

BUDAPESTI CORVINUS EGYETEM

**A SZEMANTIKUS FOLYAMATMENEDZSMENT HASZNOSÍTÁSI
LEHETŐSÉGE AZ ÜZLETI FOLYAMATOK TUDÁSALAPÚ
FEJLESZTÉSÉBEN**

Ph.D. értekezés

Varga Krisztián

Budapest

2014

Varga Krisztián

Információrendszerek Tanszék

Témavezető: Dr. Gábor András

© Varga Krisztián

BUDAPESTI CORVINUS EGYETEM

Gazdaságinformatika Doktori Iskola

A SZEMANTIKUS FOLYAMATMENEDZSMENT HASZNOSÍTÁSI
LEHETŐSÉGE AZ ÜZLETI FOLYAMATOK TUDÁSALAPÚ FEJLESZTÉSÉBEN

Ph.D. értekezés

Varga Krisztián

Budapest

2014

Tartalomjegyzék

1	Bevezetés.....	1
1.1	A kutatás célja.....	1
1.2	Az alapvető kutatási probléma.....	1
1.3	A kutatási kérdések, kihívások.....	2
1.4	A kutatás jellege, alkalmazott módszertan.....	3
1.4.1	A társadalomtudományi kutatások alapjai.....	4
1.4.2	Feltáró és igazoló kutatások – induktív vagy deduktív logika.....	4
1.4.3	Kvalitatív és kvantitatív kutatások.....	5
1.4.4	Esettanulmány alapú kutatás.....	6
1.5	A kutatás jelentősége.....	8
1.6	A kutatás előzményei, kapcsolódó kutatások.....	9
1.7	A dolgozat felépítése.....	11
1.8	Köszönetnyilvánítás.....	12
2	Elméleti áttekintés.....	14
2.1	Üzleti folyamatmenedzsment.....	14
2.1.1	Üzleti folyamat.....	14
2.1.2	Üzleti folyamatmenedzsment.....	16
2.1.3	A folyamatmenedzsment rendszer.....	17
2.1.4	Folyamatalapú szervezet.....	21
2.2	Folyamatmodellezés.....	24
2.2.1	A folyamatmodellek hasznosítási lehetőségei.....	26
2.2.2	Folyamatmodellezési módszertanok.....	29
2.2.3	Folyamatmodellező eszközök.....	39
2.2.4	Folyamatmodellek transzport formátumai.....	46
2.3	Folyamatfejlesztés.....	50

2.3.1	A folyamatfejlesztés tradicionális módjai	50
2.3.2	Üzleti folyamatok újjáalakítása/újraszervezése (BPR)	50
2.3.3	Folyamatos folyamatfejlesztés/optimalizálás (CPI).....	52
2.3.4	A BPR és CPI összehasonlítása.....	54
2.3.5	Lean menedzsment	55
2.3.6	Six Sigma	57
2.3.7	A Lean menedzsment és a Six sigma összehasonlítása.....	59
2.3.8	Folyamat-érettségi modell	59
2.4	Kapcsolódó tudásmenedzsment fogalmak ismertetése	61
2.4.1	Ontológia	61
2.4.2	Ontológia-leíró nyelvek.....	63
2.4.3	Ontológia-építés, <i>ontology learning</i>	64
2.4.4	Folyamatontológia.....	66
2.4.5	Kompetencia.....	66
2.5	Szemantikus folyamatmenedzsment	67
2.5.1	A szemantikus folyamatmenedzsment életciklusa és főbb területei	68
2.5.2	A szemantikus folyamatmodellezés eszközei és megoldásai.....	68
2.5.3	A szemantikus folyamatimplementáció eszközei és megoldásai	69
2.5.4	A szemantikus folyamat-végrehajtás eszközei és megoldásai	70
2.5.5	A szemantikus folyamatelemzés eszközei és megoldásai	70
2.5.6	A kutatás és előzményeinek elhelyezése a szemantikus folyamatmenedzsment életciklusában	71
3	Gyakorlati megoldás felvázolása.....	73
3.1	A gyakorlati megoldás áttekintése, a megoldás során felhasznált projektek és esetek ismertetése.....	73
3.1.1	Felsőoktatási intézmény folyamatainak összevetése a felsőoktatás működésének referencia folyamataival	74

3.1.2	Poszt-operatív kórházi ellenőrzési folyamat összevetése a nemzetközi standardokkal	75
3.2	Folyamatmodellező nyelvek, módszertanok vizsgálata	76
3.2.1	A folyamatmodellek szükséges és elégséges adattartalma.....	77
3.2.2	Folyamatmodellezés a Hallgatói kérvénykezelést vizsgáló eset során .	79
3.2.3	Folyamatmodellezés a Poszt-operatív ápolást vizsgáló eset során.....	80
3.3	Folyamatmodellekből történő tudáskinyerés vizsgálata	82
3.3.1	Az Adonis XML exportformátumának ismertetése	82
3.3.2	Adattranszformáció, szövegbányászati módszerek vizsgálata	82
3.3.3	A Poszt-operatív ápolást vizsgáló eset tudáskinyerésének előkészítése	83
3.4	Tudásreprezentáció	84
3.4.1	A folyamatontológia metamodellje	84
3.4.2	A folyamatontológia fejlesztett verziójának metamodellje	85
3.4.3	Folyamatontológia létrehozása a Hallgatói kérvénykezelést vizsgáló eset során	87
3.4.4	Folyamatontológia létrehozása a Poszt-operatív ápolást vizsgáló eset során	89
3.5	Tudástranszfer és folyamat-összehasonlításon alapuló fejlesztés.....	90
3.5.1	Az ontológiák összehasonlítása a Hallgatói kérvénykezelést vizsgáló eset során	91
3.5.2	Ontológia-alapú folyamat-összehasonlítás a Poszt-operatív ápolási folyamat esetén	95
3.6	A változások kezelésének vizsgálata	99
4	Összefoglalás	100
4.1	A disszertáció eredményei	100
4.2	Továbbfejlesztési lehetőségek.....	102
5	Forrásjegyzék	104
6	Idegen szavak és rövidítések jegyzéke	115

7	Mellékletek.....	116
7.1	Az Adonis XML értelmezése, a számunkra releváns tag-ekkel	117
8	Publikációs lista.....	120

Ábrajegyzék

1. ábra: A dolgozat kutatási területeinek és irodalmi áttekintésének összekötése	11
2. ábra: Folyamatmenedzsment rendszer felépítése (Scheer et al., 2002, idézi Szabó, 2009)	17
3. ábra: Az IFUA Horvát & Partners szerinti folyamatmenedzsment rendszer (Vári, é.n.).....	19
4. ábra: PMLC folyamatmenedzsment rendszer felépítése (Szabó, 2012)	20
5. ábra: A folyamatmodellek adattartalma	27
6. ábra: A folyamatmodellek hasznosítási lehetőségei	28
7. ábra: Az IDEF3 diagramjainak szimbólumai.....	32
8. ábra: Egyszerű Petri-háló tokenekkel.....	33
9. ábra: Egyszerű és kiterjesztett eseményvezérelt folyamatlánc	34
10. ábra: Egyszerű folyamat BPMN jelölésrendszerrel modellezve (White, 2004) .	36
11. ábra: A Gartner 2010-es „Magic Quadrant” kutatásának eredménye a folyamatmenedzsment eszközökre nézve (Gartner, 2010)	39
12. ábra: ARIS nézetek (Scheer, 1997 nyomán)	43
13. ábra: Az Oracle BPM Studio modellezési objektumai	45
14. ábra: A különböző folyamatmenedzsmenthez kötődő szabványok és formátumok történelmi fejlődése.....	48
15. ábra: Az ADONIS ADL szintaktikája (részlet)	49
16. ábra: Folyamatok átszervezése (Horváth et. al., 2006, 31. old).....	51
17. ábra: Folyamatoptimalizáló team (Horváth et. al., 2006, 117. old).....	53
18. ábra: A Lean öt fő alapelve	56
19. ábra: CMMI szintek (Ebizq, 2011)	60
20. ábra: A szemantikus folyamatmenedzsment életciklusa (Wetzstein et al., 2007)	68
21. ábra: A szemantikus folyamatmenedzsment területén megvalósult kutatás-fejlesztési projektek (Gábor - Szabó, 2013).....	71
22. ábra: A "Hallgatói kérvénykezelés" folyamatának kezdete	80
23. ábra: A folyamatontológia kezdeti modellje	85
24. ábra: A folyamatontológia kiegészített verziója	86
25. ábra: Adonis XML export egy részlete	88
26. ábra: Az átalakító XSLT szkript egy részlete	88

27. ábra: A folyamatontológia Protege programban megjelenítve	89
28. ábra: A "Nővér" ontológia a STUDIO rendszerben megjelenítve	90
29. ábra: Szakterületi és folyamatontológia együtt a Protege rendszerben.....	90
30. ábra: Az ontológia-összehasonlítás eredménye	97
31. ábra: Az összehasonlító jelentés.....	98

Táblázatok jegyzéke

1. táblázat: A tradicionális és a folyamatalapú szervezetek összehasonlítása	21
2. táblázat: A folyamatmodellezési lehetőségek összehasonlítása.....	25
3. táblázat: A folyamatmodellezési módszertanok általános összehasonlítása	38
4. táblázat: EPC és BPMN folyamatmodell megfeleltethetőség Adonis-ban	41
5. táblázat: A folyamatmodellező eszközök összehasonlítása	46
6. táblázat: A BPR és a CPI összehasonlítása	55
7. táblázat: A Lean menedzsment és a Six Sigma összehasonlítása (Koczor, 2009)	59
8. táblázat: A HEFOP projekt referencifolyamata alapján létrehozott új tevékenységek	93
9. táblázat: A HEFOP projekt referencifolyamata alapján megváltoztatott tevékenységek	94
10. táblázat: A HEFOP projekt referencifolyamata alapján megváltoztatott végrehajtó	94
11. táblázat: Ontológia lekérdezésre használt DL query-k	96

1 Bevezetés

1.1 A kutatás célja

A kutatás a vállalatokban meglévő munkakörök betöltéséhez szükséges tudás és kompetencia feltérképezésének újszerű lehetőségeit és hasznosítási lehetőségeit vizsgálja. Jelenleg a legtöbb vállalat esetében a munkakörökhöz tartozó feladatok nem, vagy csak rosszul strukturáltak érhetőek el. A munkaköri leírások mindig csak egy adott időpillanatban tekinthetőek érvényesnek, és a bennük szereplő szöveges elemek nem tartalmazzák mindazon kompetenciák és tudáselemek listáját, melyet a végrehajtónak valójában birtokolnia kell.

A kutatás célja olyan informatikai és tudásmenedzsment eszköz kialakítása, amely képes a szervezeti tudásvagyont a szervezet folyamatait leképező folyamatmodellekből kinyerni, azt ontológiába leképezni, majd ezt a humán szereplők és az eredeti folyamat fejlesztésére hasznosítani. A fejlesztés alapja a folyamat referenciatípusokkal, best practice-ekkel való összehasonlítása, kiemelt fontosságot adva a feltárt tudáselemeknek és kompetenciáknak. A folyamatmodellek fejlesztése így nem csak a tradicionális BPR vagy CPI alapokon valósul meg, hiszen a klasszikus elvek kiegészülnek egy tudásalapú összehasonlítás fejlesztésre használható eredményével is.

A feladat végrehajtásához szükséges ismereteket a folyamatmodellnek tartalmaznia kell. Ennek megléte és kinyerése a kutatás egyik fő megoldandó feladata. Egy másik, szintén vizsgálendő kérdés, milyen kompetenciákra lehet az ismereteket leképezni.

1.2 Az alapvető kutatási probléma

A kutatásomban a folyamatmenedzsment területéről indulva vizsgálom a tudástranszfer lehetőségeit a szervezetek életében. Célom egy működő megoldást létrehozni, mely a vállalati folyamatmodellek hasznosítását teszi lehetővé a tudástranszfer különböző területein.

A javasolt megoldás esetén a kiindulási pont a folyamatmodell, melyben megadásra kerülnek az érintett munkakörök (mint végrehajtó, döntéshozó – a RACI mátrix megfelelő kitöltésén keresztül). Amennyiben a munkakörhöz kapcsolódó folyamatok naprakészek, akkor az ezen modellekre épülő tudásfeltáró rendszer is naprakészen

mutatja be a munkavállalóktól elvárt ismereteket, illetve valós képet ad a folyamatok fejlesztési lehetőségeiről.

A folyamatmodell megfelelő elkészítése után nincs szükség a munkakörökhöz szükséges dokumentációk több forrásból való összekeresésére, hiszen a folyamatmodell maga tartalmazza a végrehajtásához szükséges elvárt tudáselemeket. A folyamat megfelelőségének értékeléséhez a referencia folyamatok vagy iparági sztenderdek megszerzése pedig már a (linked) open data területére irányítja a kutatást.

1.3 A kutatási kérdések, kihívások

Kutatásomban a folyamatmenedzsment és a tudásmenedzsment területeit kapcsolom össze, célom ugyanis a folyamatmodellekből indulva feltárni a munkakörök betöltéséhez szükséges tudáselemeket, illetve ezek kinyerésének módszereit, majd ezekből ontológia építésével a tudástranszfer megvalósításához konkrét megoldást adni.

Kutatásom problémamegoldó, feltáró jellege miatt nem fogalmazok meg statisztikai módszerekkel igazolható hipotéziseket. A disszertáció életciklus szemléletben megoldandó feladatok sorozatát adja meg, melyeken keresztül elérhető a bevezetőben említett cél. A feladatok csoportjai azonban olyan mérföldköveket határoznak meg, melyekre igazolható elméletek építhetők.

