

Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Távközlési és Médiainformatikai Tanszék

Vállalati kommunikációt elemző rendszer tervezése és megvalósítása

Muráti Ákos
JNC2FC

Konzulens
Kardkovács Zsolt

Diplomaterv

2010

Nyilatkozat

Alulírott Muráti Ákos, a Budapesti Műszaki és Gazdaságtudományi Egyetem hallgatója kijelentem, hogy ezt a diplomatervet meg nem engedett segítség nélkül, saját magam készítettem, és a diplomatervben csak a megadott forrásokat használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Budapest, 2010. május 14.

Muráti Ákos

Kivonat

Diplomatervem tárgya a vállalati kommunikáció elemzése. A cél egy olyan *üzleti intelligencia (BI)* rendszer tervezése és megvalósítása, amely a vállalati (pl. e-mailes) kommunikáció elemzését támogatva teszi lehetővé a szervezet struktúrájától eltérő csoportok feltérképezését, ügymenetek nyomon követését és az egyes szereplők terheltségének mérését.

A bevezető részben nemzetközi szakirodalom alapján ismertetem az *üzleti intelligencia* fogalmát, fogalmának kialakulását és a szakterület jelentőségét. Bemutatom a napjainkban alkalmazott *üzleti intelligencia* rendszereket, majd a fejezet végén arra a megállapításra jutok, hogy jelenleg nem érhető el olyan *BI* megoldás, ami hatékonyan támogatja a vállalati igényeknek megfelelő kommunikáció-elemzést.

Dolgozatom stratégiai és elemző szakaszában bemutatom a vállalati kommunikációs környezet elemzésének témakörét és lehetőségeit, melyekre alapozva kidolgozom az elemzés koncepcióját. Elemzési módokat és azokhoz vizualizációs lehetőségeket definiálok, valamint egy fogalmi modellt is felállítok a kommunikáció-elemzésre.

Röviden ismertetem a vállalatokon belül tipikusan előforduló kommunikációs csatornákat, majd részletesen ismertetem az e-mail-elemzés lehetőségeit, az abból kinyerhető információkat. Kifejtem a feldolgozás feladatait és problémáit, továbbá bemutatok és értékelek egy e-mail elemző rendszert, a *Deep Email Miner* alkalmazást.

A rendszerterv bemutatása során részletesen ismertetem a megvalósítandó funkciókat, a tervezői döntések hátterét, és bemutatom a rendszer architektúráját és komponenseit is. A korábban definiált fogalmi modell alapján felépítem a rendszer logikai adatmodelljét, majd az elemzés-orientált adatbázisok szokásos tervezési szempontjait figyelembe véve tervezem meg az elemzőrendszer adatbázisának csillag-sémáját.

Diplomatervem 6. fejezetében az elemzőrendszer megvalósítását és megvalósítási környezetét ismertetem. Bemutatom a rendszer különböző funkcióit ellátó modulok implementációját, és ismertetem azokat az eszközöket és komponenskönyvtárakat, amelyeket felhasználtam.

Dolgozatom az elkészült rendszer értékelésével zárom, melynek során részletesen kitérek az egyes modulok: az üzenetek gyűjtését, elemzését és ábrázolását megvalósító komponensek értékelésére. A tesztek mellett képernyőképekkel illusztrálok az eredményt, hiszen a vizualizáció fontos része az elemzőrendszernek.

Szándékaim szerint az elkészült kommunikáció elemző rendszer jövőbeli továbbfejlesztésével és az e-mail mellett további kommunikációs-csatornákra elkészített üzenetgyűjtő komponenseivel a vállalatok számára hasznos és teljeskörű elemző rendszerré válik.

Abstract

The subject of my thesis is the analysis of corporate communication. The purpose of this paper is to design and implement a *Business Intelligence (BI)* system which supports the discovery of those communication groups who are different from the organization structure, the tracing of specific cases and the load measurement of each communication participant.

In the introduction I present the concept and significance of *Business Intelligence* and those corporate demands which have led to the evolution of *BI*, and I also enumerate some *BI* solutions, which are being applied in our days. At the end of this chapter I declare that there is no available *BI* solution which could support corporate communication analysis in its entirety.

The strategic and analysis sections of my paper demonstrate the general corporate communication environment and its possibilities from the aspect of analysis, and I set up the concepts and principles of the communication analysis based on those. I define some analytical method and visualization possibilities and I also design the semantic model of the communication analysis.

Henceforward I briefly summarize the typical corporate communication channels and then I specify the details and possibilities of e-mail analysis and those information which can be extracted from it. I describe the required tasks and problems of the e-mail processing, then I introduce and evaluate an e-mail analyzer application called *Deep Email Miner*.

In the chapter of System Design, I specify the system architecture, the components and functions to be implemented and I also describe the decisions I made during the planning period. Based on the previously defined semantic model, I created the logical data model of the system and I also designed the star-schema of my system's database considering the design principles of analytical-databases.

Section 6 describes the development and implementation of my system, including the implemented functions, developed modules and those tools, environments and component libraries which have been used for the implementation.

The final part of my thesis evaluates and summarizes the engineering and development results of my Communication Analyzer Systems. During the evaluation I specify the details of message collection, analysis and visualization components, illustrating with measurements and screenshots as the visualization components being important parts of the system.

My intention is that the Communication Analyzer System - with further improvements and more communication channel collection modules beyond e-mail - be a useful, efficient and complete solution for corporate communication analysis.

Tartalomjegyzék

1. Bevezető	1
2. Az üzleti intelligencia	2
2.1. Vállalati információs rendszerek fejlődése.....	2
2.2. Az üzleti intelligencia fogalma	5
2.3. Trendek és megoldások napjainkban.....	8
3. Stratégiai fázis	12
3.1. Környezet felmérése	12
3.1.1. Vállalati környezet és kommunikáció kapcsolata	12
3.1.2. Tipikus vállalati kommunikációs csatornák	13
3.2. A vállalati kommunikáció-elemzés lehetőségei	15
3.3. Az elemzési igények és ábrázolás koncepciója	16
3.3.1. Kapcsolati háló	16
3.3.2. Téma – Részrtvevő térkép	16
3.3.3. Folyamattérkép	17
3.4. Megvalósítandó rendszerfunkciók összefoglalása.....	18
4. Elemzési fázis.....	19
4.1. E-mailek az elemzés szemszögéből.....	19
4.1.1. Történeti áttekintés	19
4.1.2. Szabványok és ajánlások	19
4.1.3. Üzenet-formátum részletei	21
4.2. Természetes fogalmak összefoglalása	23
4.3. Kommunikáció-elemzés fogalmi modellje.....	24
4.3.1. Általános kommunikációs modell	24
4.3.2. Kommunikációs folyamat szintjei	26
4.3.3. A tartalomkezelés szintjei.....	27
4.4. Funkciók meghatározása	28
4.4.1. Gyűjtés, megfigyelés	29
4.4.2. Feldolgozás.....	29
4.4.3. Tárolás.....	29
4.4.4. Elemzés	29
4.4.5. Hozzáféérés	30
4.5. A feldolgozás feladatai és problémái.....	30
4.5.1. Személyek azonosítása	30
4.5.2. Csoportok azonosítása	30
4.5.3. Szálak azonosítása	31
4.5.4. Többszörös feldolgozás elkerülése.....	31
4.5.5. Időbeli kauzalitás biztosítása	31
4.5.6. Tartalom kibontása	31
4.5.7. Tartalmak kivonatolása	32
4.5.8. Vállalati környezet előismerete	32
4.6. Jelenlegi elemző megoldások áttekintése	32
4.6.1. Deep Email Miner	32
4.7. Elemzési fázis összefoglalása	33

5. Tervezési fázis.....	34
5.1. Tervezői döntések.....	34
5.1.1. Háromrétegű ügyfél-kiszolgáló architektúra.....	34
5.1.2. <i>Web Service</i> alapú kliens-szerver kapcsolat.....	34
5.1.3. E-mailek gyűjtése és feldolgozása.....	35
5.1.4. Környezet kiválasztása.....	36
5.1.5. Adatbázisszerver kiválasztása.....	36
5.2. Alrendszerek és modulok meghatározása.....	36
5.2.1. E-mail betöltő alrendszer.....	37
5.2.2. Közös Rendszerkönyvtárak és osztályok.....	38
5.2.3. Elemzés kiszolgáló.....	39
5.2.4. Kliens alkalmazás.....	40
5.2.5. Elemzési adatbázisszerver.....	41
5.3. Adatbázis tervezése.....	41
5.3.1. Logikai adatmodell.....	41
5.3.2. Fizikai adatmodell.....	42
5.3.3. Csillag-séma kialakítása.....	43
5.4. Külső komponensek.....	45
5.4.1. Sun JavaMail TM API.....	45
5.4.2. Microsoft GLEE – MS AGL.....	45
5.4.3. Oracle Data Provider.....	45
6. Rendszer fejlesztése	46
6.1. Környezet ismertetés.....	46
6.2. Implementáció.....	46
6.2.1. E-mail betöltő alrendszer.....	46
6.2.2. Közös rendszerkönyvtárak és osztályok – caCoreLibrary.....	47
6.2.3. Elemzés kiszolgáló - caServiceHost.....	48
6.2.4. Kliens alkalmazás – caClient projekt.....	49
7. Értékelés.....	51
7.1. E-mail betöltő alrendszer értékelése.....	51
7.1.1. Teljesítmény értékelése.....	51
7.2. Elemzés értékelése.....	52
7.2.1. Elemzési modulok bemutatása.....	52
7.2.2. Ábrázolás értékelése.....	56
7.2.3. Elemzés hálózati értékelése.....	57
8. Összefoglalás.....	58
A függelék – Elemzés és Tervezés	59
A1. MIME főtípusok.....	59
A2. Logikai adatmodell attribútumai és kulcsai.....	60
A3. caEmailBufferParser projekt implementációja.....	61
A4. Közös rendszerkönyvtár.....	62
A5. Elemzési eredményosztályok.....	63
B függelék – A rendszer használata.....	64
C függelék – Képernyőképek	66

Ábrajegyzék

1. ábra – Vállalati szerepkörök és támogatórendszereik a '90-es években és napjainkban.....	8
2. ábra – Kapcsolati háló koncepcionális ábrája	16
3. ábra –Téma – Résztvevő térkép koncepcionális ábrája.....	17
4. ábra – Felderített folyamattérkép koncepcionális ábrája.....	17
5. ábra –Interneten keresztül történő e-mail kézbesítés koncepcionális ábrája	20
6. ábra –Kommunikáció általános modellje	25
7. ábra – Kommunikáció-elemzés fogalmi modelljének sematikus ábrája	25
8. ábra – Vállalati kommunikációt elemző rendszer funkcionális hierarchiája.....	28
9. ábra – Kommunikációt elemző rendszer architektúrája és funkcionális felosztása.....	36
10. ábra –Logikai adatmodell Entitás – Relációs Diagramja	42
11. ábra – Tervezett Csillag-séma Entitás- Reláció Diagramja.....	43
12. ábra – Az elemző adatbázis csillag-sémája	44
13. ábra – A rendszer általános elemzőfelülete	53
14. ábra – Kapcsolati háló vizualizációs modulja	54
15. ábra – Téma-résztvevő elemzés vizualizációs modulja.....	55
16. ábra – Folyamat felderítés vizualizációs modulja	56
A1. ábra – Az e-mail pufferező modul (caEmailBufferParser) UML osztálydiagramja	61
A2. ábra - Közös rendszerkönyvtár fontosabb osztályai és enumerációi	62
A3. ábra – Elemzési eredményosztályok.....	63
C1. ábra – Kapcsolati háló elemzés eredménye – személy nézet, irányítás nélküli mód.....	66
C2. ábra – Kapcsolati háló elemzés eredménye – csoport nézet, irányított mód	66
C3. ábra – A rendszer által regisztrált ismeretlen résztvevők listája	67
C4. ábra – Résztvevő cserélő eszköz képernyőképe	67
C5. ábra – 'Muráti Ákos' résztvevő rekordja és címei.....	68
C6. ábra – 'Muráti Család' nevű csoport tagjai és csoportjai.....	68

Táblázatjegyzék

1. táblázat – 'Muráti Ákos' kifejezés kódolásai egy e-mail fejléc mező értékeként	22
2. táblázat – Gyakori e-mail fejléc mezők és jelentésük	22
3. táblázat – E-mailek tartalmi elemeit leíró mezők és jelentésük	23
4. táblázat – E-mail betöltő alrendszer által használt puffer tábla	38
5. táblázat – Fejlesztéshez és teszteléshez használt hardver- és szoftverkonfiguráció	46
6. táblázat – Közös rendszerkönyvtár moduljai és szerepük	47
7. táblázat – Közös rendszerkönyvtár fontosabb osztályai és szerepük	47
8. táblázat – Kiszolgáló oldali rendszeradminisztrációs eljárások	48
9. táblázat – Kommunikáció-elemzést megvalósító eljárások és eredménytípusaik	49
10. táblázat – Kliens funkciókat megvalósító űrlapok és leírásuk	49
11. táblázat – Ábrázolást megvalósító vezérlők osztályai és leírásaik	50
12. táblázat – E-mail betöltő alrendszer modulok tesztjének mérési eredményei	52
13. táblázat – GLEE ábrázolási teljesítménye különböző csomópont és él konfigurációk esetében	56
14. táblázat – Egy kommunikációs elemzési üzenet alatt átvihető résztvevők száma	57
A1. táblázat – MIME főtípusok és magyarázataik néhány gyakori példával	59
A2. táblázat – Logikai adatmodell egyedei, kulcsai és attribútumai	60

1. Bevezető

A kommunikáció ősidők óta az emberi tevékenység mindennapos része. Segítségével képesek vagyunk kifejezni gondolatainkat, érzéseinket és szándékainkat, ezáltal az együttműködés, hatáskifejtés és visszacsatolás alapvető eszköze.

Céljaink elérésében kulcs szerepe van annak, hogy milyen módon, mennyire sikeresen vagyunk képesek kommunikálni – vagy éppen elhallgatni – cselekedeteink hatását és szándékainkat, hiszen tetteink önmagukban nem elegendőek személyiségünk külső megítélésében.

Nagy általánosságban ugyanez elmondható a vállalatok esetében is, a különbség csupán annyi, hogy egy szervezetben a vezetők által kijelölt célok elérése nem csak egy, hanem jóval több ember együttes munkájától, fellépésétől és kommunikációjától függ, amelyek összehangolása és követése nehéz feladat.

Napjainkban számtalan eszköz és technológia áll a szervezetek rendelkezésére, amelyek hatékonyan támogatják a munkavégzést, a munkafolyamatok automatizálását, az erőforrások felhasználását és segítséget nyújtanak az adatok, információk és a tudás előállításában, kinyerésében vagy terjesztésében, elősegítve a döntési helyzetek megfelelő, jól informált előkészítését, a szereplők együttműködését, kommunikációját és a mobilitást.

Az üzleti intelligencia megoldások széleskörű terjedésével a vállalatok egyre szélesebb és részletesebb betekintést kaphatnak belső és külső környezetük részleteiről, viszonyairól és működéséről, azonban jelenleg még nincs olyan széles körben elérhető általános megoldás, amely az emberi kommunikációs folyamatok feltérképezésére és elemzésére szolgálna.

Diplomamunkám célja a fenti problémakör megvizsgálásával egy olyan modell és rendszer megtervezése és elkészítése, amely képes a vállalati kommunikáció hatékony leírására és további elemzés céljából történő felhasználására modern mérnöki megoldások felhasználásával.

A feladat bonyolultságát és lehetőségeimet figyelembe véve a számtalan kommunikációs csatorna közül a bemeneti oldal implementálását az e-mailek gyűjtésére és feldolgozására szűkítettem, melynek egyedi problémakörét részletesen bemutatom, ám a jövőbeli fejlesztéseket szem előtt tartva a modell és a rendszer is fel lett készítve más csatornákról érkező üzenetek leírására és elemzésére.

Ugyanakkor diplomamunkámnak nem célja a kommunikáció tartalmi részeinek szemantikus értelmezése, hiszen e problémakör önmagában is hatalmas, amelyre különálló tudományág épül. A tartalmi kivonatolást ezért csak a tervezésnél vettem figyelembe a későbbi megvalósítás céljából.

2. Az üzleti intelligencia

Az üzleti intelligencia megoldások nagyban segítik a vállalatok működésének hatékonyságnövelését és javítják a tervezhetőségét, amit a versenyelőny megszerzésének egyik fő eszközeként jellemeznek. Valamennyi IT piackutató felmérése alapján az informatikai-vezetők évek óta az egyik első számú prioritásként nevezik meg az üzleti intelligencia megoldások alkalmazását és fejlesztését. Ahhoz, hogy tisztában lehessünk e fogalom jelentésével, meg kell értenünk azokat az igényeket és körülményeket, amelyek a különböző vállalati információs rendszerek kialakulásához és végül az üzleti intelligencia megjelenéséhez vezettek.

2.1. Vállalati információs rendszerek fejlődése

E fejezetben röviden összefoglalom az üzleti intelligencia rendszerek létrejöttében szerepet játszó Vállalati Információs Rendszerek fejlődésének fontosabb állomásait *Dave Ensor*, *Ian Stevenson* [1], a *Laudon* házaspár [2], *Daniel J. Power* [3], *Raffai Mária* [4], *Kővári Attila* [5] és *Kókai Béláné* [6] munkái alapján.

Munkamegosztás szempontjából egy vállalatban belül különböző funkcionális területeket különböztethetünk meg, mint például a pénzügyi-számviteli, kereskedelmi, humánerőforrás, stb.. Ugyanakkor a szervezeti struktúra vertikális felbontás alapján pedig általánosan 4 szintre osztható, melyben a legalsó, operatív szinten a szervezet alapvető feladatainak, tranzakcióinak és munkafolyamatainak ellátása történik. A következő szinten a funkcionális vezetők végzik feladataikat, akik a területüknek megfelelően az operatív tevékenység koordinálásával és ellenőrzésével foglalkoznak. A döntés-előkészítést végző vezetők, a hierarchia következő lépcsőfokát képviselik, akik a vállalat stratégiai céljainak megfelelően egy magasabb szinten ellenőrzik és hangolják össze annak tevékenységét és ezeknek megfelelően határoznak meg középtávú célokat és adnak javaslatokat a felsővezetés számára a hosszú távú célok kijelölésére. A vállalat függőleges felosztásának csúcsán a stratégiai – azaz felsővezetői – szint található, ahol a hosszú távú stratégiai és tulajdonosi célok kijelölése történik. A fentieket figyelembe véve érthető, hogy a szervezetek minden egyes szintjén és területén szükség van a pontos és megfelelő információkra, melyek előállítása, tárolása, tisztítása, rendszerezése, összegzése és elemzése hatékony eszközök támogatása nélkül lehetetlen feladat.

A modern értelemben vett üzleti intelligencia rendszerek fejlődése az 1960-as évek második feléig vezethető vissza. Ebben az időben a fejlett adatfeldolgozó technikáknak köszönhetően a szervezetek már rendelkeztek olyan egymástól elszigetelt rendszerekkel, amelyek lehetővé tették a vállalat ügyfél-

menedzsmentjét¹, tranzakcióinak², folyamatainak³, adminisztratív feladatainak automatizált ellátását⁴ és hatékonyabb erőforrás kezelését⁵ is. Összefoglaló néven az ilyen megoldásokat Végrehajtást Támogató Rendszereknek, azaz *Operation Support System (OSS)* megoldásoknak hívjuk, melyek fejlettebb utódai végzik napjainkban is a vállalatok operatív tevékenységének döntő részét. Bár az Integrált Vállalatirányítási Rendszerek technológiai és koncepcionális téren is jelentős fejlődésen mentek keresztül, csak korlátozott mértékben képesek a vállalat egészét érintő kérdések megválaszolására, és a felsőbb szinteken történő vezetői munka támogatására. Az *OSS*-ek elsősorban arra lettek optimalizálva, hogy működésük során nagy mennyiségű adatot rögzítsenek, kezeljenek és dolgozzanak fel a vállalat folytonos és megbízható működését biztosítva, így sem adatszerkezetükben, sem pedig erőforrásaikban nem képesek megfelelően támogatni a vezetői igényeknek megfelelő időszakos vagy ad-hoc elemzések végrehajtását vagy azok több szempontból történő prezentálását.

A fenti problémák figyelembevételével születtek meg az első az úgynevezett vezetői információs rendszerek (*Management Information System, MIS*), melyek modern értelemben az első jelentéskészítő eszközöknek tekinthetők. Kezdetben ezek a szoftverek gyakorlatilag egyszerű, interakció nélküli jelentésgeneráló szoftverek voltak, melyek részben az *OSS*-ek, részben pedig saját adatbázisukra támaszkodva készítettek előre definiált igények alapján riportokat a magasabb szinten lévő vezetők számára. Legnagyobb hátrányuk az volt, hogy lekérdezések és jelentések összeállításához olyan elemző és tervező szakemberek segítségére volt szükség, akik az üzleti igényeket és kérdéseket képesek voltak az adatok szintjére lefordítani.

Az *IBM* a vezetői információs rendszerek fénykorában kezdett egy olyan interaktív rendszer fejlesztésébe, amely könnyen érthető és kezelhető felületet adott hétköznapi felhasználók kezébe, az adatok geológiai szempontból történő analizálására és megjelenítésére. A szoftver a *Geodata Analysis and Display System (GADS)* nevet viselte, amit geológiai adatokat elemző és megjelenítő rendszerként fordíthatunk magyarra. A hetvenes évek második felére a terméket már több rendőrség, tűzoltóság, önkormányzat és közlekedési társaság is alkalmazta különböző területi, övezeti vagy útvonal-tervezési feladatok megoldásához.

Az eltérő ábrázolási és elemzési szempontok mellett ugyancsak ebben az időben kezdett egyre növekvő és tudatosabb figyelem irányulni a felső üzleti döntések informatikai támogatására. A szervezet-kutatás önálló tudományággá fejlődésének és az üzleti tevékenység rendszer alapú felfogásának köszönhetően kialakultak a különböző döntéstámogató rendszerek (*Decision Support*

¹ CRM – *Customer Relationship Management*, azaz Ügyfélkezelési rendszer

² TPS – *Transaction Processing System*, azaz Tranzakció-feldolgozó rendszerek

³ PCS – *Process Control System*, WF - *Workflow System*, azaz folyamatirányító és munkafolyamat rendszerek

⁴ OAS – *Office Automation System*, azaz irodaautomatizálási rendszerek.

⁵ ERP – *Enterprise Resource Planning*, azaz vállalatirányítási rendszer

System, DSS). A *DSS* nagy előnye abban jelentkezett, hogy fejlettebb, felhasználóbarát felületeivel a végfelhasználók kezébe adta a rendszerben lévő információkkal kapcsolatos jelentések összeállításának és ábrázolásának lehetőségét, ugyanakkor egyszerre támogatta a fizikai adatokon és az arra épülő logikai modellen megfogalmazott lekérdezések kezelését is.

A döntéstámogató és vezetői információs rendszerek értelmezése az évek-évtizedek alatt többször is átértékelődött, ugyanakkor fejlődésük törvényszerűnek tűnik az eltérő és folyamatosan változó vállalati igények mellett. A gyártók ezeket az eltérő igényeket *MIS* és *DSS* megoldásaik köré szerveződő különböző funkciócsomagokkal próbálták megoldani. Ezek a középvezetői szinteken általánosságban megfelelőek voltak, de a felsővezetők teljes vállalatra vonatkozó információs igényeit, már nem tudták teljes mértékben kielégíteni. E fontos igény pótlására alakultak ki az *Executive Information System (EIS)*⁶ megoldások, melyek személyre szabottan nyújtottak a felsővezetők számára magas szintű és aggregált pénzügyi valamint statisztikai információkat a vállalat egészére vonatkozóan.

A mainframe⁷ rendszerek letűnésével, a gyors többprocesszoros szerverek és személyi számítógépek előretörésével a vállalatok informatikai rendszerei is átalakultak. A szerver és kliens oldali megvalósítások informatikában jól megfigyelhető ciklikussága a központi kiszolgálók, adatbázisok és végfelhasználók szerepeit és lehetőségeit is átformálta. A terminálokat felváltották a munkaállomásokon futó kliensek, melyek segítségével az egyes felhasználók testre szabhatták elemzési környezetüket, tárolhatták lekérdezéseiket és egyedi nézeteket definiálhattak a szerver által felkínált lehetőségeken kívül. Az egyénileg testreszabott és szerepkörök szerinti elemzés támogatása fokozatosan beépült a kiszolgáló oldali logikába, a felhasználók jobb menedzselhetősége végett. A logikai rétegek felett az egyes szerepköröknek megfelelően kialakult az üzleti modell rétege is, amely a logikai modellt fordítja le az egyes szerepköröket ellátó felhasználók által használt üzleti fogalmi kifejezésekre.

Az információs rendszerek fejlődése megkövetelte az adatbázis-kezelő rendszerek (*Database Management System, DBMS*) fejlődését is, hiszen teljesítményük kulcsfontosságú a hatékony működéshez. A kezdeti szigetszerű rendszerek adatbázisaikkal a teljes vállalatot átfogó kérdések megválaszolására sem technikailag, sem pedig adattartalmukban nem voltak képesek vagy elegendőek. A *DSS* rendszerekkel együtt, azok elemzéseit kiszolgálva megjelentek olyan központi adatbázisok – adattárházak –, melyek a szervezet különálló tranzakciós adatbázisaiból vették adataikat. Ezzel a megközelítéssel egyrészt megoldották a tranzakciós rendszerek elemzések miatt bekövetkező felesleges terhelésének problémáját, másrészt saját, kimondottan elemzésekre kialakított adatbázis

⁶ A szakirodalom néhány helyen *Executive Support System (ESS)* néven is hivatkoznak e rendszerekre

⁷ Nagy teljesítményű központi számítógép, melyek erőforrásait és számítási kapacitását a terminálon rákapcsolódott felhasználók időosztásos elven használták

sémák használatával a lekérdezések komplexitását is csökkentették. Az évek folyamán mind az analitikus adattárolási és modellezési módszereket, mind pedig az ezek kiszolgálását támogató adatbázisszerverek képességeit is folyamatosan tökéletesítették, melyek végül elvezettek az *Online Analytical Processing (OLAP)* technológia kialakulásához. Az *OLAP* adatbázismotorokat és az ezekre épülő végfelhasználói eszközöket kimondottan úgy tervezték, hogy az összetett elemzési feladatokat minél egyszerűbb legyen megadni és velük hatékonyan lehessen végrehajtani, miközben segít a felhasználónak végigfűrkészni akár magasabb szintű összegzett vagy alacsonyabb szintű részletezett adatokról is legyen szó. Az *OLAP* technológia jellemzően támogatja az adatok több – például idő vagy tér - dimenzió menti elemzését, hierarchikus és idősoros lekérdezését, valamint a „mi van akkor, ha...” típusú kérdések felvetését és elemzését is.

Az *OLAP* adattárházak olyan adatbázisok, melyeket kimondottan elemzési céllal hoztak létre egy adott téma köré felépítve. Jellemzőjük, hogy az adattárházba kerülő adatok nem változnak, történeti jellegűek és forrásaik tranzakciós rendszerek. Az adattárházak és a forrás-adatbázisok eltérő sémája miatt szükség van olyan szoftverekre, melyek a forrásadatokat kinyerésük után azokat struktúrájukban, típusukban, mértékegységeikben és felbontásukban konzisztensen átalakítják és betölti a központi elemző adatbázisba. Az e célt szolgáló eszközök elnevezése a kivonatolás (*extract*), átalakítás (*transform*), és betöltés (*load*) funkcióikból adódó angol szavak rövidítése, vagyis az *ETL* lett.

