gyakorlatban ilyenkor egy **körkapcsolt** (circuit-switched) kommunikációs csatorna jön létre, amely két végpont között biztosítja az adatátvitelt. Ez az útvonal egy meghatározott kapacitású, kétirányú kommunikációs csatorna, amely a beszélgetés teljes időtartama alatt dedikáltan fenntartott marad. A **telefonközpontok kapcsolóként** (switch) működnek, és biztosítják, hogy az adatok a megfelelő útvonalon haladjanak a két végpont között. A hívás befejezése (például a telefon letétele) után az útvonal felszabadul, és más hívások számára újra felhasználhatóvá válik. Más körkapcsolt hálózati példa lehet a rádiókommunikáció, például építkezéseken használt adóvevők. Ebben az esetben a felhasználók azonos frekvenciára (**csatornára**) hangolódnak, és mindenki hallja az adott csatornán folyó kommunikációt. Itt az adatok megosztott csatornán (shared channel) keresztül kerülnek továbbításra, de a csatornaváltás nem automatikusan, hanem manuálisan történik.
- A **számítógépes hálózatok** esetében nem jönnek létre állandó kapcsolatok (körök), mint a telefonhálózatokban. Ehelyett az adatátvitel dinamikusan osztott átviteli úton történik, amelyet az összes hálózati eszköz közösen használ. A számítógépes hálózatokban az adatátvitelt **csomagkapcsolás** (packet switching) biztosítja.
    - Nincs dedikált csatorna, az összes hálózati eszköz megosztja a sávszélességet.
    - Az adatokat csomagokra (pakettekre) bontják, amelyeket egyedi címzéssel látnak el.
    - A csomagok különböző útvonalakon haladhatnak, és a célállomásnál állnak össze újra.

  Hogy elkerüljék az adatcsomagok összekeveredését, az adók megfelelő csomagformátumot használnak, amelyben az adatok

tartalmazzák: A címzett azonosítóját (**IP-cím**) és az adó azonosítóját. Ha a címzett más hálózatban található, a csomagot egy **útválasztóhoz** (router) küldik. Az útválasztó egy speciális eszköz, amely csomagokat fogad, feldolgozza és a megfelelő irányba továbbítja. Az útválasztók a **store & forward** elvet követik, azaz először elraktározzák a beérkező csomagokat, majd sorban továbbítják őket a megfelelő célállomás felé.

- Az **ipari hálózatok** a számítógépes hálózatok egy speciális kategóriáját képezik, amelyek ipari alkalmazásokban és gyártási folyamatok vezérlésében használtak (pl. gyártósorok, numerikus vezérlésű szerszámgépek, robotok). Az ilyen hálózatok esetében kiemelten fontosak a következők.
    - **Valós idejű kommunikáció** biztosítása, azaz az egyes eszközök meghatározott időintervallumokban hozzáférhessenek a hálózathoz (deterministic access).
    - **Prioritási mechanizmusok**, amelyek lehetővé teszik, hogy kritikus eszközök elsőbbséget élvezzenek a kommunikációban.
    - **Központi vezérlés** (master/slave elv), de ugyanakkor az eszközök közötti közvetlen kommunikáció is biztosított kell legyen (peer-to-peer).
    - **Megbízható, ipari környezetre optimalizált kábelezés** és infrastruktúra, hogy elkerüljék a zavarokat és az adatvesztést.

  Az ipari hálózatok példái:
    - Profibus, Profinet (automatizálásban),
    - Modbus, CAN bus (ipari kommunikációban),
    - EtherCAT (precíziós valós idejű vezérlésben).

A továbbiakban kizárólag a számítógépes és telekommunikációs hálózatokkal foglalkozunk, mivel napjainkban ezek egyre erősebben összeolvadnak (**konvergálnak**). Ez a konvergencia abban nyilvánul meg, hogy az alkalmazások már nem tesznek különbséget a között, hogy az adatátvitel telekommunikációs vagy számítógépes hálózaton keresztül történik. Az egyik oldalon ma már teljesen természetes, hogy a számítógépes hálózatokon keresztül telefonálhatunk (pl. IP-telefonia, VoIP, internetes hívások). A másik oldalon a telekommunikációs hálózatokat is használjuk számítógépes hálózatok elérésére, és egyes telekommunikációs infrastruktúrák kifejezetten adatátvitelre vannak fenntartva a számítógépes hálózatok számára. Ez a fejlődési irány azt mutatja, hogy a telekommunikáció és az informatika egyre inkább összeolvad, és a jövőben még szorosabb integráció várható.