A megoldandó feladatok, nagyobb kutatási területek a következők:

- 1. Folyamatmodellező nyelvek, módszertanok vizsgálata:** Mi a folyamatmodell szerepe a kutatásban? Milyen követelményeket fogalmaz meg a folyamatmodellezéssel szemben a tudásintenzív és IKT-alapú környezet? Milyen kapcsolatban áll a folyamatmodell a szervezeti tudásvagyonnal, hogyan azonosíthatóak a tevékenységekhez köthető tudáselemek? A mi értelmezésünkben a tudásterületeket a tevékenységekhez kötjük, melyeket különböző munkakörökhöz, majd ezeken keresztül szerepkörökhöz kötünk.
- 2. Tudáskinyerés folyamatmodellekből:** A tárolt adatok milyen módon nyerhetőek ki a modellekből? Ha már tudjuk, hogy mik és hogyan kerülnek tárolásra a folyamatmodellekben, akkor ezeket valamilyen módon, és valamilyen formában ki lehet és kell nyerni. Ez lehet strukturált leíró nyelv, vagy script-es megoldás.

További vizsgálati kérdés, hogy miként lesz a kinyert adatokból strukturált tudáselem-halmaz? A kinyert adatok kezdetben csak szövegek, ideális esetben munkakörönként csoportosítva. Ebből akarjuk a munkakör elvégzéséhez szükséges tudásterületeket kinyerni, és majd egy ontológiában tárolni.

3. **Tudásreprezentáció:** Ennek egyik lehetséges megoldása az ontológia. Folyamat- és szakterületi (*domain*) ontológia kerül megkülönböztetésre a kutatásban. A kettő közötti kapcsolat létrehozása is a megoldandó feladatok közé tartozik. A nyers szövegből történő ontológia-építés - a mai napig még - nem automatizálható feladat. Az *ontology learning* terület foglalkozik az ide tartozó módszerekkel, ezeket meg kell vizsgálni és ki kell választani a feladathoz leginkább illeszkedőt.
4. **Tudástranszfer és hasznosítási lehetőségek:** Milyen területeken, milyen egyéb rendszerekkel működhet együtt az eddig felvázolt megoldás?
 - A hasznosítás egyik területe az emberi erőforrás-gazdálkodás, hiszen vállalati folyamatokra épülő kiválasztó és képzőrendszer hozható létre.
 - Cél, de az előzőből kiadódó nyereséggént is felfogható a szervezeti tudás artikulálása (feltárása) és hosszú-távú megőrzése (intellektuális vagyon kezelése).
 - A hasznosítás további területe a folyamatok fejlesztése, azok strukturális vizsgálatán túlmutató tudásalapú **összehasonlító** elemzésekre támaszkodva.
5. **A változások kezelésének vizsgálata:** Mi történik, ha változik a folyamatmodell? Hogyan kezelhető a folyamatontológia ennek megfelelő változása, miként értelmezhető a változó folyamatok fejlesztése? Hogyan valósítható meg a dinamizálás, életciklus szemlélet? Mennyiben fenntartható, illetve mennyiben képes önkorrekcióra a felvázolt megoldás?

1.4 A kutatás jellege, alkalmazott módszertan

Az informatika tudományok tudományágban akkreditált doktori iskolákban született dolgozatok esetében gyakori, hogy megoldandó feladatot definiálnak, nem kerülnek hipotézisek kimondásra, hanem helyettük kutatási kérdések, problémák sorozatát állítja fel a jelölt, és azokat oldja meg. Az ilyen kutatások kutatás-fejlesztés területébe tartoznak, és nem az igazolás vagy csupán a feltárás a céljuk, hanem egy elméletileg is megalapozott, de működő megoldás létrehozására törekcszenek. Ennek megfelelően a

végcél érdekében kisebb részekre kell bontani a kutatást, így definiálva a kutatási kérdéseket és kihívásokat, majd ezek megoldásával lehet felépíteni a teljes kutatási célt. A hipotézisek igazolását célzó értekezésekhez képest itt a probléma meg nem oldása nem fogadható el eredményként, hiszen az egyértelmű kudarcot jelent. A kihívások, kutatás kérdések azonosításával a feladat jól operacionalizálható lesz, és pontosan láthatóvá válik, ha a kitűzött célt elértük, vagy ha nem.

1.4.1 A társadalomtudományi kutatások alapjai

A kutatások alapvetően egy területen teljesen új elméletet feltárását célozzák a már ismeret elméletek, összefüggések más szempontú vizsgálatán keresztül, vagy pedig már azonosított, de még nem bizonyított elméleteket próbálnak igazolni, így bővítve a terület ismeretanyagát. E két cél kétféle logikát követel meg: az igazoló kutatások deduktív logikát, míg a feltáró kutatások az induktív logikát követik.

1.4.2 Feltáró és igazoló kutatások – induktív vagy deduktív logika

Az igazoló megközelítés a kutatási terület elfogadott elméleti háttéréből következtetett feltételezések, hipotézisek tesztelésére alkalmas. Ehhez deduktív logikát használ, melyet a kutatási elméletek tesztelésére alkalmaznak, hipotézisekből kialakítva az elméleteket. Jól látható tehát, hogy az igazoló kutatások során elkerülhetetlen hipotézisek alkotása, ezt követheti a kutatás megfigyelő része, majd a hipotézisek értékelése.

Feltáró megközelítést célszerű használni minden olyan esetben, amikor a kutatás területe teljesen vagy jelentős részben felderítetlen. A feltáró jellegű kutatások tipikusan három célból készülnek: a téma jobb megértését biztosítják, egy későbbi alaposabb kutatás megvalósíthatóságát tesztelik, és további kutatások számára fejlesztenek alkalmazható módszereket (Szabó, 2000). Az olyan kutatási területen, ahol ez a megközelítés megfelelő, tesztelhető hipotézisek megfogalmazása gyakran korai volna, és a folyamat, amelyen keresztül az elmélet fejlesztése zajlik, természetesen kevésbé szigorú (Babbie (2003), Benbasat et. al. (1987)). A feltáró kutatások az induktív logikára épülnek, mely szerint az elméleteket a kutatás adatainak elemzésén keresztül lehet kifejleszteni, általánosítás révén.

A Ph.D értékezéseket vizsgálva megjegyzendő például, hogy nem állít hipotéziseket Klimkó sem doktori disszertációjában (Klimkó, 2001), hanem ezek helyett megfogalmazza a kutatáshoz kötődő elvárásait. Kiemeli azonban, hogy ezt műve induktív volta miatt engedheti meg magának, hiszen értekezése nem igazoló jellegű. „Hipotézisigazoló, deduktív jellegű vizsgálatok nem szerepelnek a kérdések között, a kérdések jellege induktív. Ezért esik szó a kutatási kérdésekben "elvárásokról", és nem hipotézisekről.” (Klimkó, 2001 pp 6.)

Jelen kutatás feltáró jellegű, logikája induktív. Az értekezés során kutatási kérdéseket és feladatokat fogok azonosítani hipotézisek helyett, a kérdések fontosságát fogom indokolni és a bennük foglalt célok elérésén keresztül indoklom a témám fontosságát. A megoldott feladatokon keresztül igazolom a disszertáció témájának relevanciáját.

1.4.3 Kvalitatív és kvantitatív kutatások

Módszertani szempontból a szervezetelemzési módszereknél megszokott kvantitatív és kvalitatív megközelítéseket vehetjük alapul (Balaton - Dobák 1991). A kvantitatív módszerek alkalmazása során matematikai-statisztikai megoldásokat alkalmaznak az adatfeldolgozásban, tehát olyan kutatások esetén alkalmazható, ahol nagymennyiségű, mérhető adat áll rendelkezésre.

A kvalitatív módszerek használata akkor indokolt, ha a tudományterület mélyebb összefüggéseit akarjuk feltárni vagy megérteni – de nem számszerű adatok elemzésével. Olyan terület vizsgálatára alkalmasak ezek a módszerek, ahol még nincs kiforrott ismeretbázis, vagy ahol egy probléma, feladat megoldása a cél, és ezen keresztül épül az elmélet is.

A módszerek alkalmazásának hátrányai, esetleges egyoldalúságuk miatt a módszertani triangulációt (több különböző kutatási módszer és perspektíva használatát ugyanannak a kérdésnek az elemzésére) érdemes használni (Balaton - Dobák 1991). A trianguláció fajtái:

- kvantitatív módszereken belül többféle eljárás használata,
- kvalitatív módszereken belül többféle eljárás használata, illetve
- kvantitatív és kvalitatív módszerek kombinációja.

Jelen kutatás alapvetően kvalitatív módszereket használ, hiszen feltáró, induktív logikára épül, nagymennyiségű, mérhető adat megléte nélkül.

1.4.4 Esettanulmány alapú kutatás

Yin szerint a kutatás alapvető stratégiái a következők lehetnek (Yin 1994):

- kísérleten alapuló;
- kérdőíves felmérésen alapuló;
- másodlagos elemzésen alapuló;
- történeti elemzésen alapuló;
- esettanulmány feldolgozáson alapuló stratégia.

Yin szerint akkor érdemes esettanulmányokat használni, amikor "...hogyan és miért kérdéseket teszünk fel olyan jelenbeli események kapcsán, melyeket a kutató kevésbé képes kontrollálni" (Yin, 1994 pp. 9).

Az esettanulmány a vizsgált jelenséget annak természetes környezetében vizsgálja, többféle adatgyűjtési módszert alkalmaz kis számú vizsgálati alannyal kapcsolatban (Benbasat et. al. 1987).

Az esettanulmányok használata más módszerek helyett akkor javasolható, amikor a kutatás tárgyát képező koncepciók és kapcsolataik nem vizsgálhatók izolált módon. Ilyen szituáció esetén csak az esettanulmány módszere garantálja az elégséges mélységet az elmélet kibontakoztatásához. Ez a módszer komoly hagyománnyal rendelkezik az IT irodalomban (Lee, 1989).

Az esettanulmány megközelítésnek számos erőssége van: átfogó perspektívát nyújt, és képes a probléma mélyebb és teljesebb megértésére. Segítségével felfedezhetők olyan összefüggések is, amelyek más kutatási módszer alkalmazásával rejtve maradnának (Babbie, 2003, Galliers, 1992). Benbasat et. al. (1987) értékes meglátásokat tesz az esettanulmányokon alapuló kutatással kapcsolatban, amely, mint idiografikus kutatás a saját kontextusában igyekszik megérteni a problémát.

Benbasat et. al. (1987) összefoglalja az esettanulmányokra alapozott kutatási stratégia főbb jellemzőit:

- A jelenséget a természetes kontextusban vizsgálja
- Többféle adatgyűjtési eszközt alkalmaz
- Egy vagy néhány vizsgálati egységre vonatkozik
- Exploratív jellegű
- Nem alkalmaz kísérleti kontrollt vagy manipulációt
- Nem specifikálja előre a függő és független változókat
- Az eredmények nagyban függenek a vizsgálatot végző személy integráló képességétől
- A vizsgálat közben az adatgyűjtési módszerek változhatnak
- A jelenségek természete, oka a kérdéses, nem az előfordulásuk gyakorisága

Az esettanulmányok vonatkozhatnak egyetlen vagy sokféle esetre, és számtalan elemzési szint lehetséges a kutatáson belül. Az esettanulmányok rendszerint kombinált adatgyűjtési módszerekre (archívumok, interjúk, kérdőívek, megfigyelések) támaszkodnak, és az eredmények kvalitatívak és kvantitatívak egyaránt lehetnek.

Az esettanulmány módszerét legalább három cél elérésére lehet alkalmazni (Eisenhardt 1989):

- illusztrációs céllal (az elmélet megvilágítására)
- alkalmazható elmélet konstruálására
- már kifejlesztett elmélet tesztelésére is.

Az esettanulmányok alkalmazhatók annak értékelésére is, hogy az elmélet főbb koncepcióit támogatja-e a gyakorlat. Eisenhardt (1989) és Benbasat et al. (1987) részletes útmutatást biztosít az esettanulmányok alapján történő elméletépítő kutatás megtervezéséhez.

A módszerhez kapcsolódó bármely veszély elkerülése érdekében öt kritériumnak kell teljesülnie (Babbie, 2003):

- viszonylag semleges, neutrális célt kell kitűzni,
- ismert adatforrásokat kell használni,
- adekvát időtávot kell vizsgálni,
- ismert adatgyűjtési módszereket kell alkalmazni és
- biztosítani kell a konzisztenciát a jelenleg elfogadott ismeretekkel.

Az esettanulmány alapú kutatás nagy előnye a rugalmasság, mivel kölcsönhatást tesz lehetővé az adatgyűjtés és az adatelemzés között. Ennek a megközelítésnek kimagasló a validitása: koncepciók definiálása helyett az esettanulmányok részletes illusztrációval szolgálnak.

A megközelítés számos hátránnyal is járhat: ritkán ad pontos leírást egy nagy populáció állapotáról, és a következtetések inkább javaslatok, mint definitív konklúziók. Az esettanulmány alapú kutatás megbízhatósága is lehetséges probléma. Az általánosíthatóság szintén problematikus az esettanulmány alapú kutatás számára: a megfigyelések és mérések személyes természete olyan eredményekhez vezethet, ami nem feltétlenül replikálható mások által. Másodszor, a mély és átfogó megértés is nehezebben általánosítható, mint a szigorú mintán és szabványosított méréseken alapuló eredmények. Harmadszor, a minta torzításának lehetősége nagy (Babbie, 2003).

Jelen kutatás feltáró jellegéből adódóan esettanulmány alapú megközelítést alkalmaz, és az itt szerzett tapasztalatokon keresztül jut közelebb a felvázolt kutatási kérdések és célok megoldásához.

1.5 A kutatás jelentősége

A kutatás fő jelentősége abban rejlik, hogy az eddigi, főként statikusként használt folyamatmodelleket dinamikussá teszi. A folyamatmodellek használatára eddig az volt a jellemző, hogy feltérképező, vagy fejlesztési céllal készültek, de mindkét esetben vagy egy aktuális, vagy egy kívánt állapotot modelleznek. A modellezés eredményét aztán sok esetben szabályzatok, leírások elkészítéséhez használják a vállalatok, melyek szintén statikus jellegűek.