A '90-es évek második felében az *OLAP* adattárházakra épülő *DSS* rendszerek olyan kiterjedt szolgáltatásokkal és elemzési potenciállal rendelkeztek, melyek jóval túlmutattak a klasszikus értelemben vett informatikai döntéstámogatáson. A *DSS*, mint elnevezés akarva-akaratlanul is, de szűkítette és behatárolta e rendszerek felhasználását és főleg eladhatóságának lehetőségeit, ezáltal a piac mintegy kikövetelte egy régi-új értelmezés és elnevezés, a *Business Intelligence (BI)* használatát.

2.2. Az üzleti intelligencia fogalma

A *Business Intelligence* fogalmát, elsőként *H.P. Luhn* vezette be az *IBM Journal* 1958. októberi számában megjelent „*A Business Intelligence System*” cikkében. *Luhn* szerint az üzleti intelligencia rendszerek „*olyan feldolgozó rendszerek lesznek, melyek az összegyűjtött adatokat és dokumentumokat automatikusan képesek lesznek dekódolni, kinyerni azok információtartalmát, és a szervezet igénycsoportjainak megfelelően érdeklődési profilokba sorolni*” [7]. Bár *Luhn* cikkében még a vállalatok által felhalmozott és létrehozott papíralapú dokumentumokban, mikrofilmekben és képernyőképekben gondolkodott, elképzelése messzemenően meghaladta korát és több évtizeddel előre vetítette az adattárházak és tudásbázisok működését.

A modern értelemben vett *Business Intelligence* fogalom első megjelenéséhez azonban még több mint három évtizedet kellett várni. Szakmai körökben az üzleti intelligencia atyjaként emlegetett *Howard Dresner* egy 1989-es értekezésében a *Gartner Inc.* elemzőjeként az akkor népszerű *DSS* és *EIS*

rendszerek képességeit összegezve egy olyan fogalmat szeretett volna alkotni, amely általánosabb definíciót nyújt az információk mennyiségi elemzéséről, felhasználók szélesebb körét érintve. [8]

Az üzleti intelligencia fogalmának széleskörű elterjedésére – ahogy azt a vállalati információs rendszerek fejlődésénél is láthattuk -, a '90-es évek végéig várni kellett, azt követően azonban az információs technológiák egyik legkeresettebb és legjövödelmezőbb ágazatává vált. Az elmúlt évtizedben számos könyv, cikk, értekezés és beszámoló jelent meg a témával kapcsolatban. Alábbiakban az ezek által nyújtott definíciókat felsorolva próbálok meg általános képet kialakítani a *Business Intelligence* 21. századi fogalmáról.

A *Drexel University* és *New Jersey Institute of Technology* egyetem professzorai „*Business Intelligence Techniques*” című könyvükben az üzleti intelligenciát az alábbiakban definiálták. „*Olyan adatelemző, jelentéskészítő és lekérdező eszközök, amelyek képesek segíteni az üzleti felhasználóknak abban, hogy átvergődjenek az adatok tengerén, értékes információkat előállítva.*” [9]

A vállalati információs rendszerekkel több, mint 20 éve foglalkozó *Laudon* házaspár az Üzleti intelligenciát gyakorlatilag a döntéstámogató rendszerekkel tekinti azonosnak, míg *Daniel J. Power* „*A Brief History of Decision Support System*” (magyarul „*A döntéstámogató rendszerek rövid története*”) című munkájában gyakorlatilag ugyanerre a következtetésre jut.

Az *IBM* weboldalán hasonló filozófiát fogalmaz meg, mivel véleményük szerint az „üzleti intelligencia megoldások olyan szoftverek, amelyek összekötik az embereket az információval, hogy döntéseiket a lehető legegyszerűbb módon hozhassák meg.”

Rod Newing a *BI* megoldások széleskörű elterjedéséről írt „*Higher intelligence*” című cikkében az alábbiakban határozta meg fogalmunkat. „*Az üzleti intelligencia egy koncepció, amely körülbelül húsz évvel ezelőtt indult önálló útjára az eredetileg vezetőknek készült döntéstámogató rendszerektől. ... A szervezetek azóta rájöttek, hogy döntéshozóknak a vállalat minden szintjén és területén szükségük van naprakész, releváns és pontos információkra, legyen szó akár értékesítésről, termelésről, kiskereskedelemről, logisztikáról vagy más szervezeti-egységről.*” [10]

A *HP Business Intelligence* blogjában *Vicky Farell* a *BI*-t egy újgenerációs rugalmas jelentéskészítő és elemző eszközként említi, ami az évek során önálló életre kelt. Véleménye szerint a *Business Intelligence* rendszerek – az *OLAP* technológiák segítségével - webes portálokon keresztül az informatikai képzettség nélküli felhasználók kezébe adja a lekérdezések összeállítását, vizualizálását és megosztását, többnyire historikus adatokon. [11]

A *Microsoft Magyarország* az üzleti intelligenciával kapcsolatban az alábbiakat írja weboldalán: „*Vállalati szervezete az üzleti intelligenciát hívhatja segítségül a versenyelőnyt és nyereségességet biztosító döntéshozatal támogatásához. Az üzleti intelligencia révén integrálhatja, tárolhatja, elemezheti, és jelentések készítéséhez használhatja fel az adatokat, ami teljesebb betekintést, és végső soron jobb üzleti teljesítményt biztosít.*”

A téma súlyát és aktualitását az is mutatja, hogy a világ egyik neves IT elemző és tanácsadó vállalata, a *Gartner Inc.* egy a saját CIO felmérése alapján a 2009-es évre a Top-10 IT stratégiai technológiák egyikeként a *Business Intelligence* –t nevezte meg. Elemzéseik alapján az üzleti intelligencia alkalmazása közvetlenül pozitív hatással lehet a vállalatok üzleti teljesítményére és drámaian javíthatja a kitűzött célok elérését az üzletmenet minden szintjén - a vállalat stratégiai döntéseitől az operatív feladatok szintjéig. [12]

Egy másik neves IT piacelemző vállalat, az *IDC* egy 2009-es *Business Intelligence* megoldásokat vizsgáló tanulmányában a *BI* fogalmát már pontosabban és kézzel foghatóbban definiálta [13]. *Vesset* és *McDonough* – a tanulmány szerzői - az üzleti intelligencia piacot két részre osztották: általános célú- (*QRA*⁸) illetve speciális elemző⁹ rendszerekre.

Az általános célú *QRA* szoftverek kategóriája magában foglalja a kifejezetten ad-hoc adathozzáféréshez tervezett lekérdező, multi-dimenzionális elemző, valamint jelentéskészítő eszközöket és vezérlőpultokat is. Jellemzőjük, hogy mind az IT, mind pedig az üzleti felhasználók könnyen tudják alkalmazni e megoldásokat. Ugyanakkor ide sorolhatók a szerver oldali *OLAP* technológiák és modellező eszközök, valamint az előre konfigurált szoftvercsomagok is, melyek tipikus funkcionális területek üzleti modelljeit és transzformációs eszközeit tartalmazzák.

A speciális elemzőrendszerek alatt olyan adatbányászati és statisztikai eszközöket értünk, amelyek minták, hasonlóságok, kapcsolatok és rejtett információk feltérképezésével foglalkoznak, vagy komplexebb feldolgozást igényelnek egy hagyományos *QRA* tevékenységnél. E kategóriába sorolhatók például neurális hálók modelljét alkalmazó elemzőrendszerek, vagy az olyan statisztikai szoftverek is, amelyek olyan speciális alkalmazási területekre fókuszálnak, mint az ipari kísérletek, klinikai tesztelések, kutatási- vagy valós idejű statisztikai-adatok elemzése.

A fentieket összefoglalva tehát elmondhatjuk, hogy az üzleti intelligencia fogalmán egy olyan koncepciót értünk, ami biztosítja a különböző stratégiai szerepköröket ellátó személyek munkájához, döntéseikhez szükséges releváns és kellő részletezettségű információkat a rendelkezésére álló adatok elemzése alapján, a vállalat jobb teljesítményének érdekében. Üzleti intelligencia megoldásokon pedig olyan eszközöket és technológiákat értük, amelyek a vállalati adatvagyon felhalmozásától kezdve a felhasználók rugalmas informálásáig bezáróan e koncepció megvalósulását támogatják.

Ide tartoznak többek között az operatív és más külső forrásokból származó adatok gyűjtését, transzformálását és betöltését támogató ***ETL eszközök***, az ***elemzések végrehajtását támogató***

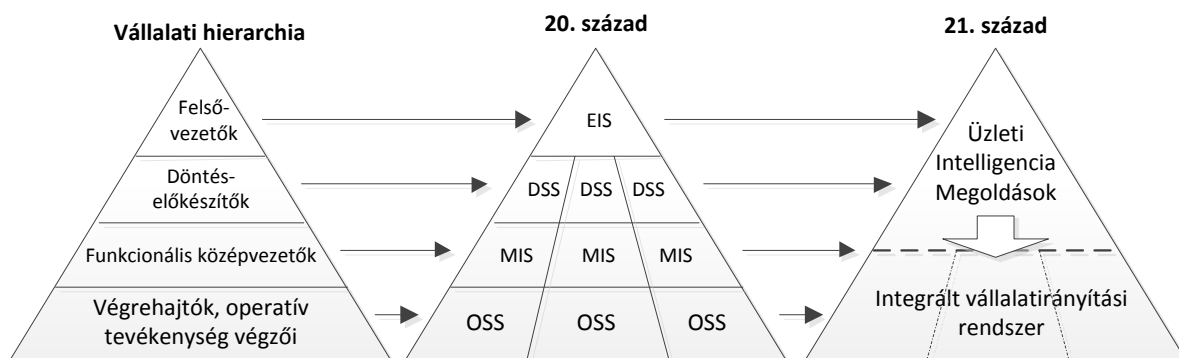
⁸ *QRA* - A szakirodalom így hivatkozik a hagyományos lekérdező, jelentéskészítő és elemző azaz „*Query, Report and Analysis*” megoldásokra

⁹ A szerzőpáros az általam használt speciális elemző rendszerekre az angol *Advanced Analytics* fogalommal hivatkozik

adatbázis és adattárház megoldások, a modellezést és optimalizálást segítő adatmenedzsment szoftverek és az összefüggések feltárására, előrejelzések készítésére szolgáló matematikai vagy statisztikai módszereket alkalmazó *adatbányászati eljárások* is. Ide sorolhatók azok a *vékony és vastag-kliens megoldások* is, melyek az üzleti és informatikai felhasználók igényeinek egyaránt megfelelő *lekérdező, jelentéskészítő és elemző felületeket* valósítanak meg, valamint az eredmények *ábrázolását, finomítását és terjesztését* támogató eszközök is.

2.3. Trendek és megoldások napjainkban

A 20. század végéig a *MIS*, *DSS* és *EIS* rendszerek ugyan fokozatosan átvették és beépítették az újabb és hatékonyabb technológiákat, de, mint önálló rendszerek, fokozatosan tűnnek el és olvadnak be (1. ábra) az üzleti intelligencia fogalmkörébe és eszközkészletébe. Ugyanakkor a 21. század üzleti szoftvereivel szemben támasztott követelmények közül a széleskörű, integrált funkcionalitás mellett a mobil és kényelmi szolgáltatások is már alapelvárásnak számítanak.



1. ábra – Vállalati szerepkörök és támogatórendszereik a '90-es években és napjainkban

Az újabb dinamikus webes technológiák megjelenésével a vállalati szinten drága és nehezen menedzselhető kliens oldali alkalmazásokat felváltották a szinte bárhonnán elérhető interaktív vékony kliens megoldások, melyek testreszabható webes portálokon keresztül szolgálják az egyes szerepköröknek megfelelő felhasználói igényeket.

A modern *BI* megoldásokban az ad-hoc elemzések mellett továbbra is igény mutatkozik a statikus követelményekkel rendelkező, naprakész és személyes riportok generálására. Az üzleti intelligencia rendszerek többsége éppen ezért rendelkezik olyan prezentációs modullal, amely a felhasználó által kért friss adatokon generált riportokat bizonyos eseményekhez vagy időpontokhoz kötve automatikusan eljuttatja a címzettekhez, gyakorlatilag tetszőleges formátumban, legyen az egy egyszerű *PDF* jelentés, *Microsoft Excel* táblázat, *XML* állomány, interaktív *Flash* vagy egy weboldal.

Megfigyelhető tendencia, hogy az üzleti intelligencia megoldások által prezentált információk jól hasznosíthatóak a vállalat szinte összes szintjén. A hierarchia felsőbb – vezetői és döntéshozói -

szintjei mellett a *BI* megoldások alkalmazása kezd egyre inkább teret nyerni a mindennapi operatív feladatok ellátásában résztvevő szinteken is. [10]

A jelenlegi trendek alapján mondhatjuk, hogy a jövőben nagy fejlődés várható a *BI* rendszerek kooperációs lehetőségeinek fejlesztésében, ahol a prezentált információkat a résztvevők közösen, együttműködve értelmezhetik és szabadon transzformálhatják saját megszokott irodai szoftvereiken a központi kiszolgáló terhelése nélkül.

Az *IDC* korábban említett tanulmánya az üzleti intelligencia eszközök első számú beszállítójának az *SAP AG* szoftveróriást nevezte meg 2008-as piaci részesedése alapján. A sorban öt követő gyártók a *SAS*, *IBM*, *Oracle* és *Microsoft*, melyek együtt a *BI* világpiac majdnem 60 %-át uralják, ezért érdemes e vállalatok elképzeléseit és megoldásait is figyelembe venni az üzleti intelligencia témájának vizsgálatakor. Itt érdemes megemlíteni, hogy a fenti gyártók által szállított szolgáltatások több mint 80%-ban a *Vesset* és *McDonough* által definiált általános célú végfelhasználói *BI* eszközöket takarja.

A világ vezető integrált vállalatirányítási rendszereinek beszállítója, az *SAP AG* egyben az üzleti intelligencia piac legnagyobb részesedéssel rendelkező szereplője, amely az *SAP NetWeaver BW* megoldásával teljes körű szoftveres *BI* szolgáltatást nyújt ügyfelei részére. A szoftvercsomag többek között saját fejlesztésű *OLAP* technológiára épülő adattárház megoldását, a heterogén rendszerek integrálását segítő *ETL* eszközeit, valamint a vállalati infrastruktúrához igazodó *SAP Enterprise Portal* webes keretrendszert tartalmazza. A felhasználók önkiszolgáló webes felületeken vagy a kimondottan erre kifejlesztett lekérdező, elemző és jelentéskészítő eszközök segítségével végezhetik ad-hoc elemzéseiket vagy készíthetik el jelentéseiket, vezérlőpultjaikat és ábráikat, melyeket a kollaborációs és publikációs platformokon keresztül terjeszthetnek. A vállalat piaci részesedését biztosítva 2007-ben felvásárolta az üzleti elemző szoftverekre szakosodott, *Crystal Reports* jelentéskészítő szoftveréről híres *BusinessObjects* –et.

A *SAS Institute Inc.* *Enterprise Business Intelligence* és *Business Visualization* megoldásaiban saját *OLAP* alapú adattárházával, integrációs és elemző eszközeivel kínálja üzleti intelligencia szoftvereit. A végfelhasználók szerepkörüknek megfelelően egy testre szabható önkiszolgáló portálon vagy vastag kliensen keresztül végezhetnek historikus vagy prediktív elemzéseket, adat- és szövegbányászatot vagy készíthetnek statisztikákat, előrejelzéseket a jobb és gyorsabb döntések elősegítéséhez.

Az *IBM* adattárház és tartalommenedzsment megoldása az *Information Management* csomag részeként elérhető, saját fejlesztésű *Infosphere Warehouse* és *Smart Analytics System* adattárház és elemző szoftverekre épül, amelyek tartalmi elemzéseket, adat- és szövegbányászatot, valamint integrációs feladatok ellátását is támogatják. Ugyanakkor a *Cognos* 2007-es akvizíciója során szerzett *Cognos Business Intelligence* megoldása *OLAP* képes vagy relációs adatforrásra épülő valódi üzleti

intelligencia szoftvercsomag, amelyben a felhasználók önkiszolgáló webes felületen futtathatnak lekérdezéseket, ad-hoc elemzéseket, illetve készíthetnek jelentéseket és vezérlőpultokat is. Az információk megosztását a rendszer az *IBM Cognos Go!* szolgáltatásain keresztül támogatja, legyen szó e-mailküldésről, Microsoft Office integrációról, vagy mobil eszközön való megjelenítésről.

Az *Oracle Inc.* az alábbiakban foglalja össze üzleti intelligencia megoldását: „Az *Oracle Business Intelligence (BI)* az alaptechnológiák és alkalmazások legátfogóbb portfólióját kínáló megoldás, amely az összefüggések felismerésén alapul, átgondolt vállalatirányítást szolgálja piacvezető BI-alkalmazásokkal, üzleti adatelemző platformmal és adattárházakkal.” [14] A vállalat *OLAP* opciót is alkalmazó saját *Oracle Database* adatbázis-kezelő megoldása mellett *Oracle Warehouse Builder* névre keresztelt adattárház tervező, modellező és *ETL* eszközével igyekszik kielégíteni a felhasználók igényeit. Az *Oracle Business Intelligence Suite* szoftvercsomag kiválóan együttműködik heterogén *OLAP* vagy relációs-adatbázisokkal és egyéb adatforrásokkal. A felhasználók a nagyvállalati infrastruktúrába jól illeszthető önkiszolgáló portálon keresztül végezhetik előre definiált és ad-hoc elemzéseiket, vagy készíthetnek vezérlőpultokat és jelentéseket. Az információkat a *BI Publisher* és *Delivery* komponensek segítségével, tetszőleges formátumban és módon oszthatják meg munkatársaikkal vagy partnereikkel, akár mobileszközökön keresztül is. Az *Oracle* olyan analitikai alkalmazásokat is értékesít, amelyek lefedik a specifikus üzleti folyamatok és iparágak széles spektrumát, továbbá lehetőséget biztosítanak geológiai adatok ábrázolására is. A cég évek óta erősíti pozícióját a BI megoldások piacán, amelyet a *Siebel* 2005-ös, a *Hyperion* 2007-es felvásárlásával alapozott meg. A *Sun Microsystems* 2009-es akvizíciójával a szoftveres megoldások mellett már teljes körű hardver-szolgáltatásokat is nyújt.

A *Microsoft* az utóbbi években rengeteget tett *Business Intelligence* szolgáltatásainak fejlesztése érdekében, melyet a BI technológiára specializálódott *ProClarity* 2006-os felvásárlásával alapozott meg. A *Microsoft Business Intelligence* megoldásában az adattárházak menedzselését, elemzését és külső rendszerek integrációját a gyárilag beépített *OLAP* technológiával és jelentés-generáló mechanizmusával felvértezett¹⁰ új generációs *SQL Server* adatbázismotor és *ETL* eszközök biztosítják. A *SharePoint Server* önkiszolgáló webes felületein a felhasználók szerepköreiknek megfelelően szabhatják testre oldalait és nézeteiket, megoszthatnak információkat, készíthetnek vezérlőpultokat és – akár más felhasználókkal együttműködve – kérdezhetik le, elemezhetik és ábrázolhatják az eredményeket a *Microsoft Excel* képességeit felhasználva. A *Microsoft* a geológiai információk megjelenítésére saját webes térképészeti megoldását, a *Bing Maps* –et alkalmazza.

¹⁰ Az említett funkciókat a *Microsoft SQL Server Analysis Services* és *SQL Server Reporting Services* szolgáltatások látják el

A létező megoldások mellett azonban a felhasználók egyre többször találhatnak új lehetőségeket az üzleti intelligencia rendszerek felhasználására – az ipari termelés monitorozásától kezdve, a kórházi felépülési mutatókon keresztül, az ügyek részleteiben kutató ügyvédekig. [15]

Napjainkban a vállalati levelezés, üzenetküldés és telefonálás a cégen belüli és kívüli kommunikáció egyik legfontosabb tényezője, így a jövőben megjelenhet az igény a tényleges kommunikációs folyamatok feltérképezésére és azok üzleti szempontból történő felhasználására is.

Diplomatervem aktualitását az jelenti, hogy jelenleg nyilvánosan nem ismert olyan általánosan használható üzleti intelligencia megoldás, amely képes lenne a kommunikációs csatornák figyelésével, üzenetek gyűjtésével és a kommunikáció elemzésével a vállalatok számára üzleti szempontból használható információkat előállítani.

3. Stratégiai fázis

E fejezet célja a vállalati kommunikáció-elemzés problémakörének feltárása, a vállalati kommunikációs környezet sajátosságainak és lehetőségeinek felmérése a felmerülő elemzési igények figyelembevételével.

3.1. Környezet felmérése

3.1.1. Vállalati környezet és kommunikáció kapcsolata

A vállalatok gazdasági tevékenységet folytató szervezetek, amelyek fogyasztói szükségleteinek kielégítéséhez és szolgáltatások nyújtásához belső struktúrájukat valamilyen munkamegosztás alapján alakítják. A szervezetek többsége profit-orientált berendezkedésű, ami megköveteli az erőforrások hatékony és optimális felosztását, melyben a kommunikációnak kulcsszerepe van. Minősége jelentősen meghatározza a működés milyenségét és kialakult mechanizmusaival még a vezetési struktúra jellemzőire is hatással lehet. *Albert Tucker* híres foglydilemma problémájával mutat rá *Bakacsi Gyula* arra, hogy kommunikáció nélkül a „foglyok”, egy mindannyiuknak hasznos cél érdekében képtelenek összehangoltan cselekedni, vagyis kvázi szervezetként működni. [16]

A szervezeti magatartás kutatása az utóbbi időben olyan tudományterületté vált, amely a szervezeteken belül a magatartásra gyakorolt egyéni, csoportos- és strukturális hatásokat vizsgálja a célból, hogy ezeket az ismereteket a szervezet hatékonyságának növelésére alkalmazza. Röviden összefoglalva azzal foglalkozik, hogy a szervezeti struktúrát és működést létrehozó személyek és csoportok mit tesznek a szervezetben és hogyan befolyásolják annak teljesítményét. A szervezeti magatartás nagy hangsúlyt fektet többek között a munkakörök kialakítására és a munkavégzés szervezésére, amelyek a termelékenységgel, az emberi teljesítménnyel és a menedzsmenttel függnek össze. [16]

Russell Ackoff a modern szervezet-kutatás úttörője szervezeti-kommunikációs modelljében közvetlen összefüggést mutatott ki a hibás kommunikáció és a szervezet rossz hatékonyságát előidéző hibás döntések között. A csoportos kommunikációt vizsgáló *Janet Beavin Bavelas* pszichológus pedig arra mutatott rá először, hogy feladatokat teljesítő csoportok esetében a kialakított kommunikációs sémák miként jelölnek ki hatalmi pozícióba kerülő embereket, hogyan alakítják a csoport struktúráját, milyen módon hatnak a tagok megelégedettségére a csoportban elfoglalt helyük alapján és nem utolsósorban miként hatnak a feladat megoldására.

A jobb szervezeti teljesítmény érdekében a csoportok kialakításához, átszervezéséhez és az alulról szerveződő vezetés kijelöléséhez a vállalatoknak a jövőben szükséges lesz a belső és külső kommunikáció vizsgálatára a szervezeti pszichológiai hatások figyelembevételével.

A vállalatoknak legalább annyira kell figyelniük a kommunikációra és az ezzel összefüggő működésükre, folyamataikra, mint amekkora hangsúlyt fektetnek a pénzügyi tervezésre vagy a piac felmérésére, mivel a kommunikáció-elemzés a modern szervezetfejlesztés hangsúlyos tényezőjévé vált.

3.1.2. Tipikus vállalati kommunikációs csatornák

Nyilvánvaló, hogy a vállalati kommunikáció elemzése csak akkor valósulhat meg, ha a szervezet képes a megfelelő csatornákon keresztül zajló kommunikációs folyamatokat és azok szereplőit az elemzési céloknak megfelelően észlelni és regisztrálni. Ezt csak abban az esetben lehet gépekkel megoldani, ha a kommunikáció átviteli közeget képesek vagyunk figyelni és azon keresztül a résztvevőket is azonosítani.

A felhasználók manapság számtalan különböző módon kommunikálhatnak egymással, amely a csatornák figyelését első pillantásra megnehezítheti. Ugyanakkor a távközlés és informatika konvergenciájának köszönhetően, az eltérő kommunikációs csatornák közötti különbségek elmosódnak és az egységes digitális, IP alapú megoldások felé tolódnak el, melyek ellenőrizhetősége egyre egyszerűbb és olcsóbb megoldásokat igényel.

Az alábbiakban a vállalkozások által leginkább alkalmazott kommunikációs csatornákat és jellemzőiket veszem soron.

Telefon

A legtöbb nagyobb vállalat már rendelkezik valamilyen saját vezetékes vagy VOIP telefonközponttal egyrészt belső kommunikációjának egyszerűsítése, másrészt pedig hívásköltségeinek csökkentése és kézbe tartása érdekében. Az ilyen szervezeteknél elterjedt megoldás, hogy az alkalmazottak saját vonallal vagy mellékkel rendelkeznek, melyek kezeléséről a központi címtár gondoskodik. A címtárak lehetőséget biztosítanak mobil-munkaállomások esetén a telefonvonalak dinamikus kiosztásához, ugyanakkor a hívások szereplői ezáltal könnyen azonosíthatóak lesznek bárhol is végzik éppen munkájukat.

Egy telefonbeszélgetés egy vagy több személy között folytatódhat, ahol a szereplők akár menet közben is becsatlakozhatnak vagy kiléphetnek, melyek tényét a telefonközpontok rögzíthetik. Már manapság is számtalan olyan üzleti terület létezik – például ügyfélszolgálatok -, ahol a hívások tényén kívül magát a beszélgetés tartalmát is rögzítik a későbbi visszahallgathatóság céljából. A beszélgetések tartalmi feldolgozásához a modern hang és beszéd-felismerési eljárásokkal lehetőség nyílik azok szöveges címkézésére is a további elemzést segítő.

A lehetőségek felsorolásakor természetesen nem szabad megfeledkezni a mobiltelefon megoldásokról sem. A vállalatok szemszögéből a relatíve olcsó és gyakorlatilag infrastrukturális költségek nélkül fenntartható hangátviteli és adat-szolgáltatások vonzó alternatívát jelentenek a saját telefonközpontok

üzemeltetésével szemben. Ugyanakkor a mobiltelefon hálózatokon keresztül bonyolított telefonbeszélgetések az előfizető vállalatok számára általában nem érhetők el, mivel e megoldásokban használt technológiák és eszközök külső szolgáltatók kezében vannak, melyek tevékenysége állami engedélyhez kötött.

E-mail

A vállalatok másik, leginkább népszerű kommunikációs megoldása az elektronikus levelezés. Mivel a vállalatok szinte kivétel nélkül rendelkeznek saját levelezőszerverekkel, a küldött és fogadott e-mailek figyelése egyszerű feladat. A szereplők azonosítása a fenntartott címtárak segítségével ugyancsak egyszerű feladat a belső címek és csoporttagságok ismeretében, a domain-en kívüli e-mail címek azonosítására pedig a vállalatok által karbantartott CRM adatbázisok használhatók.