Mivel a kommunikációt lebonyolító eszközök eltérőek lehetnek, különböző hálózatokban helyezkedhetnek el, a kommunikáció során különféle átviteli utak használhatók, és az alkalmazások eltérő szolgáltatásminőséget (QoS) igényelhetnek, szükségessé vált egy olyan hálózati koncepció kidolgozása, amely alkalmazkodik ezekhez a változatos feltételekhez, miközben az alkalmazások számára transzparens marad. A cél nem egy monolitikus rendszer létrehozása volt, mint például a hagyományos telefonhálózatok esetében, amelyek megbízhatóan biztosítják a hangátvitelt, de nehezen bővíthetők új alkalmazásokkal. Ehelyett azt az elvet fogadták el, hogy a kommunikációs funkciókat logikai rétegekre bontják, ahol minden réteg egy adott, pontosan meghatározott feladatot lát el. A rétegek hierarchikus elrendezése, azokhoz kapcsolt funkcionalitás és a világosan meghatározott szabályok alkotják a **hálózati modellt**. A rétegzett modell egyik legfontosabb példája az ITU (International Telecommunication Union) által kifejlesztett X.200 szabvány, amelyet az ISO (International Standard Organization) is elfogadott, és ISO 7498 szabványként rögzített. Ez a modell a Nyílt Rendszerek Összekapcsolási Referenciamodellje (*Open Systems Interconnection Reference Model*), röviden **OSI modell** vagy ISO/OSI. A szabvány összesen hét réteget határoz meg, amelyeket két fő csoportba osztanak.

- Felső négy réteg → a végpontok közötti kommunikációért felelősek (alkalmazás-orientált rétegek).
- Alsó három réteg → a kommunikációs alrendszerhez tartozik, és azt határozza meg, hogyan történjen az adattovábbítás a hálózaton belül, akár különböző hálózatok összekapcsolásával is.

A modell ez mellett kétféle hálózati eszközt különböztet meg.

- Végponti csomópontok (end nodes) – azokat az eszközöket jelenti, amelyek közvetlenül egymással kommunikálnak (pl. számítógépek, szerverek, okostelefonok).
- Közbenső rendszerek (intermediate systems) – olyan hálózati eszközök, mint az útválasztók (routerek), kapcsolók (switch-ek) és átjárók (gateway-ek), amelyek biztosítják az adatok célba juttatását.

Az OSI (Open Systems Interconnection) referencia modell hét különálló rétegre osztja a hálózati kommunikáció folyamatát, ahol minden réteg egy adott feladatot lát el, és csak a közvetlenül felette és alatta lévő réteggel lép kapcsolatba.

- Az **alkalmazási réteg** közvetlen kapcsolatot biztosít az alkalmazások és a hálózat között. Ez a réteg gondoskodik az olyan

hálózati szolgáltatásokról, amelyekre a felhasználói programoknak szüksége van (például fájlátvitel, e-mail hozzáférés, böngészés). Tipikus példák:

- fájltovábbítás (FTP – File Transfer Protocol),
- elektronikus levelezés (SMTP, POP3, IMAP),
- webszolgáltatások (HTTP, HTTPS).