Az informatikai rendszerek fejlesztésekor, amikor egy folyamat megértésén keresztül az elvárások rögzítése, a követelményspecifikáció elkészítése a cél, akkor szintén statikus jellegű a modellek hasznosulása.

Ezzel szemben, a kutatásban felvázolt rendszer a folyamatmodellezésben a dinamikát hangsúlyozza azáltal, hogy a folyamatmodellezés célját kiterjeszti. A folyamatmodell az alapját képezi a rendszernek, ezért a folyamatmodell változásával a rá épülő rendszer is változni képes. Emellett, mivel a fő cél a szervezeti tudásvagyon azonosítása a folyamatmodellekben, ezért a vállalat változásait a folyamatmodelleken

keresztül azonosítva dinamikusan tudja szabályozni a megfelelő munkaerő kiválasztását, illetve maguknak a folyamatoknak a ki- és átalakítását.

Az ontológia használatával a felvázolt rendszer bármilyen szakterületen hasznosítható, hiszen a folyamatmodell „cserélhető”, de a funkcionalitása ugyanúgy elérhető marad.

1.6 A kutatás előzményei, kapcsolódó kutatások

A kutatás szervesen illeszkedik az utóbbi évtizedek során a Budapesti Corvinus Egyetem Információrendszerek Tanszékén megvalósult doktori és Európai Unió kutatások sorába.

A Tanszéknek évek óta profiljába vág az információmenedzsment, tudásmenedzsment, folyamatmenedzsment oktatása és kutatása. A legtöbb kolléga kötődik doktori disszertációjának témájával a tudásmenedzsment területéhez. A hangsúly idővel a tudás feltérképezésének eszközei (Klimkó, 2001) és a tudásmenedzsment szerepe (Fehér, 2004b) felől az ontológiák építése (Kő, 2004) és oktatásban való hasznosítása (Vas, 2007) felé tolódott el. Ezzel együtt az ERP rendszerek és a szemantikus folyamatmenedzsment illetve a web szolgáltatások (Ternai, 2009), illetve a szintén ontológia alapú workflow rendszerek (Kovács, 2010) is a kutatási témáink közé kerültek. Végül legfrissebb eredményeink a kompetencia-alapú toborzás (Kismihók, 2012), illetve a munkaerő-piaci igények és az egyetemi képzés kimeneti kompetenciáinak ontológia alapokon való összehasonlításának (Szabó, 2011) kutatásában jelentkeztek.

A fenti kutatások a legtöbb esetben egy-egy Európai Unió kutatás-fejlesztési projekt során keletkeztek. Az elméleti eredmények mellett fontosnak tartom a gyakorlati eredmények ismertetését is, mert azokon talán még inkább látszik a területek stratégiai integráltsága.

A STUDIO¹ nevű ontológia-alapú adaptív tesztrendszer adja az egyik pillérét a portfóliónak. Itt az ötlet az, hogy egy szakterületi ontológiában leképezésre kerül egy terület fogalmi hálójára, a fogalmak ismeretét feleletválasztós tesztek ellenőrzik, és egy-

¹ STUDIO: <http://corvinno.com/web.nsf/do?open&lang=en&page=proj-studio>

egy jó válasz esetén egyre részletesebb kérdésekkel. A tanuló a rossz válaszokhoz tartozó fogalmak listáját kapcsolódó tananyagokkal együtt megkapja, így fejlesztheti önmagát. Ez a rendszer így alkalmazkodik a különböző tanulási stílusokhoz, minden tanuló számára csak a releváns tananyagokat jeleníti meg, és az ontológia használata miatt bármilyen területen hasznosítható. A nehézséget az ontológia karbantartása okozza, ami csak manuálisan oldható meg (Vas et al., 2009 és Kismihók et al., 2009).

A Bologna-folyamat elindulásával felmerült a kérdés, hogy hogyan érhető el, hogy a mester-szintű hallgatókkal szemben azonos kompetenciákat várhassanak el a felsőoktatási intézmények, még ha a hallgatók különböző intézményekből, különböző alapidplomával is érkeznek (Gábor – Szabó, 2009). Ez vezetett el a kompetencia megfeleltetés területére, ekkor kezdtük vizsgálni a kimeneti és az elvárt kompetenciák közötti különbséget, illetve ezek összeegyeztethetőségét (Kő et al., 2011 és Vas – Kovács, 2008). A SAKE² projektben olyan ontológia-alapú rendszer került kifejlesztésre, amely kontrollálható módon valósítja meg az információszállítást, és már a szemantikus eszközök használatát hirdeti (Kovács, 2008).

A szemantikus eszközök használatával együtt másik kutatási témánk a folyamat alapú rendszerfejlesztés volt (ez két terület a szemantikus folyamatmenedzsmentben egyesül). Vizsgáltuk az ontológia-alapú rendszerfejlesztés területét (Ternai, 2011), majd az eBest³ projekt során kifejlesztettünk egy megoldást, amelyik a magas szintű folyamatmodellből automatikusan állít elő működőképes workflow-t az ontológia használatával (Ternai – Török, 2011). Ez a megoldás is a folyamatmenedzsment dinamikusabbá tételének egyik megvalósulása, hisz a folyamatmodell átalakításával a workflow újragenerálható, így dinamikus követi a folyamatmodellt (Kő – Ternai, 2011).

Végül, a munkaerő-piaci hasznosítás irányába mutat az OntoHR⁴ projekt, melynek keretén belül a munkaerő toborzás és kiválasztás támogatása volt a cél. A STUDIO rendszernél leírt elvek szerint a munkakörhöz tartozó elvárt kompetenciák és a munkára

² SAKE: "Semantic enabled Agile Knowledge-based E-government" (FP6 IST 027128)

³ eBest: "Empowering Business Ecosystems of Small Service Enterprises to face" (FP7 Capacities/SME 243554)

⁴ OntoHR: "Ontology based competency matching" (504151 - LLP -1 - 2009-1-HU-LEONARDO-LMP)

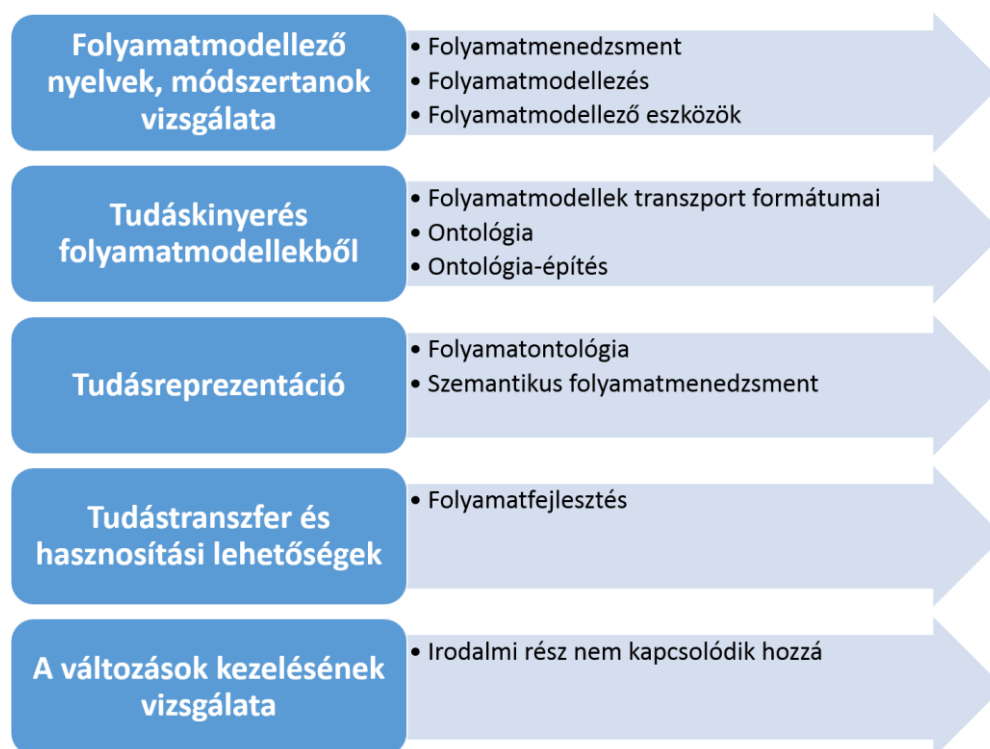
jelentkező személyek saját kompetenciái kerülnek összevetésre a feleletválasztós teszteken keresztül (Kismihók – Mol, 2011).

Az itt leírt eredményeket a 2.5.6 alfejezetben a szemantikus folyamatmenedzsment életciklusához fogom kötni.

1.7 A dolgozat felépítése

Az első fejezetben ismertetem a kutatás alapvető célját és témáját, kijelölöm a főbb kutatási kérdéseket. Itt írok a kutatás előzményeiről és jelentőségéről, illetve a használt kutatási módszertanról is.

A második fejezetben áttekintem a témában jelentősnek számító irodalmat; a folyamatmenedzsmentről, folyamatmodellezésről, a folyamatfejlesztés klasszikus lehetőségeiről, majd a tudásmenedzsment számomra fontosabb területéről, az ontológiákról és a kompetenciaelméletről, végül a két terület összekapcsolódásaként létrejövő szemantikus folyamatmenedzsmentről. Ebben a fejezetben részletesen vizsgálom a kutatási téma aktuális kérdéseit is, így mutatva be a ma legfejlettebbnek számító megoldásokat, elméleteket.



1. ábra: A dolgozat kutatási területeinek és irodalmi áttekintésének összekötése

Az 1. ábra azt mutatja be, hogyan illeszkedik a szakirodalmi áttekintés a dolgozat gyakorlati témájához. A folyamatmenedzsment és folyamatmodellezés elemzése azért szükséges, mert a javasolt megoldás folyamatmodelleket tekint kiindulási pontjának. A folyamatmodellekben azonosítja a végrehajtáshoz szükséges tudáselemeket, amit szakterületi ontológiába képez le. Ezért fontos az ontológia és kompetencia fogalmának áttekintése is. A folyamatmodellből folyamatontológia készül, ehhez áttekintem a szemantikus folyamatmenedzsment irodalmát. Végül a hasznosítás az folyamatmodellek összehasonlító elemzésén keresztül valósul meg, és a folyamatfejlesztést célozza. A változások kezelésének vizsgálata pedig megteremti a folyamatmodellekből induló dinamikus, tudásalapú fejlesztés lehetőségét.

A harmadik fejezetben a gyakorlati munka bemutatására helyezem a hangsúlyt. Ismertetem az általam fejlesztett megoldás részleteit, és a megoldás bemutatásán keresztül válaszolom meg a kutatási kérdéseket. A kitűzött feladatok sikeres megoldását a kutatás kérdések megoldásaként fogadom el, és ezeken keresztül igazolom a téma fontosságát is.

A negyedik fejezetben összegzem a kutatást, kitekintést adok a későbbi munkára, feladatokra nézve, és igazolom munkám jelentőségét.

A dolgozatot zárja az előforduló rövidítések és idegen szavak jegyzéke, illetve a források tételes jegyzéke.

1.8 Köszönetnyilvánítás

Köszönöm a dolgozatom elkészítésében nyújtott segítségét dr. Gábor Andrásnak, témavezetőmnek, aki megtalálta a nekem való témát, és végigvezetett a cseppet sem könnyű, ám annál hosszabb úton.

Köszönöm az értékes szakmai segítséget, melyet Borbásné dr. Szabó Ildikótól, Dr. Ternai Katalintól és Török Mátyástól kaptam. Ők még a tanácsstalanság pillanataiban sem engedtek csüggedni, „harci kedvük” sok mélyponton segített át.

Köszönöm értékes tanácsait dr. Fehér Péternek és dr. Klimkó Gábornak, akik idejüket nem kímélve folyamatosan visszajelzéseikkel láttak el, és ezzel járultak hozzá a dolgozat folyamatos fejlődéséhez.

Szintén köszönöm a Budapesti Corvinus Egyetem Információrendszerek Tanszéke többi munkatársának, hogy nem hagyták, hogy feleslegesen hosszan húzzam magát a kutatást.

Végezetül pedig köszönöm édesanyámnak és menyasszonyomnak, hogy tudták, mikor legyenek mellettem és azt is, hogy mikor ne. A támogatásuk és szakmai segítségük mindig jókor talált rám.

2 Elméleti áttekintés

Az elméleti áttekintés során irodalomszemlét adok a kutatási terület elméleti háttéréről. Az érintett fogalmak és területek: folyamatmenedzsment, folyamatmodellezés, folyamatmodellezési módszertanok, folyamatfejlesztés, szemantikus folyamatmenedzsment. A szemantikus folyamatmenedzsment terület elején ismertetem az ontológia és kompetencia fogalmakat is, mert a kutatás során kiemelt fontosságúak.

2.1 Üzleti folyamatmenedzsment

2.1.1 Üzleti folyamat

A különböző szerzőktől származó „folyamat” definíciók mind egyetértenek abban, hogy a folyamat egy olyan tevékenység-lánc, melynek során inputok átalakítása történik meg outputokká, valamilyen értékteremtési célnak alárendelten. Az üzleti folyamat pedig a fenti átalakítást valósítja meg, üzleti környezetben, üzleti szereplők esetén.

Wesner és szerzőtársai (Wesner et al, 1994) a kimeneteket termékként és szolgáltatásként, míg felhasználóként egy másik személyt vagy folyamatot határoznak meg, és kiemelik annak szerepét is, hogy a folyamat során fontos szerepet kapnak az abban szerepet játszó humán és egyéb erőforrások is. Hammer és Champy (Hammer-Champy, 2000) ezzel szemben erőteljesen a vevőnek történő értékteremtéssel azonosítják a folyamat outputját, míg Davenport (Davenport, 1993) kiemeli annak fontosságát is, hogy a folyamat strukturált és mért, és időben, illetve térben kiterjedt lehet. Tenner és DeToro (Tenner – DeToro, 1998) az értékteremtés helyett érték növelésről ír, és kiemeli a folyamatot alkotó emberek, módszerek és eszközök kombinálásának fontosságát.