Az elektronikus levelezés sajátossága - a hagyományos levelezéstől eltérően -, hogy a feladó által küldött üzenet minden egyes címzettnél önálló entitásként jelenik meg, a terjedési viszonyoktól függően akár egymástól eltérő időpontokban. Előfordulhat például olyan eset, hogy egy betelt postafiók esetén az egyik szereplő csak napokkal később kapja meg üzenetét, ami az elemzés során figyelembevett válaszidők tekintetében nagy jelentőséget hordozhat. Ugyanakkor az egyes fogadó felek ugyanazt az üzenetet eltérő kézbesítési módokkal, közvetlenül, másolatként vagy titkos másolatként is megkaphatják, melyek az értelmezés szempontjából más-más jelentéssel bírhat.

Tartalom szempontjából az e-mail az egyik legváltozatosabb kommunikációs forma, hiszen az általa hordozott üzenet tartalmazhat sima vagy HTML szöveget, beágyazott tartalmat, de csatolmányként tetszőleges fájlokat is képes tárolni, melyet a strukturálatlan és összetett tartalom dekódolásakor figyelembe kell venni a későbbi elemzés szempontjából.

Azonnali üzenetküldés

Az azonnali üzenetküldés, - angolul *Instant Messaging (IM)* - napjaink egyik legelterjedtebb szöveges ugyanakkor valós-idejű kommunikációs megoldása. Általában meglévő e-mail vagy VOIP alkalmazásokhoz szokták társítani és jellemzőjük, hogy a felhasználók jelenléti állapotukat közzétéve élőben cseveghetnek rövid szöveges üzeneteken keresztül, de bizonyos megoldások akár a fájlok küldését is lehetővé teszik.

Az e-mailekhez hasonlóan, az azonnali üzenetek naplózása és elemzése ugyancsak könnyen megvalósítható vállalati kereteken belül, amennyiben a vállalat már rendelkezik valamilyen saját azonnali üzenetküldő szolgáltatással.

Fax

A fax a papír alapú vállalati működés egyik legöregebb és legegyszerűbb kézbesítési formája, mely a dokumentumok továbbítását telefonos hálózaton teszi lehetővé. A faxok megfelelő gyűjtése ugyanakkor csak akkor lehetséges, ha a vállalati fax vagy telefonhálózat az e-mailekhez hasonlóan központi rendszeren keresztül kerülnek továbbításra. A feladó és címzett személyek azonosítása a

telefonszámok vagy mellékek központi címtárból történő kiolvasásával, vagy a faxszerverek azonosítási mechanizmusainak használatával lehetséges.

A faxok tartalmi elemzés szempontjából bizonyos mértékben az e-mailekre hasonlítanak, hiszen a fax szerverek megjelenésével a közvetlen küldés előtt elfogott faxok szinte tetszőleges formátumú dokumentumok, fájlok lehetnek – melyek tartalmi elemzése a csatolmányokhoz hasonlóan megoldható –, azonban a telefonhálózaton keresztüli továbbítás után a fogadott faxok már képi információként állnak csak rendelkezésre, melyek értelmezése OCR¹¹ technológiák bevonását igénylik.

3.2. A vállalati kommunikáció-elemzés lehetőségei

A kommunikáció vizsgálatával feltérképezhetők a kommunikáció résztvevőinek, vagyis a szervezet tagjainak és csoportjainak belső és külső kapcsolatai, valamint a kommunikáció során ellátott tipikus szerepeik, melyek által olyan összefüggések fedezhetők fel, amik a szervezeti felépítésből nem feltétlen olvashatók ki.

A kommunikáció tartalmának megfelelő kivonatolásával és szövegbányászati módszerekkel megállapíthatók azok a témák, amik az adott személyekhez, vagy csoportokhoz köthetők, melyek által lehetőséget biztosítanak élő téma-személy térképek készítésére is a szervezeten belül.

A szervezet kommunikációjának elemzésével egy olyan felülnézeti képet kaphatunk folyamatainkról, melyekben az adott ügyek könnyen nyomon követhetők és időben felismerhetők az olyan kommunikációs elakadások, melyek hátráltathatják a vállalat működését.

Nem utolsó sorban a kommunikációs csatornák és azok szereplői a vállalat erőforrásai, melyek mennyiségi elemzésével kimutatható azok terheltsége. Ezeket az adatokat felhasználva az erőforrások átszervezésével javítható azok hatékonyabb kihasználtsága.

Összefoglalásul elmondható, hogy a vállalati kommunikáció mennyiségi és minőségi elemzésével lehetőséget kapunk kommunikációs erőforrásaink terhelési és teljesítménybeli vizsgálatához, segítségével feltérképezhetjük **szervezetünk tagjainak belső és külső kapcsolatait, téma-résztvevő térképeit** és nem utolsó sorban nyomon követhetünk konkrét **ügymeneti folyamatokat** is.

¹¹ OCR – *Optical Character Recognition*, azaz optikai karakterfelismerő technológia, ami lehetővé teszi képeken lévő karakterek felismerését és szöveggé alakítását.

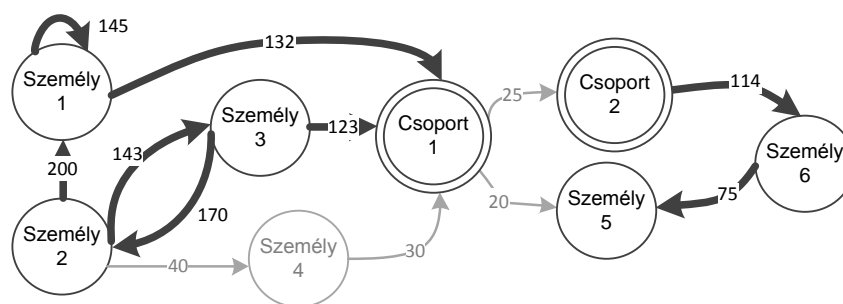
3.3. Az elemzési igények és ábrázolás koncepciója

Az elemzést végző felhasználóknak lehetőséget kell biztosítani a kommunikáció fontosabb tulajdonságai alapján történő részletes szűrésekre. Ilyen jellemző a kommunikáció időpontjának behatárolása egy időintervallum megadásával, a kommunikáció szereplőinek, akár konkrét szerepükkel – feladó vagy címzett – történő szűkítése, vagy üzenetek témájának kijelölése.

Az alábbiakban a különböző elemzési eredmények ábrázolásának koncepcionális terveit ismertetem. Az ábrázolások kidolgozásánál az értelmezést elősegítve fontos szem előtt tartani, hogy az egyes szimbólumok a különböző környezetekben is egységesen jelenjenek meg. Ezért a további ábrázolások során a személyeket szimpla körrel, a csoportokat pedig dupla vonalú körrel fogom jelölni, a témákat téglalappal, a hangsúlyosabb kommunikációs momentumokat pedig félkövér betűtípussal vagy vastagabb éllel jelölöm.

3.3.1. Kapcsolati háló

A kommunikációban résztvevő szereplők kapcsolatának legkézenfekvőbb ábrázolására kapcsolati-hálót alkalmazok, amely egyaránt tartalmazhat külső és belső személyeket és csoportokat is. A 2. ábrán látható egy általam elképzelt tipikus kapcsolati háló. A kommunikáció egyes szereplőit a gráf csomópontjai prezentálják. Két csomópont között akkor található él, ha közöttük történt kommunikáció. Az él irányítottsága a kommunikáció irányát jelöli, míg annak súlyozása azt mutatja meg, hogy a két szereplő hányszor kommunikált egymással.

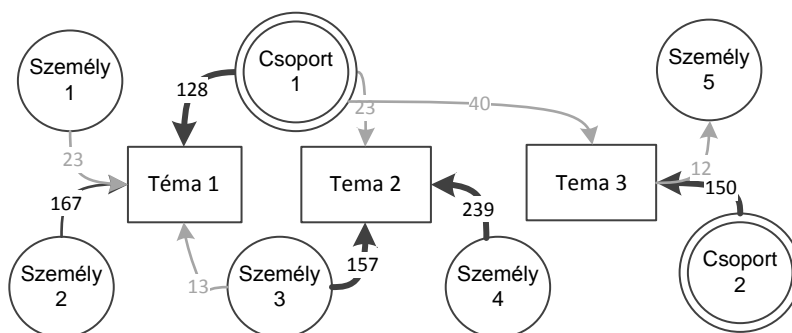


2. ábra – Kapcsolati háló koncepcionális ábrája

Az élek a súlyok által történő vastagításával grafikusan könnyen kiemelhetők és leolvashatók az egymással gyakran kommunikáló szereplők, melyek halmazát a továbbiakban klikknek nevezzük.

3.3.2. Téma – Résztvevő térkép

Az üzenetek tartalmának megfelelő módon történő kivonatolásával - kulcsszavainak logikai szinten történő csoportosításával - felismerhető azok témája. E témák és a kommunikációban résztvevő személyek és csoportok összekötésével kaphatjuk meg a téma – résztvevő térképet, melynek ábrázolási koncepcióját a 3. ábra mutatja.



A téma-részrtvevő térkép olyan gráf, amelyek kétféle csomópontot tartalmazhat: témát vagy résztvevőt prezentáló csomópontot. Él csak téma és résztvevő között futhat és azt a jelentést hordozza, hogy az adott személy az adott témával kapcsolatban állt egy másik személlyel folytatott kommunikációja során. Amennyiben az él a személytől a téma felé mutat, az azt jelenti, hogy a kommunikációt a személy kezdeményezte ahhoz a témához kapcsolódóan. A témából a személy fele mutató él azt a jelentést hordozza, hogy a kommunikáció során az adott témában a kommunikáció résztvevője a fogadó fél szerepét töltötte be. Az élek súlyozása és vastagítása a kapcsolati hálókhoz hasonlóan a kommunikáció gyakoriságát jelöli.

A szervezetek megosztott munkavégzése során gyakran merülhet fel igény a párbeszéd felülről történő vizsgálatára, amelyből megállapítható, hogy egy adott ügymenet kommunikációs folyamata éppen hol tart, vagy mely résztvevőknél akadt el. A kommunikáció során az egyes üzenetek külön példányokban és akár eltérő időpontokban jelennek meg címzetteknel. Ugyanakkor a kommunikációs tevékenység során a résztvevők válaszaikkal és továbbított üzeneteikkel újabb és újabb példányokat hoznak létre, melyek egymásra hivatkoznak. A 4. ábra egy ilyen, a résztvevőknél külön-külön példányként megjelenő üzeneteket és kapcsolataikat ábrázolja az idő-tengely mentén.

3.4. Megvalósítandó rendszerfunkciók összefoglalása

A cél egy olyan kommunikációt elemző üzleti intelligencia rendszer megvalósítása, amely a végfelhasználók számára egyszerű és hatékony elemző megoldást nyújt a kommunikációban szereplő **részrtvevők kapcsolatainak elemzéséhez, a téma-resztvevő kapcsolatok feltárásához és ügymeneti folyamatok felderítéséhez.**

A kapott eredményeket a rendszernek az elemzés ábrázolási koncepciójának megfelelően kell **ábrázolnia.**

Az elemzések során lehessen szűrni az olyan általános kommunikációs jellemzőkre, mint adott résztvevők körére, a rendszer üzemeltetője által definiált témákra, vagy, hogy a kommunikáció mely időintervallumba esett.

A **témák központi kezelése** a kommunikációs folyamatok vállalati szinten egységes logikai csoportosítását hivatott megvalósítani.

Ugyanakkor feladat az **e-mailek központi gyűjtése**, az elemzés szempontjából fontos kommunikációs **részletek (pl. szereplők, témák, stb.) feldolgozása** és későbbi **elemzés** céljából történő **eltárolása**. Az e-mailek megfigyelésének elősegítéséhez a levelezőszerver a rajta áthaladó leveleket egy mappába fogja gyűjteni.

A rendszernek alkalmasnak kell lennie a résztvevők azonosításához használt **elérhetőségek**, valamint ezek **tulajdonosainak** és **csoporttagsági** adatainak kezelésére is.

A tervezés során figyelembe kell venni, hogy a rendszer későbbi fejlesztések keretében **más kommunikációs csatornákon történő kommunikáció elemzésére** is alkalmas legyen, ám ezek megvalósítása már nem tárgya ennek a dolgozatnak.

4. Elemzési fázis

E fejezet célja az előző stratégiai részben ismertetett célok, fogalmak és funkciók pontosítása és azok kisebb funkcionális összetevőkre bontása, valamint az elemzés fogalmi modelljének felállítása.

4.1. E-mailek az elemzés szemszögéből

Ahogy azt a bevezetőben már jeleztem, diplomamunkám és a megvalósított kommunikáció elemző rendszer elsősorban az e-mailek gyűjtésére, feldolgozására és elemzésére fókuszál, ezért ebben az alfejezetben az e-mailek technikai sajátosságait ismertetem.

4.1.1. Történeti áttekintés

Az internetes levelezés hosszú évtizedek alatt fejlődött azzá, amit ma e-mailnek nevezünk, és emberek milliárdjai használnak. A számítógépeken keresztül forgalmazott elektronikus levelezés az 1960-as évekig vezethető vissza. Kezdetben az ugyanazon *mainframe*-ekre csatlakozott terminál felhasználók alkalmazták az egymás közötti üzenetek cseréjére, majd később különböző megoldások segítségével lehetővé vált több egymással összekötött szerver közötti kommunikáció is. Az Internet őseinek tekintett *ARPANET* hetvenes évekbeli térnyerésével együtt terjedt az elektronikus levelezés népszerűsége is. A már meglévő hálózatok fokozatos összekapcsolásával megjelent az igény az addig szeparáltan működő levelezőrendszerek összekötésére, együttműködésére és egy közös elektronikus levelezési platform kidolgozására. A ma is használatos „@” jel bevezetését egy bizonyos *Ray Tomlinson* nevéhez köthetjük, aki e szimbólum felhasználásával címezte meg első e-mail üzenetét, elválasztva egymástól a címzett felhasználó nevét és annak számítógépnevét.

4.1.2. Szabványok és ajánlások

Az Internetes protokoll-szabványok kidolgozásáért felelős *Internet Engineering Task Force (IETF)* csoport 1982-ben az *ARPANET* Szöveges Üzenet-formátum szabványa¹² alapján megalkotta az *RFC 822* szabványt, ami csaknem 20 évig volt a hivatalos Internetes Üzenet-formátum (*Internet Message Format*) szabványa.

Ezzel a formátum-szabvánnyal azonban az az alapvető probléma, hogy csak 7-bites *ASCII* karakterek használatát engedélyezi, amely lehetetlenné teszi az eredeti kódtáblán kívüli speciális és nemzeti karakterek - például ékezetes betűk – használatát. 1996-ban e hiányosság pótlására alkotta meg az *IETF* az *RFC 2045 – 2049* szabványtervezeteket, a különböző kódolású és formátumú tartalmak

¹² A dokumentum *RFC 733 – Standard for the format of ARPA Network Text Messages* címen jelent meg

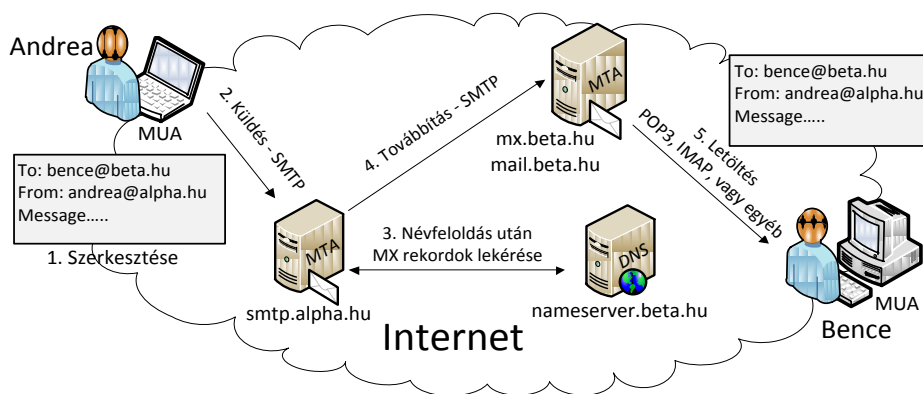
kezelésére, amit *Multipurpose Internet Mail Extensions (MIME)* névvel emlegetnek, amit magyar nyelvre *Többfunkciós Internet Üzenet-kiterjesztésként* fordíthatunk.

A *MIME* szabványok [17] segítségével a levelező alkalmazások kétféleképpen is kódolhatják a nem ASCII tartalmakat. Az egyik megoldás a többnyire sima szövegek átvitelére alkalmazott *quoted printable* formátum, míg a másik, bináris tartalmak – például csatolmányok – 7-bites leírásához általában a *base64* tartalomkódolót szokták használni.

A *quoted printable* alapötlete, hogy bármely 8-bites jel leírható kódjának hexadecimális számként történő ábrázolásával. E kódolásnál a szöveg *ASCII* karaktereit tartalmazó részeit önmagával, míg az *ASCII* kódtáblán kívül eső karaktereket egy „=” jel után saját hexadecimális kódjukkal írjuk le. Az önábrázolás alól természetesen kivételt képez maga az „=” jel, illetve a sor végi nyomtatásban nem megjelenő (*whitespace*) karakterek, melyeket szintén hexadecimális kódjukkal kell leírni.

A *Base64* kódolás elve a fentiekkel szemben a bitek bináris-átcsoportosításán alapul. A kódolandó 8-bites jelek bináris kódját egymás mellé írva 24 bites csoportokra osztjuk, ezután a csoportokból sorban veszünk 6-bitenként bináris számokat, melyek értékét *ASCII* karakterekké alakítjuk. Ezzel a megoldással egy 8 bites tartalom 3 szimbólumát 4 darab *ASCII* karakterrel írhatjuk le.

Az utóbbi években ugyanakkor több kiegészítés is készült a fenti dokumentumokhoz, így a 2001-ben elkészült és 2005-ben javasolt szabvánnyá előlépett, *RFC 822* –t felváltó *RFC 2822*, és a *MIME RFC* szabványtervek további tartalmi és biztonsági módosításai is megjelentek. Az *IETF* az Üzenet-formátum szabványosításán túl az e-mailek hálózaton történő kézbesítési protokollján a *Simple Mail Transfer Protocol (SMTP)* szabványosításán is dolgozik.



5. ábra –Interneten keresztül történő e-mail kézbesítés koncepcionális ábrája

Az e-mailek kézbesítése *SMTP* –n keresztül az 5. ábra alapján, az alábbi módon történik. Első lépésben a feladó megszerkeszti üzenetét, amit levelező kliense¹³ a megfelelő Internet Üzenet-

¹³ A végfelhasználók által elérhető különböző e-mail kliensekre vagy webmail megoldásokra a szakirodalom általában *Mail User Agent (MUA)* kifejezéssel hivatkozik

formátumra alakítva továbbít egy e-mailküldő szolgáltatást nyújtó szervernek. Amennyiben az e-mail domain-en kívüli címre továbbítandó, a levelező szerver a céltartomány levelezőszerverének¹⁴ névfeloldása után *SMTP* -n továbbítja azt. A helyi levelezőszerverekről a címzettek levelező kliensükkel különböző protokollokon keresztül tölthetik le üzeneteiket.

4.1.3. Üzenet-formátum részletei

Az *IETF* által szabványosított formátumú e-mailek két részre oszthatók, a fejlécre illetve az üzenet törzsére, melyekre a szakirodalom *header* illetve *body* kifejezésekkel hivatkozik [18].

E-mailek fejléc része

Az üzenetek fejléc része strukturált szöveges információkat tartalmaz, amelyeket mezőknek hívunk. A mezők többek között a levél feladójáról, címzettjeiről, kézbesítések módjáról, feladásának idejéről, az üzenet tárgyról, azaz gyakorlatilag bármiről tárolhatnak információkat, amit az egyes levelező kliensek és szerverek beleírnak.

A fejléc struktúrája alapján minden mező új sorban kezdődik, melynek legelső karaktere egy nyomtatható *ASCII* karakter, amely egyúttal a mező nevének első karaktere is. A mező értékét annak nevétől a „:” (kettőspont) karakter választja el és addig tart, amíg egy új sor első karaktere egy nyomtatható *ASCII* karakter nem lesz, vagyis addig, amíg nem kezdődik egy új mező. A fentiek értelmében egy mező értéke lehet több soros is, amennyiben az új sorokban szereplő értékek nyomtatásban nem megjelenő karakterrel – például szóközzel, vagy tabulátorral - kezdődnek.

Az e-mailek fejléc részével kapcsolatos pontos szintaktikai megkötéseket az *RFC 5322* szabvány részletezi, ami az *RFC 2822* által nyitva hagyott kérdéseket hivatott tisztázni, melyek közül az e-mail címek tárolására alkalmas mezőket érdemes megemlíteni. A cím mező értékei a tényleges e-mail címen kívül opcionálisan tartalmazhatják az ahhoz tartozó nevet is. Ebben az esetben a cím-értéke a **megnevezés_kifejezése <cím>** formátumot követi, melyben megnevezés kifejezését minden esetben szimpla vagy dupla idézőjelek közé kell tenni. Amennyiben a mező több cím-értéket is tárolhat, a fenti formátumú címeket egymástól vesszővel elválasztva kell leírni.

Gyakran – például az üzenet tárgya vagy éppen egy címzett neve esetében - előfordulhat, hogy egy fejléc mezőben olyan értéket kell tárolnunk, ami nem írható le *ASCII* karakterekkel. Ilyen helyzetekben kell alkalmaznunk a korábban már említett tartalomkódolási eljárásokat, melyeket mezők esetében **=?KARAKTERKÉSZLET?KÓDOLÁS_MÓDJA?KÓDOLT_SZÖVEG?=** formátumban kell megadni. A nyitó és záró szimbólumok között található három paramétert a kérdőjel választja el egymástól, melyben az első a kódolt szöveg karakterkészletét, a második a kódolás módját, a harmadik pedig a konkrét kódolt szöveget tartalmazza. A kódolás módja *quoted printable* esetében

¹⁴ A szakirodalom azokra a szerverekre, amelyek részt vesznek az e-mailek Interneten történő továbbításában *Mail Transfer Agent (MTA)* kifejezéssel hivatkozik.

egy **Q** betű, *Base64* esetében pedig egy **B** betű lehet. Ezek alapján a nevemet egy e-mail cím mező értékeként az 1. táblázatban bemutatott módokon lehet *ASCII* karakterekként ábrázolni az alapján, a használt kódlap és kódoló függvényében.

1. táblázat – 'Muráti Ákos' kifejezés kódolásai egy e-mail fejléc mező értékeként

Kódlap	Kódolás módja	Mező ASCII értéke
iso-8859-2	quoted printable	=?iso-8859-2?Q?'Mur=Elti =C1kos'?=
utf-8	quoted printable	=?utf-8?Q?'Mur=C3=A1ti =C3=81kos'?=
iso-8859-2	base64	=?iso-8859-2?B?J011cuF0aSDBa29zJw==?=
utf-8	base64	=?utf-8?B?J011csOhdGkgw4Frb3Mn?=

Az e-mailek fejléceiben leggyakrabban használt és az elemzőrendszer szempontjából fontos mezőket, rövid leírásukkal a 2. táblázatban foglaltam össze.

2. táblázat – Gyakori e-mail fejléc mezők és jelentésük

Mezőnév	Jelentése
To	A címzett(ek) cím listáját tartalmazó mező
From	Az e-mail üzenet szerzőjének címe
Sender	Az üzenet tényleges feladóját jelölő mező, amennyiben az üzenet szerzője más, mint a feladó – például egy megbízott.
Cc	Cím(ek)re érkező üzenet másolat
Bcc	Cím(ek)re érkező titkos másolatot
Reply-To	Cím, amelyre a választ várja a feladó
Delivered-To	Az a cím, melyre az eredeti üzenetet átirányítva lett a kézbesítés során (pl levelezési lista egy konkrét tagja)
Date	Levél elküldésének dátuma
Subject	Levél tárgya
Message-ID	Levél szerver oldali egyediazonosítója, mely a többszörös kézbesítések elkerülésé
In-reply-to	Arra a MessageId –ra hivatkozik melyre az adott üzenet válaszként lett szerkesztve.
References	Arra a MessageId-ra hivatkozikMelyik M_ID-ra hivatkozva
Received	A levél továbbításában résztvevő szerverek általában feljegyzik az e-mailek fejlécében, hogy mely másik szervertől, mikor és milyen módon kapták az üzenetet továbbításra. E mezők segítségével visszakövethetők az e-mailek útja az Interneten.

E-mailek tartalmi része

A tárgy mező kivételével az e-mail teste hordozza az üzenet szöveges, beágyazott és csatolt, - egyszóval teljes - tartalmát 7-biten kódolva, a *MIME* szabványokban definiált módokon. Egy e-mail teste lehet sima szöveg (*Simple Message*), de akár többféle tartalmat (*Multipart Message*) is hordozhat magában, melyek akár különböző módokon is lehetnek kódolva. Az üzenetek tartalmi elemeit a 3. táblázatban ismertetett mezők írják le és jelölik ki az e-mail testében.

3. táblázat – E-mailek tartalmi elemeit leíró mezők és jelentésük

Mezőnév	Rövid leírása
Content Type	Tartalmi elem MIME típusát tartalmazó mező, amely információ szolgáltat az általa hordozott adat típusáról. <u>Paraméterei lehetnek:</u> <i>charset</i> : Szöveges tartalom esetén a karakterkészletet jelöli. <i>boundary</i> : Az adott tartalmi elemet határoló sorokat adja meg
Content-Disposition	Szövegbeágyazott (<i>inline</i>) vagy csatolt (<i>attachment</i>) tartalmat jelölő mező. Paraméter tartalmazhatja a csatolmány nevét is (<i>filename</i>)
Content-Transfer-Encoding	Az adott tartalmi rész kódolási eljárását írja le (pl.: quoted-printable vagy base64)

Az e-mailekben szereplő tartalmak megfelelő kezeléséhez és megjelenítéséhez az úgynevezett *MIME* típusok¹⁵ nyújtanak segítséget a dekódolást követően. A *MIME* típusok hasonló szerepet látnak el, mint az egyes operációs rendszerekben használatos fájlkiterjesztések, vagyis az állomány tartalmát hivatottak jelezni. Mivel az informatikában ismert számtalan fájlformátum csaknem mindegyikéhez tartozhat saját *MIME* típus, ezért annak kidolgozói a típusok leírót hierarchikusan határozták meg. A hierarchia legfelső szintjén a tartalmak típusaik alapján 8 főcsoportba¹⁶ sorolták, melyek mindegyike további altípusokra bontható.

Az egyes főtípusokat rövid magyarázattal és néhány fontosabb példával az *A függelék A1 táblázatában* ismertetem. Az e-mailek feldolgozása során a tartalmak kibontásához fel kell készítenünk rendszerünket a legkülönbözőbb üzenet-struktúrák és *MIME* típusok feldolgozására.

4.2. Természetes fogalmak összefoglalása

Az alábbiakban összefoglalom a vállalati kommunikációs környezet feltárásakor felmerült természetes fogalmakat és kifejezéseket.

Személy, tag, csoport

A szervezetben valamilyen tevékenységet végző személyeket a szervezet tagjainak nevezzük. A személyek tevékenységi körül, feladataik, helyzetük, vagy tetszőleges közös tulajdonságuk alapján különböző csoportokba tartozhatnak. Egy személy természetesen több csoportnak is tagja lehet. Itt jegyzendő meg, hogy egy személy csoporthoz tartozása időben nem állandó, így egy személy csoporttagsága egy adott időintervallumban értelmezendő.