- A **megjelenítési réteg** felelős az adatok megfelelő formázásáért, hogy azok érthetőek legyenek a célrendszer számára. Ennek során biztosítja a kódolást, titkosítást és tömörítést.
- A **viszonyréteg** biztosítja és kezeli a két eszköz közötti kommunikációs kapcsolatot (relációt). Ez lehetővé teszi az adatátvitel szinkronizálását, valamint ellenőrzőpontok létrehozását, hogy hiba esetén az adatok újra küldhetők legyenek az utolsó helyes állapotból.
- A **szállítási réteg** gondoskodik az adatok csomagokra bontásáról és azok hibamentes továbbításáról. Ez a réteg biztosítja, a beérkező adatcsomagok megfelelő sorrendben állítását, valamint ellenőrzi az adatvesztést és újraküldi az elveszett csomagokat. Tipikus példák:
  - TCP (Transmission Control Protocol) – megbízható, kapcsolat-alapú adatátvitel,
  - UDP (User Datagram Protocol) – gyors, de nem garantált adatátvitel.
- A **hálózati réteg** végzi az adatok címzését és útvonalválasztását (routing), ha az adatokat különböző hálózatokon keresztül kell továbbítani a küldőtől a címzettig. Az adatcsomagokat (packets) a hálózati réteg továbbítja, és ezek a csomagok különböző útvonalakon haladhatnak a hálózatban. Az OSI modell ezt datagram szolgáltatásnak nevezi, mivel nem garantálja, hogy a csomagok ugyanazon az útvonalon haladnak. Tipikus példák:
  - IP (Internet Protocol) – csomagok továbbítása különböző hálózatok között,
  - ICMP (Internet Control Message Protocol) – hálózati hibák visszajelzése.
- Az **adatkapcsolati réteg** felel az adatok keretekbe (frame) csomagolásáért, és biztosítja azok biztonságos továbbítását a hálózaton belül. Ez a réteg vezérli az eszközök közötti közvetlen adatátvitelt egy adott fizikai hálózaton belül. Fontos szerepei, hogy hibajavítást és csomagellenőrzést (pl. CRC – Cyclic Redundancy Check) végez és MAC (Media Access Control) címek kezelését végzi, valamint meghatározza, hogy az adott pillanatban, az

állomások közül melyik állomás használhatja a csatornát adásra. Tipikus példák:
    - Ethernet (IEEE 802.3),
    - Wi-Fi (IEEE 802.11),
    - PPP (Point-to-Point Protocol).
- A **fizikai réteg** végzi az elektromos, optikai vagy rádiófrekvenciás jelek formájában történő adatátvitelt, a hálózat fizikai közegén keresztül. Ez a réteg határozza meg a kábelezési szabványokat, csatlakozókat és a jelátviteli technológiákat. Tipikus példák:
    - rézvezetékes átvitel (pl. Ethernet UTP kábel),
    - optikai szálas átvitel (pl. Fiber Optic),
    - vezeték nélküli átvitel (pl. Wi-Fi, Bluetooth, 5G).

Az OSI referencia modell gyakorlati alkalmazása akkor vált lehetővé, amikor az egyes rétegekhez konkrét protokollokat és technológiákat rendeltek, amelyek a szabványok szerint működnek. A fejlesztés több irányban haladt tovább.

- Az OSI modell egyes rétegeihez már meglévő ipari szabványokat és technológiákat rendeltek hozzá. A fizikai és adatkapcsolati rétegek esetében például az ITU (International Telecommunication Union) V és X sorozatú protokolljai kerültek alkalmazásra. A fizikai és adatkapcsolati rétegek minősége meghatározza az elérhető átviteli sebességet, amely napjainkban 10 Gbps is lehet. Például:
    - **V.90 protokoll** (56 kbps adatátvitel telefonvonalon),
    - **IEEE 802.x szabványok** – Ethernet, Wi-Fi, stb.,
    - **ANSI X3T12 szabványok** – később ISO 8802 néven váltak szabvánnyáé.
- Az OSI modell struktúrája alapján új protokollok is kidolgozásra kerültek. Pl., a **MAP** (Manufacturing Automation Protocol) – ipari automatizálási környezetben alkalmazott hálózati protokollkészlet.
- Bizonyos gyártóspecifikus protokollok is megjelentek, amelyek nem közvetlen OSI-alapúak, de kompatibilisek vele. Például:
    - **SPX/IPX** (Sequenced Packet Exchange/Internet Packet Exchange) – Novell Netware rendszerekhez,
    - **NetBIOS/NetBEUI** (Network Basic Input/Output System) – Microsoft hálózatkezeléshez.
- Egyes szabványok az OSI modellt leegyszerűsített formában alkalmazzák, és más technológiákat használnak a gyakorlati működéshez. **TCP/IP protokollkészlet** (Transmission Control Protocol / Internet Protocol), amely eredetileg az Egyesült Államok

Védelmi Minisztériumának (DoD) heterogén hálózatokhoz szánt fejlesztése volt, ma az internet globális szabványa. A TCP/IP egy rugalmasabb, kevesebb réteget tartalmazó modell, amely világszerte elterjedt, az OSI modell helyett.