A fentieket kiegészítve a folyamatok, és kiemelten a vállalati üzleti folyamatok rendszert alkotnak; egymástól tehát nem elszigetelten mennek végbe, mégis pontosan kijelölt kiindulóponttal és véggel kell rendelkezniük. A folyamatok végrehajtása során megkülönböztetünk rutin, végrehajtási tevékenységeket, és döntési tevékenységeket, vagy pontokat, melyek különböző folyamatágakat aktiválnak. A folyamatok végrehajtása során nem minden tevékenységnek kell szekvenciálisan lefutnia, lehetnek

közöttük egymással párhuzamosan végezhetőek, vagy olyanok, melyek egy döntés hatására nem futnak le.

Az üzleti folyamatokat Raffai három kategóriába sorolja: végrehajtási, irányítási és ellátási folyamatokra (Raffai, 1999). A végrehajtási folyamatokon azokat a tevékenységláncolatokat érti, melyek az outputot előállítják. Ide sorolhatók a kulcsfolyamatok, illetve az ezekhez kapcsolódó egyéb folyamatok, melyek az output előállításában közvetlenül játszanak szerepet. Az irányítás a fenti folyamatokat fogja össze, koordinálja azokat, erőforrásokat allokal egyes tevékenységekhez. Az irányítási folyamatok jelölik ki a megvalósítandó üzleti célt, és amennyiben szükséges, beavatkoznak a cél elérését támogató folyamatokba. Az ellátási (támogató) folyamatok pedig megteremtik azt a környezetet, mely a végtermék előállításához szükséges végrehajtó és irányító folyamatok működéséhez szükségesek, valamint biztosítják a működéshez szükséges erőforrásokat.

Egy másik, a folyamatok időbeli eltérését alapul vevő tagolás szerint a folyamatokat négy kategóriába sorolhatjuk: valós idejű (azonnali), operatív (néhány óra, nap), taktikai (néhány nap, hét) és stratégiai (hónapok, évek). Megállapítható, hogy ebben a sorrendben egyre nagyobb a folyamatok kimutatható pénzügyi hatása a szervezetre vetítve, valamint általában az egyes folyamatokhoz kapcsolódó folyamatgazdák és felelősök szervezetben elfoglalt pozíciója is egyre magasabb (Fehér, 2004a).

Az egyes folyamatok osztályozása nem triviális feladat és nem is létezik teljesen elfogadott folyamatmodell vagy keretrendszer. Raffai (1999) az APQC modelljét emeli ki, mely működési (operatív) és menedzsment (stratégiai) folyamatosztályokat azonosít, s ezek alatt 12 főfolyamatot nevez meg. Ez az általános séma aztán vállalatokként testre szabható (APQC, 2012).

A folyamatok a hagyományos, funkcionális elven működő vállalatoknál legtöbbször felszabdaltak, láthatatlanok, nem menedzselték, nem mérték és nem ellenőrzöttek (Hammer, 2001). A jó folyamat egyszerű, könnyen átlátható, világosan látható, hogyan kapcsolódik más folyamatokhoz, és mérhető. Ezen felül egyértelműen kiderül, hogy mit vár el az ügyfél, vagyis mi számára az érték. Világos a folyamat-gazda személye, illetve kihasználja a technológia nyújtott lehetőségeket (Horváth et. al., 2006).

2.1.2 Üzleti folyamatmenedzsment

A menedzsment egyfajta irányítást jelent, az erőforrások optimális koordinálásának feladatát, melynek lényege a szervezeti célok elérése. A menedzsment az a tevékenységsorozat, melyet egy vagy több személy vagy rendszer végez a szervezeti tevékenységek, erőforrások koordinálása és összehangolása céljából (Boda, 2006).

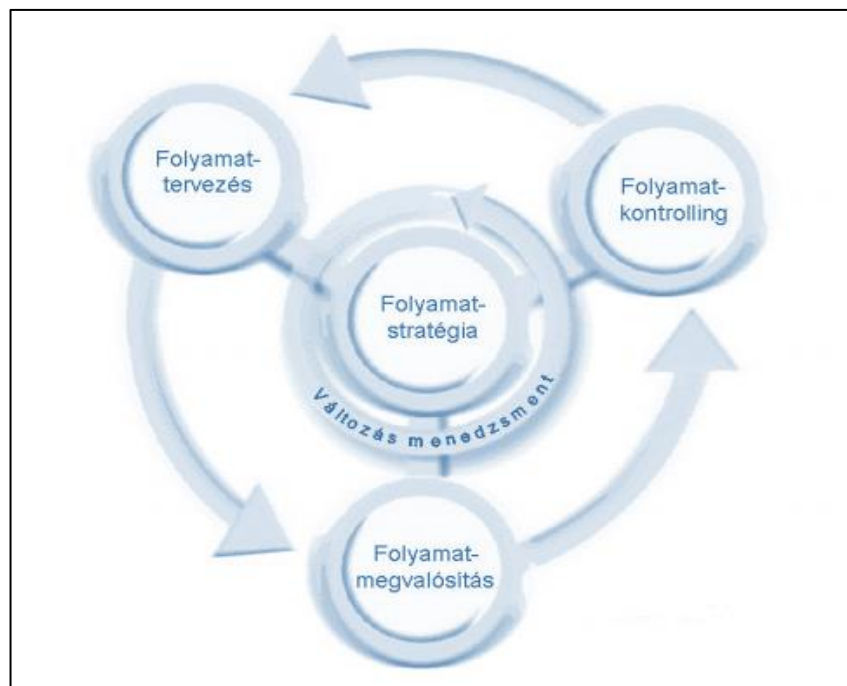
A menedzsment definícióját a folyamatokra vonatkoztatva a folyamatmenedzsment területére jutunk. A fogalomra vonatkozóan az irodalomban szintén több definíció található. Gartner szerint a folyamatmenedzsment egy olyan tudomány, ami az üzleti folyamatokat eszközökként kezeli, amik közvetlenül hozzájárulnak a vállalat teljesítményéhez a működési kiválóság és az üzleti agilitás megteremtésével (Gartner, 2012). Bácsfalvi és társai a fogalmat a folyamatmenedzsment részterületeivel írják le: folyamatok tervezésével, szabályozásával, dokumentálásával és fejlesztésével foglalkozó menedzsment területeként. A folyamatmenedzsment céljaiként pedig a folyamatok kialakítását, kézben tartását és fejlesztését nevezik meg (Bácsfalvi et al., 2001). Szabó definíciója szerint a folyamatmenedzsment a szervezet azon tudása és képessége, mellyel a belső folyamatait kialakítja és fejleszti, melynek célja a költséghatékonyság növelése, az egyszerűsítés és a versenyelőny tartós megőrzése (Szabó, 2010). Németh hangsúlyozza a folyamatmenedzsment komplex mivoltát is, amikor egy olyan integrált, összefüggéseket kezelő megközelítmódként és irányítási feladatként határozza meg, ami egyidejűleg foglalkozik a szervezeti és technológiai kérdésekkel (Németh, 2008). Scheer (Scheer et al., 2002) a folyamatmenedzsment céljaként a folyamat-kiválóság (*Business Process Excellence*) elérését nevezik meg, míg van der Aalst a meghatározásában már tetten érhetőek a folyamatmenedzsment legfontosabb feladatai a folyamatok életciklusán keresztül: [a folyamatmenedzsment célja] „az üzleti folyamatok tervezésének, bevezetésének, kontrolljának és elemzésének támogatása megfelelő módszertanok, technikák és szoftverek alkalmazásával, bevonva az embereket, szervezeteket, alkalmazásokat és dokumentumokat, illetve egyéb információforrásokat is” (van der Aalst et al., 2003).

A fenti definíciók jól meghatározzák a folyamatmenedzsment fogalmát. Egyértelműen látszik, hogy egy olyan menedzsment ágról van szó, amely a vállalatok üzleti folyamataival foglalkozik azok teljes életciklusán keresztül, ehhez támogató

eszközöket és erőforrásokat, illetve vegyes információforrásokat felhasználva, különböző pozitív hatások érdekében.

2.1.3 A folyamatmenedzsment rendszer

A folyamatok számossága, bonyolultsága, és egymástól való függése miatt a kezelésük nehézkes, megfelelő felső szintű szemlélet nélkül nem megoldható. Ezt a nehézséget segít megoldani a folyamatmenedzsment rendszer (vagy a folyamatok életciklusának) tudatos használata, amely Scheer szerint öt fő részből áll (Scheer et al., 2002), és felépítését a 2. ábra szemlélteti.



2. ábra: Folyamatmenedzsment rendszer felépítése (Scheer et al., 2002, idézi Szabó, 2009)

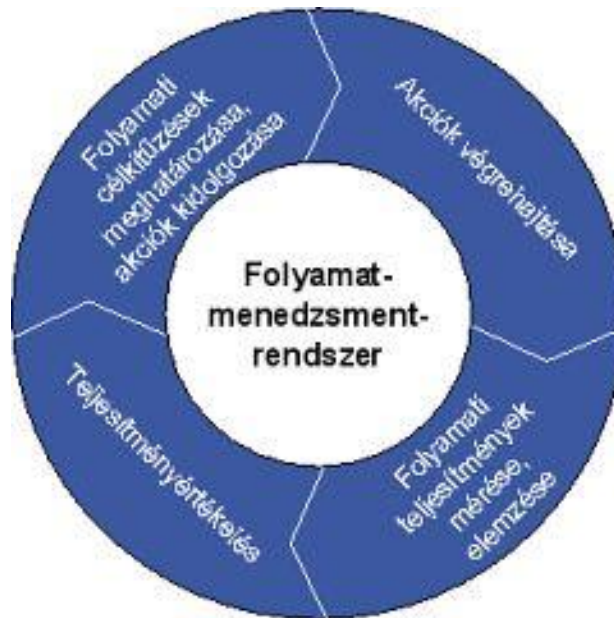
A folyamatmenedzsment rendszer öt része (Scheer et al., 2002):

- A folyamatmenedzsment rendszer alapja az **üzleti folyamatstratégia**, mely folyamatokkal kapcsolatos célok meghatározását takarja. A folyamatstratégia a szervezeti folyamatok hosszú távú terve. Tartalmazza a folyamatmenedzsment céljait, illetve azt is, hogy ezeket hogyan lehet elérni. Nevezhetjük őket kritikus sikertényezőknek. Továbbá a folyamatstratégia része, hogy analizáljuk a külső körülmények által nyújtott lehetőségeket illetve veszélyeket.

- A **folyamattervezés** során a meglévő folyamatok értékelése és újak megtervezése a feladat. A fázis a jelenlegi folyamatok vizsgálatával kezdődik. A vizsgálat után meg kell határozni azokat a folyamat attribútumokat, melyek alapján eldönthető, hogy miként kell átalakítani a jelenlegi folyamatrendszert. Ez magában foglalja a törlendő, átalakítandó és új folyamatok meghatározását is. Eközben természetesen a folyamatok kapcsolódásai is világossá válnak. Az új folyamatok meghatározása után újra elemzések következnek.
- A **folyamatok megvalósításakor** az előbb létrehozott terveket a gyakorlatba is átültetik. Ez az a fázis, ahol minden olyan változást, amit a folyamattervezés során meghatároztunk, beépítjük a mindennapi működésbe. Nagyon fontos lehet az előre meghatározott mutatószámok nyomon követése, ugyanis nem biztos, hogy a terv minden lehetőséget számba vett. A mutatószámok előrevetíthetnek olyan kritikus változásokat, melyek negatív vagy pozitív hatással lehetnek a folyamatokra. Utóbbi esetében a tovagyrűző hatások vizsgálatán kívül nincs más feladatunk, viszont potenciális negatív hatások esetén itt még beavatkozhatunk.
- A **folyamat-kontrollingnak** köszönhetően lehetőség nyílik az eredmények mérésére, értékelésére, visszacsatolásra. Az itt keletkező eredményeket hasznos lehet tárolni és elemezni, mert azok fontos építőelemei lehetnek a következő tervezési szakasznak.
- Mivel mindez folyamatos változást jelent a vállalat életében, szükséges, hogy megfelelő **változásmenedzsment** támogassa ezeket a tevékenységeket, mely a változások kezeléséhez kapcsolódó technikákat, eszközöket és folyamatokat jelenti.

Az IFUA Horváth & Partners elemzője, Vári Attila a BPM rendszert négy főbb alkotórészre osztja. Az első a folyamatokkal szembeni elvárások meghatározása. Korábban tisztáztuk, hogy a folyamatok soha nem saját magukért vannak, mindig valamilyen célt szolgálnak. Ezt a célt határozzuk meg ebben a szakaszban. Ugyanitt kerül sor az akciótervek kidolgozására. A második elem a kidolgozott akciók végrehajtása. Ezek általában valamilyen folyamatfejlesztési akciók. Ez után fontos tudnunk, hogy az akciók akkor és úgy teljesülnek-e, ahogy az a tervekben szerepelt. Az ellenőrzés során meg kell vizsgálnunk, hogy a fejlesztésekkel a kívánt hatást érjük-e el

vagy sem. Amennyiben nem, úgy visszajutunk a rendszer első eleméhez, újra kell terveznünk a folyamatokat. Végül a folyamatmenedzsment-rendszert össze kell egyeztetnünk a szervezet teljesítményértékelő rendszerével, hiszen csak így láthatjuk a valós eredményeket (Vári, ism.). A rendszert szemlélteti a 3. ábra.



3. ábra: Az IFUA Horvát & Partners szerinti folyamatmenedzsment rendszer (Vári, é.n.)