¹⁵ Az általam használt *MIME* típus kifejezésre az angol nyelvű szakirodalom *MIME-Type* vagy *MIME Media Type* néven hivatkozik

¹⁶ A 9. *MIME* típus az *example* nevet viseli, amely csupán bemutatató és tesztelési célokat szolgál, élő környezetben nem fordul elő.

Kommunikáció, kommunikációs csatorna, kézbesítési mód

Azt a folyamatot vagy eseményt tekinthetjük kommunikációnak, melynek során egy adott személy vagy csoport valamilyen információt közöl egy vagy több –nem feltétlenül más – személynek vagy csoportnak valamilyen kommunikációs csatornát felhasználva.

Kommunikációs csatornán azt a közeget értjük, melyen keresztül a kommunikációs folyamat létrejön a résztvevő felek között. Ugyanakkor az egyes csatornákon az információ a felekhez különböző módokon juthatnak el, melyek az információ tartalmára nem, de az értelmezésre hatással lehetnek. Ilyen például az e-mailek esetében közvetlen, másolat vagy titkos másolat szerinti levelezés.

Üzenet, kivonat, téma

A kommunikációs folyamat során közölt információ fizikai megjelenése valamilyen konkrét formában, ami az egyes kommunikációs csatornák képességeitől függően lehet hang, szöveg, kép vagy tetszőleges fájl. Minden üzenet az információtartalmától, a kontextustól és az előzményektől függően valamilyen jelentést hordoz, ami összefoglalható pár kulcskifejezéssel, melyet kivonatnak hívunk, melyeket nagyobb témákba sorolhatunk.

Résztvevő, szereplő, feladó, adó, küldő, fogadó, vevő

A kommunikációs folyamatban résztvevő személyeket vagy csoportokat értjük e fogalmak alatt. Egy kommunikációs folyamatban résztvevő feleket szereplőknek vagy résztvevőnek nevezünk, melyben kétféle szerepet láthatnak el. A feladónak, adónak vagy küldőnek nevezzük azt a felet, aki a kommunikáció során egy üzenetet megoszt a másik féllel, vagyis a fogadó féllel, vagy más néven vevővel.

Címek, elérhetőségek

A különböző kommunikációs csatornákon az egyes résztvevők eléréséhez ismerni kell azok címeit, telefonszámait, ami alapján az üzenet kézbesíthető számukra, melyek röviden elérhetőségeknek nevezünk. Egy elérhetőség természetesen a kommunikációs csatornához szorosan kötődik, ugyanakkor egy résztvevőnek több elérése is lehet, akár egy kommunikációs csatornán is.

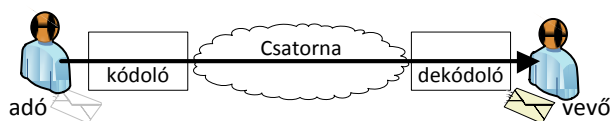
4.3. Kommunikáció-elemzés fogalmi modellje

A cél egy olyan speciális kommunikációs modell létrehozása volt, amely az előző bekezdésben röviden ismertetett elemzési lehetőségek felderítéséhez nyújt megfelelő környezetet, azok egyedi és közös tulajdonságait figyelembe véve.

4.3.1. Általános kommunikációs modell

A klasszikus kommunikációs modell szerint (6. ábra) a küldő üzenetét kódolás után egy átviteli közegen (csatornán) továbbítja a fogadó fél felé, aki az üzenetet dekódolva fogadja azt.

Nyilvánvaló, hogy az emberi kommunikáció gépi elemzését, csak úgy valósíthatjuk meg, ha a küldő és fogadó között képesek vagyunk az üzenetet hordozó jelkészletet elfogni és értelmezni. Ebben az esetben az elemzést végző gép is egy fogadó félként jelenik meg modellünkben.

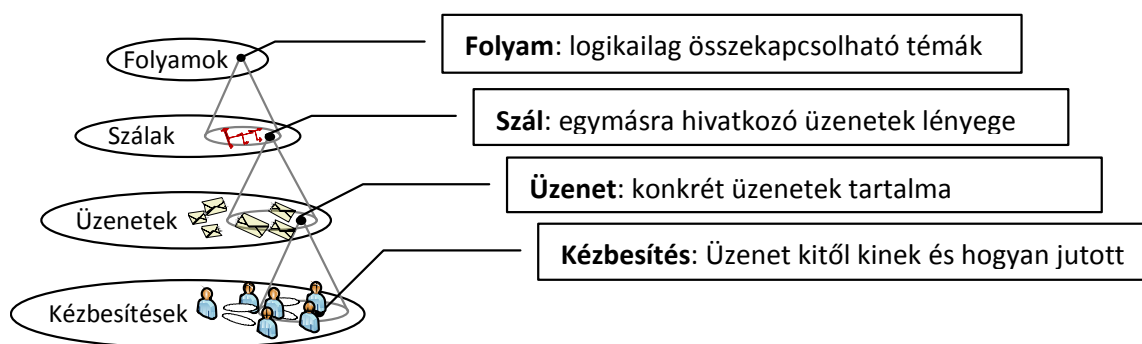


6. ábra –Kommunikáció általános modellje

Az emberi kommunikáció csupán gépekkel történő minőségi elemzése azonban lehetetlen feladat. Soha nem lehetünk képesek olyan eszköz építésére, mely csupán a kommunikációs csatornán keresztül minden esetben képes rekonstruálni a fogadó fél által értelmezett üzenetet. Ez annak köszönhető, hogy az emberi kommunikáció egy dinamikus, állapottal rendelkező tevékenység, melynek helyes értelmezése lehetetlen az *előzmények* és *kontextus* ismerete nélkül. Az értelmezési folyamat közben gyakran találunk olyan hivatkozásokat, utalásokat, amelyek a *kontextus* vagy *előzmények* ismeretében korábban már értelmezésre kerültek– azonban üzenetünkben explicite nem szerepelnek. Az ilyen elemeket a szakirodalom *tacit*, azaz néma elemeknek nevezi, melyek értelmezéséhez bonyolult tartalomkezelési, szövegbányászati, nyelvfeldolgozási és szemantikai módszerekre van szükség.

Belátható, hogy ez a kommunikációs modell nem alkalmas a fenti elemzésekhez, ugyanis nem nyilatkozik a kommunikáció *előzményeiről*, *kontextusáról*, de magáról az üzenet tartalmáról sem, ami a kommunikációs folyamat értelmezésének fontos része.

Ennek eredményeként kialakítottam egy hierarchikus fogalmi modellt (7. ábra), amely megkülönbözteti a *kommunikációs folyamat* és a *tartalomkezelés* egyes szintjeit, de ezeket együtt alkalmazza az elemzési célok eléréséhez.



7. ábra – Kommunikáció-elemzés fogalmi modelljének sematikus ábrája

4.3.2. Kommunikációs folyamat szintjei

Kézbesítések szintje

A kézbesítés kommunikációs modellünk legalsó szintje, amely a kommunikációt egy küldő és egy feladó esetére vezeti vissza, leírva, hogy az adott üzenet mikor és milyen módon került továbbításra egy adott kommunikációs csatornán. Ez a megközelítés azt jelenti, hogy többszereplős kommunikáció esetén modellünkben egy adott üzenet továbbítása a feladótól annyi kézbesítésként jelenik, ahányan azt ténylegesen fogadják. Az elemzések szempontjából tehát e szint képviseli a tényleges szereplők közötti kommunikációs kapcsolatokat és azok pontos jellemzőit.

Itt érdemes megjegyeznünk, hogy a kommunikáció résztvevői személyek vagy csoportok lehetnek. Egy üzenetnek egy küldője van, de több fogadója is lehet, ám csak a pillanatnyi értelmezés kérdése, hogy a feladót vagy címzettet egy személynek vagy egy csoportnak, azaz egy csoport nevében vagy éppen egy csoport részére küldött üzenetnek tekintjük.

Kézbesítési módon a kommunikációs csatornán keresztül történő továbbítás módját értjük, amelyek például e-mailek esetében azt jelenti, hogy egy címzett az adott üzenetet a feladótól közvetlenül, másolatként vagy rejtett másolatként kapta-e meg.

A kézbesítések ez által leírhatók a *feladó*, a *címzett*, az *üzenet*, a *feladás és kézhezvétel ideje*, valamint a *kommunikációs csatorna* és a *kézbesítési mód* megadásával.

Üzenetek szintje

Az üzenetek olyan önálló entitások, melyek a kommunikáció tényleges tartalmát hordozzák. Ezek – a kommunikációs csatornán keresztül továbbítható üzenetek lehetőségeihez mérten – gyakorlatilag tetszőleges típusúak lehetnek, legyen szó akár beszédről, akár írott szövegről, képről vagy videóról. Ezek a tartalmak, ahogy azt már korábban ismertettem a kommunikáció előzményeire és kontextusára explicit és implicit módon építve értelmezhetők a résztvevők által.

Nyilvánvaló, hogy egy többszereplős kommunikációs folyamatban a kézbesítések során egy adott üzenet kerül továbbításra minden egyes résztvevőhöz, ezért egy üzenet szorosan kapcsolódik annak kézbesítéseéhez is.

Az elemzés minősége attól függ, hogy az egyes üzenetek teljes tartalmát miként tudjuk kivonatolni, azaz milyen módon vagyunk képesek a teljes tartalom értelmezése után annak lényegi részeit az előzmények és kontextus ismeretében bizonyos kulcsszavakkal kiemelni és általános témáját meghatározni.

Az üzenetek ez által *teljes tartalmukkal* kivonatolt *kulcsszavaikkal*, *témájukkal*, illetve annak *kézbesítéseivel* jellemezhetők.

Szálak szintje

Szálnak nevezzük az üzeneteknek olyan fa struktúráját, melyben a gyökeret leszámítva minden üzenet a fában az őt megelőző - eggyel alacsonyabb szinten lévő szülő - üzenetre hivatkozva jöttek létre a kommunikáció során, tehát az ezek közötti összefüggés az *előzményre* hivatkozás miatt egyértelműen kimutatható. Egy üzenet, ha nem kötődik direkt egyetlen levélhez sem, önmagában is alkothat egy szálát.

A szálak szintje ezek ismeretében a kommunikációs folyamat előzményeit explicite leíró kapcsolatok szintje. Ez jellemezhető a szálát alkotó *üzenetek* fájával, amely így közvetve magában foglalja a szál összesített *kulcsszavait* és *témáit*, valamint a kézbesítések szintjén keresztül annak *szereplőit* egyaránt.

Folyamok szintje

A kommunikációs folyamat legfelső szintjén a folyamatok állnak: folyamokon azon szálak összességét értjük, amelyek témájukban valamilyen logikai szinten azonosak, ezáltal találhatók közöttük valamilyen szorosabb témabeli összefüggési kapcsolat. E kötés azonban nem olyan egyértelmű, mint a szálaknál megismert explicit hivatkozás, ezért ezek feltérképezéséhez már komolyabb *kontextus* értelmezési és szövegbányászati eljárások szükségesek.

A folyamatok szintjének jellemzői gyakorlatilag nem különböznek a szálaknál ismertektől, hiszen itt is egymással kapcsolatban álló üzenetekről van szó, azonban a folyamatok esetében ez a kapcsolat implicit és logikai természetű.

4.3.3. A tartalomkezelés szintjei

Ahogy láthattuk a kommunikációs-folyamatok különböző szintjeinek alkalmazása megköveteli a tartalmak eltérő részletezettségű kezelését is, melyek alapján három tartalomkezelési szintet határozhatunk meg.

Teljes tartalom szintje

A tartalomkezelés legrészletesebb szintje. Az adó által küldött üzenet tartalmi részeinek összességét értjük, ami magában foglalja a kommunikáció teljes szövegét és esetleges csatolmányait is.

Kivonatok szintje

A kivonatok szintjén az üzenetek teljes tartalmát annak típusától függően jellemzően valamilyen módszerrel szövegesen prezentáljuk, amelyből szövegbányászati módszerekkel kulcsszavakat hozhatunk létre, melyek a teljes tartalom lényegi részeit írják le. A kivonatok elkészítése kétféleképpen történhet a rendszer intelligenciájától függően:

Nem informált kivonatolás során a kulcsszavakat csak teljes tartalom vizsgálata alapján határozzuk meg a különböző kifejezések előfordulása vagy szövegsűrűsége alapján.

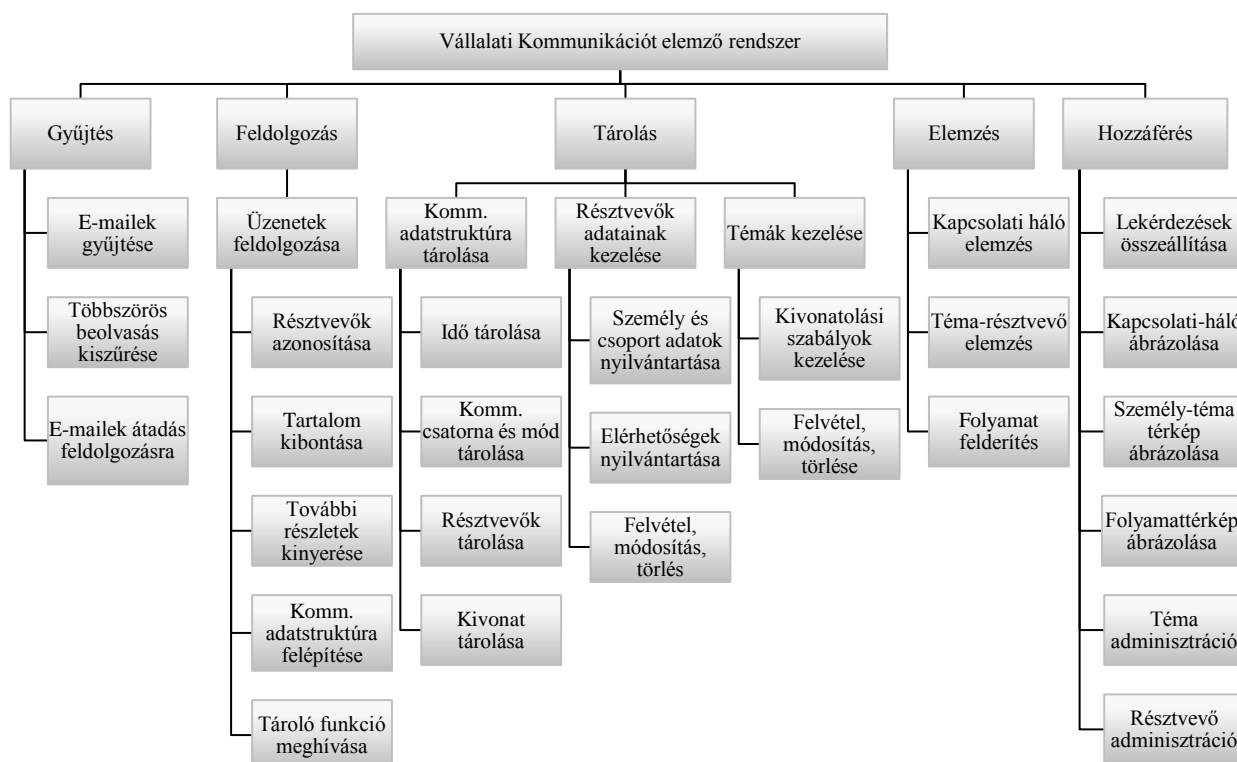
Informált kivonatolás során azonban a tartalmak elemzésekor, már eleve egy adott *kontextushoz* és *előzményhez* tartozó kulcsszavakat próbálunk észlelni, - például a kommunikáció szál ismeretében, - ezáltal a kivonat nem csak a teljes tartalom, hanem a kommunikációs folyamatban betöltött szerepe és kapcsolata alapján is jellemezhetővé válik.

Témák szintje

A témák kezelése elemzési modellünk legabsztraktabb tartalomkezelési szintje, melyben a tartalmak leírása a kivonatolt kulcsszavak alapján, már csak azok témájának megjelölésére szorítkozik, ezáltal lehetővé válik azok logikai csoportosítása. Az elemzés során alkalmazott kivonatolási szabályok, vagy fejlettebb tartalomkezelés esetében ontológia¹⁷ legalább kezdeti felépítése szükséges a helyes működéshez, mely alapvetően emberi intelligenciát és beavatkozást igénylő feladat.

4.4. Funkciók meghatározása

A stratégiai bekezdésben ismertetett rendszerfunkciókat a 8. ábrán látható módon öt nagy és azon belül számos kisebb funkciócsoportra osztottam.



8. ábra – Vállalati kommunikációt elemző rendszer funkcionális hierarchiája

¹⁷ Az *ontológia* számítástechnikai értelemben a valóság egy területéről vagy annak egy sajátos nézőpontjáról alkotott viszonylag kimerítő és szabatos fogalmi séma, melynek szerepe az összetartozó fogalmak kapcsolatainak, viszonyainak és szabályainak szemantikai szempontból történő leírása.

4.4.1. Gyűjtés, megfigyelés

A funkció szerepe az adott kommunikációs csatornán áthaladó üzenetek gyűjtése és átadása a feldolgozást végző funkciós egységnek, megoldva az üzenetek többszörös beolvasásának elkerülését is. Dolgozatomban esetében a megvalósítás csak e-mailek gyűjtésére korlátozódik, de ez a funkcionális egység fogja később tartalmazni a további kommunikációs csatornák figyelését végző funkciókat is.

4.4.2. Feldolgozás

Itt történik a megfigyelés során gyűjtött kommunikációs adatok feldolgozása, melynek során az alábbi feladatköröket kell ellátni:

- Résztevők azonosítása a kommunikációból kinyert elérhetőségek alapján
- Tartami részek kinyerése a különböző tartalomdekódolók segítségével. Elkészül az üzenet kommunikációs modellbeli megfelelője.
- Kommunikáció szempontjából egyéb fontos adatok kinyerése, mint például a csatornán történő kézbesítés módja és a kommunikáció időpontja, stb..
- Kommunikációs modellnek megfelelő adatstruktúra felépítése
- Kommunikációs adatstruktúra átadása elemzés céljából történő tárolása

4.4.3. Tárolás

Ez a funkcionális egység valósítja meg a kommunikációs elemzések szempontjából fontos összes adat tárolását és szolgáltatását, ami az alábbi feladatköröket érinti:

- Kommunikációs adatstruktúra tárolása, ami tartalmazza a kommunikáció szereplőit, tartalmát, idejét és a csatornán történő kézbesítés módját is
- Tárolja a résztvevők, vagyis a személyek és csoportok adatait, tagsági információit és elérhetőségi adatait a címzések feloldásához.
- Témák kezelése, melynek során a rendszer üzemeltetője karbantarthatja a kommunikáció logikai kategorizálásában résztvevő kivonatolási szabályokat

4.4.4. Elemzés

A stratégiai szakaszban ismertetett három elemzési módot megvalósító funkció, mely az elemzéseket a tároló funkció által szolgáltatott kommunikációs adatokon végzi.

- Kapcsolati háló elemzés, melynek feladata a kommunikáció során egymással kapcsolatban álló résztvevők felkutatása
- Téma-résztvevő elemzés, melyben a kommunikáció során felmerült témák és a kommunikáló résztvevők kapcsolatának feltárása a feladat
- Folyamat felderítés, melynek feladata egy adott ügymenethez kapcsolódó kommunikáció felderítése

4.4.5. Hozzáférés

A végfelhasználók számára elérhető felületek, melyek a rendszerhez történő hozzáférést biztosítják. A lekérdezés, ábrázolás és az adminisztráció alapfunkcióit látja el, melyeken keresztül az alábbi funkciókat lehet elérni.

- Lekérdezések összeállításához szükséges felületek elkészítése a különböző elemzések futtatásához
- Kapcsolati háló elemzés összeállítása, indítása és eredményeinek ábrázolása
- Téma-résztvevő elemzés összeállítása, indítása és eredményeinek ábrázolása
- Folyamat felderítés összeállítása, indítása és folyamattérkép ábrázolása
- Témakezelés adminisztrációs felületeinek megvalósítása a tároló funkcionálisegység résztvevő adatok kezeléséhez
- Résztvevő adminisztráció felületeinek megvalósítása a tároló funkcionálisegység témakezeléséhez

4.5. A feldolgozás feladatai és problémái

4.5.1. Személyek azonosítása

Az e-mailek feldolgozása során az elsődlegesen megoldandó feladat a személyek azonosítása, melyhez szükség van egy olyan nyilvántartásra, mely tartalmazza a szervezet tagjainak és külső partnereinek személy és e-mail cím megfeleltetéseit.

Az azonosítás az üzenetek fejléc részében szereplő e-mail címekhez tartozó személyek kikeresésével történik az elemzési sémába illesztés előtt. A feldolgozási időben történő üzenet-személy társítás azért lényeges, mert előfordulhat, hogy egy szervezet működése során egy adott e-mail címet egy munkakör megváltozásával egy másik személynek ad át. Ebben az esetben az azonos címet használó, de időben eltérő két külön személy által küldött és fogadott e-maileket az elemzőrendszer a megfelelő személyekhez fogja társítani.

4.5.2. Csoportok azonosítása

A csoportok kommunikációjának elemzése összetett probléma. Az esetek többségében nem deríthető fel egyértelműen, ha egy adott személy több csoportnak is tagja, így a személy által küldött üzenetről általában nem dönthető el, hogy melyik csoport nevében eljárva küldte a feladó. Az üzenetek és csoportok társítása ezért feldolgozási időben nem oldhatók meg, azokat elemzések során, a személyek felderítésekor kell figyelembe venni azok időben aktuális csoporttagságait alapján, melyhez szükség van egy csoporttagság nyilvántartásra is.

4.5.3. Szálak azonosítása

Szükséges az egyes üzenetek egymásra történő direkt hivatkozásainak feldolgozási időben történő azonosítása, mely lehetővé teszi a szálak építését, így az elemzés idejére már készen állhatnak a logikai szintű kapcsolatok vizsgálatához. A szálak azonosítása közvetlenül lehetséges a feldolgozandó üzenetek *References* és *In-Reply-To* fejléc mezőiben szereplő üzenetazonosítók kinyerésével, melyek egyúttal a hivatkozás módját is elárulják.

4.5.4. Többszörös feldolgozás elkerülése

Előfordulhat, hogy az egyes üzenetek a fizikai kézbesítések során valamilyen oknál fogva többszörösen is a feldolgozó alrendszerhez jutnak. Az ilyen típusú üzenetek viszonylag könnyen kiszűrhetők a fejlécben található *MessageId* mező segítségével, vagy a már feldolgozott üzenetek tartalmi lenyomat (*hash*) alapján történő összevetésével.

4.5.5. Időbeli kauzalitás biztosítása

A vállalaton kívüli hálózatokból érkező üzenetek esetében előfordulhat, hogy az eltérő időbeállítások miatt bizonyos egymásra hivatkozó üzenetek ok-okozati összefüggése időben felcserélődik. Az ilyen anomáliák elkerülése érdekében ezek javítását feldolgozási időben meg kell oldani úgy, hogy a hivatkozó üzenet küldési ideje legalább annyi legyen, mint a hivatkozott üzeneté.

4.5.6. Tartalom kibontása

A modern levelező kliensek képesek üzeneteinket a sima szöveg mellett *HTML* és akár *rich text* formátumokban, szövegbe ágyazott tartalmakkal együtt is kezelni. Az üzenetek olvashatóságát azonban biztosítani kell a gyengébb képességekkel rendelkező – például webes vagy mobil – levelezők számára is. A legtöbb kliens ezt úgy éri el, hogy a levél üzenetét duplikálva kódolja, egyrészt a feladó által beszerkesztett, másrészt pedig a csökkentett funkciójú kliensek számára olvasható módon egyszerű szöveggént. Amennyiben az üzenet rendelkezik szövegbe ágyazott tartalmakkal, úgy azokat csatolmányként is kódolják az alkalmazások.

A feldolgozás során fel kell készítenünk rendszerünket a legkülönbözőbb *MIME* típusú elemek és struktúrák dekódolására és azok kivonatolás szempontjából használható szöveges és csatolt részeinek ideiglenes tárolására is.

Ugyanakkor a többszörös kódolást alkalmazó üzeneteknél a tartalmak kibontása során figyelni kell a feldolgozáshoz leginkább felhasználható egyedi részek kinyerésére, szem előtt tartva a kivonatolás elősegítését és a gazdaságos tárhely felhasználást is.

4.5.7. Tartalmak kivonatolása

A feldolgozás során kinyert tartalmak lényegi részeinek kivonatolása az elemzés előkészítésének egyik legfontosabb része, hiszen a kivonatolás során keletkező kulcsszavak segítségével történik az egyes üzenetek magasabb szintű tematikai, logikai összekapcsolása, melyek generálásához szövegbányászati megoldások szükségesek.

4.5.8. Vállalati környezet előismerete

A feldolgozás és elemzés segítéséhez a környezeti előismeretek is szükségesek a szervezettel és a lehetséges kontextusokkal kapcsolatban. Ilyen ismeretek a korábban már említett e-mail cím, személy és csoport nyilvántartások vagy a folyamatok feltárása szempontjából, a tartalomkezelés szinteknél már említett kulcsszó és téma kivonatolási-szabályok és ontológia építése.

4.6. Jelenlegi elemző megoldások áttekintése

Az elemzőrendszer tervezését már létező kommunikációt elemző megoldások felkutatásával kezdtem. Bár rengeteg olyan alkalmazást találtam, amely képes e-mailek vállalati szintű összegyűjtésére és a tartalmukban történő tetszőleges keresésre, de egy kivételével nem találtam olyan megoldást, ami a tartalomkezelésen túl, az ebben a fejezetben definiált elemzési szempontoknak valamilyen szinten is megfelelt volna.

4.6.1. Deep Email Miner

A *Deep Email Miner* egy nyílt forráskódú, JRE 5 verzióra épülő e-mail elemző megoldás, amely alapvetően kétféle funkciót lát el. Képes kirajzolni az üzenetek alapján a személyek kapcsolati hálóját illetve képes listázni a különböző szálak üzeneteit.

A projekt ugyanakkor nem foglalkozik – vagy legalábbis publikusan ad megoldást – a szabványos *Internet Message* formátumban lévő e-mailek feldolgozására, mivel azt adottnak tekinti. Az e-mailek egy egyszerű 3 táblás adatbázis sémában vannak letárolva, az *üzenetek*, *személyek* és a *kézbesítések* egyedeivel. Az üzenetek tartalma csak sima szöveg lehet, csatolmányokat, tartalomkonténereket és más szövegeket nem képes kezelni. A szoftver a csoportok kezelését nem támogatja és egy személyhez csak egy e-mail tartozhat, így sajnos a több címmel rendelkező személyek ugyan azonos névvel, de több csomópontként jelennek meg.

A szoftver kapcsolati háló ábrázoló felülete ugyanakkor kidolgozott, lehet vele szűrni időintervallumra, kézbesítési módra és arra, hogy egy kapcsolat minimum hány üzenetből álljon. A finomításához azonban személyekre nem állítható be szűrőfeltétel.

Az üzenetek címkézése manuális, vagyis a felhasználók belátására van bízva. Összesen kétféle címkét alkalmazhatunk: magán vagy üzleti. Az üzenetek kereséséhez természetesen tartozik egy kulcsszó

modul is, ami a nyílt forráskódú *Word Vector Tool* könyvtárat használja, azonban az csak az alkalmazás memóriájába betöltött üzenetekben tud keresni, mivel a kulcsszavak nem kerülnek tárolásra az adatbázisban.