A hálózati kommunikációhoz minden eszköznek rendelkeznie kell egy hálózati interfésszel, amely lehetővé teszi az adatkapcsolatot a hálózat többi eszközével. (Számítógépek esetén a **NIC** (hálózati kártya – *Network Interface Card*), amely lehet alaplapra integrált vagy PCI-bővítőkártya. Laptopoknál korábban PCMCIA vagy ExpressCard csatlakozások, ma már leginkább Wi-Fi és USB adapterek. Telekommunikációs kapcsolatoknál modemek és hálózati adapterek.)
Minden hálózati interfész rendelkezik egy egyedi fizikai címmel, a MAC-címmel (Media Access Control Address). Az Ethernet hálózati eszközök 48 bites MAC-címmel rendelkeznek, amelyet hexadecimális formátumban írnak le (pl. 00-50-7F-29-A7-AB).
A hálózatok különböző eszközöket használnak az adatok továbbítására és irányítására. Ezek az eszközök az OSI modell különböző rétegeiben működnek, és meghatározott feladatokat látnak el.

- A fizikai rétegben működő eszközök az adatjelek továbbításával és erősítésével foglalkoznak. A **repeater** felerősíti a gyengülő jelet, amely az átvitel során csillapodhat a kábel hosszától és a fizikai környezettől függően. Ezt a megoldást olyan helyeken alkalmazzák, ahol a hosszú kábelezés miatt szükséges a jelminőség fenntartása. Egy **hub** több eszközt kapcsol össze egy hálózati szegmensen belül. Minden beérkező adatot továbbít minden csatlakoztatott eszköz felé, így a hálózatban lévő összes eszköz ugyanazt az információt kapja meg.
- Az adatkapcsolati rétegben működő eszközök segítik a hálózati forgalom kezelését és az adatátvitel hatékonyságát. A **híd** (**bridge**) egy olyan eszköz, amely két vagy több hálózati szegmenst köt össze. Képes az adatcsomagokat (frame) továbbítani az egyik szegmensből a másikba, ha a célállomás az adott szegmensben található. Ha az üzenet a saját szegmensben lévő eszköznek szól, akkor a híd nem küldi tovább a másik szegmensbe. A **switch** a bridge továbbfejlesztett változata, amely nem csak két, hanem több szegmens között is képes az adatok irányítására. Intelligensen továbbítja az adatokat, azaz csak annak az eszköznek küldi el a csomagot, amelynek szól. Ez jelentősen csökkenti az

adatütközések[48] (collision[49]) számát, mivel az egyes portok különálló kommunikációs csatornákat alkotnak.

- A **hálózati** rétegben működő eszközök gondoskodnak az adatok célba juttatásáról különböző hálózatok között. A **router** a hálózati forgalom irányításáért felelős. A csomagokat az IP-címek alapján továbbítja, nem csak egy helyi hálózaton belül, hanem különböző hálózatok között is (például otthoni hálózat → internet). Eltérően a switchektől, amelyek csak egyetlen hálózati szegmensen belül dolgoznak, a router képes különálló hálózatokat összekapcsolni. A routerek a hálózati útválasztási táblák (routing table) segítségével döntenek arról, hogy az adott csomagot milyen útvonalon kell továbbítani. (Szabó, B.; Márföldi, E. 2011) (Herdon, M. 2016) (Juhász, S. 2011)

[48] A hubok minden csatlakoztatott eszköz felé továbbítják az adatokat, ami sok ütközést és felesleges adatforgalmat okoz. A switch ezzel szemben célzottan továbbítja az adatokat, ami optimalizálja a sávszélességet és csökkenti az ütközéseket. A hubokat felváltották a switchek, mert hatékonyabb adatkezelést biztosítanak.

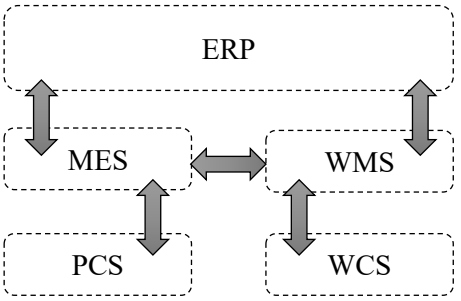
[49] Ütközés (collision) akkor következik be egy hálózatban, amikor két vagy több eszköz egyszerre próbál adatot küldeni ugyanazon a kommunikációs csatornán. Az ütközés olyan hálózati probléma, amely lassítja a kommunikációt, mert az adatokat újra kell küldeni. Modern hálózatokban (Wi-Fi, switchelt Ethernet) már szinte teljesen kiküszöbölték.