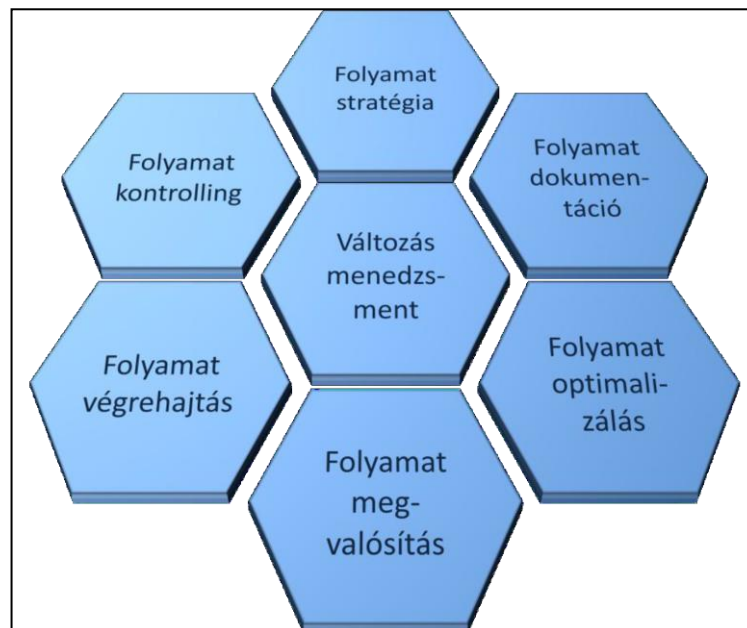
A folyamatmenedzsment életciklus (*Process Management LifeCycle*) szintén egy folyamatmenedzsment rendszer, mely a korábban már részletezett feladatokat hét nagyobb szakaszra bontja, a 4. ábrán láthatóak szerint.

A PMLC folyamatmenedzsment rendszer egyes szakaszai (Szabó, 2012):

- **Folyamatstratégia:** Az üzleti folyamatmenedzsment rendszer az üzleti folyamatstratégiából indul ki. A folyamatstratégia tartalmazza a folyamatmenedzsment céljait, amely számol a kritikus sikertényezőkkal.
- **Folyamat dokumentáció:** E fázis lényege a folyamatok felmérése és modellezése.
- **Folyamatoptimalizálás:** A meglévő folyamatok elemzését követően meg kell határozni a lehetőségeket. Az új folyamatok megtervezése után megvalósíthatósági-, illetve a költség/haszonelemzést érdemes készíteni.
- **Folyamat megvalósítás:** Ebben a fázisban az eddigi tervek alapján megvalósul a folyamatfejlesztés. A bevezetés során nagyon fontos a mérőszámok és mutatószámok nyomon követése, hogy idejében fény

derüljön a céltól való eltérésekre. Ez magába foglalja a folyamatok megújulását, a struktúra átalakítását, a meglévő IT rendszerek módosítását, vagy új rendszerek bevezetését.

- **Folyamat végrehajtás:** A végrehajtás fázisban veszi kezdetét a folyamatok működtetése. A megfelelő hosszú távú működtetés feltétele a folyamat kontrollig.
- **Folyamat kontrollig:** A folyamatok és eredmények rendszeres ellenőrzése, mérése, és értékelése nagyon fontos a szervezet hatékonyságának további fejlesztése érdekében. A kulcs teljesítménymutatók (KPI - Key Performance Indicator) nagy segítséget nyújtanak az eredmények értékelésében.
- **Változás menedzsment:** A folyamatok változtatása – csakúgy, mint egy új rendszer bevezetése, vagy bármilyen jelentősebb újítás – olyan szintű változás a szervezet életében, melyet tudatosan kell előkészíteni, levezényelni, illetve értékelni a későbbi, hasonló projektek sikeres végigvitele érdekében.



4. ábra: PMLC folyamatmenedzsment rendszer felépítése (Szabó, 2012)

A fent említett folyamatmenedzsment rendszerek mindegyikében megtalálható az a fázis, vagy fázison belüli elem, amely által megtörténik a szervezeti folyamatok felmérése és rögzítése, azaz modellezése. A folyamatmodellezés tehát a folyamatmenedzsment rendszer fontos lépése, amelynek segítségével valamilyen

módszertan szerint leképezésre kerülnek a vállalati folyamatok, hogy azok a szervezetek tacit tudásából annak explicit tudásává váljanak.

A folyamatmodellezési technikák sokszínűsége, illetve a disszertáció témája megköveteli a folyamatmodellezés részletes vizsgálatát, amit a következő fejezetben teszek meg.

2.1.4 Folyamatalapú szervezet

A technológia fejlődésével és a vállalatok növekedésével szükségszerűvé vált, hogy a ma már tradicionálisnak nevezett szemléletet felváltsa a folyamatok felismerésén és azonosításán alapuló rendszer. Az új szemlélet napjainkra szinte teljesen beépült a vállalatok működési rendjébe, kiszorítva a hagyományos rendszer elemeit.

Arthur R. Tenner, Irving J. DeToro véleménye szerint a folyamatra orientált szervezetet legkönnyebben úgy lehet bemutatni, ha összevetjük az előzőleg alkalmazott módszerekkel, hiszen itt látható meg, hogy mennyivel korszerűbbek és hatékonyabbak az ez által létrejövő megoldások.

Arthur R. Tenner, Irving J. DeToro véleménye szerint a folyamatra orientált szervezetet legkönnyebben úgy lehet bemutatni, ha összevetjük az előzőleg alkalmazott módszerekkel, hiszen itt látható meg, hogy mennyivel korszerűbbek és hatékonyabbak az ez által létrejövő megoldások.

	Tradicionális szemlélet	Folyamatra orientált szemlélet
Középpont	Vezető	Vevő
Elsődleges kapcsolat	Parancsnoki lánc	Vevő – beszállító
Orientáció	Hierarchikus	Folyamat
Döntéshozó	Menedzsment	Minden dolgozó
Vezetési stílus	Tekintélyelvű	Résztevéő

1. táblázat: A tradicionális és a folyamatalapú szervezetek összehasonlítása

A táblázat alapján megállapítható, hogy a korábban megszokott lineáris felépítésű szervezetekben az alapfeltevés, hogy a feladatok végrehajtása általában vertikális rendszerben folyik. A döntéshozás és a feladatok végrehajtása jól látható módon elkülönül. Míg a döntési helyzetben lévő szervezeti elemek a hierarchia legtetijén helyezkednek el, addig a gyakorlati végrehajtók, csak az ő parancsaik alapján végzik a munkájukat, a döntésekbe, akár az adott szintjükön belül is, nincs beleszólásuk. Ennek

köszönhető, hogy a legfontosabb pozícióban az éppen szereplő vezető áll, amely az elsődleges kapcsolatként a parancsnoki láncsal kapcsolja össze a szervezetben elhelyezkedő többi egységet. A hierarchia rendkívül jelentős szerepet tölt be az így felépült vállalatok életében, ennek köszönhetően erősen tekintélyelvű a rendszer, ahol a menedzsment szerepel csak a döntéshozók listáján. A kívánt feladatok végrehajtására így a parancsnoki láncra végigfutó utasítások alapján működik a szervezet, míg bottom-up irányban csak a felmerülő incidensek, problémák szerepelnek, amelyek megoldása ugyancsak a vezetői szinten elhelyezkedők feladata. (Németh, é.n.)

Ezzel szemben alakult ki a folyamatra orientált szemlélet, ahol a középpont szerepét átveszi a vevő, valamint az elsődleges kapcsolat helyére a parancsnoki lánc helyett, a vevő és beszállítói lánc kerül. Így az az elmélet kerül előtérbe, amely szerint a folyamatok nem pusztán a vállalaton belül történnek, hanem átlépik annak határait, ezáltal a vevő és beszállító is a folyamat részévé válik, így fontos szerepet töltenek be a folyamat egésze szempontjából, hiszen a cél a vevő számára való értékteremtés. A vertikálisan haladó munka helyére pedig a horizontális kerül. Ezáltal akik az előző rendszerben csak végrehajtották a feladatokat, a saját szintjükön döntési jogot is kapnak, annak az elvnek alapján, hogy ők dolgoznak az adott szinten. A döntések és a problémák megoldása így kevésbé időigényes, valamint a teljes folyamatról több információval rendelkeznek. Ehhez természetesen hozzá kapcsolódik a felelősség megosztása is, amely ez által már nem csak a menedzsment szintjén jelenik meg. A legnagyobb nehézsége ugyanakkor e szemléletnek, hogy a folyamatok logikája nem kapcsolódik egyértelműen a szervezet szerkezetéhez. Míg a folyamatok, ahogy említettem, átlépik a szervezet határait, addig a felépítés általában funkcionális vagy divízionális szerkezetű. Ennek hatására a hagyományos szemléletben minden résztvevő csak addig felügyelt a folyamatokra, ameddig az funkcionálisan vagy divízionálisan hozzá tartoztak. Ezáltal nem voltak elkülöníthető felelősök és a kapcsolódási pontokon problémák jelentek meg, amelyek információhiány esetén késlekedést, hatékonyság csökkenést okoztak. Ennek megoldására a folyamatalapú szemlélet esetén minden folyamathoz tartozik egy folyamatgazda, aki felelős az érintett folyamat futásának minőségéért és probléma esetén azok kijavításáért. (Németh, é.n.)

A folyamatok többféle módon is csoportosíthatóak. Akár felhasználás helye szerint szervezeten belül, fizikai / telephelyhez kötött beosztás vagy felhasznált inputok

minősége alapján, néhány példát említve csak. Gyakori kategorizálás a szerep szerinti kategóriák kialakítása, irányítási folyamatok, alapfolyamatok és támogató folyamatok.

Első csoportban találhatóak azok a folyamatok, amelyek több folyamat működését határozzák meg (kontrolling, stratégiai folyamatok), majd a stratégiai célok megvalósításához szükséges tevékenységek (értékesítés, marketing), végül pedig azok a folyamatok, amelyek segítséget nyújtanak az előző két csoportba soroltak működéséhez (HR, IT támogatás).

A folyamat alapú szemlélet megjelenik a minőségirányítás alapelvei között is. Itt is ugyanaz a folyamat definíció jelenik meg, mely szerint a tevékenységeket folyamatokkal leírni úgy lehet, hogy a bementeket az előírt módokon, olyan outputtá alakítják, amelyek értéket képviselnek a továbbiakban, legyen az érték akár a következő folyamatszámára, vagy a vevő számára. Az alábbi általános követelmények szerepelnek a rendszerben (Németh, é.n.):

- A folyamatokat azonosítani kell és le kell írni;
- Meg kell adni a folyamatok sorrendjét és kölcsönhatásait;
- Meg kell határozni a folyamatok működtetéséhez és szabályozásához szükséges kritériumokat és módszereket;
- Gondoskodni kell a folyamatok működtetéséhez és méréséhez szükséges erő –és információforrásokról;
- Rendszeresen mérni és elemezni kell a folyamatokat;
- Gondoskodni kell a folyamatok állandó fejlesztéséről.

Főbb szempontok, mint látható, a folyamatok minél átfogóbb megismerése, ellenőrzése, közöttük fennálló kapcsolatok bemutatása, valamint kezelése a folyamatok előtt, közben és után. Bár a szervezetek nagy része eltérő módon kezdett el a folyamatok szabályozásával foglalkozni, ennek hatására több szabályozás is kialakult például: ISO TS 16949, ISO 14001. (Németh, é.n.)

Tekintve, hogy ezek a szabályozások nem térnek el alapjaikban, inkább kiegészítik egymást, így napjainkban a szervezetek összetett rendszerek kialakítását tűzték ki célul, hogy több keretrendszernek való megfelelés kiegészítse egy rendszer használatának hiányosságait. Ezen alapelvek alapján jöhet létre, a jelenleg is alkalmazott

folyamatmenedzsment rendszer, amelynek leghatékonyabb használatához nyújt segítséget a folyamatmodellezés módszertana.

2.2 Folyamatmodellezés

A modell nem más, mint a való világ egy részének egyszerűsített példánya. Az üzleti folyamatmodell a Business Dictionary szerint: „Egy adott üzleti tevékenységhez tartozó funkciók szekvenciális ábrázolása.” (WebFinance, Inc, 2013). A WfMC definíciója túlmegy a funkciókon, ugyanis értelmezése szerint a folyamatmodell meghatározza a folyamat tevékenységeinek sorrendjét és mindazon erőforrásokat (gépi vagy humán), melyre a folyamat lefutása során szükség van (WfMC, 1999). Továbbiakban a WfMC definícióját fogadjuk el.

A modellezés célja, hogy felmérjük, majd pedig elemezhessük és javíthassuk a folyamatainkat. A folyamatmodellek különböző információkat hordozhatnak és változó lehet a befogadó fél is, így nem mindegy, hogy milyen szemszögéből nézve készítjük el őket. A folyamatmodellek alapvetően két csoportra bonthatóak. Léteznek As-Is modellek, amik a jelenlegi helyzetet mutatják be, és To-Be modellek, amik a kívánt szituáció ábrázolását takarják. Igazán izgalmas a kettő közötti átmenet megragadása.

A folyamatok leírására többféle lehetőségünk is van, és legtöbbször ezek kombinációját használják a felelős személyek, ahelyett, hogy egyetlen egyhez ragaszkodnának. Készülhet szöveges folyamatleírás vagy táblázatos leírás is, de a legcélravezetőbb a grafikus, modell-orientált leírás, amely segítségével sokkal egyértelműbben, és átláthatóbban ábrázolhatjuk az adott folyamatokat.