A *Deep Email Miner* sem koncepciójában, sem pedig architektúrájában nem fedi az általam kitűzött célokat, mivel csoportkezelés, személyszűrés, csatolmányok, tetszőleges címkék és adatbázis alapú perzisztens kulcsszó-keresés hiányában nem alkalmas vállalati kommunikáció elemzésére. Azonban kapcsolati-háló felületén alkalmazott megoldásait rendszerem tervezésekor és fejlesztésekor mindenféleképpen figyelembe kívántam venni.

Összefoglalásul - témám aktualitását hangsúlyozva - elmondhatjuk, hogy jelenleg nem létezik olyan nyilvánosan elérhető eszköz, amely alkalmas lenne vállalati kommunikáció elemzésére.

4.7. Elemzési fázis összefoglalása

A kommunikáció-elemzés témakörének feltárása során ismerttettem a vállalati kommunikáció-elemzés lehetőségeit és jelentőségét a modern szervezetfejlesztésben. Kiemeltem három elemzési módot, - a **kapcsolati háló elemzést**, a **téma-résztevő elemzést**, és a **folyamat felderítést** - és ezekhez definiáltam ábrázolási koncepciókat is. Ezek fogják alkotni a megvalósítandó elemzőrendszer végfelhasználói elemző felületeinek megjelenítési módjait.

Az általános kommunikációs modell és a kiemelt elemzési módok figyelembe vételével megalkottam a kommunikáció elemzés fogalmi modelljét, melyben magát a kommunikációs folyamatot négy, a tartalomkezelést pedig három szintre osztottam. Ezekhez kapcsolódva definiáltam olyan kulcsfogalmakat, mint a kézbesítés, részttevő, üzenet, szál, folyam, tartalom, kivonat és téma, melyek a logikai adatmodell egyedeit hivatottak előkészíteni.

Bemutattam a leggyakoribb vállalati kommunikációs csatornákat és részletesen kifejtettem az elektronikus levelezés működését és üzenetformátumának fontosabb részleteit, meghatározó szabványait és ajánlásait. Ismerttettem ez e-mail üzenetekből kinyerhető, elemzés szempontjából hasznos fejléc mezőket és kitértem a tartalmak dekódolásának részleteire is, melyek a megvalósítandó e-mail feldolgozás fontos részei lesznek.

A vállalati környezet, a fogalmi modell és a kommunikációs csatornák ismeretében definiáltam a feldolgozás fő feladatait és problémáit, melyeknek a személyek, csoportok és szálak azonosítását, az időbeli kauzalitás biztosítását, a többszörös feldolgozás elkerülését, a tartalmak kibontását, kivonatolását és a környezeti előismeretek kezelését neveztem meg.

Az elemzési fázisban ugyanakkor kutattam jelenlegi kommunikációt elemző megoldások után is, melyeket figyelembe kívántam venni az elemzőrendszer megvalósításához. A kutatás eredményeként egyetlen hasonló alkalmazást találtam, a *Deep Email Miner*-t, melyet részletesen szemügyre is vettem.

A felmérés végén megállapítottam, hogy a szoftver csak nagyon kis részben fedi le a dolgozatomban kitűzött elemzési célokat. Témám aktualitását hangsúlyozva végül arra a megállapításra jutottam, hogy jelenleg nem létezik olyan nyilvánosan elérhető eszköz, amely alkalmas lenne vállalati kommunikáció elemzésére.

5. Tervezési fázis

E fejezet célja az előzőekben ismertetett funkciók modulokba szervezése és pontos megtervezése, valamint a fogalmi modell egyedeire és kapcsolataira épülő logikai adatmodell és elemzési adatbázis megtervezése.

5.1. Tervezői döntések

5.1.1. Háromrétegű ügyfél-kiszolgáló architektúra

A rendszer erőforrásainak elosztásakor az elsődleges szempont, hogy minden komponens a saját feladatát a lehető leghatékonyabban láthassa el. Éppen ezért szétválasztottam az adatok fizikai kezelését és a ráépülő alkalmazási logikát, valamint a végfelhasználók által használható prezentációs felületek rétegét.

A kiszolgáló oldalon az adatok fizikai kezelését ellátó adatbázisszerver és egy köztes réteg, az alkalmazási logika kiszolgálója kerül kialakításra, mely a hálózaton rákapcsolódó ügyfelek elemzési kéréseit hajtja végre. A kliens oldalon már csak a kért és szükséges információk jelennek meg, melyeket a felhasználók szabadon transzformálhatnak a kiszolgáló további terhelése nélkül.

A rétegek közötti interfészek átviteli kapacitása a megoldás kritikus pontja, mert míg a kiszolgáló oldalon feltételezhetjük az adatbázisszerver és alkalmazásszerver közötti megfelelő sávszélességet, a kliensek és az alkalmazás-szerver közötti kapcsolatról már nem mondható el ugyanez. Kliens oldalról vékony sávszélességet feltételezve a kommunikációt a lehető legtömörebben kell megvalósítani, melyet a már elküldött információk kliens oldali újrahasznosításával és gyorsítótárazásával tervezem megoldani.

5.1.2. *Web Service* alapú kliens-szerver kapcsolat

A weben történő együttműködés egyik legelterjedtebb módja a *Web Service* –k használata, melyek lehetővé teszik távoli eljárások paraméterezett hívását és válaszok küldését *HTTP* –n keresztül *XML* dokumentumok segítségével. A kiszolgáló eljárásainak hívása így gyakorlatilag bármilyen klienssel megvalósítható, ezért a kliens és szerver közötti együttműködést a platform-független *Web Service* megoldásra tervezem.

5.1.3. E-mailek gyűjtése és feldolgozása

Gyűjtés módja

A kommunikációs üzenetek gyűjtése a leghatékonyabban a továbbítási-láncba történő belépéssel oldható meg, hiszen üzembeállítása jóval kevesebb beavatkozást igényel, mint a felhasználók postafiókjainak elérése. Ez utóbbi megoldással ugyanis az a probléma, hogy a betöltő alrendszernek minden egyes megfigyelt postafiókhoz hozzá kell férnie, amelynek adminisztrálása nehezen megoldható. Ezzel szemben az első megoldásban elég a kimenő és beérkező levelek kiszolgálóit figyelni egy-egy beékeléssel a teljes csatorna elemzéséhez. Az e-mailek megfigyelésének elősegítéséhez a levelezőszerver a rajta áthaladó leveleket egy mappába fogja gyűjteni, melyeket a pufferelő modul a feldolgozó és betöltő modulnak továbbít.

Gyűjtés és feldolgozás külön modulban

Bár mindkét funkció az e-mailek elemzőrendszerbe töltését látja el, külön modulban történő megvalósításuk szükséges. Az e-mailek gyűjtése a kommunikációs csatornába történő belépéssel azt jelenti, hogy a gyűjtést végző modul a levelezőszerveren fut, azonban a feldolgozást erőforrás igényes feladatai miatt egy másik gépen ajánlott elvégezni a levelezőszerver indokolatlan terhelését elkerülve.

Puffer tábla és aszinkron feldolgozás

Ugyanakkor a gyűjtés elsődleges feladata, hogy az elfogott üzeneteket átadja a feldolgozást végző modulnak, azonban ez az átadás nem történhet szinkron módon, hiszen a feldolgozás (a résztvevők, tartalmak és egyéb információk kibontásával) jóval időigényesebb feladat. Az *e-mail pufferelő modul* egy puffer szerepét betöltő adatbázis táblába fogja beszúrni a feldolgozásra váró üzeneteket, melyeket később a feldolgozó modul innen vesz el.

A puffer tábla ugyanakkor a betöltési fázisnál fontos **többszörös feldolgozás elkerülését** is segíti a táblára vonatkozó megkötésekkel, így az ilyen típusú üzenetek már beszúrási időben el lesznek utasítva.

Közvetlen adatbázis elérés

A rendszer megvalósításánál alapvető szempont, hogy a feldolgozás során az adatokat úgy kell előkészíteni, hogy az elemzésekhez a hatalmas halmazok eléréséhez a lehető legkevesebb adatbázis műveletre legyen szükség. Ugyanakkor a feldolgozás és a sémába történő beszúrás költséges és időigényes feladat, melyet úgy kell időzíteni, hogy képes legyen az üzenetek feldolgozására és betöltésére úgy, hogy az újonnan érkező, feldolgozásra váró üzenetek ne torlódjanak fel.

A fentieket figyelembe véve a gyűjtő és feldolgozó funkciókat ellátó modulok a gyorsabb rendszerbe töltés érdekében közvetlenül az adatbázist fogják elérni az alkalmazásszerver megkerülésével.

5.1.4. Környezet kiválasztása

A gyűjtési alrendszer tervezésekor kiemelt szempont a különböző levelezési kiszolgálókhoz történő beépülés, ezért platform-függetlenség figyelembevételével a Java környezetre esett a választás.

A kiszolgáló és kliens oldali logikák megvalósításánál kimondottan a nagyvállalati és kiszolgáló rendszerek megvalósítására alkalmas környezetet kerestem, melyek üzemeltetése és karbantartása egyszerű, ezért e modulok megvalósítását *Microsoft Windows* platformra és *.NET keretrendszerre* tervezem.

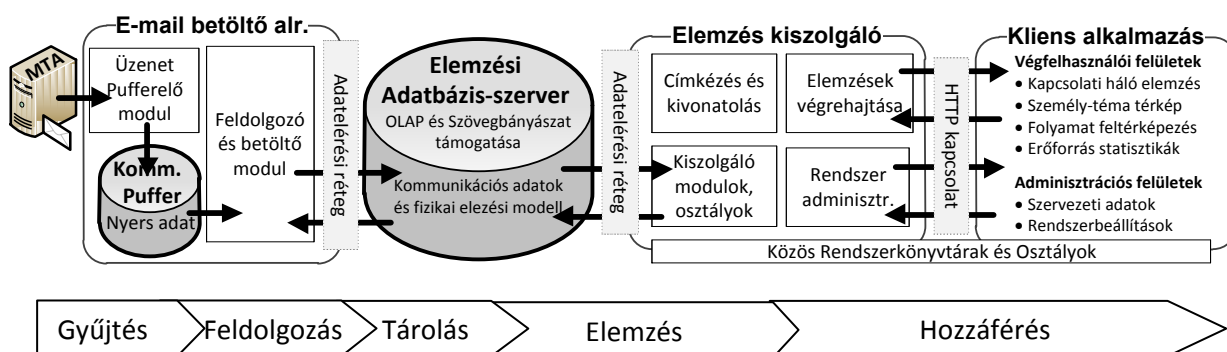
5.1.5. Adatbázisszerver kiválasztása

Vállalati kommunikáció során naponta folyamatosan akár több ezer üzenet is áthaladhat, melyek átalakítása, feldolgozása, kivonatolása, tárolása, indexelése és elemzése hatalmas erőforrásokat emészt fel. Az adatbázisszerver megválasztásakor kiemelt szempont, hogy olyan *DBMS* szolgálja ki az elemzőrendszert, amely képes nagy mennyiségű adat hatékony kezelésére. Az általános adatbázis funkciók mellett támogatnia kell a többdimenziós elemzéseket, és a kivonatolási funkciók megvalósításához rendelkeznie kell szövegbányászati megoldással is. Döntésem ezért az *Oracle Database 11g* megoldásra esett, mely *OLAP* és *Text* opciók telepítésével megbirkózik a fenti feladatokkal.

5.2. Alrendszerek és modulok meghatározása

A modulok meghatározásakor első sorban a hasonló rendeltetésű funkcionális elemek és a tervezői döntéseknél meghatározott szempontokat vettem figyelembe.

A modulokat az architektúrális döntések miatt 4 alrendszerbe szerveztem, melyeket a 9. ábra szemléltet. Az ábra alján a rendszerrel párhuzamosan ábrázoltam az öt fő funkcionális területet is.



9. ábra – Kommunikációt elemző rendszer architektúrája és funkcionális felosztása

Az e-mail betöltő alrendszer biztosítja az e-mailek gyűjtését, feldolgozását és beszúrását. Az elemzési adatbázis gondoskodik az elemzési adatstruktúra és az ahhoz kapcsolódó adatok tárolásáról. Az

elemzési kiszolgáló látja el az üzleti logika szerepét, mely a kliensektől érkező elemzési és adminisztrációs kéréseket szolgálja ki. A kliens alkalmazás feladata a végfelhasználók számára egyszerű és kényelmes felület nyújtása az elemzési és adminisztrációs feladatok ellátásához, valamint az elemzési eredmények ábrázolásához.

Az egyes modulok és osztályok részletes definícióját a dolgozat terjedelmét figyelembe véve nem ismertetem, azonban az elemzés szempontjából fontosabb alkotók osztálydiagrammját az A függelékben részletezem.

5.2.1. E-mail betöltő alrendszer

Ez az alrendszer tekinthető kommunikációt elemző üzleti intelligencia rendszerünk *ETL* eszközének.

E-mail Pufferelő modul

Az *e-mail pufferelő modul* szerepe az üzenetek gyűjtése a továbbításában résztvevő szerverektől és betöltése egy puffer táblába.

Az általunk felépített környezetben a levelezőszerverek továbbítási láncába belépve az e-maileket egy mappába gyűjtik. Az *e-mail pufferelő modul* ütemezett módon futtatva a beállított mappából az üzeneteket egyenként kiolvastva elkészíti azok *hash* kulcsát és betölti a *puffer adatbázis táblájába* feldolgozásra váró státuszjelzéssel. Amennyiben egy levél már szerepel a *puffer táblában*, úgy az adatbázis elutasítja a beszúrást.

A modul a paramétereket parancssorból vagy konfigurációs fájlból is átveheti, ahol megadható, hogy az üzenetek beolvasása mely mappából történjen, pufferelés után törölődjenek-e az üzenetek és természetesen az adatbázisok hozzáférése is itt állítható be. Alapértelmezésként a pufferbe töltött üzeneteket a modul automatikusan törli a mappából.

A megvalósítandó feladatok összefoglalva:

- Paraméterek beolvasása parancssorból vagy konfigurációs fájlból
- Beszédes mód a futás részletes nyomon követéséhez és naplók generálásához
- Üzeneteket tartalmazó mappa e-mailjeinek beolvasása és tartalmukból *hash* képzés
- E-mailek *hash* kulcsokkal történő betöltése a *puffer táblába*
- Beállítástól függően, a pufferelt üzenetek törlése a forrásmappából

Kommunikációs Puffer

Szerepe a feldolgozásra váró levelek tárolása és itt történik a levelek többszörös feldolgozás elleni védelme, mivel a pufferelendő levelek beolvasása során a teljes üzenet tartamából egy *hash* képződik, amelynek fenntartott mezőt elsődleges, egyedi kulcsaként használunk (*BUFFER_ID*). Amennyiben egy üzenet már bent van a *puffer táblában*, vagyis a *hash* értéke már szerepel valamelyik *BUFFER_ID*

mezőben, úgy az adatbázis az egyediség sérülése miatt jelezni fog és megakadályozza az ismételt beszúrást.

4. táblázat – E-mail betöltő alrendszer által használt puffer tábla

CAD_RAWDATA		Puffer tábla
BUFFER_ID	VARCHAR2 (400 byte)	Az üzenet HASH kulcsát tároló szöveges mező egyediség megkötéssel
DATA	CLOB	A teljes, feldolgozásra váró nyers e-mail
STATUS	NUMBER (1, 0)	Ez a mező jelzi, hogy egy e-mail feldolgozásra vár-e
CATEGORY	NUMBER (1, 0)	A kommunikációs csatorna azonosítóját tároló mező
COMM_ID	VARCHAR2 (400 byte)	A feldolgozó modul által visszaírt e-mail <i>MessageID</i> – ellenőrzési célokból

E-mail feldolgozó és betöltő modul

Az *e-mail feldolgozó és betöltő modul* lekérdezi és feldolgozza a puffer tábla még feldolgozatlan nyers kommunikációs üzeneteit, ezek alapján felépíti a kommunikációs adatstruktúrát, majd beszúrja az elemzési adatbázisba.

A modul a paramétereket parancssorból vagy konfigurációs fájlból is átveheti, melyeken keresztül az adatbázisok hozzáférése állítható be.

A megvalósítandó feladatok összefoglalva:

- Paraméterek beolvasása parancssorból vagy konfigurációs fájlból
- Beszédes mód a futás részletes nyomon követéséhez és naplók generálásához
- Feldolgozatlan e-mailek lekérdezése *a puffer táblából*
- E-mailek feldolgozása
 - E-mail *MessageId* fejléc mezőjének kiolvasása
 - Összes címzett és kézbesítési mód kiolvasása
 - Címek alapján résztvevők azonosítása és ismeretlen résztvevők regisztrálása
 - Tartalmi részek (tárgy, szöveg, csatolmány és beágyazott elemek) kibontása
 - E-mail számbeli előzményének és hivatkozási módjának kiolvasása
 - Számbeli előzmény időpontjának és az aktuális e-mail időpontjának egyeztetése (kauzalitás megőrzése)
- Kommunikációs adatstruktúra felépítése és beszúrása az elemzési adatbázisba

5.2.2. Közös Rendszerkönyvtárak és osztályok

Az architektúrális felépítés miatt megosztott elemzési és hozzáférési funkciókat megvalósító könyvtárak, melyeket a kliens és szerver közösen használ, mind a hálózati, mind pedig a belső műveletek során.

Magába foglalja az alábbi osztályok és komponensek egységes kezelését:

- Résztvevőt megvalósító osztályok, kapcsolódó enumerációk, résztvevők gyorsítótárázási megoldása és kapcsolatleíró osztályok
- Kommunikációs modellt megvalósító osztályok és kapcsolódó enumerációk
- Elemzésekhez használt egységes lekérdezést megvalósító osztályok
- Elemzések eredményeit prezentáló osztályok
- Listák - például témák, csatornák, kézbesítési módok – univerzális tárolását, gyorsítótárázását megvalósító osztályok és eljárások
- Kliens és szerver közötti eseményvezérelt üzenetkezelés osztályai

5.2.3. Elemzés kiszolgáló

Az elemzés kiszolgáló *Web Service* interfészen keresztül fogadja és szolgálja ki a kliensektől érkező elemzési és adminisztrációs kéréseket. Feladata az elemzések futtatása, vállalati adattörzs kezelése, valamint a téma és kivonatolási szabályok karbantartása az adatbázis szövegbányászati képességeire támaszkodva.

Az alrendszer az adatbázisszerverhez egy saját adatelérési rétegen keresztül csatlakozik, melyek a közös rendszerkönyvtárak moduljait használva az alábbi szolgáltatásokat építi.

Elemzési szolgáltatások – Elemzés végrehajtó modul

- Kapcsolati háló elemzés végrehajtása
- Téma-résztvevő elemzés végrehajtása
- Folyamat feltérképezés végrehajtása

Adminisztratív szolgáltatások – Adminisztrációs modul

Résztvevő osztály fölé emelt funkciók

- Résztvevő példány létrehozása, módosítása, betöltése
- Résztvevő példányok listájának lekérdezése, szűrőfeltételekkel
- Tagsági adatok kezelése - adott résztvevő csoportjainak és csoport esetében tagjainak lekérdezése, mentése
- Résztvevők elérhetőségének lekérdezése, módosítása
- Ismeretlennek jelölt résztvevő cseréje ismert résztvevőre, elérhetőség regisztrálásával

Téma és kivonatolási szabályok fölé emelt szolgáltatások

- Témák listájának lekérdezése
- Kivonatolási szabály létrehozása, módosítása, lekérdezése, futtatása
- Kivonatolási szabályok listájának lekérdezése

Egyéb feladatok és szolgáltatások

- *Web Service* interfész megvalósítása a kliensekkel történő kommunikációhoz
- Szerver üzeneteinek továbbítása a kliensek felé
- Kapcsolati adatok továbbítása a kliens felé
- Sessionkezelés kliensenként a dedikált adatkapcsolat biztosításához
- Adatelérési réteg megvalósítása
- Adatelérési réteg és egyéb osztályok közötti konvertálást segítő eljárások

5.2.4. Kliens alkalmazás

A végfelhasználók elemző és adminisztrációs felületeit megvalósító kliens-oldali alkalmazás, melyen keresztül az elemzési kiszolgáló által nyújtott elemzési és adminisztrációs szolgáltatások elérhetővé válnak.

Elemzési felületeivel a felhasználók egyszerűen állíthatják össze és futtathatják lekérdezéseiket, elemzéseiket, valamint a kapott eredmények a korábban ismertetett ábrázolási koncepciók alapján ábrázolja. Az ábrázolhatók transzformálhatók személyek vagy csoportok szerint is, valamint az élek irányítottága is ki- és bekapcsolható az elemző szándékától függően.

Az adminisztrációs felületek a rendszer karbantartói számára biztosítják a vállalati résztvevők, tagsági adatok, elérhetőségek, és kivonatolási szabályok kezelését.

Ezáltal a kliensben megvalósítandó funkciók és feladatok

Elemzési szolgáltatások

- Elemzések összeállítására alkalmas egyszerű felhasználói felület megvalósítása
 - Résztvevők egyszerű keresése, listázása, csoportosítása és kijelölése szűrésekhez
 - Témák szerinti szűrés
 - Szűrés időintervallum alapján
 - Szűrés a kommunikáció gyakorisága alapján
- Elemzési kérések küldése a kiszolgáló megfelelő távoli eljárásához és kiszolgáló válaszának fogadása *Web Service* –en keresztül
- Korábban már lekérdezett adatok helyi gyorsítótárazása
- Kapcsolati háló ábrázolása
 - Kapcsolatok többféle nézetben történő megjelenítése: személy-személy, csoport-személy, csoport-csoport
 - Kapcsolati élek irányított és irányítatlan megjelenítése
- Téma-résztvevő térképek ábrázolása
 - Kapcsolatok többféle nézetben történő megjelenítése: téma-személy, téma-csoport
 - Kapcsolati élek irányított és irányítatlan megjelenítése
- Folyamattérkép ábrázolása

Adminisztrációs szolgáltatások

- Kivonatolási szabályok lekérdezése és listázása
- Kivonatolási szabály megjelenítésére, szerkesztésére, mentésére és futtatására alkalmas felület megvalósítása
- Részrtvevők lekérdezése, listázása
- Részrtvevő megjelenítésére, szerkesztésére, mentésére alkalmas felület
 - Személy és csoport adatok szerkesztése
 - Részrtvevőhöz tartozó elérhetőségek felvétele, szerkesztése, törlése
 - Részrtvevő csoporttagsági adatainak felvétele, szerkesztése, törlése
 - Ismeretlen résztvevők azonosítását szolgáló résztvevő cserélő eszköz

5.2.5. Elemzési adatbázisszerver

A központi adatbázisszerver biztosítja a kommunikációs modell fizikai tárolását és elemzését. A gyors és hatékony elemzések érdekében olyan adatbázismotort alkalmazunk, amely *OLAP* és *szövegbányászati* megoldásokat is alkalmaz. Az elemzéshez azonban az adatbázisszerver mellett szükség van a kommunikáció szempontjából jól használható logikai adatmodellre és adatbázis sémára is, melyek tervezését a következő részben ismertetem.

5.3. Adatbázis tervezése

5.3.1. Logikai adatmodell

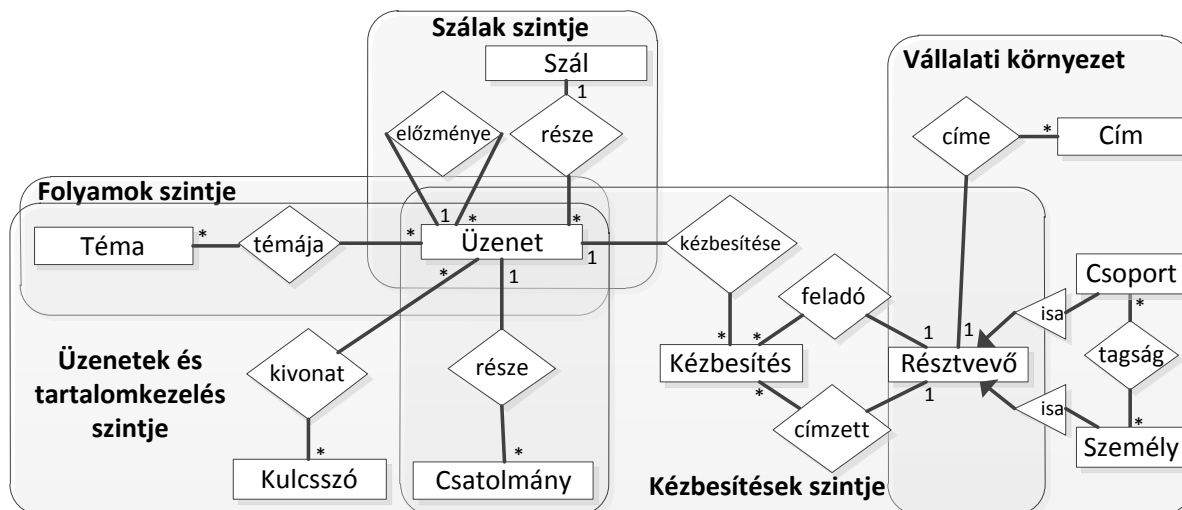
A logikai modell tervezése elsődleges célja az elemzési fejezetben ismertetett fogalmi modell lefordítása egyedek, attribútumok és kapcsolataik szintjére, melyek alapján az adatbázis séma könnyen felépíthető. Hagyományos tranzakciós adatbázisok tervezése esetén az elemzési fázisból a tervezési fázisba kerülő modell kulcskövetelménye, hogy az redundancia-mentes, azaz normalizált legyen. A normalizálás segít olyan adatbázisokat létrehozni, melyek nem tartalmazznak szükségtelenül redundáns adatokat, így a beszúrások, módosítások és törlések által nem jöhetnek létre olyan következtetlenségeket, melyek tárolt adataink későbbi használhatóságát rontanák az adatmanipuláció, a teljesítmény és a helyesség tekintetében.

Ugyanakkor meg kell jegyeznünk, hogy adatbázisunkat jelen esetben elemzési és lekérdezési célokra fogjuk használni, amely sok tábla, kapcsolat, rendezés és csoportosítás esetében rengeteg processzoridőt emésztene fel a normalizált forma miatt. Éppen ezért meg kell találni az ésszerű megoldást a normalizált forma és a leggyakoribb lekérdezések kapcsolási költségeinek optimalizálására.

Az adatbázis séma tervezéséhez ezért azt a módszer alkalmazom, hogy a logikai modellt redundancia-mentesítem, majd a leggyakoribb lekérdezési szempontokat figyelembe véve bizonyos részeit

denormalizálom a táblák kapcsolási költségeinek minimalizálásához, vagyis a teljesítmény növelésének érdekében.

A fentiek értelmében az előző fejezetben ismertetett fogalmi modell elemeiből és jellemzőiből a 10. ábrán látható módon elkészítettem a modell *Entitás-Relációs Diagrammját (ERD)*. A séma jobb átláthatósága érdekében a kulcsok és attribútumok listáját az ábrán nem jelöltem, ám az *A függelék A2* pontjában ezeket is ismertetem. A logikai adatmodell és a fogalmi modell megfeleltetésének szemléltetésére az entitások és kapcsolatok körül téglalapokkal jelöltem a megvalósított szintet.