Alkalmazások az integrált, automatizált termelési és raktározási tevékenységek irányítására

A **gyártásirányítási rendszer** (MES – *Manufacturing Execution System*) az alábbi területeket támogatja: erőforrás-kezelés és alokáció, operatív gyártástervezés és ütemezés, gyártásfelügyeleti irányítás, dokumentumkezelés, adatgyűjtés, adatintegráció és archiválás, munkaerő-menedzsment, minőségirányítás, folyamatirányítás, termeléskövetés, teljesítmény-elemzés és értékelés. Az **automatizált gyártási rendszer** (PCS – *Production Control System*) gépekből és berendezésekből áll, beleértve a robotokat, számjegyvezérlésű gépeket (CNC) és gyártósorokat, amelyek előre meghatározott programok alapján működnek.

A **raktárkezelő rendszer** (WMS – *Warehouse Management System*, néha IMS – *Inventory Management System*) a következő területeket támogatja: a raktárszerkezet és elrendezés jellemzőinek nyilvántartása, termékek fogadása és raktározása, kiszedési, csomagolási és szállítási folyamatok irányítása, minőségellenőrzés nyomon követése a teljes raktári működés során, a raktárban tárolt termékek minőségének folyamatos ellenőrzése. Az **automatizált raktárirányítási rendszer** (WCS – *Warehouse Control System*) olyan raktártechnológiai megoldásokat foglal magában, amelyek lehetővé teszik a raktári tételek automatikus mozgatását. Egy ilyen rendszer példája a KIVA Systems, amelyet például az Amazon.com elektronikus kereskedelmi platform is alkalmaz.

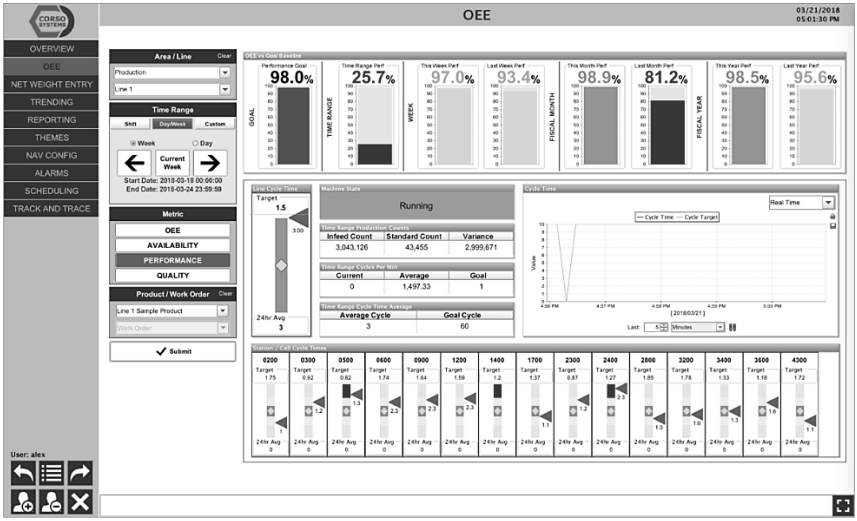
A vállalati információ áramlás és rendszerek közötti adatkapcsolatok jelenleg az ISA-95 (Enterprise-Control System Integration) szabvány szerint vannak szabványosítva. Ez a szabvány négy alapvető információ típust határoz meg, amelyek az egész vállalatra kiterjedő tranzakciós rendszerektől az automatizált gyártási és raktárkezelő rendszerekig terjednek, a gyártásirányítási (MES) és raktárirányítási (WMS) rendszereken keresztül**.** Ezek az alábbiak.



M.2. ábra: A termelés- és raktárirányítás integrációja
Forrás: (Gála, L.; Pour, J.; Šedivá, Z. 2015)

- **Termékspecifikáció.** (Mit kell megadni a gyártás elindításához? Mit kell be- vagy kiszállítani?)
- **Gyártási/raktározási kapacitások specifikációja.** (Milyen erőforrások és raktárterületek állnak rendelkezésre?)
- **Gyártási, raktározási és kiszállítási tervek specifikációja.** (Mit, milyen mennyiségben és mikor kell legyártani, betárolni vagy kiszállítani?)
- **Gyártási eredmények specifikációja.** (Mi, hogyan és mikor került legyártásra, betárolásra vagy kiszállításra?)