	Előnyök	Hátrányok
Szöveges folyamatleírás	<ul style="list-style-type: none"> egyszerűen előállítható (pl.: szövegszerkesztővel), a folyamat ismeretén kívül semmilyen előképzettséget nem igényel, mindenki számára könnyen értelmezhető és rugalmasan módosítható. 	<ul style="list-style-type: none"> a szövegek nem egyértelműek, felhasználók egyénileg értelmezik a leírtakat szöveges leírás alapján nem lehet további elemzéseket végrehajtani nehezen ellenőrizhető a teljes körűség nincs mód a leírt folyamatok kiértékelésére átfutási idők

Táblázatos folyamatleírás		<p>optimalizálásnak elemzése nehézkes</p> <ul style="list-style-type: none"> • folyamatok közötti összefüggések nehezen értelmezhetőek • folyamatköltségek nyomon követése nem lehetséges • folyamatlépések felelőseinek meghatározása nehézkes • folyamatok karbantartása nehezen kivitelezhető
	<ul style="list-style-type: none"> • Jól strukturált • A szövegesnél jobban áttekinthető • A folyamat érintettjei, input-jai és output-jai jobban azonosíthatóak 	<ul style="list-style-type: none"> • aktualizálás meglehetősen költséges • az összes vállalati folyamat nem írható le egyetlen táblázattal • továbbra is nehézkes a kiértékelés és változtatások átvezetése • táblázatba történő táblázatba történő leírás átláthatatlanná teheti a folyamatokat • több folyamatot érintő optimalizálási lehetőségek felismerése nehézkes
Grafikus folyamatmodell	<ul style="list-style-type: none"> • áttekinthető, könnyebben olvasható és hamarabb értelmezhető • könnyebb a modell megismertetése és kommunikálása szervezeten belül és kívül • főként releváns információkat tartalmaz • a szoftver segítségével ellenőrizni, szimulálni lehet folyamatot 	<ul style="list-style-type: none"> • egységes folyamatábrázolás hiányában a szisztematikus elemzés és összehasonlítás teljesen lehetetlenné válik • alapvető ábrázolási szabályok ismerete szükséges • nehézkes a kiértékelés és változtatások átvezetése

2. táblázat: A folyamatmodellezési lehetőségek összehasonlítása

A 2. táblázat áttekinti a folyamatmodellezési módszertanok előnyeit és hátrányait. Az első típus, a szöveges leírás rendkívül egyszerű megoldás, könnyen kivitelezhető, azonban ez a módszer többféle hibát is magában hordoz, mint például a szövegek egyéni értelmezéséből adódó eltéréseket, könnyen átláthatatlanná válik, nehezen

ellenőrizhető és értékelhető, stb. Ez a módszer a másik kettő alapjának is tekinthető, hiszen ezek célja a szöveges módszer hibáinak kiküszöbölése.

A következő megoldás a táblázatos ábrázolás, mely már jóval strukturáltabban és átláthatóbban jeleníti meg a folyamatokat, azonban az összefüggések és komplexebb folyamatok még mindig nehézkesen ábrázolhatóak.

A legfejlettebb módszer a grafikus, modellorientált ábrázolás mely folyamatábrák segítségével mutatja be a folyamatokat, így biztosítva a modell átláthatóságát, érthetőségét. A grafikus ábrázolás elkészítésére többféle számítógépes programot is használhatunk, melyek megkönnyítik az elkészítést, de ebből adódóan egy hátrányt is magában hordoz a módszer, mégpedig azt, hogy a többféle ábrázolási módból adódóan, nehezen értelmezhető modellek jöhetnek létre. Ennek kiküszöbölésére azonban már bevezetésre kerültek olyan modellező nyelvek, szoftverek, melyek egységesítik az ábrázolás módját, ilyenek például az UML szabvány, vagy az ARIS termékcsalád (és módszertan).

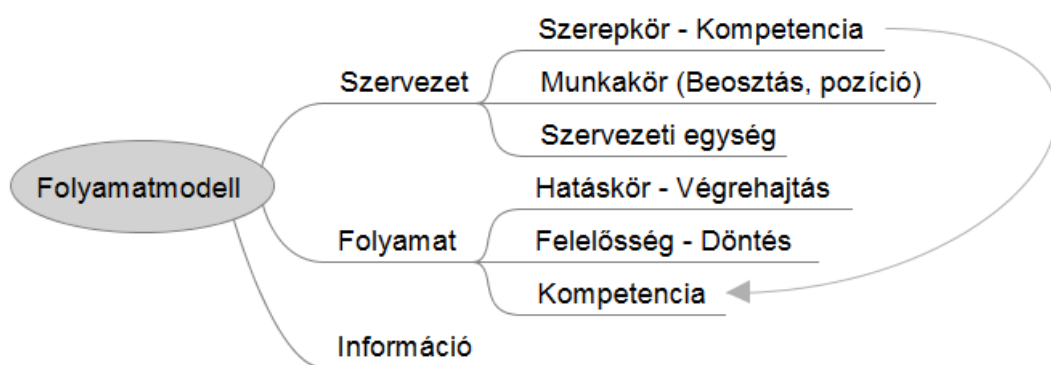
Az üzleti folyamatok modellezésével az üzleti stratégia és az informatikai rendszerek között olyan kapcsolat hozható létre, ami nagyban hozzájárulhat üzleti értékünk növeléséhez. A modellek készítése sok szempontból is hasznunkra válhat: egységes nyelvezetet szolgáltat a felhasználók között, a jelenlegi és jövőbeli folyamatok dokumentációjának elkészítésben is a segítségünkre lehet, valamint elemzésre is használhatjuk. Mérhetjük és javíthatjuk a folyamatok költségét, sebességét és rugalmas workflow-kat hozhatunk létre, amik alkalmazkodnak a későbbi változásokhoz. Az üzleti folyamatmodell segítségével javíthatjuk a folyamat által létrehozott outputot és annak elkészítésére is hatással lehetünk.

2.2.1 A folyamatmodellek hasznosítási lehetőségei

Kutatásomban a folyamatmenedzsment és a tudásmenedzsment területeit kapcsolom össze, célom ugyanis a folyamatmodellekből indulva feltárni a munkakörök betöltéséhez szükséges tudáselemeket, illetve ezek kinyerésének módszereit, majd ezekből ontológia építésével a tudástranszferhez való hozzájárulást elősegíteni.

A folyamatmodellekben ugyanis leírásra kerülnek a szervezeti folyamatokat alkotó tevékenységek, a szükséges kimenetek és bemenetek, a végrehajtók, és a támogató IT

rendszerek. A végrehajtókat a szervezeti hierarchia megadásával szokás egyértelműsíteni, ebben a hierarchiában kerül leképezésre a szervezet struktúrája. A szervezeti egységekhez alárendelt szervezeti egységek, azokhoz munkakörök (beosztások, pozíciók), majd ezekhez szerepkörök tartoznak. A szerepkörökhöz felelőségek és hatáskörök tartoznak; az első megadja a szerepkör döntési lehetőségeit, míg a második a szerepkör végrehajtási feladatait adja meg. A munkakörökhöz munkaköri leírás társítható szerepköreinek végrehajtott feladatai és a hatásköre alapján, és így megadható - többek között - a munkakör betöltéséhez szükséges tudásterületek leírása. A folyamatmodellek adattartalmát mutatja be az 5. ábra.



5. ábra: A folyamatmodellek adattartalma

A fent ismertetett felelőségek és hatáskörök a folyamatmodellekben RACI mátrix segítségével adhatóak meg egyértelműen. A RACI mátrix négy kifejezés betűszava, melyek a tevékenység során betöltött szerepet jelképezik:

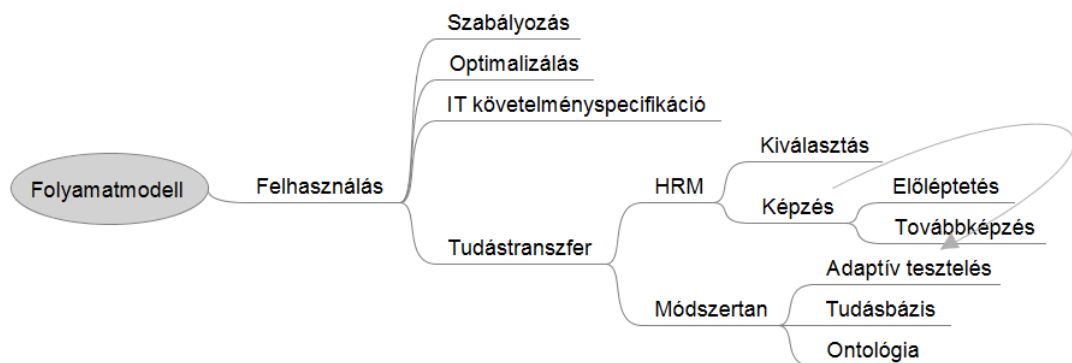
- a Responsible, azaz a Végrehajtó szerepkör (az, aki az adott tevékenységet véghezviszi);
- az Accountable, azaz a Felelős szerepkör (az, aki a feladat végrehajtását jóváhagyja, vagy aki a végrehajtásért elszámoltatható);
- a Consulted, azaz a Konzultált szerepkör (az, aki a tevékenységben indirekt, tanácsaival, vagy társként vesz részt)
- az Informed, az a Tájékoztató szerepkör (az, akinek a tevékenység végrehajtása során, vagy a végrehajtásról tudomást kell szereznie).

A RACI mátrix a folyamatmodellekben egyértelművé teszi a felelőségeket és hatásköröket, ezzel együtt azonban egy tevékenységhez több szerepkört is kapcsolhat. A

különböző szerepekhez pedig eltérő tudáselemek és kompetenciák tartozhatnak, melyeket egy jól felépített folyamatmodellnek tartalmaznia kell.

A folyamatmodellek tehát tartalmazzák azon tudáselemeket a tevékenységekhez kötve, melyeket a tevékenységek végrehajtóinak ismerniük, vagy kompetencia szinten birtokolniuk kell.

A folyamatmodellek alapvető hasznosítási lehetőségeit mutatja be a 6. ábra.



6. ábra: A folyamatmodellek hasznosítási lehetőségei

Szabályozás: egyértelműen leírásra kerül a folyamat, ügymenet, beleértve minden érintettet és végrehajtót, döntéshozót. Ez innentől előírassá tehető. A legtöbb vállalat a folyamatmodelljeit arra használja, hogy egyszerűen, közérthetően mutassa be a munkavállalóknak, mik a feladataik, mik az elvárt feladatok.

Optimalizálás: A folyamatmenedzsment fontos területe a felmért folyamatok fejlesztése. Erre kínál lehetőséget a folyamatok radikális átalakítása (BPR – *Business Process Reengineering* vagy *Redesign*), vagy a folyamatos fejlesztés (CPI – *Continuous Process Improvement*) módszere. Mindkettő az aktuálisan megvalósuló folyamatok fejlesztését tűzi ki célul, ám a BPR radikálisan alakítja át azokat, míg a CPI során apró változásokkal történik meg a folyamatok átalakítása.

IT követelményspecifikáció: Egy folyamatmodell az IT fejlesztés követelményeinek alapja lehet, hiszen egyértelműen láttatja, hogy miket kell megvalósítania a szoftvernek, azaz tartalmazza a funkcionális követelményeket.

Tudástranszfer: Ez két részre osztható, humánerőforrás-menedzsment és módszertani oldalra, melyek azonban szorosan összefüggenek.

- **Emberi erőforrás menedzsment (HRM – *Human Resources' Management*)** oldalról nézve, a folyamatmodell a kiválasztásnál segít, elsősorban a belőle nyerhető kompetenciáknak való megfelelés értékelése útján.. Emellett a **személyzetfejlesztés (továbbképzés)** és **karrierút-tervezés (előléptetés)** területen is segítséget nyújt az, ha látható, hogy melyik pozícióban/szerepkörben minek kell eleget tenni, és arra hogyan lehet felkészíteni az alkalmazottakat.
- A **módszertani** oldalt tekintve, egy terület folyamatainak feltérképezése során az **ontológia** segítségével leképezhetővé válik a terület fogalmi modellje, erre pedig többféle alkalmazás, pl. **adaptív tesztrendszer** építhető. Korábban említettem, a tesztrendszer így lesz a kompetenciaalapú fejlesztésre is alkalmas.

A kutatás során a folyamatmodellek tudástranszfer célokra való hasznosítását fogom vizsgálni. A cél tehát a folyamatmodellekben tárolt tudáselemek kinyerése, majd ezek strukturált tárolása és humán erőforrás menedzsment célokra való felhasználása. Ahhoz, hogy a célt elérjem, először meg kell vizsgálni a folyamatmodellezési módszertanokat, hiszen a kutatás során ezek közül azt kell kiválasztani, amelyik a legjobban illeszkedik a kutatás céljából eredő elvárásokhoz. Ezután megvizsgálom a folyamatmodellező eszközöket, mert a modellezési nyelvek mellett a modellező környezet is magában hordoz bizonyos tulajdonságokat, melyek a kutatás célját tekintve fontosak lehetnek. A modellező szoftver kiválasztásánál szintén a kutatás célja kell, hogy vezessen.

2.2.2 Folyamatmodellezési módszertanok

A folyamatmodellezés kialakulása és fejlődés során több folyamatmodellezési módszertan is kialakult. Ezek különböző célok esetén nyújtanak megfelelő megoldásokat.

A különböző módszertanok fő irányultságuk alapján a következő nagy csoportokba sorolhatóak (Li – Chen, 2009 és Hanrahan, é.n.):

Funkcióorientált módszertanok, mint az adatfolyam diagram és IDEF0. Ezek a módszerek az ábrázolt jelenség funkcióját próbálják megragadni, céljuk egy-egy folyamat céljának és értelmének, ezzel együtt lefutásának magas szintű ábrázolása. Ezek a leíró modellek a főbb tevékenységeket azonosítják, és azok bemeneteit, kimeneteit és feldolgozási lépéseit ábrázolják.

Adatorientált módszertanok, mint az entitás-kapcsolat diagramok és IDEF1x modellek. Ezek a diagramok a folyamatok információfeldolgozására, adatmozgásaira koncentrálnak. Figyelembe veszik a különböző szereplők információ- és kommunikációs igényeit is. Segítségükkel megfelelően szervezett adatelosztási, és információmenedzsment folyamatok és eszközök tervezhetőek.

Objektumorientált módszertanok, mint az UML és az IDEF4. Az objektumorientált szemlélet terjedésével ezek a modellek is megjelentek a tervezési, ábrázolási technikák között. Céljuk absztrakt modellek és magas fokon újrahasznosítható minták tervezése és azonosítása. Az UML eszközkészletéből a tevékenység diagramok (OMG, 2010) és viselkedés diagramok (Bichler et al., 1997) a leginkább alkalmasak a folyamatmodellezésre.