10. ábra –Logikai adatmodell Entitás – Relációs Diagramja

5.3.2. Fizikai adatmodell

A lekérdezéseink teljesítményének javítása érdekében szükséges a logikai modellünk átalakítása és kapcsolatainak egyszerűsítése. Az üzleti intelligencia rendszerek adattárház megoldásainál leggyakrabban alkalmazott adatbázis séma a csillag-séma, mely az elemzések szempontjából, egy központi ténytáblából és a körülötte elhelyezkedő csoportosítási, szűrési és egyéb kifejtési szempontokat tartalmazó dimenziótáblákból áll.

A ténytábla sorai adatrekordokat tartalmaznak, melyek a dimenziótáblákhoz egy-egy idegen kulccsal kapcsolódnak, így ezeken keresztül elérhetővé válnak a dimenzió attribútumai. A ténytábla soraiban található adatokat tehát úgy értelmezhetők, hogy megkeressük a dimenziótáblákban a kulcsoknak megfelelő rekordokat.

E megoldás előnye, hogy egy ilyen adatstruktúrában a többszörös áttétes táblakapcsolások számának csökkenésével vagy kizárásával a lekérdezések készítése és futtatása is egyszerűbbé válik, hiszen egy jól felépített csillag-sémában a ténytáblához kapcsolódó dimenziótáblákat pár egyszeres táblakapcsolással közvetlenül elérhetjük.

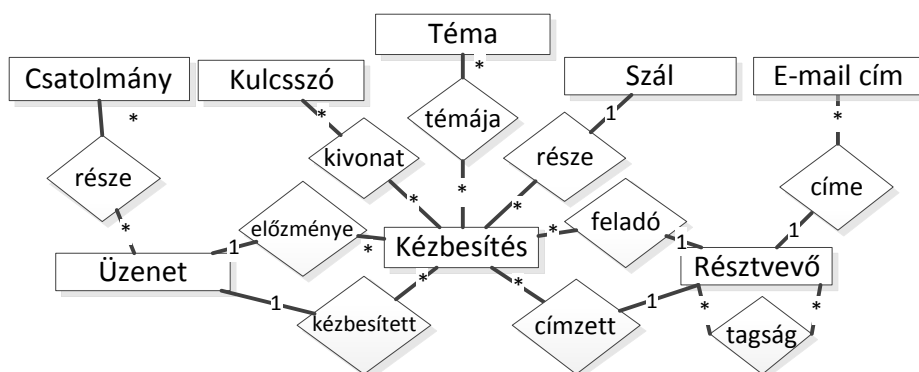
A résztvevők főtípus egyede, a személy és csoport ISA kapcsolata miatt összetett helyzetet eredményez, ami a lekérdezésünk teljesítményét ugyancsak rontja. Az ilyen típusú kapcsolat kiküszöböléséhez a személy és csoport egyedet megszüntettem, ám attribútumaikat összevonva felvettem a résztvevő egyedéhez és bevezettem egy új attribútumot az altípus jelölésére, mellyel megkülönböztethető, hogy személyről vagy csoportról van-e szó.

Az elemzés-gyorsulásának azonban ára van, mivel a kapcsolatok átszervezésével bizonyos attribútumok redundáns tárolása szükséges. Ez ugyan megnöveli az adatmanipulálás költségeit, de elemzőrendszerünknel ez nem lényeges szempont, hiszen a kommunikációs adathalmazon menet közben csak bővítést végzünk.

5.3.3. Csillag-séma kialakítása

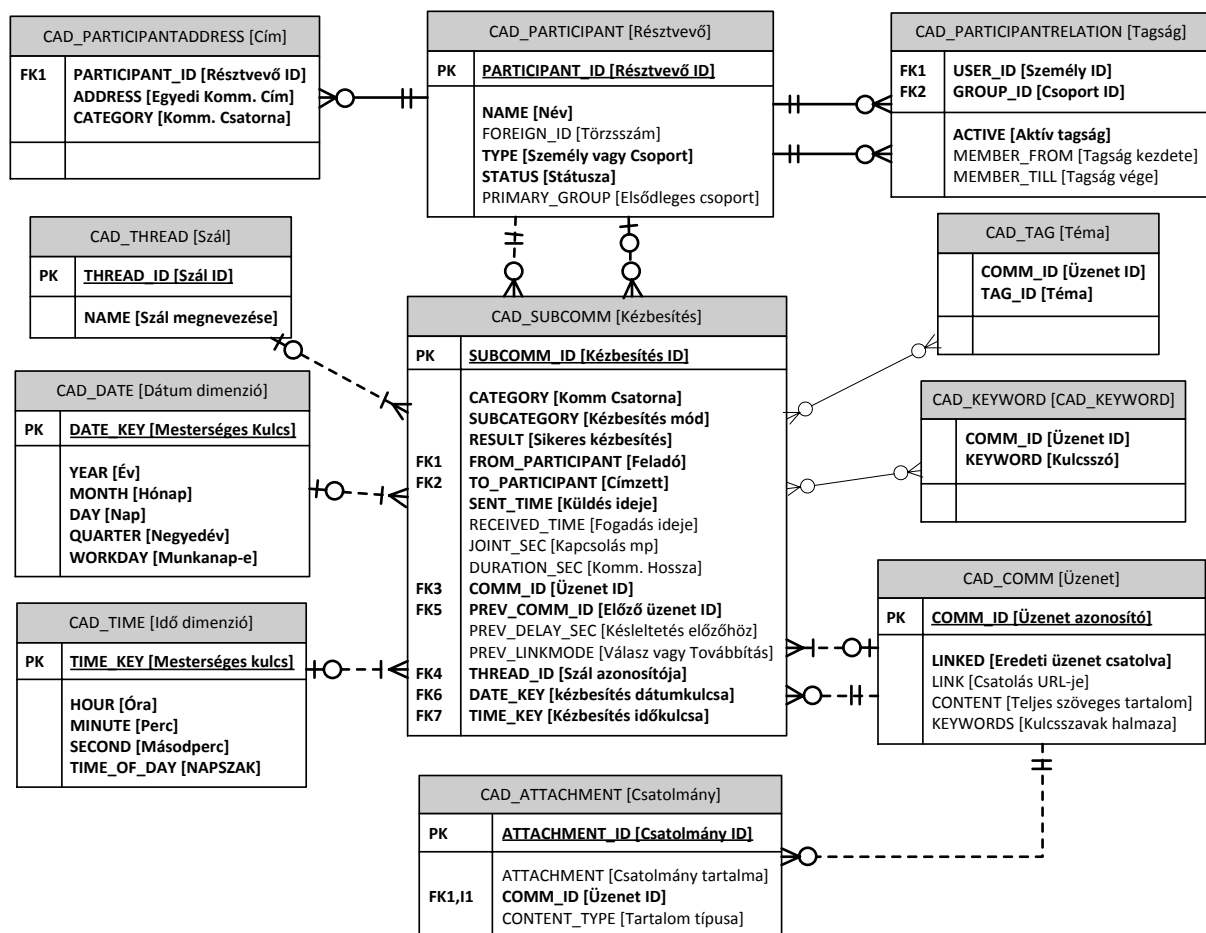
Ténytábla

Az elemzések szempontjából leglényegesebb egyedek a *résztvevők*, az érintett *témák*, *kulcsszavak*, a *szálak* és természetesen maguk a *kézbbsítések*, mivel az utóbbi határozza meg, hogy ki, kivel, mikor és miről kommunikált. Ezek alapján az elemzőrendszer csillag-sémáját a 11. ábrán látható módon a *kézbbsítések* ténytáblája köré építettem, melynek dimenziói a kommunikáció *résztvevői* - a feladó és címzett szerepében -, annak *témái*, *kulcsszavai* és *szála* lett.



11. ábra – Tervezett Csillag-séma Entitás- Reláció Diagramja

A 12. ábrán látható, hogy a kézbbsítések ténytáblája a korábbi kapcsolatok felépítéséhez szükséges idegen kulcsok, valamint még két új idő és dátum dimenzió idegen kulcsának bevétele miatt ugyan kiszélesedett, de az így kialakított séma jóval egyszerűbbé vált az eredeti logikai modellnél, így a lekérdezések is egyszerűbben végrehajthatóvá váltak.



12. ábra – Az elemző adatbázis csillag-sémája

A dátum és idő dimenziók bevezetésével az elemzőrendszer képes lesz olyan elemzések végrehajtására és összehasonlítására, melyek napszakokra, munkanapokra, vagy akár negyedéves adatokra vonatkoznak. Ugyanakkor a dátumfüggvények használatának mellőzése miatt a kézbítások ténye is gyorsabban lekérdezhetővé vált az egyes attribútumok szerint.

Kulcsok

A kézbítés, üzenet, csatolmány és résztvevő táblák esetében az elsődleges kulcsokat (*primary key - PK*) mesterségesen *GUID*¹⁸-ok generálásával mesterségesen osztottam ki, míg a dátum dimenzió elsődleges kulcsának az *ÉÉÉÉHHNN* dátumformátumú, az idő dimenzió pedig az *ÓÓPPMM* időformátumú értékeket választottam számként ábrázolva, egyediséget biztosító megszorítással. Ennek köszönhetően a dátumintervallumokra történő szűrés vissza lett vezetve az egész számok közötti relációkra.

¹⁸ *GUID - Globally Unique Identifier* – globális egyedi azonosító. Olyan álvéletlen azonosító, melyben az egyedi kulcsok száma annyira nagy (2^{128}) hogy két azonos szám generálásának a valószínűsége gyakorlatilag nulla.

Idegen kulcsok (*foreign key* - *FK*) akkor jelennek meg, ha egy kapcsolatban a tábla a gyermek szerepét tölti be, ami jól megfigyelhető sémánkban a tényátlák és a csoport-tagság, illetve résztvevő-cím kapcsolatok definiálásakor.

Indexek

Az indexek felvételekor azt az elvet követtem, hogy azokra az attribútumokra kell indexet építeni, melyekre a lekérdezések a leggyakrabban irányulnak, ezért nyilvánvalóan a kulcsok mellett, a *téma*, *kulcsszó*, *üzenet azonosító*, valamint a *dátum* és *idő* dimenziók attribútumaira is definiáltam indexeket.

5.4. Külső komponensek

Az elemzőrendszer tervezésekor a megvalósítandó feladatok ismeretében megvizsgáltam, hogy vannak-e olyan kész megoldások, melyeket lehet és érdemes is felhasználni a megvalósítás során.

5.4.1. Sun JavaMail™ API

A *Sun Microsystems* által fejlesztett *JavaMail™ API* [19] egy keretrendszert biztosít az *IETF* által szabványosított e-mail üzenetek létrehozásához, levelező kiszolgálókról történő lekérdezéséhez, tárolásához és feldolgozásához. A nyers üzenetek feldolgozását megkönnyítő komponens képes olyan *Java* objektumokat példányosítani, melyek segítségével az e-mailek fejléc mezőit és tartalomkonténerait kényelmesen és gyorsan ki lehet nyerni.

5.4.2. Microsoft GLEE – MS AGL

Az elemzési eredmények ábrázolási lehetőségeinek megvalósításához egy gráf alapú interaktív prezentációs réteget, a *Microsoft Research* által fejlesztett *Microsoft Automatic Graph Layout (MSAGL)* .NET referenciacsomag, egy korábbi verzióját a *Microsoft Graph Layout Execution Engine (GLEE)*-t választottam, amely kereskedelmi és üzleti felhasználást kivéve szabadon használható. A komponens segítségével a fejlesztők komplett eszközugyteményt kapnak gráfok vektoros megjelenítésére és interaktív eseménykezelésre.

5.4.3. Oracle Data Provider

A .NET keretrendszerben fejlesztett komponensek Oracle adatbázissal történő együttműködéséhez az *Oracle Data Provider .NET* csomagját választottam. A komponensgyűjtemény más driverektől eltérően natív módon támogatja az *Oracle* specifikus adattípusokat is [20].

A csomag tartalmaz serveroldali menedzsment eszközökkel együttműködő eszközöket, melyek segítséget nyújtanak az adatbázisok tervezéséhez, adminisztrációjához, tárolt-eljárások hibajavításához, ugyanakkor lehetőséget biztosít akár a futás időben generált lekérdezések finomhangolásához is *Microsoft Visual Studio 2005* és *2008* fejlesztői környezetekben.

6. Rendszer fejlesztése

6.1. Környezet ismertetés

Ebben a szakaszban ismertetem a tervezésnél kiválasztott környezetekhez és külső komponensek fejlesztéshez és teszteléshez használt pontos verziókat, fejlesztőeszközöket és számítógép konfigurációt.

A *Java* komponensek fejlesztéshez a *NetBeans IDE 6.8*, míg a *.NET keretrendszer* alatt futó komponensek implementálásához a *C#* nyelvet és a *Microsoft Visual Studio 2008* integrált fejlesztői környezetet használtam. A fejlesztési és tesztelési hardverkörnyezetként saját notebookomat használtam, így a teljes fejlesztési konfigurációt az 5. táblázat tartalmazza.

5. táblázat – Fejlesztéshez és teszteléshez használt hardver- és szoftverkonfiguráció

Számítógép	HP Compaq nw8240
Processzor	Intel Pentium M, 2,13 GHz (32bit)
RAM	2048 MB
Operációs Rendszer	Windows 7 Enterprise 32-bit (6.1 build 7600)
Java környezet	Java 6 Update 20
.NET környezet	Microsoft .NET Framework 3.5.7283
Adatbázisszerver	Oracle Database 11g - 11.1.0.7.0
Egyéb komponensek	Oracle Developer Tools for Visual Studio 11.1.0.0720 Oracle JDBC Driver 11.1.0.7.0 Sun JavaMail API 1.4 Microsoft Research GLEE 1.0

6.2. Implementáció

6.2.1. E-mail betöltő alrendszer

E-mail Pufferelő modul – *caEmailBufferer* projekt

Ez az egyszerű *Java SE 6* konzolalkalmazás valósítja meg az e-mailek pufferelését végző modult, amely a szerveren lévő leveleket a fájlrendszerből kiolvastva betölti a puffer táblába *Oracle Database 11g JDBC Driver* segítségével.

E-mail feldolgozó és betöltő modul – *caEmailBufferParser* projekt

Ez a *Java SE 6* konzolalkalmazás látja el az *caEmailBufferer* által a puffer táblába betöltött, még feldolgozatlan státusszal rendelkező e-mailek feldolgozását és tárolását az elemző adatbázisba.

A puffer táblában az üzenetek nyers *Internet Message* formátumban vannak rekordonként tárolva. A feldolgozásra váró e-maileket lekérdezés után egy saját *SimpleMail* osztállyal reprezentálom, ami a *Sun JavaMail API MimeMessage* osztályát példányosítva nyeri ki az elemzési modellhez szükséges

adatokat az általam fejlesztett eljárásokkal. A felépített elemzési adatstruktúra elemzési adatbázisba történő beszurását a *SimpleMail* osztályból származó *SimpleMailDatabaseObject* végzi. Sikeres beszurást követően a puffer táblában az aktuális rekord státusza feldolgozott állapotba kerül.

A megvalósított modul osztálydiagramját és további működését az *A függelék A3.* pontja tartalmazza.

6.2.2. Közös rendszerkönyvtárak és osztályok – caCoreLibrary

A megvalósított rendszerkönyvtár a kliens és szerver által használt osztályokat és eljárásokat tartalmazza egy közös *DLL*-ben. A logikai modell által definiált egyedek és a rendszer működése szempontjából lényeges osztályok és eljárások funkciójuk alapján modulokba vannak szervezve, melyeket a 6. táblázat, míg az elemzés szempontjából fontosabb osztályokat a 7. táblázat részletezi.

6. táblázat – Közös rendszerkönyvtár moduljai és szerepük

Rendszerkönyvtár	Szerepe
caAnalysisResults	Elemzések eredményeit leíró osztályok gyűjteménye
caComm	Kommunikációs modell és az azzal kapcsolatban álló enumerációk, osztályok és eljárások gyűjteménye
caParticipant	Résztvevők és az ahhoz kapcsolódó osztályok és statikus eljárások gyűjteménye
caRelation	Személyek és csoportok közötti kapcsolatokat prezentáló osztályok
caConvert	Saját osztályok és tulajdonságaik statikus konvertáló eljárásai
caGlobalList	A kliens oldali listák - például témák, csatornák, kézbesítési módok – univerzális tárolását, gyorsítótárazását és grafikus prezentációját ellátó osztályok és eljárások
caMessageService	Kliens és szerver közötti üzenetkezelést biztosító eseménykezelést támogató, osztályok és statikus eljárások gyűjteménye
caSQL	Összeállított hierarchikus feltételek SQL lekérdezésekké alakító osztályok gyűjteménye
caSystem	Egyéb rendszerosztályok és enumerációk könyvtára

Az implementálás során a kollekciók és halmazok hatékonyabb kezelése céljából a *.NET 3.5* keretrendszer beépített generikus lista osztályát használtam.

7. táblázat – Közös rendszerkönyvtár fontosabb osztályai és szerepük

Osztály vagy Enumeráció	Szerepe
caCommCategory	Kommunikáció csatornák listája
caCommSubcategory	Kommunikációs csatornák kézbesítési módjait jelölő lista
caParticipantType	Résztvevők típusát jelölő lista (személy vagy csoport)
caParticipant	Résztvevő egyedet leíró osztály
caParticipantObject	Résztvevő egyed kiterjesztett osztálya, gyorsítótárazáshoz és megjelenítéshez
caParticipantAddress	Résztvevő címét reprezentáló osztály
caSubCommItem	Kézbesítés egyedet leíró osztály
caSubCommItemObject	Kézbesítés egyed kiterjesztett osztálya, gyorsítótárazáshoz és megjelenítéshez
caTaggingRule	Kivonatolási szabályt megvalósító osztály
caMessage	A rendszer által generált üzenet osztály eseményvezérelt üzenetkezeléshez
caGlobalList	A kliens oldali listák - például témák, csatornák, kézbesítési módok – univerzális tárolását, gyorsítótárazását és megjelenítését ellátó osztály

6.2.3. Elemzés kiszolgáló - caServiceHost

Az elemzés kiszolgáló rendszerét egy önmagát hostoló .NET Windows Communication Foundation Service-ként implementáltam, mely elindítása után a gép 8511-es portján szolgálja ki a HTTP –n érkező Web Service hívásokat a caServiceLibrary szolgáltatásaira építve.

caServiceLibrary WCF Service projekt

A szerver szolgáltatásait a WCF Service Library – Service nevű osztály végzi az IService interfész implementálásával, melynek adminisztrációs eljárásait a 8. táblázat, elemzési eljárásait pedig a 9. táblázat ismerteti. A kliensek által kezdeményezett hívások szerver oldalon minden esetben új adatelérési osztályt példányosítanak, így az egyes elemzési és adminisztrációs műveletek egymástól elkülönítetten hajthatók végre.

Adminisztrációs modul

Az adminisztrációs modul eljárásai a rendszerkönyvtárak alapobjektumai szintjén nyújtanak összetett személy, csoport és cím-kezelési, valamint a caTaggingRule osztályra épülő kivonatolási szolgáltatásokat.

8. táblázat – Kiszolgáló oldali rendszeradminisztrációs eljárások

Szerep	Web Service eljárás	Eredmény osztálya
Új adatkapcsolat nyitása	Connect	caMessage
Szerver üzenetek letöltése	GetMessages	List<caMessage>
Résztvevő példány lekérése	LoadParticipant	caParticipant
Résztvevő példány mentése	SaveParticipant	caParticipant
Résztvevő lista lekérdezése	LoadParticipantList	List<caParticipant>
Résztvevő csoportjainak lekérése	LoadParticipantGroups	List<caParticipant>
Résztvevő csoportjainak mentése	SaveParticipantGroups	List<caParticipant>
Résztvevő tagjainak lekérése	LoadParticipantMembers	List<caParticipant>
Résztvevő tagjainak mentése	SaveParticipantMembers	List<caParticipant>
Résztvevő címeinek lekérése	LoadParticipantAddress	List<caParticipantAddress>
Résztvevő címeinek mentése	SaveParticipantAddress	List<caParticipantAddress>
Résztvevő cseréje másira	ReplaceParticipant	caMessage
Témák listájának lekérdezése	LoadTagList	List<string>
Téma törlése	DeleteTag	caMessage
Kivonatolási szabály lekérése	LoadTaggingRule	caTaggingRule
Kivonatolási szabályok listája	LoadTaggingRuleList	List<caTaggingRule>
Kivonatolási szabály mentése	SaveTaggingRule	caTaggingRule
Kivonatolási szabály futtatása	RunTaggingRule	caMessage

Kivonatolási szabályok– caTaggingRule osztályok

A kiszolgáló logika szövegbányászati feladatait az Oracle Text opció szolgáltatásaira építve terveztem megoldani, melyek támogatására a caTaggingRule osztályokat hoztam létre. Ezek gyakorlatilag olyan objektumok, amelyek előre elkészített SQL utasításokat foglalnak magukba kulcsszavak és témák generálásához.

Elemzés-végrehajtás

Az elemzések a kiszolgáló 9. táblázatban szereplő eljárásain keresztül végezhetők el. Az elemzések mindegyike saját eredményosztállyal rendelkezik. Ezek a kisebb sávszélességű hálózatokon is csak az eredmény szempontjából leglényegesebb információkat tartalmazzák, a többletinformációkat pedig a kliensek gyorsítótára biztosítja. A három elemzési eredményosztály mezőit az *A függelék A5. pontja* tartalmazza.

9. táblázat – Kommunikáció-elemzést megvalósító eljárások és eredménytípusaik

Elemzés	Web Service eljárás	Eredmény osztálya
Kapcsolati háló elemzés	GetParticipantRelationAnalysisResultList	caRelationAnalysisResult
Téma-részrtvevő elemzés	GetTagParticipantAnalysisResultList	caTagParticipantAnalysisResult
Folyamat fekerítés	GetFlowAnalysisResultList	caFlowAnalysisResult

Adatelérési réteg - caDataAccessLayer projekt

Az adatok fizikai kezelését egy köztes adatelérési réteggel oldottam meg, amely a kiszolgáló szolgáltatásszintű eljárásaiból kerülnek meghívásra a *caDatabaseService* osztály példányosításával. Az adatelérési réteg az elemi műveleteket adatbázis-specifikus hívásokká alakítja az *Oracle Data Provider .NET* osztályaira és szolgáltatásait használva.

6.2.4. Kliens alkalmazás – caClient projekt

Az elemzőrendszer vastag-kliens megoldása egy *Windows Forms* projekt, amely saját osztályain kívül a *caCoreLibrary*, *caGraphLibrary* könyvtárakra épül és a szerverrel történő kommunikációhoz a *ServiceClient* osztályt használja.

Az elemzési és adminisztrációs funkciókat külön űrlapok látják el, melyek a főablakban *MDI* gyermekként jelennek meg. A *Forms* névtérben szereplő űrlapok (10. táblázat) tetszőleges példányban futhatnak, melyek mindegyike a főablak kapcsolat példányát másolva kommunikál a szerverrel.

10. táblázat – Kliens funkciókat megvalósító űrlapok és leírásuk

Ellátott funkció	Windows Forms űrlap
Kliens alkalmazás főablaka – MDI szülő	FormMain
Kapcsolati-háló elemzés	FormParticipantRelationAnalysis
Folyam felderítés	FormFlowAnalysis
Téma-részrtvevő elemzés	FormTagRelationAnalysis
Kapcsolatkezelés	FormConnect
Részrtvevő részleteinek űrlapja	FormParticipant
Részrtvevő kezelőfelülete	FormParticipantMgmt
Részrtvevő cserélő eszköz	FormParticipantReplaceTool
Kereső felületekhez Részrtvevő-választó	FormParticipantSelector
Kivonatolási szabály részleteinek űrlapja	FormTaggingRule
Kivonatolási szabályok kezelőfelülete	FormTaggingRuleMgmt

A gyakran használt űrlapfunkciók ellátásához - mint például a résztvevők listázása, kiválasztása, szűrése, valamint az általános keresőfelület kialakítása - saját vezérlőket fejlesztettem, melyek a projekt *Controls* névterében találhatók.

Eredmények grafikus prezentációja – *caGraphLibrary*

Az űrlapok eredményeinek ábrázolását megvalósító és interaktív elemzést lehetővé tevő *Windows Forms* vezérlők projektje a *caGraphLibrary*, melynek osztályai generálják az elemzési fejezetben koncepcionálisan felvázolt ábrázolási módoknak és tervezési feladatoknak megfelelő grafikus kimeneteket. A három elemzési módhoz fejlesztett három *Windows Forms* vezérlő osztályt a 11. táblázat tartalmazza.

11. táblázat – Ábrázolást megvalósító vezérlők osztályai és leírásaik

Ábrázolási mód	Windows Forms vezérlő osztály
Kapcsolati háló	<i>caRelationGraph</i>
Téma-résztvevő térkép	<i>caCompetencyDigaram</i>
Folyamattérkép	<i>caCommFlow</i>

A vezérlők fejlesztésében a *Microsoft GLEE* által biztosított osztályokat és GUI vezérlőket használtam. Az elemzések eredményhalmazát .NET generikus listaként tároltam és a gráfok felépítését, valamint a memóriában lévő halmazok kezelését *LINQ for Objects*¹⁹ megoldással valósítottam meg.

¹⁹ *LINQ* – *Language Integrated Query*, A Microsoft .NET keretrendszer beépített lekérdező nyelve, amely az SQL-hez hasonló szerkezettel képes objektumhalmazokon és adatobjektumokon különböző kiválasztási és adatmanipulálási műveleteket végrehajtani

7. Értékelés

Az elemzési és tervezési fázisban kitűzött célok és feladatok sikeresen megvalósításra kerültek, így elkészült a kommunikációt elemző rendszer négy alrendszere, az e-mail betöltő, az elemzési kiszolgáló, az elemzési adatbázisszerver és a kliens alkalmazás is, a szövegbányászati funkciók kivételével az összes funkciót lefedve.

Az egyes osztályok, modulok és modulcsoportok működését külön-külön és összekapcsolásuk után is teszteltem mesterségesen előkészített teszteseteken, míg a rendszer egészének működését saját levelezésem hibakeresési módban és empirikus módszerekkel is ellenőriztem, ami így a hibák folyamatos javítása után helyesnek bizonyult.

A továbbiakban a feladatkiírásának megfelelően az E-mail betöltő és az Elemzési és Hozzáférési alrendszer funkcióin keresztül értékelem az e-mailek feldolgozását, reláció-elemzését és azok kapcsolati-hálós megjelenítését végző komponenseket.

7.1. E-mail betöltő alrendszer értékelése

Az e-mailek fájl-forrás-ból történő feldolgozását az előző fejezetben ismertetett *caEmailBufferer* és *caEmailBufferParser* Java modulok végzik. A levelezés továbbítási láncából egy mappába gyűjtött e-mailek pufferelesét az előbbi, míg a puffereelt üzenetek tényleges feldolgozását és elemzési-modellnek megfelelő átalakítását és betöltését az utóbbi *JAR* konzol-alkalmazás végzi.

7.1.1. Teljesítmény értékelése

Az alrendszer teljesítménybeli tesztelését különböző méretű és számosságú üzenethalmazokon teszteltem. A mérések a puffereelő és feldolgozó komponensek sorosított futtatásával, beszédes módban szolgáltatott futási idők alapján készültek, melyek eredményeit a 12. táblázat tartalmazza.