Az integrációs követelmények abból a szükségletből fakadnak, hogy gyorsabban továbbítsák az információkat a gyártás, a raktárak és az ügyféligények kiszolgálására összpontosító rendszerek között. Ennek célja, hogy a gyártási és raktározási folyamatok hatékonyan alkalmazkodjanak a változó vevői igényekhez. (Gála, L.; Pour, J.; Šedivá, Z. 2015)

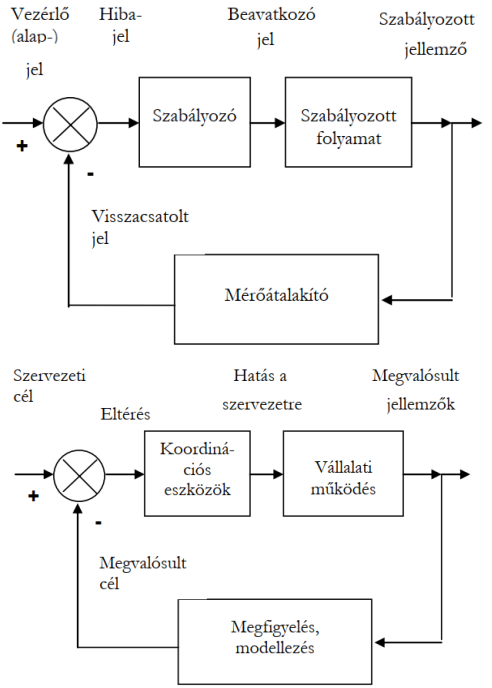


M.3. ábra: A Corso Systems egyedi OEE (Overall Equipment Effectiveness) Dashboardja, amely a termelési sebességet és az azt befolyásoló számos tényezőt, valamint a teljesítménycélokat mutatja.
Forrás: https://corsosystems.com/manufacturing-execution-systems-guide

A visszacsatolt rendszerek

A visszacsatolt rendszer fogalmával már találkoztunk a modellezés kapcsán (lásd, a 9.3. ábra: Modellezési folyamat), bár ott még nem neveztük meg így. Visszacsatolt rendszereknek azokat nevezzük, amelyeknél a kimenet egy része visszajut a bemenethez, és ez által befolyásolja a rendszer további működését. Két alapvető típust különböztetünk meg:

- **negatív visszacsatolás**, amikor a kimenet csökkenti vagy korigálja a bemenetet, főként az ingadozások csillapítása érdekében;
- **pozitív visszacsatolás,** ahol a kimenet erősíti a bemenetet, ami a változások, illetve ingadozások felerősítéséhez vezet. Ennek tipikus példája a pánikvásárlás[50].



M.4. ábra: A visszacsatolásos rendszer (fent) és vállalatvezetés mint visszacsatolt rendszer (lent)
Forrás: (Sziray, J.; Gaul, G. 2006)

[50] Pl., háború közeledtével az emberek elkezdenek nagyobb mennyiségben vásárolni. Az üzletek ezt észreveszik, és nagyobb árakat állítanak be. Emberek látják az árnövekedést, de félnek, hogy később még drágább lesz, tehát még többet vásárolnak most.

A rendszerelmélet visszacsatolásos modellje a vállalat működésére is alkalmazható. Ebben az értelmezésben a szabályozott folyamat maga a vállalati működés, amelynek eredményeként különféle, a vállalat állapotát leíró adatok – az ún. megvalósult jellemzők – keletkeznek. Ezeket megfigyeléssel és modellezéssel dolgozzuk fel, s így határozható meg a megvalósult cél, amelyet összevetünk a vállalati tervezés során kijelölt szervezeti céllal. Ha a kettő nem egyezik, eltérés keletkezik, amelyet koordinációs eszközök alkalmazásával kell korigálni, hogy a működés a kívánt irányba terelődjön. Például, ha a szervezeti cél az üzleti év végére kitűzött nyereség, a modell számviteli eljárásokkal méri a tényleges eredményt. Jelentős eltérés esetén módosítani kell a koordinációs eszközöket, például a prémiumrendszer feltételeit. A visszacsatolt rendszerek jók lehetnek olyan esetekben, amikor eltérést kell korigálni, ha piaci zavar alakul ki, vagy szervezeti célokat kell mérni és értékelni. Ez a vállalatirányításon kívül a tervezésben, a kockázatkezelésben vagy gazdaságpolitikában használható. (Sziray, J.; Gaul, G. 2006)