Folyamatorientált módszertanok, mint IDEF3, Petri háló, Eseményvezérelt folyamatlánc (EPC), BPMN. Ezek a módszertanok alapvető céljukként folyamatok ábrázolására jöttek létre. Megközelítésükben mégsem azonosak, a folyamatoknak különböző részeit hangsúlyozzák és ennek megfelelően más nézőpontokat részesítenek előnyben.

A szakirodalom bőségesen foglalkozik ezekkel a technikákkal. Petri-hálók és felső szintű Petri hálók elméletével és gyakorlatával foglalkozik Reisig több művében (Brauer – Reisig, 1986, Reisig - Rozenberg, 1998), eseményvezérelt folyamatláncokkal megalkotója, Scheer (Staud, 2001 és Scheer - Nüttgen, 2000), a BPMN-nel White (White, 2004). Ezek a megoldások a folyamat, vagy az annak alapján készített/készülő rendszer viselkedését írják le. Ezáltal jó alapot adnak az alternatív folyamat leírások közötti választásra (Betz et al., 2006). De a legtöbbször csak átalakítások után alkalmasak működő programok, szolgáltatások létrehozására. Emiatt is tekintjük ezt a folyamatmenedzsment statikus használatának.

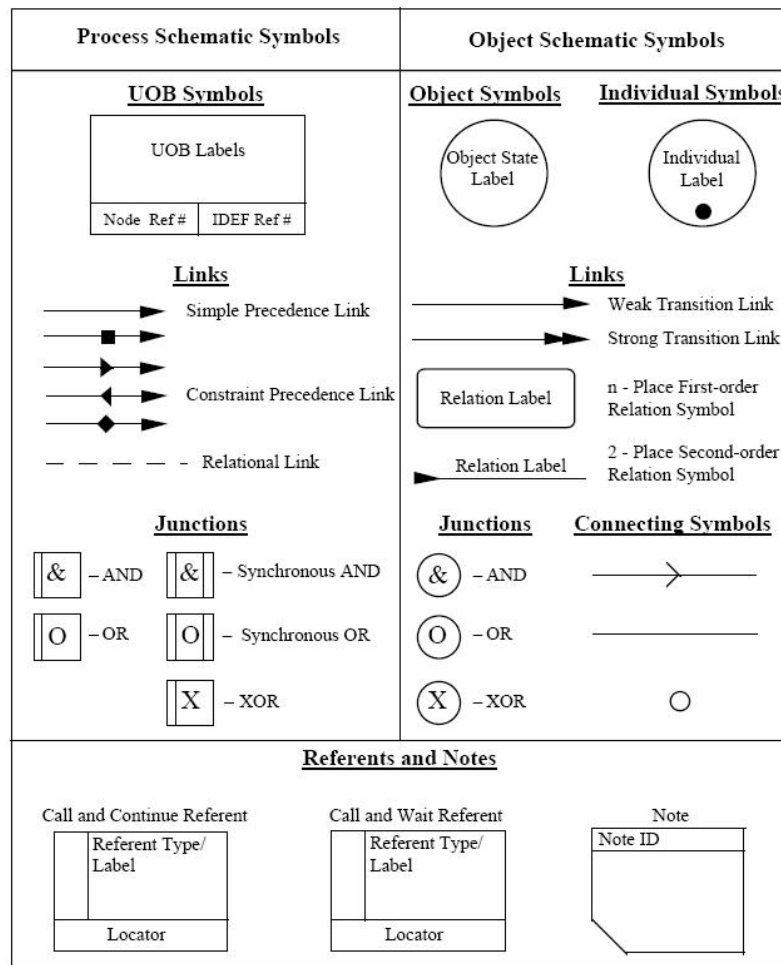
A folyamatmodellezés területén a mai napig nincs egyetlen, mindenki által elfogadott modellezési nyelv, ráadásul meg kell különböztetni egymástól az ábrázoló nyelveket és a végrehajtó nyelveket. A SOA (*Service Oriented Architecture* – szolgáltatásorientált architektúra) megjelenésével - ami az interneten elérhető webszolgáltatások megjelenését is eredményezte – egyre erősebb lett az igény arra, hogy a webszolgáltatások programnyelve közeledjen a folyamatmodellezési nyelvekhez, hiszen, ha a folyamatmodellben már leírásra került a program logikája, akkor ideális lenne, ha abból kinyerhető lenne maga a kész program is. Ez már a folyamatmenedzsment és a folyamatmodellek egyik dinamikus alkalmazása.

A folyamatmodellek üzleti alkalmazások alapjaként valós hasznosítása természetesen nem új igény. Több nyelv ezt a célozza, mint például a BPEL (Arkin et al., 2005, OASIS, 2007).

A továbbiakban Li – Chen és Hanrahan négyes csoportosítás közül a folyamatorientált módszertanokat ismertetem, mégpedig azért, mert az üzleti életben az általános célú folyamatmodellezés terén ezek a legelterjedtebbek. Az üzleti elemzők és a rendszertervezők is megértik a jelölésrendszerüket, és az üzleti életben előszeretettel használják is.

2.2.2.1 IDEF3

Az IDEF3 folyamatdefiníciós módszertan egy rendszer viselkedésének leírására szolgál elsődlegesen. A módszertan alkalmas egy általános rendszer, egy informatikai rendszer, vagy akár egy vállalat, mint rendszer működésének leírására is, így lehetőséget ad folyamatok modellezésére is.



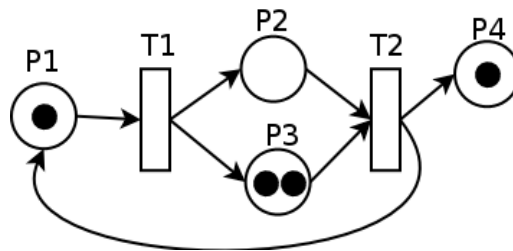
7. ábra: Az IDEF3 diagramjainak szimbólumai

A folyamatok teljességének ábrázolásához az IDEF3 két diagramot kínál: a Tevékenységfolyam Leírást (*Process Flow Description*, PFD) és az Objektumállapotváltás Leírást (*Object State Transition Description*, OSTD). A PFD írja le a tevékenységek sorrendjét, míg az OSTD az objektumok engedélyezett átmeneteit ismerteti egy adott folyamat során. Ezáltal a PFD a folyamat vázát adja, míg az OSTD a be- és kilépési pontokat adja meg, illetve az adatáramlást definiálja (Hanrahan, é.n.). A két diagram által használt szimbólumokat mutatja be a 7. ábra.

2.2.2.2 Petri háló

A Petri-háló diszkrét elosztott rendszerek matematikai ábrázolására szolgál. A Petri-hálókat az 1960-as években Carl Adam Petri vezette be. Mivel ez az ábrázolás az egy időben lezajló események megjelenítésére alkalmas, az automataelmélet általánosításának tekinthető, és folyamatok ábrázolására csak korlátozottan, vagy fenntartásokkal alkalmas.

A Petri-háló helyekből, átmenetekből és irányított élekből áll. Az élek kötik össze a helyeket az átmenetekkel és fordítva, ugyanakkor a helyek és az átmenetek saját csoportja között nincsen közvetlen élel megvalósított kapcsolat, azaz a Petri-háló irányított páros gráfok. Az egyes helyeken tetszés szerinti számú *token* fordulhat elő, amely tokenek akkor kerülnek át a következő helyre, ha az átmenethez vezető élek mindegyikén az átmenet feltétele teljesül. A 8. ábra egy egyszerű Petri-hálót ábrázol néhány tokennel.



8. ábra: Egyszerű Petri-háló tokenekkel

Jól látható, hogy a Petri-háló, bár folyamatok tevékenységeinek és azokat kiváltó eseményeknek a modellezésére alkalmasak, mégsem képesek komplex, többszereplős folyamatrendszerek közérthető, egyszerűen interpretálható ábrázolására.

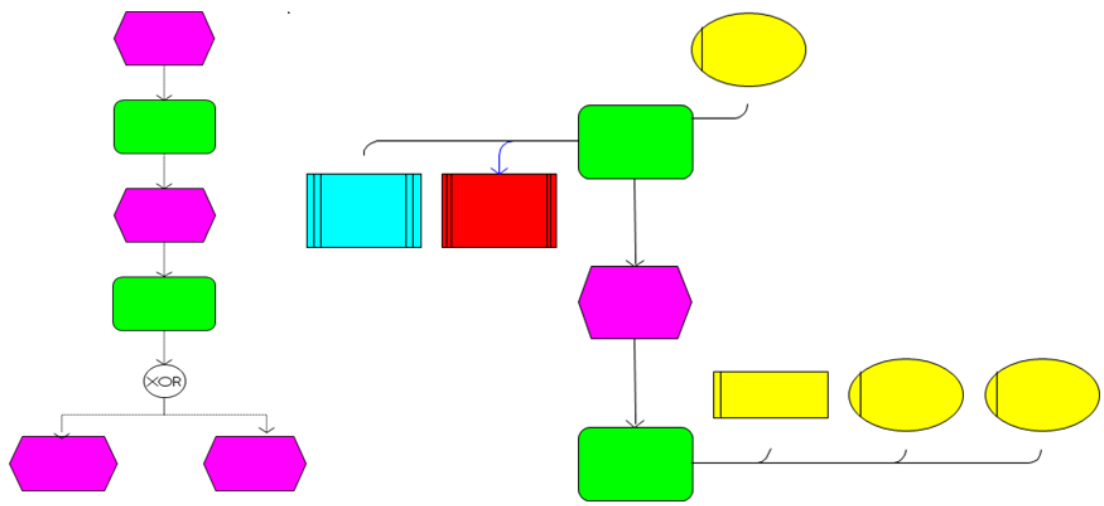
2.2.2.3 Eseményvezérelt Folyamatlánc – Event-driven Process Chain (EPC)

Az Eseményvezérelt Folyamatláncot 1992-ben fejlesztették ki Németországban a University of Saarland. A népszerűségének az oka, hogy egyetlen egy diagramon képes bemutatni a tevékenységek, események, logikai kapcsolók, szervezeti-, adat- és alkalmazási rendszereket leíró objektumokat. Itt a hangsúly azon van, hogy nemcsak a tevékenység kerül ábrázolásra, hanem a hozzátartozó erőforrások is, vagyis a szerepkörök, IT rendszerek és az adatok.

Az eseményvezérelt elnevezés onnan ered, hogy a funkciók és az események váltakozásával épül fel a folyamatmodell. Az objektumok fő jellemzője, hogy idő és logikai struktúrában kerülnek be a folyamatábrába. A folyamatdiagram teljessége szempontjából kétféle lehet. Az egyik a “karcsú” modellezés, ami csak az időbeli és a logikai kapcsolatokat helyezi el a folyamatmodellben. A kibővített eseményvezérelt folyamatlánc integrálja a funkciók és adatok, illetve a termék vagy szolgáltatás és szervezeti ábra közti statikus kapcsolatokat. (Scheer, 1999)

Az esemény meghatározza a folyamatban elhelyezkedő és információval rendelkező objektum állapotát. Ez az állapot meghatározhatja a folyamat lefolyását: elindítja azt, döntések utáni állapotokat jelöl, vagy a befejezését adja meg. Az események tevékenységeket váltanak ki, míg a tevékenységek eseményeket eredményeznek. A tevékenységekben zajlik a feldolgozás, itt történhet meg az erőforrások felhasználása és állapotának változása. A szervezeti elemek és tevékenységek között különböző kapcsolatok állhatnak fenn. Ezeket az összekötő nyilak attribútumaként adhatjuk meg. Ezek a kapcsolatok (végrehajt, dönt, közreműködik, információt ad, információt kap, stb.) azt jelölik, hogy a szervezeti objektumok mennyire érintettek a tevékenység során. Az adatok és a tevékenységek közötti input és output összefüggéseket a nyilak segítségével ábrázoljuk. Egy tevékenységhez még köthetünk informatikai rendszereket is, amik azt jelölik, hogy a feladat végrehajtásához milyen információs rendszert használtunk (Scheer, 1999).

Az egyszerű eseményvezérelt folyamatlánc csak tevékenységeket és eseményeket tartalmaz, míg a kiterjesztett EPC-ben megjelenítésre kerülhet minden további információ (ezt mutatja be a 9. ábra).



9. ábra: Egyszerű és kiterjesztett eseményvezérelt folyamatlánc

Az egész üzleti folyamatnak kulcstényezői a logikai operátorok. Minden olyan helyen szükséges a logikai kapcsolók használata, ahol több él találkozik, illetve hagyja el az eseményt vagy a tevékenységet. A logikai operátorok a folyamat lefutásában az elágazások logikai tartalmát jelölik. Az ÉS kapcsoló segítségével ábrázolhatóak a párhuzamosan zajló folyamat részek. A KIZÁRÓ VAGY operátornál választanunk kell

egyetlen folyamat ágat, amin tovább folytatódik az üzleti művelet. Míg a MEGENGEDŐ VAGY operátor azt a bizonytalanságot fejezi ki, mely szerint a folyamat akár több irányba is elágazhat, de az is lehet, hogy csak egy ágon megy tovább. A logikai operátorokhoz kapcsolódó eseményeknek és tevékenységeknek csak egy bemenő és kimenő kapcsolata lehet. Tehát ha a modellezés során több ág is fut egy Eseménybe, vagy egy tevékenységbe, akkor a logikai operátorokat kell alkalmaznunk.

A komplex folyamatrendszer átívelnek a vállalatokon, és ezt jelezni kell tudni. Erre szolgálnak a folyamatkapcsolók, melyek teljes folyamatokat kötnek össze. Jelezni kell, ha egy folyamat egy korábbi folytatása, és támaszkodik például egy ott létrejött termékre, vagy ha egy folyamat egy másikba kerül átirányításra. Megkülönböztetjük még a beágyazott folyamatokat, amikor egy különálló folyamat egy másik részeként fut le, a benne foglalt tevékenységek tehát logikailag ezen időre a főfolyamat tevékenységévé válnak (Scheer, 1999).