A mérési táblázat minden egyes sora egy tesztfutást jelent. Az első sor az üresjáratú tesztfuttatást jelenti, ahol a modulok elindultak, de nem találtak sem betöltendő, sem pedig feldolgozandó üzenetet. Az *A* oszlop tartalmazza az egy futás alatt beolvasott e-mailek számát, a *B* oszlop pedig ezeknek az e-maileknek a teljes méretét KB-ban. *C* oszlop mutatja, hogy az adott e-mail halmazon a puffereelő modul mennyi ideig futott, a *D* pedig ugyanezt a feldolgozó modul futási idejére. Az *E* oszlop tartalmazza a két modul futási idejének összegét. *F* oszlop a relatív futási időt tartalmazza, ami a teljes futási idő a rendszer üresjáratú idejével csökkentett értéket jelenti ($E-2945$ ms), mely a rendszer bemelegedési idejét ezáltal kiveszi a további számításokból. *G* oszlop egy üzenet átlagos méretét adja meg (B/A), míg a *H* oszlop egy üzenet átlag, relatív beszúrási idejét (F/A). A mérés legérdekesebb értéke az *I* oszlop, amely azt mutatja, meg, hogy átlagosan mennyi relatív időt igényel 1KB adat beszúrása a rendszerbe.

12. táblázat – E-mail betöltő alrendszer modulok tesztjének mérési eredményei

A	B	C	D	E	F	G	H	I
E-mail (db)	Teljes méret (KB)	Pufferelő futása (ms)	Feldolgozó futása (ms)	Összes Beszúrási Idő (ms)	Relatív beszúrási idő (ms)	Egy Üzenet átlag mérete (KB)	Egy Üzenet relatív átlag besz. ideje (ms)	1 KB Átlag relatív besz. idő (ms)
-	-	1 550	1 395	2 945	-	-	-	
1	1	1 595	1 643	3 238	293	1,0	293,00	293,00
5	5	1 715	2 283	3 998	1 053	1,0	210,60	210,60
10	10	1 723	2 507	4 230	1 285	1,0	128,50	128,50
25	25	2 828	3 201	6 029	3 084	1,0	123,36	123,36
1	2	1 695	1 895	3 590	645	2,0	645,00	322,50
5	10	1 734	1 979	3 713	768	2,0	153,60	76,80
10	20	1 739	2 910	4 649	1 704	2,0	170,40	85,20
25	50	2 047	2 910	4 957	2 012	2,0	80,48	40,24
1	4	1 598	1 640	3 238	293	4,0	293,00	73,25
5	15	1 643	1 890	3 533	588	3,0	117,60	39,20
10	30	1 719	2 406	4 125	1 180	3,0	118,00	39,33
25	90	3 007	4 047	7 054	4 109	3,6	164,36	45,66
1	5	1 406	1 656	3 062	117	5,0	117,00	23,40
5	36	1 609	1 891	3 500	555	7,2	111,00	15,42
10	70	2 203	2 047	4 250	1 305	7,0	130,50	18,64
25	192	2 922	2 657	5 579	2 634	7,7	105,36	13,72
1	300	1 688	1 781	3 469	524	300,0	524,00	1,75
5	967	3 063	2 531	5 594	2 649	193,4	529,80	2,74
10	2 264	5 656	3 797	9 453	6 508	226,4	650,80	2,87
25	5 076	10 407	6 625	17 032	14 087	203,0	563,48	2,78
1	1 058	2 547	2 343	4 890	1 945	1 058,0	1 945,00	1,84
5	5 214	7 968	5 031	12 999	10 054	1 042,8	2 010,80	1,93
10	11 614	20 203	9 281	29 484	26 539	1 161,4	2 653,90	2,29
20	30 195	45 500	22 313	67 813	64 868	1 509,8	3 243,40	2,15
1	3 635	3 703	4 547	8 250	5 305	3 635,0	5 305,00	1,46
50	35 372	25 484	26 111	51 595	48 650	707,4	973,00	1,38

A fejlesztési hardver és szoftver konfiguráción végzett mérések eredményeinek vizsgálatával elmondható, hogy az üzenetek 1KB-ra eső átlagos beszúrási ideje – a komponensek indulási idejét is figyelembe véve – (I oszlop) annál gyorsabb, minél nagyobb az egyszerre beolvasott üzenethalmaz mérete (A, B oszlop).

Természetesen a futási idők javíthatók, ha a puffereleési és feldolgozási modulok soros feldolgozás helyett párhuzamosan vannak futtatva, mivel így az előző párhuzamos fázisban puffereelt üzeneteket a feldolgozó modul az újabb puffereleési ciklussal párhuzamosan képes feldolgozni és beszúrni az elemzési adatbázisba.

Megállapítható, hogy a rendszer szűk keresztmetszete a modulok indítási és listázási fázisa, ami az üresjáratú tesztnél figyelhető meg, hiszen a moduloknak külön-külön is körülbelül 1,5 másodperc kellett, mire elindultak és lekérdezték a betöltésre vagy feldolgozásra váró e-mailek üres listáját.

7.2. Elemzés értékelése

7.2.1. Elemzési modulok bemutatása

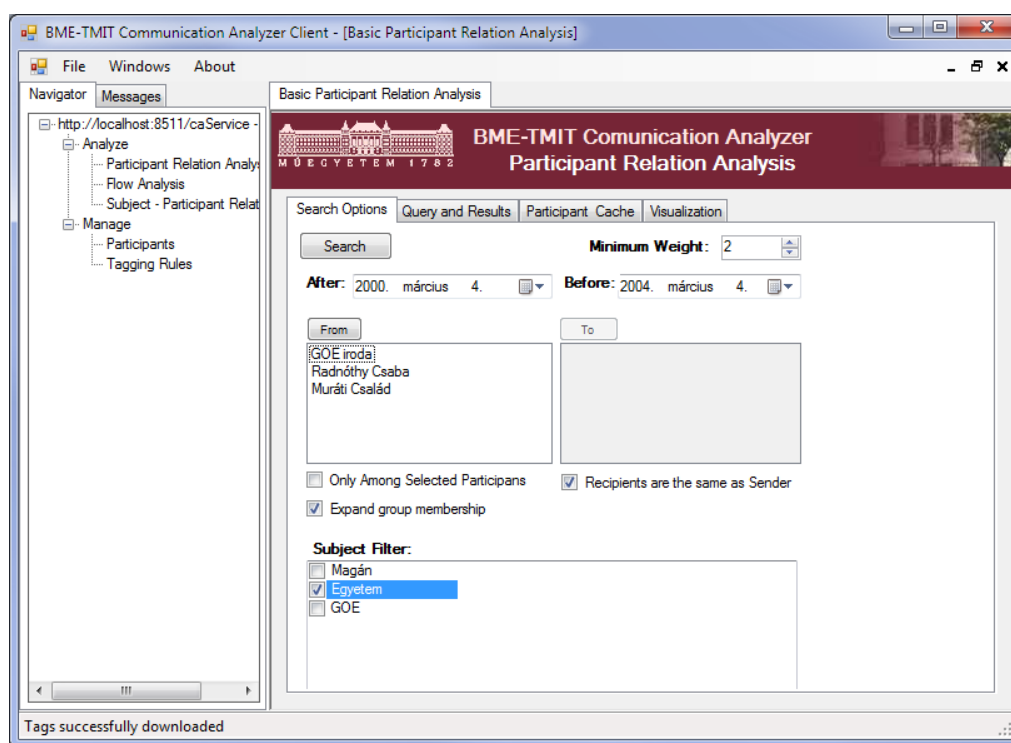
Az megvalósított kliens alkalmazás elemzési és adminisztratív funkcióit a végfelhasználók bejelentkezés után érhetik el a bal oldalt elhelyezkedő fa struktúrájú menüből. Az elemzés menü alatt

érhető el a tervezés során kitűzött három elemzési mód, a kapcsolati háló elemzés, a téma-részrtvevő elemzés és a folyamat felderítés.

Az elemzések a terveknek megfelelően megoszlik az elemzési-kiszolgáló és kliens között. Az elemzések végrehajtását kizárólag a kiszolgáló látja el, a lekérdezések összeállítása, részletekkel történő kiegészítése és ábrázolás pedig a kliens erőforrásait használja. A témák listájának letöltése és gyorsítótárazása a kliens indulásakor történik, ami a továbbiakban így nem terheli a szerver erőforrásait.

Elemzőfelület ismertetése

A terveknek megfelelően a megvalósított kliens az összes elemzéshez egységes szűrési felületet (13. ábra) biztosít, melyen a felhasználók gyorsan és egyszerűen állíthatják be elemzési szempontjaikat. Itt adható meg, hogy az elemzés milyen időintervallumban történjen, milyen témákban keressen, konkrét résztvevők megjelölésével, valamint az is megadható, hogy minimum hány üzenet vegyen figyelembe két résztvevő kapcsolatának feltáráskor.

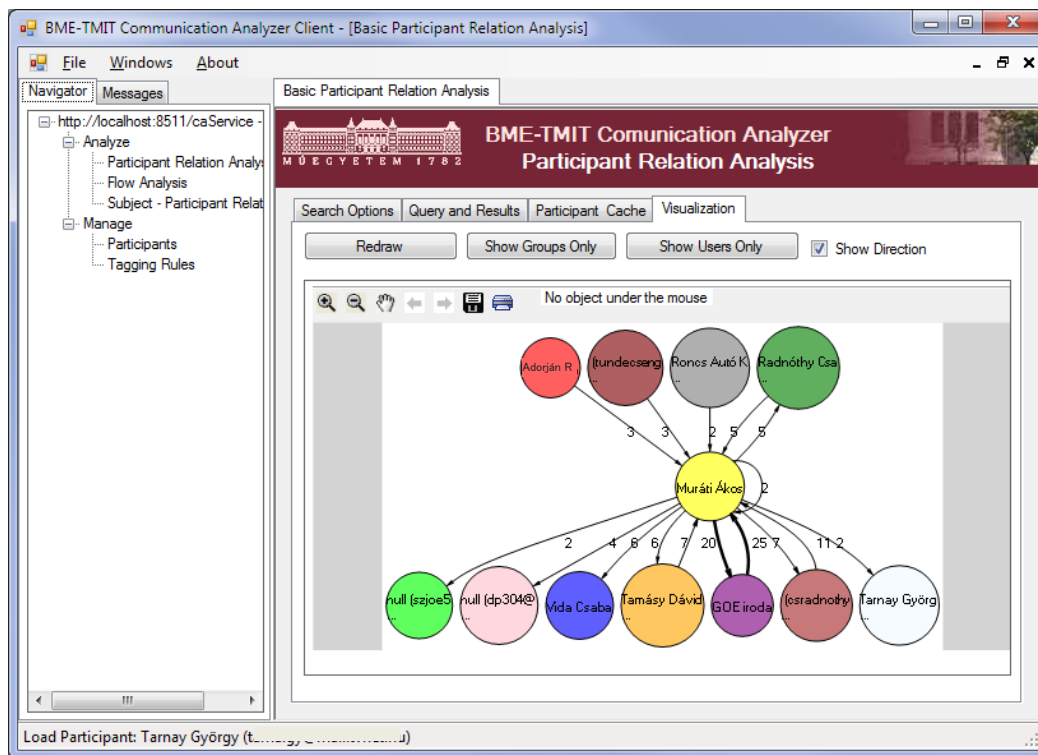


13. ábra – A rendszer általános elemzőfelülete

Kapcsolati háló elemzés bemutatása

A kapcsolati háló elemző modul az általános elemzési szempontok beállítása után összeállítja lekérdezését, majd az elemzési kérést eljuttatja a szervernek. A kiszolgáló a kérésnek külön adatbázis-kapcsolatot dedikálva elvégzi az elemzést, majd a kis sávszélességre kialakított redukált eredményhalmazt visszaküldi a kliensnek, amik ezután az eredmények oldalon jelennek meg.

A vizualizációk oldalán az elemzés eredményét ábrázolja a tervezési céloknál kitűzött módon (14. ábra). Ezen a felületen a felhasználó az ábrázolást tovább finomíthatja az alapján, hogy a szereplőket személyek vagy csoportjaik alapján kívánja megjeleníteni vagy, hogy számít-e a kommunikáció iránya. A kapcsolati élek automatikus vastagításával a rendszer kiemeli a gyakran kommunikáló klikkeket és a végeredmény akár képként is exportálható.

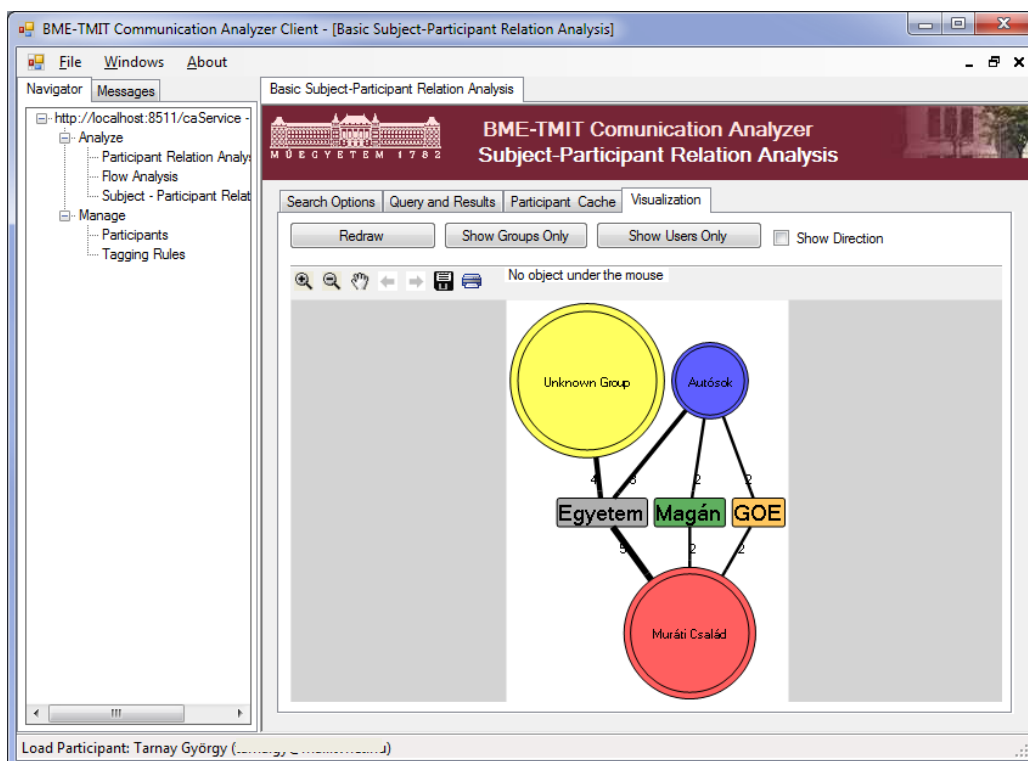


14. ábra – Kapcsolati háló vizualizációs modulja

(A 14. ábra 'Muráti Ákos' résztvevővel kapcsolatban álló személyeket jeleníti, melyen a rendszer feltüntette a kapcsolatok irányát is. A képről leolvasható a vastagított éleket figyelembe véve, hogy az adott elemzésnél a GOE Irodával gyakran kommunikált a szereplő.)

Téma-résztvevő elemzés bemutatása

A téma-résztvevő kapcsolat a résztvevő elemzési modulhoz hasonlóan épül fel. Az elemzési szempontok beállítása után a kliens kérését a szerver felé továbbítja, majd a kapott eredményeket a tervezési szempontoknak megfelelően ábrázolja a vizualizációk oldalán (15. ábra).



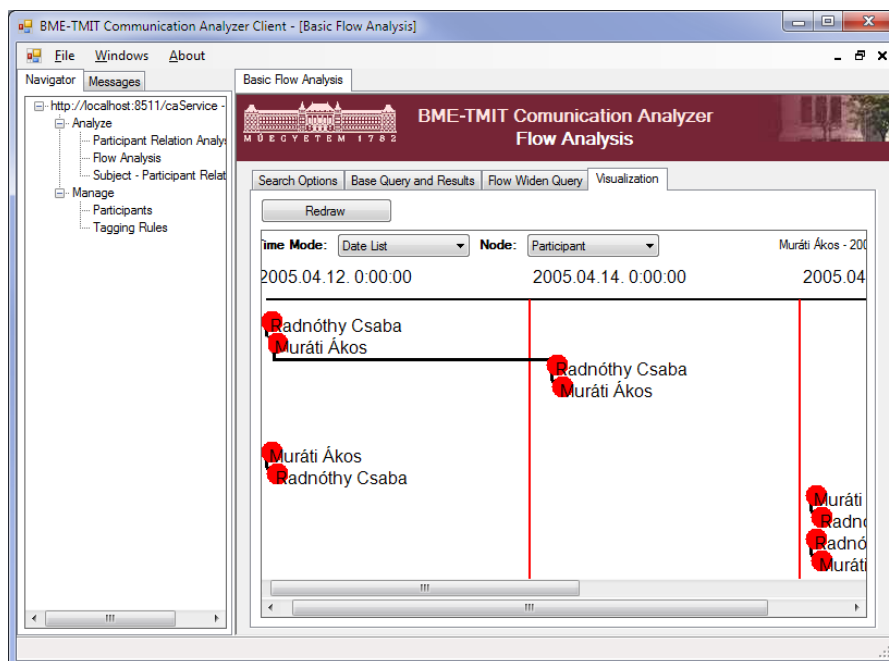
15. ábra – Téma-résztevő elemzés vizualizációs modulja

(A 15. ábra olyan elemzést ábrázol, melyben a résztvevőket csoportjaik alapján jelenítettük meg, így leolvasható, hogy azok milyen témákban kommunikálnak. A vastagított éleknek köszönhetően leolvasható, hogy az Egyetem a leggyakoribb téma a három csoport között.)

Folyamat felderítés

A folyamat felderítés során az elemzési szempontok beállítása után az elemzés eredményeinek lekérdezése hasonlóan történik, a máik két elemző modulhoz, azonban az ábrázolás módja itt eltérő. A rendszer az egyes üzenetpéldányokat az ügymentben elfoglalt helyük alapján az időtengely mentén fa struktúrában ábrázolja (16. ábra). Az ábrázolási módok között választható, hogy az üzenetpéldányok vagy azokat birtokló résztvevők által történjen a vizualizáció.

(A 16. ábrán egy folyamat felderítés eredménye látható, ahol a piros körök ábrázolják az egyes üzenetpéldányokat és az ábra felső részén elhelyezkedő vízszintes tengely mutatja az ábrázolt üzenetpéldányok dátumait. Az üzenetpéldányok között futó élek az jelölik, hogy egy példány a másiknak a folytatása a kommunikációban. Ezáltal az ábráról leolvasható, hogy Muráti Ákos 2005.04.12 –én kapott egy üzenetet, melyre csak két napos késéssel válaszolt. Ugyanakkor az e-mailre még aznap kapott választ, ám a párbeszéd ott meg is állt.)



16. ábra – Folyamat felderítés vizualizációs modulja

Az elemzésekhez tartozó további képernyőképek és magyarázatok a *C függelékben* található.

7.2.2. Ábrázolás értékelése

Résztevő- és téma-résztevő kapcsolat ábrázolása

A kliens vizualizációs moduljait a *Microsoft GLEE* komponenseire építettem, melyek a *.NET* generikus lista osztályaira épülnek, így technikai felsőkorlátot az általuk tárolt élek és csomópontokra csak a futtató hardver ad.

Ugyanakkor teszteltem ezeknek a komponenseknek abból a szempontból, hogy mennyire gyorsan képesek megjeleníteni a bementként kapott gráfot. A tesztelés során véletlenszerűen generáltam N csomópontos gráfot, melyben egyszer N , $2N$, majd $5N$ darab élét véletlenszerűen húztam be a csomópontok között. A gráf generálás idejét nem, csak az ábrázolás idejét mértem a teszt során, melynek eredményeit a 13. táblázat tartalmazza.

13. táblázat – GLEE ábrázolási teljesítménye különböző csomópont és él konfigurációk esetében

Csomópontok száma (db)	N élű gráf kirajzolása (ms)	$2N$ élű gráf kirajzolása (ms)	$5N$ élű gráf kirajzolása (ms)
0	0	0	0
1	156	-	-
2	156	312	-
5	468	625	937
10	312	468	1718
25	625	2968	5156
50	1250	6781	15312
100	3437	22813	30625
200	6875	35968	38843
250	14687	41687	43750
500	25937	76875	101250
1000	40156	155468	223281

A teszt eredményei alapján meg lehet jósolni, hogy egy átlag 200 fős vállalat esetében, ha minden alkalmazott átlagban 5 emberrel tartja gyakran a kapcsolatot, akkor a cég teljes kapcsolati hálójának kirajzolása körülbelül 3 másodpercet vesz igénybe a tesztkonfiguráción.

7.2.3. Elemzés hálózati értékelése

A megvalósított *WCF Web Service*-en küldhető üzenetek maximális méretét a *WCF* alapértelmezésben 65KB-ban határozza meg, tehát az a kliens és a szerver közötti 1 kommunikáció felső korlátja. Ebből az értékekből kiszámolható az egyes elemzési műveletek alatt átvihető eredmények is. A méréshez a *Microsoft Visual Studio WCF Test Client* eszközt hívtam segítségül, melyben a generált *Web Service* hívás és válasz *XML* üzenetek pontosan vizsgálhatóak. A 14. táblázat e mérések eredményét tartalmazza.

14. táblázat – Egy kommunikációs elemzési üzenet alatt átvihető résztvevők száma

A Elemzés	B Üres válasz mérete (byte)	C Átlagos eredmény sor méret (byte)	D Egy eredmény sorban kódolt kapcsolat (db)	E Egy üzenetben átlagosan átvihető összes kapcsolat (db)
Kapcsolati háló elemzés	5062	504	2	238
Téma-résztvevő elemzés	5042	422	2	284
Folyamat felderítés	4995	596	1	101

Az *A* oszlop jelöli, hogy milyen elemzés eredményéről van szó, a *B* oszlop az üres válasz méretét tartalmazza, ami nagyjából megfelel a válasz *XML* boríték méretének. A *C* oszlop azt jelöli, hogy egy átlagos eredmény sorban mekkora a mérete, míg a *D* oszlop azt mondja meg, hogy egy ilyen eredmény sorban hány résztvevő azonosítója vihető át. Az *E* oszlop ez alapján azt mondja meg, hogy a teljes 65KB méretű üzenet esetében maximum hány résztvevő vihető át egyszerre, az *XML* boríték méretét nem számolva ($\sim(65000-B)*D/C$).

Tehát az elemzőrendszer lekérdezéseiben maximum az *E* oszlop értékeinek megfelelő kapcsolat vihető át, ami a jelenlegi konfiguráció technikai korlátja az elemzési üzenetkezelésre nézve. Ezzel visszafelé kiszámítható, hogy egy olyan lekérdezés esetében melyeknél átlagosan 5 kapcsolat van minden résztvevő között, a rendszer csak 238/5, vagyis 47 résztvevőt tud egyszerre megjeleníteni, a résztvevő kapcsolat elemzés esetében.

A korlát természetesen növelhető az alapértelmezett *WCF* üzenet méretkorlát növelésével, de arányaiban a mért értékeknek megfelelően nem fog változni.

8. Összefoglalás

Diplomatervemben a feladatkiírásnak megfelelően az üzleti intelligencia fogalomkörének és aktuális megoldásainak ismertetésével, valamint a vállalati környezet és kommunikáció lehetőségeinek, problémakörének elemzésével reális célokat tűztem ki egy vállalati kommunikációt elemző rendszer tervezéséhez és megvalósításhoz.

Megalkottam egy általánosan használható fogalmi-modellt és az erre épülő elemzési és ábrázolási koncepciókat is vállalati kommunikáció elemzéséhez. Diplomamunkám ugyanakkor a sokféle kommunikációs csatorna figyelembevételével nagy és összetett feladatnak bizonyult, ezért dolgozatom csak az e-mail gyűjtés és feldolgozás részleteinek tervezésére és megvalósítására tér ki.

Az elemzési fázisban ismertetett funkciók alapján a rendszer tervezésekor a vállalati IT környezet sajátosságait és a modern mérnöki megoldásokat figyelembe véve felépítettem a rendszer architektúráját, megterveztem logikai adatmodelljét és adatbázis sémáját, továbbá kidolgoztam az egyes modulok és komponensek részleteit is.

A rendszer implementációja során a megtervezett modulokat és komponenseket a szövegbányászati modulok kivételével kifejlesztettem, így az öt fő funkcionális egységet megvalósítottam, melynek eredményeként elkészült a kommunikációt elemző rendszer.

A rendszer az általánosan használható kommunikációs modell segítségével más kommunikációs csatornára épülő, betöltő alrendszerek bekapcsolását is támogatja, így a jövőben elkészülhet egy a teljes vállalatot lefedő kommunikációt elemző rendszer is, ami az e-mailek, azonnali üzenetek és faxok mellett képes lesz akár a telefonbeszélgetések szöveggé alakítására és elemzésére is.

A fejlesztés során az egyes modulokat, komponenseket és alrendszereket folyamatosan teszteltem, ellenőriztem és javítottam, aminek eredményeként a rendszer működése végül helyesnek bizonyult a kitűzött funkcionális célok megvalósításával.

Az értékelés során a rendelkezésemre álló konfiguráción elvégzett mérések alapján ismertettem az e-mail betöltő alrendszerének teljesítményét és annak fokozásának lehetőségeit, valamint a klienssel végezhető elemzések és ábrázolások teljesítményét és technikai korlátait, ugyanakkor rámutattam ezek kiküszöbölésének lehetőségeire is.

Szándékaim szerint az elkészült kommunikáció elemző rendszer jövőbeli továbbfejlesztésével és az e-mail mellett további kommunikációs-csatornákra elkészített üzenetgyűjtő komponenseivel a vállalatok számára hasznos és teljeskörű elemző rendszerré válik

A függelék – Elemzés és Tervezés

A1. MIME főtípusok

Az IETF által kidolgozott MIME szabványok tartalmi elemeinél használatos nyolc MIME főtípus és rövid magyarázataik példákkal illusztrálva.