UTF-8 kódtábla (részlet) és Unicode karakterek

M.3. táblázat: UTF-8 kódtábla – kivonat az ASCII kompatibilis tartományból (U+0021–U+007E)

Decimálisan	Hexadecimálisan	UTF-8 Hexadec.	Karakter	Decimálisan	Hexa decimálisan	UTF-8 Hexadec.	Karakter	Decimálisan	Hexa decimálisan	UTF-8 Hexadec.	Karakter
…	…	…		65	U+0041	41	A	98	U+0062	62	b
33	U+0021	21	!	66	U+0042	42	B	99	U+0063	63	c
34	U+0022	22	"	67	U+0043	43	C	100	U+0064	64	d
35	U+0023	23	#	68	U+0044	44	D	101	U+0065	65	e
36	U+0024	24	$	69	U+0045	45	E	102	U+0066	66	f
37	U+0025	25	%	70	U+0046	46	F	103	U+0067	67	g
38	U+0026	26	&	71	U+0047	47	G	104	U+0068	68	h
39	U+0027	27	'	72	U+0048	48	H	105	U+0069	69	i
40	U+0028	28	(	73	U+0049	49	I	106	U+006A	6A	j
41	U+0029	29	)	74	U+004A	4A	J	107	U+006B	6B	k
42	U+002A	2A	*	75	U+004B	4B	K	108	U+006C	6C	l
43	U+002B	2B	+	76	U+004C	4C	L	109	U+006D	6D	m
44	U+002C	2C	,	77	U+004D	4D	M	110	U+006E	6E	n
45	U+002D	2D	-	78	U+004E	4E	N	111	U+006F	6F	o
46	U+002E	2E	.	79	U+004F	4F	O	112	U+0070	70	p
47	U+002F	2F	/	80	U+0050	50	P	113	U+0071	71	q
48	U+0030	30	0	81	U+0051	51	Q	114	U+0072	72	r
49	U+0031	31	1	82	U+0052	52	R	115	U+0073	73	s
50	U+0032	32	2	83	U+0053	53	S	116	U+0074	74	t
51	U+0033	33	3	84	U+0054	54	T	117	U+0075	75	u
52	U+0034	34	4	85	U+0055	55	U	118	U+0076	76	v
53	U+0035	35	5	86	U+0056	56	V	119	U+0077	77	w
54	U+0036	36	6	87	U+0057	57	W	120	U+0078	78	x
55	U+0037	37	7	88	U+0058	58	X	121	U+0079	79	y
56	U+0038	38	8	89	U+0059	59	Y	122	U+007A	7A	z
57	U+0039	39	9	90	U+005A	5A	Z	123	U+007B	7B	{
58	U+003A	3A	:	91	U+005B	5B	[	124	U+007C	7C	|
59	U+003B	3B	;	92	U+005C	5C	\	125	U+007D	7D	}
60	U+003C	3C	<	93	U+005D	5D	]	126	U+007E	7E	~
61	U+003D	3D	=	94	U+005E	5E	^	…	…	…	
62	U+003E	3E	>	95	U+005F	5F	_				
63	U+003F	3F	?	96	U+0060	60	`				
64	U+0040	40	@	97	U+0061	61	a				

Forrás: https://www.utf8-chartable.de/, https://www.charset.org/utf-8



Univerzita J. Selyeho - Selye János Egyetem
Fakulta ekonómie a informatiky - Gazdaságtudományi és Informatikai Kar
Bratislavská cesta 3322
SK-945 01 Komárno
www.ujs.sk

Ing. Ondrej Takáč, PhD.
PaedDr. Ladislav Végh, PhD.
PaedDr. Krisztina Czakóová, PhD.

Vállalati információs rendszerek I.

Recenzenti - Recenzensek:
Dr. habil. Gombkötő Nóra
Ing. Norbert Gyurian, PhD.
Ing. Ladislav Tóth, PhD.