A1. táblázat – MIME főtípusok és magyarázataik néhány gyakori példával

MIME Főtípus	Leírás és néhány gyakori altípus
text	Szöveges tartalom – pl: text/plain : Sima szöveg text/richtext : RichText formátumú szöveg text/html : HTML tartalom text/xml : XML állomány text/csv : Karakterrel tagolt táblázat
image	Képi tartalom - pl.: text/png : Portable Network Graphics kép text/jpeg : JPEG kép
audio	Hang tartalom – pl.: audio/mpeg : MP3 hangfájl audio/x-wav : WAV állomány
video	Video tartalom – pl.: video/mpeg : MPEG video video/x-msvideo : AVI video
application	Speciális alkalmazás vagy állomány – pl.: application/octet-stream : Bináris alkalmazás pl: exe, class application/pdf : PDF dokumentum application/vnd.ms-excel : Microsoft Excel táblázat application/msword : Microsoft Word dokumentum
multipart	Összetett üzenetszerkezet jelölő főtípus – pl.: multipart/mixed : Több tartalmi elemet magába foglaló tartalmi elem multipart/alternative : Többféle módon eltárolt tartalom jelölője
model	Különböző modellek jelölője– pl: model/vrml : Virtual Reality Modelling Language felíró fájlja
message	Üzenetkezelési információk jelölésére szolgál – pl: delivery-status : kézbesítési jelentést hordozó tartalom

A2. Logikai adatmodell attribútumai és kulcsai

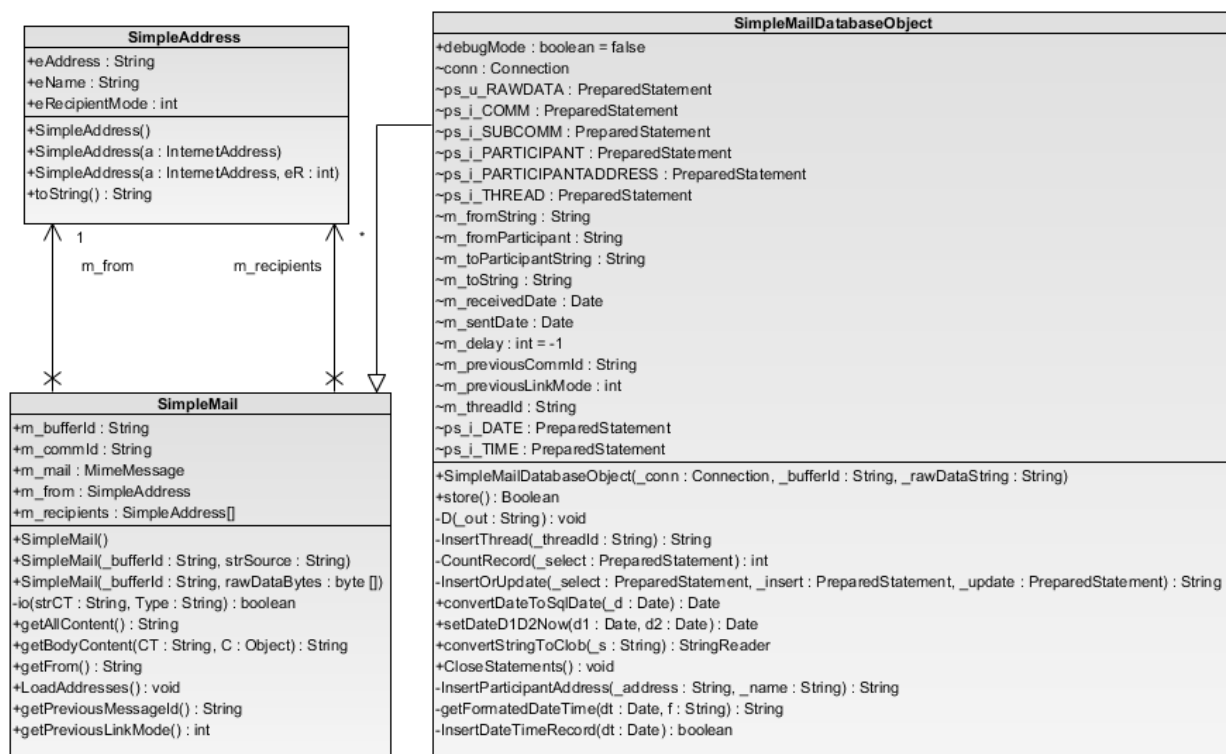
A fogalmi modell jellemzői alapján a logikai modell egyedeihez az alábbi valós attribútumokat rendeltem. Az egyedi azonosítókat aláhúzással jelöltem.

A2. táblázat – Logikai adatmodell egyedei, kulcsai és attribútumai

<div>Üzenet</div> <div><div><div>Üzenet egyedi azonosítója</div><div>Üzenet teljes szövege</div></div></div>	<div>Kézbesítés</div> <div><div><div>Kézbesítés egyedi azonosítója</div><div>Kommunikációs csatorna</div><div>Kézbesítés módja</div><div>Küldés ideje</div><div>Fogadás ideje</div></div></div>
<div>Csatolmány</div> <div><div><div>Csatolmány egyedi azonosítója</div><div>Csatolmány tartalma</div><div>Csatolmány típusa</div><div>Csatolmány fájlneve</div></div></div>	<div>Résztevő</div> <div><div><div>Résztevő egyedi azonosítója</div><div>Megjelenési név</div><div>Státusza (Azonosítatlan /Aktív /Törölt)</div></div></div>
<div>Kulcsszó</div> <div><div><div>Egyedi Kulcsszó</div></div></div>	<div>Személy ISA Résztevő</div> <div><div><div>Vállalati törzsszáma</div><div>Alapértelmezett csoportja</div></div></div>
<div>Szál</div> <div><div><div>Szál egyedi azonosítója</div><div>Szál megnevezése</div></div></div>	
<div>Cím</div> <div><div><div>Egyedi cím azonosítója</div><div>Kommunikációs csatorna</div></div></div>	<div>Csoport ISA Résztevő</div> <div><div><div>Téma</div><div>Egyedi Téma</div></div></div>

A *személy* és *csoport* egyedeknél itt jegyzendő meg, hogy azok a Résztevő egyed altípusai, ezért ISA kapcsolattal öröklik a *résztevő* egyed attribútumait. A csoport egyedeinél azért nem jelöltem attribútumokat, mivel a résztevő attribútumai pontosan lefedik ezeket.

A3. caEmailBufferParser projekt implementációja



A1. ábra – Az e-mail pufferelő modul (caEmailBufferParser) UML osztálydiagramja

A *SimpleMail* osztály végzi a *getAllContent()* függvény segítségével a levél teljes tartalmi részeinek, vagyis a tárgynak, a levél szövegének és a csatolmányok szöveges reprezentánsainak összefűzését is egy string mezőbe. Egy arra megfelelő logika a kivonatolási szabályok alapján, szövegbányászati módszerekkel a teljes tartalomból kiemeli annak kulcsszavait és meghatározza. A megvalósított rendszerben jelenleg csak a sima szöveges tartalmak kinyerése és összefűzése lett kifejlesztve, a PDF, DOC és más szöveges fájlok tartalmának dekódolása nincs megoldva.

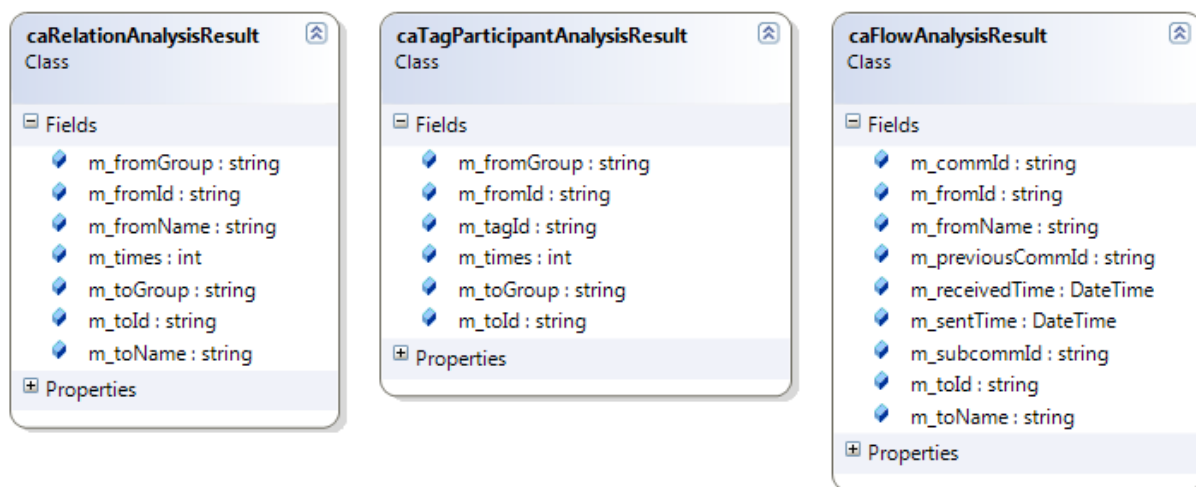
A levélben szereplő személyek beazonosítását a *SelectOrInsertParticipantAddress()* eljárás végzi az e-mailcím – résztvevő párosítások lekérdezésével. Amennyiben egy adott személy nem azonosítható e-mail címe alapján, a rendszer ismeretlen résztvevőként rögzíti, melynek későbbi feloldását a rendszer adminisztrátorára bízta.

A feldolgozás után a felépített üzenetpéldány tárolását a *SimpleMail* osztályból származó *SimpleMailDatabaseObject* végzi úgy, hogy az üzenetek *MessageID* -ja alapján – ha van neki - megkeresi az elemzési adatbázisban lévő levél szálbeli előzményét és ezek alapján kiszámolja a válaszdőket. Egy üzenet annyi példányban kerül bele a *Kézbesítés* táblába, ahány *From-To*, *From-CC* és *From-Bcc* pár képezhető belőle. Az adatbázis műveleteket az *Oracle Database 11g JDBC Driver* csomagot felhasználva *PreparedStatement* osztályokon keresztül hajtom végre.

A4. Közös rendszerkönyvár

A2. ábra - Közös rendszerkönyvtár fontosabb osztályai és enumerációi

A5. Elemzési eredményosztályok



A3. ábra – Elemzési eredményosztályok

A három megvalósított eredményosztály az eredményhalmaz 1-1 sorát prezentálják, a teljes halmazt .NET generikus lista tárolja. Ezek az osztályok az elemzés eredményének szempontjából csak a legfontosabb adatokat tartalmazzák a minimális hálózati adatforgalom érdekében.

B függelék – A rendszer használata

E függelék tartalmazza a diplomatervhez csatolt adathordozón található tartalmak ismertetését és a megvalósított kommunikációt elemző rendszer fontosabb technikai részleteit, melyek szükségesek a rendszer üzembeállításához, futtatásához és karbantartásához.

Adathordozó tartalma

A csatolt adathordozó tartalmazza e dokumentum elektronikus változatát, valamint az alábbi elemeket

Binaries mappa

E mappa tartalmazza a lefordított kliens (caClient), szerver (caServer) és az e-mail betöltő alrendszer komponenseinek binárisait (*caEmailAgents*).

Docs mappa

E dolgozathoz készült ábrák, táblázatok és egyéb dokumentumok szerkeszthető változatai

References mappa

Az elemzőrendszer megvalósításához felhasznált külső komponensek disztribúciói.

Screenshots mappa

A rendszer futásáról készült képernyőképeket tartalmazó mappa

Scripts mappa

Ez a mappa tartalmazza az elemző adatbázis sémáját, a sémát ürítő és teszt adatokkal feltöltő *SQL* fájlokat, valamint a szerver komponens *WCF* szolgáltatását engedélyező *netsh* shell scriptet is.

Sources mappa

A megvalósított rendszer Java és .NET forráskódját tartalmazó mappa.

E-mail betöltő alrendszer

Üzenetek pufferelése - caEmailBufferer

szükséges környezet és komponens:

- JRE 6 update 20
- OJDBC 6

futtatás:

java -jar caEmailBufferer.jar

parancssori paraméterek:

- property <PROPERTY FÁJL>

Beállításokat tartalmazó fájl betöltése alapértelmezésben **caEmailBufferer.settings**

- verbose

Beszédes üzemmód bekapcsolása, főleg log fájlok generálásához

- delete_buffered

Sikeresen puffertelt üzenetek törlése a forrásmappából

Üzenetek feldolgozása – **caEmailBufferParser**

szükséges környezet és komponens:

- JRE 6 update 20
- OJDBC 6
- JavaMail API 1.4

futtatás:

java -jar caEmailBufferParser.jar

parancssori paraméterek:

- `property <PROPERTY FÁJL>`

Beállításokat tartalmazó fájl betöltése alapértelmezésben **caEmailBufferParser.settings**

- `verbose`

Beszédes üzemmód bekapcsolása, főleg log fájlok generálásához

Elemzés kiszolgáló

Az adatbázis séma Oracle adatbázis alá történő telepítése és a WCF service engedélyezése után az elemzés kiszolgáló a **caServiceHost.exe** konzolalkalmazással indítható. Indítás után a konzolon megjelenik, hogy a kliensek milyen url-en csatlakozhatnak hozzá.

A szerver szolgáltatásait *Web Service*-en keresztül látja el, melynek *WSDL* –je lekérdezhető futás közben a *http://localhost:8511/caService?wsdl* url-ről.

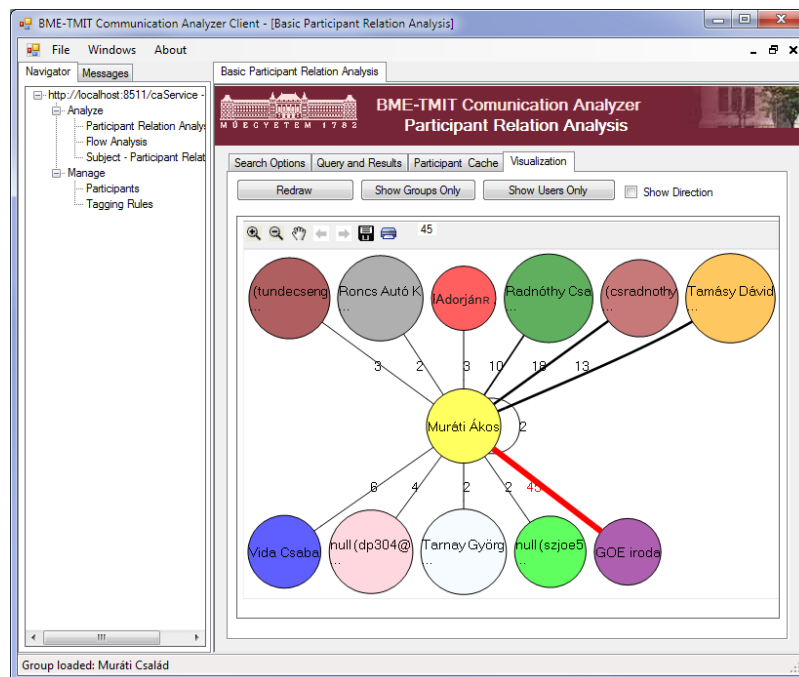
Kliens alkalmazás

A kliens alkalmazás futtatható *ClickOnce* telepítéssel vagy telepítés nélkül is. Amennyiben a felhasználó azonban az alkalmazás telepítése mellett dönt, úgy indításkor a szoftver ellenőrzi, hogy van-e frissebb verziója és amennyiben talál, felajánlja az automatikus frissítést.

A kliensek a **caClient.exe** alkalmazással indíthatók, melyek ezután a megjelenő kapcsolatkezelési felületen csatlakozhatnak az elemzés kiszolgálóra.

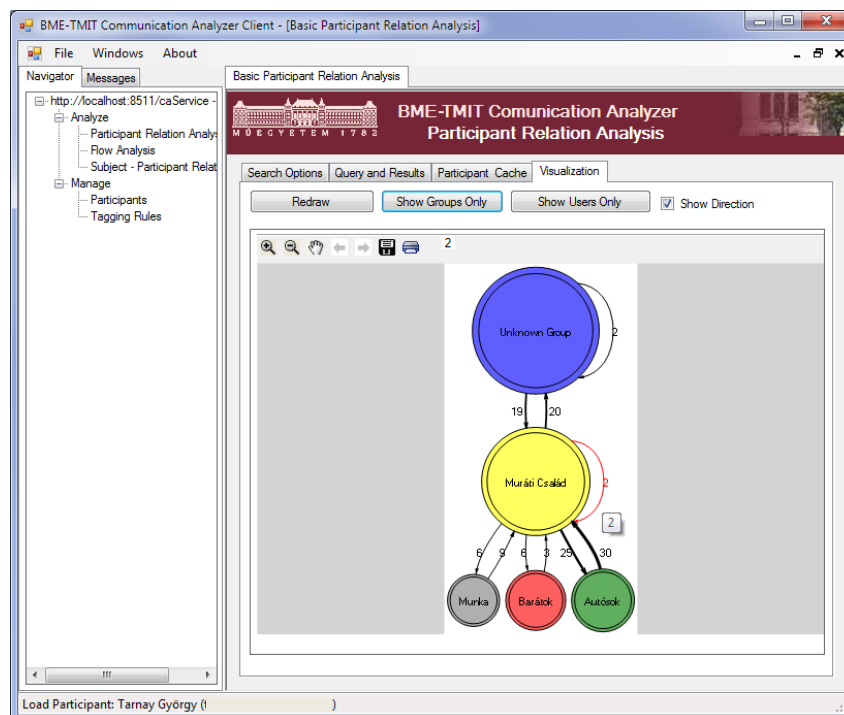
C függelék – Képernyőképek

C1. ábra – Kapcsolati háló elemzés eredménye – személy nézet, irányítás nélküli mód



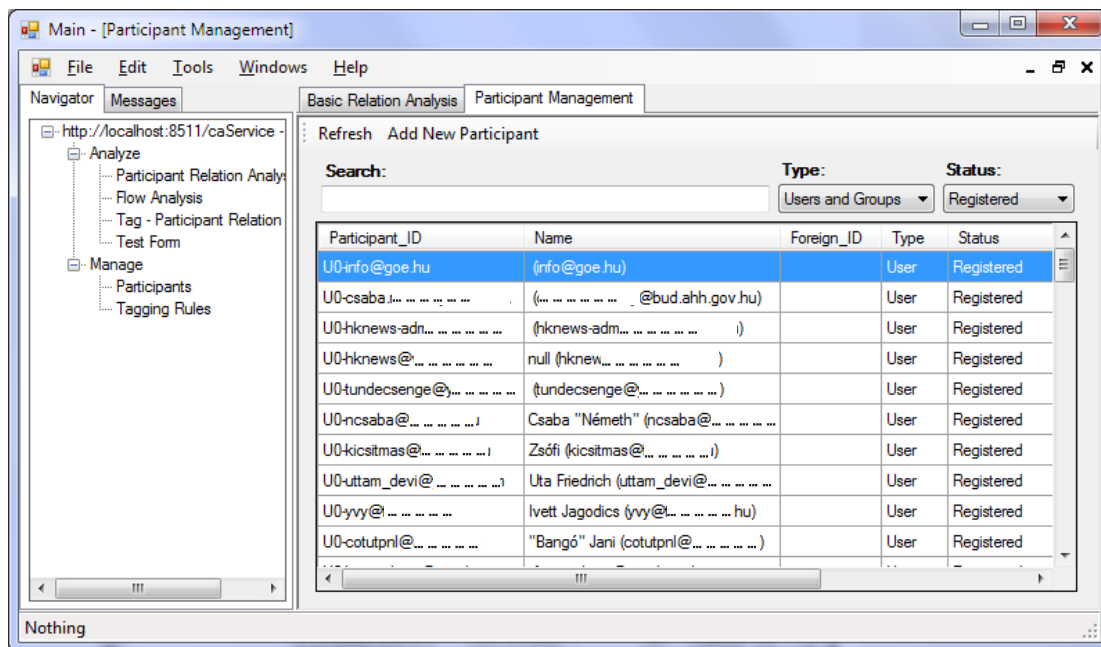
A képernyőképen egy kapcsolati háló elemzés eredménye látható, melyben a résztvevőket konkrét személyekként ábrázoltuk, a kapcsolatokat irányítását pedig kikapcsoltuk.

C2. ábra – Kapcsolati háló elemzés eredménye – csoport nézet, irányított mód



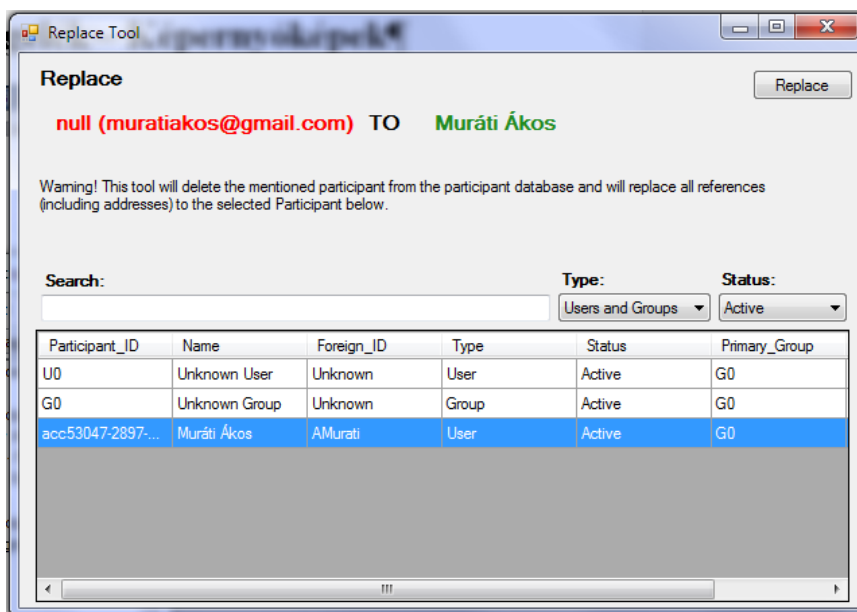
A C2 ábra a korábbi elemzéssel megegyező eredményt mutatja csoport nézetben, irányított kapcsolati élekkel. A kliens oldalon a gyorsítótárazott csoporttagságok ismeretében a program a személyeket tagsági adataik szerint csoportosította.

C3. ábra – A rendszer által regisztrált ismeretlen résztvevők listája



A C3 ábra szemlélteti, hogy ha a rendszer egy e-mailhez nem találja az adott résztvevőt, úgy ismeretlenként rögzíti **Registered** státusszal, melyet később az adminisztrátor lecserélhet vagy összefűsülhet valós felhasználói rekordokkal, a C4 ábrán látható résztvevő cserélő eszköz segítségével.

C4. ábra – Résztvevő cserélő eszköz képernyőképe



C5. ábra – 'Muráti Ákos' résztvevő rekordja és címei

The image shows two screenshots of a software window titled 'Participant - Muráti Ákos'. The left screenshot displays the 'General' tab with the following fields: Participant ID (acc53047-2897-4ec7-9796-bb3b8f7cf7d0), Type (User), Status (Active), Name (Muráti Ákos), and Foreign ID (AMuráti). The right screenshot displays the 'Addresses' tab, showing a list of addresses: 06304836850, muratiakos@hotmail.com, muratiakos@gmail.com, muratiakos@yahoo.co.uk, akos@murati.hu, and 06304836850. A 'Category' dropdown menu is open, showing options: Voice, Email, Voice, and Text. Buttons for 'Add' and 'Remove' are visible next to the dropdown.

A C5 ábra azt szemlélteti, hogy egy résztvevőnek akár több címe is lehet a rendszerben, többféle csatornához is.

C6. ábra – 'Muráti Család' nevű csoport tagjai és csoportjai

The image shows two screenshots of a software window titled 'Participant - Muráti Család'. The left screenshot displays the 'Members' tab with a list of members: Muráti Ákos and Muráti Szél Edit. The right screenshot displays the 'Member Of' tab, showing a list of groups: murati.hu domain. Buttons for 'Add...', 'Remove', and 'Set Selected to Primary' are visible. At the bottom, there is a 'Primary Group' field with the value 'murati.hu domain'.

A C6 ábrán látható, hogy a rendszerben csoportok is lehetnek tagjai más csoportoknak.

Irodalomjegyzék

1. **Ensor, Dave és Stevenson, Ian.** - *Oracle-tervezés*
Budapest: Kossuth Kiadó, 2000.
2. **Laudon, Kenneth C. és Laudon, Jane P.** -
Management Information Systems: Managing the Digital Firm (9th Edition).
Prentice Hall, 2005.
3. **Power, Daniel J.** - *A Brief History of Decision Support Systems*
DSSResource.com. [Online] 2007. március 10. [Hivatkozva: 2010. március 27.]
<http://dssresources.com/history/dsshhistory.html>
4. **Raffai, Mária** - *Információrendszerek fejlesztése és menedzselése*
Budapest: Novadat Kiadó, 2003.
5. **Kővári, Attila** – *BI Projekt: Adattárház és Üzleti intelligencia*
Üzleti intelligencia (Business Intelligence, BI) fogalma
[Online] 2007. október 27. [Hivatkozva: 2010. március 20.]
<http://www.biprojekt.hu/Uzleti-intelligencia-Business-Intelligence-BI.htm>
6. **Kókai, Béláné** - *Üzleti Intelligencia az Informatikában*
Dunaújváros: Főiskolai Kiadó, 2005.
7. **Luhn, H.P.** - *A Business Intelligence System.*
IBM Journal. 1958. október, old.: 314-319.
8. **Dresner, Howard** - *Business Intelligence at age 17.*
China Martens, Computerworld, 2006. október 23.
9. **Anandarajan, Murugan, Anandarajan, Asokan és Srinivasan, Cadambi A.**
Business Intelligence techniques: a perspective from accounting and finance
New York: Springer, 2004.
10. **Newing, Rod.** - *Higher Intelligence*
Racounter on Business Intelligence. 2009. december 1, old.: 6.
11. **Farrel, Vicky.** *HP Business Intelligence Solutions*
BI Trends and New Generation of Business Intelligence
[Online] 2010. 03 19. [Hivatkozva: 2010. 03 29.]
<http://www.communities.hp.com/online/blogs/business-intelligence-blog/archive/2010/03/19/bi-trends-and-the-new-generation-of-business-intelligence.aspx>.
12. **Gartner** - *Gartner Identifies the Top 10 Strategic Technologies for 2009*
[Online] 2008. október 14. [Hivatkozva: 2010. március 21.]
<http://www.gartner.com/it/page.jsp?id=777212>

13. **Vesset, Dan és McDonough, Brian** - *Worldwide Business Intelligence Tools 2008 Vendor Shares*
IDC paper, 2009.
14. **Oracle Magyarország** - *Az Oracle Business Intelligence megoldásai*
[Online] 2010. [Hivatkozva: 2010. 03 29.]
http://www.oracle.com/global/hu/business_intelligence/index.html
15. **Couper, Ben** - *Racounter on Business Intelligence, Introduction*
2009. december 1, old.: 1.
16. **Bakcsi, Gyula**. *Szervezeti magatartás és vezetés*
Budapest: KJK-KERSZÖV Jogi és Üzleti Kiadó, 2003.
17. **IETF, Network Working Group** - *Multipurpose Internet Mail Extensions (MIME) Part I-V*.
IETF, 1996. RFC 2045-2049.
18. **IETF, Network Working Group** - *Internet Message Format*
The Internet Society, 2001. RFC-2822.
19. **Sun Microsystems, Inc** - *JavaMail Guide for Service Providers*
Palo Alto: Sun Microsystems, Inc, 1998.
20. **Oracle Inc.** - *Oracle Data Provider for .NET Developer's Guide 11g Release 1*
Oracle Inc, 2009.
21. **Bögel, György** - *IQSqs Blog – Intelligencia*
[Online] 2006. szeptember 27. [Hivatkozva: 2010. március 18.]
<http://www.iqsys.hu/web/guest/blog/-/blogs/28816/maximized>
22. **Ward, J.** - *Principles of Information Systems Management*
New York: Routledge, 1995.
23. **Ditter, Harald** - *The Value of Business Intelligence*
Bucharest: Ensign Management Consulting, 2009.
24. **Laudon, Kenneth C. és Laudon, Jane Price** -
Management Information System: A Contemporary Perspective
New York: Macmillan Publishing, 1988.
25. **Ackoff, Russel L. és Emery, Fred**. *On purposeful systems: an interdisciplinary analysis of individual and Social Behavior as a System of Purposeful Events*
New Jersey: Aldine, 2006.
26. **Gajdos, Sándor** - *Adatbázisok*
Budapest: Műegyetemi Kiadó, 2006.
27. **Hewlett-Packard**. *Top 10 Trends in Business Intelligence for 2010*
28. **Mager, Johannes**. *Deep Email Miner Application*
Sydney: University of Technology, Sydney, 2006