Vydavateľ - Kiadó:
Univerzita J. Selyeho - Selye János Egyetem

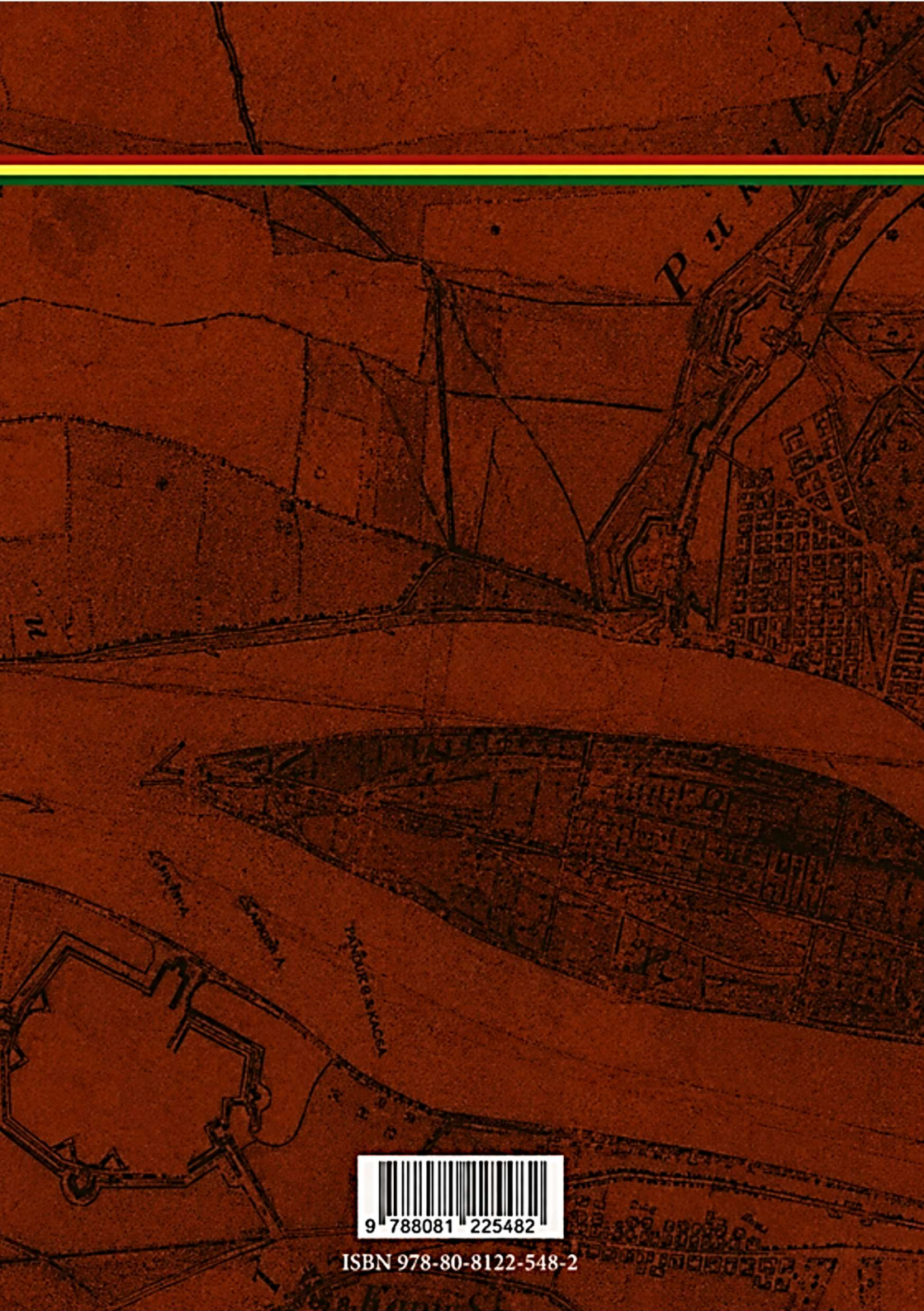
Tlačová príprava - Nyomdai Előkészítés:
Szerzők

Rozsah - Terjedelem:
12,02 AH – 12,02 szerzői ív

Rok vydania - Megjelenés éve
2026

Číslo vydania - Kiadás száma:
prvé vydanie - első kiadás

ISBN 978-80-8122-548-2 EAN 9788081225482



9 788081 225482

ISBN 978-80-8122-548-2