2.2.2.4 Business Process Modeling Notation (BPMN)

A Business Process Management Initiative (BPMI) hozta létre a *Business Process Modeling Notation*-t. Ezzel a céljuk az volt, hogy egy könnyen érthető jelölési rendszert készítsenek az üzleti élet minden érintettjének, kezdve az üzleti elemzőkkel, akik a folyamatok kezdeti tervezetét készítik el, amire alapozva a fejlesztők ezeknek a folyamatoknak a technológiai bevezetésére képesek, és végül az üzletembereknek, akik ezeket a folyamatokat irányítani, ellenőrizni fogják (BPMI, 2004). Jelenleg a jelölőrendszer a 2.0-s verzióán tart, ez az 1.2-est követte, melyet már az Object Management Group jelentett meg, és amihez képest több területen is próbáltak előrelépést produkálni. Így például megjelentek a *coreography* és *conversation* diagramok, valamint a *non-interrupting event*-ek és az *event sub-process*-ek. A módszertan egy kisebb névváltoztatáson is átesett.

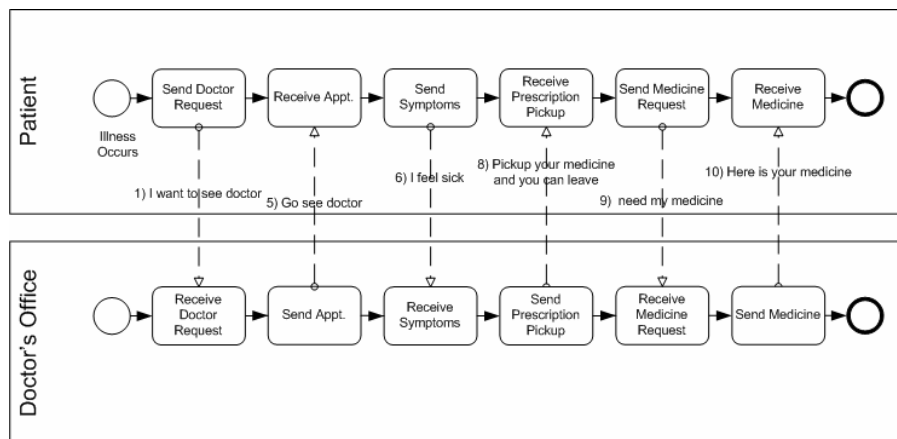
A BPMN alapvetően egy vizuális nyelv, amiben kulcstényező a megjelenés, ezért a választott alakzatokat, ikonokat minden folyamatmodellezőnek ismernie és értenie kell. A folyamatábrák olyan objektumokból állnak, melyek alaptípusain változtatni nem szabad a folyamatmodellező eszközök fejlesztőinek sem, és az esetlegesen bekerülő új objektumok sem sérthetik azokat. Ezek az elemek könnyen megkülönböztethetőek egymástól, és ezekhez az alakzatokhoz hasonlóakat használ a legtöbb modellező.

Például a tevékenységek téglalapok, a döntések csúcsra állított négyzet alakúak. Az alakzatok alapértelmezett formája nem változtatható, de szabadon színezhetőek.

A BPMN 2.0 szerint háromféle modellt különböztethetünk meg, ezek a folyamatok, koreográfia és kollaboráció diagramok (OMG, 2011).

A folyamatokon belül a nem futtatható privát folyamatok célja, hogy dokumentálják az adott folyamatot, a folyamat viselkedését, míg az futtatható privát folyamatok természetesen azzal a céllal készülnek, hogy azokat futtatni is lehessen. Abban az esetben, ha egy privát üzleti folyamat más folyamattal (folyamatokkal) vagy más résztvevővel (résztvevőkkel) kommunikál, publikus folyamatról beszélünk. Ilyenkor a privát folyamatnak csak azokat a tevékenységeit ábrázoljuk, amelyek interakcióba lépnek a másik folyamattal.

Az alapvető folyamatmodellezési objektumok különálló kategóriákba tartoznak. Ezek a csoportok jól elkülönítik a hasonló funkciójú objektumokat, és így segítséget adnak a folyamatdiagram olvasójának, hogy könnyebben felismerje az alapvető objektum típusokat, és megértse az ábrát. (White, 2004)



10. ábra: Egyszerű folyamat BPMN jelölésszerrel modellezve (White, 2004)

Az öt alapvető jelölési kategória a következő:

- Folyamat objektumok (*esemény, tevékenység, átjáró*)
- Kapcsolódási objektumok (*szekvencia folyam, üzenetfolyam, asszociáció*)
- Úszósávok (*medence, sáv*)
- Artifacts (*csoportosítás, megjegyzés*)
- Adat objektumok (*adatobjektum, bemenet, kimenet, adattár*)

A 10. ábra egy egyszerű példán keresztül mutatja be a BPMN folyamatmodellezési nyelv elemeit.

A kollaboráció két vagy több üzleti egység közötti interakciót mutat be. Medencék, folyamatok és koreográfiák is előfordulhatnak benne. A medencék jelképezik a résztvevőket, míg a köztük váltott üzenetek adják a „*Message Flow*”-t, az üzenetek áramlását.

A koreográfia az elvárt viselkedést, az üzenetváltást mutatja be a résztvevők között.

2.2.2.5 A folyamatmodellezési módszertanok összehasonlítása

A fent részletezett módszertanok közül üzleti alkalmazásban a BPMN és az EPC elterjedtebb. Az IDEF3 közelebb áll a rendszerfejlesztés területéhez, ezáltal a BPMN-hez is. Ez is magyarázhatja azt, hogy miért nem igazán elterjedt napjainkban a használata: a népszerűbb BPMN funkcionalitásában lefedi az IDEF3-at, így elsajátítása (hacsak nem egy szakterületi elvárás) nem indokolt.

A Petri-hálók alkalmazása a szakirodalomban, elméleti kutatásokban gyakran használt az erős matematikai alapok miatt. Szintén emiatt gyakran Petri-hálós alapokon működnek a folyamatszimulációt támogató megoldások is. A kutatásban azonban ezekre nem támaszkodom, emiatt most az EPC-t és a BPMN-t fogom részletesebben összehasonlítani.

A fent említett négy folyamatmodellezési módszertan általános összehasonlítását szemlélteti a 3. táblázat.

Habár mind a BPMN, mind az EPC segítségével modellezhetjük ugyanazokat a folyamatokat, sosem fogunk ugyanolyan ábrát kapni, köszönhetően a sok apró eltérésnek, ami a két módszertant megkülönbözteti, ez alatt pedig nem csak a jelölésbeli különbségeket értem.

Mindkét módszertanban jelentős szerepe van az eseményeknek és a tevékenységeknek. Az események BPMN-ben a tevékenységek következményeiként, illetve előzményeiként szolgálnak, megmutatják, hogy milyen feltételek szükségesek az adott tevékenység bekövetkezéséhez, illetve, hogy mi történik, ha elvégzésre kerül az a tevékenység. Az EPC-nél a tevékenységek és az események felváltva követik egymást, és az eseményekkel követhetjük nyomon, hogy hol tart, hogy áll éppen a folyamat. Míg

EPC-nél csak egyetlen egy objektumot használhatunk az események megjelenítésére, BPMN-ben az események típusától függően több közül is válogathatunk. A folyamat kezdetekor is eseményeket használunk, ami kiváltja magának a folyamatnak a beindulását. A BPMN esetén erre egy dedikált objektumot használhatunk, ami nem más, mint a Start Event. EPC-nél egy teljesen hétköznapi esemény szolgál kiindulópontként.

	IDEF3	Petri-háló	EPC	BPMN
Alapvető cél	Egy rendszer viselkedésének jellemzése folyamat-modellel	Diszkrét elosztott rendszerek matematikai ábrázolása	Üzleti folyamatok és kapcsolódó információk egyetlen diagramon történő modellezése	Könnyen érthető jelölésrendszer az üzleti élet minden részvevője számára folyamatok modellezésére
Diagramok	Tevékenység-folyam Leírás (Process Flow Description, PFD) Objektum-állapot-változás Leírás (Object State Transition Description, OSTD)	Petri-háló	Esemény-vezérelt folyamatlánc, melynek részletessége az elérhető "módszerekkel" (lényegében előre definiált objektum-szűrőkkel) szabható testre	Üzleti folyamatmodell, Kollaboráció, Koreográfia
Tevékenységek végrehajthatók kinyerhetőek?	Nem	Nem	Igen	Igen
Elterjedtsége az üzleti folyamatok ábrázolásában	Nem elterjedt	Kevésbé elterjedt	Magas szinten elterjedt	Magas szinten elterjedt

3. táblázat: A folyamatmodellezési módszertanok általános összehasonlítása

A tevékenységek mindkét módszertanban megjelennek, lényegi különbség azonban nem tapasztalható közöttük.

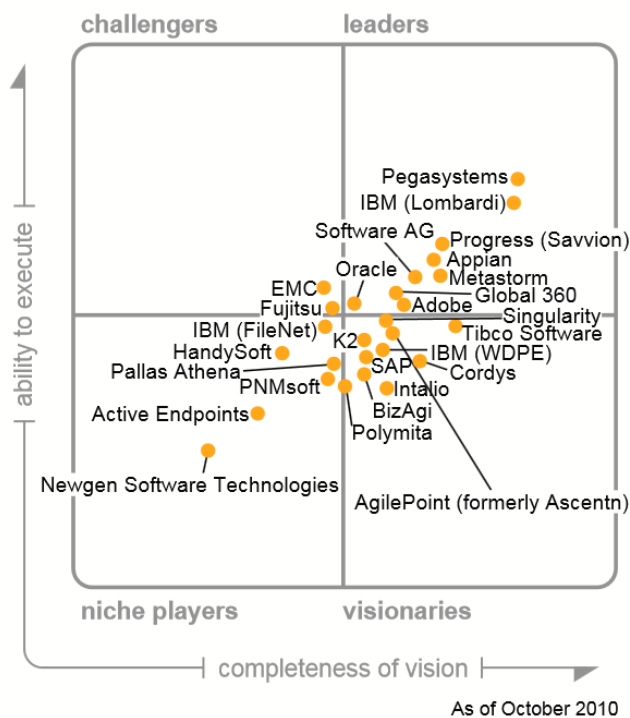
Átjárókból a BPMN-nél öt, az EPC-nél pedig három típust különböztetünk meg. Az EPC-nél az ÉS, VAGY, KIZÁRÓ VAGY lehetőségek közül választhatunk, a párhuzamos ágak pedig egy új eseményben egyesülhetnek. A párhuzamos ágak egyesülésénél BPMN-ben is ugyanez a helyzet. Fontos eltérés az EPC és a BPMN között, hogy míg BPMN-nél az elágazások után a nyilakon található a választási lehetőségek, EPC-nél az elágazások utáni események mutatják meg, hogy mik a döntési lehetőségek. A BPMN-ben megoldható egy *Loop Task* létrehozása, ami egy tevékenység ciklikusságát, megismétlését jelzi.

A BPMN-nél a szerepkörök jelölésére úszósávokat használhatunk. EPC-nél az szerepkörök jelölése a tevékenység jobb oldalán történik, az IT rendszeré pedig a bal oldalán. Ugyanígy a tevékenység jobb oldalán helyezkednek el az inputok és az outputok is.

2.2.3 Folyamatmodellező eszközök

A folyamatmodellek hasznosíthatóságának vizsgálatokor nem lehet figyelmen kívül hagyni a folyamatmodellező eszközök sajátosságait sem. Alapvetően a folyamatmodellező eszköz léte már egy adottságként kezelhető egy vállalat esetében, azonban ha módunk nyílik annak megválasztására, hogy milyen eszközt használjunk, az adhat némi könnyebbséget.

A Gartner egyik állandó felmérése a Magic Quadrant meghatározása bizonyos szektorokra nézve. A legutóbbi folyamatmenedzsment eszközökre vonatkozó ilyen kutatás 2010-ben készült el. Ennek az eredményeit figyelembe véve képet kaphatunk a folyamatmenedzsment eszközök főbb beszállítóiról, és a képességeikről. A 11. ábrán jól látható, hogy a vezetők közé került a Pegasystems, az IBM a Lombardi családjával, az Oracle, az Adobe, vagy a Software AG is.



11. ábra: A Gartner 2010-es „Magic Quadrant” kutatásának eredménye a folyamatmenedzsment eszközökre nézve (Gartner, 2010)

A következőkben áttekintem, mi alapján célszerű meghozni azt a döntést, hogy melyik folyamatmodellező eszközt használjuk, és bemutatok több ismertebb szoftvert is.

2.2.3.1 Folyamatmodellező eszköz kiválasztása

A megfelelő szoftver kiválasztása az egyik legnehezebb feladat. A megfelelő döntés megszületéséhez Coleman szerint több lépés vezet (Coleman, 2007). A kiválasztási folyamat azzal kezdődik, hogy egy szoftverértékelő csoportot hozunk létre, melynek első feladata a szervezet igényeinek felmérése lesz, ezután pedig a szóba jöhető szoftveralternatívákat kell meghatározniuk. Olyan programot kell választaniuk, amelynek formátum használata kompatibilis a vállalat többi rendszerével, ezért fontos, hogy összehasonlítsák a már meglévő információrendszer képességeit, és a bevezetendő modellező eszközét. Ugyanígy a szoftver rendszerkövetelményét is előzetesen ellenőrizni kell. Emellett az sem mindegy, hogy milyen módszertanok használatát támogatja a választott program. Oda kell még figyelni a szoftvertámogatásra, a szoftvertámogatás minőségére is, mivel a rendszer karbantartása rendkívül komplex feladat. Jelentős költséggel járhat az eszköz vállalatra szabása is, bár léteznek olyan lehetőségek, ahol nincs szükség testreszabásra, hanem azonnal használatba vehető a szoftver.

Ezek mind hasznos és megfontolandó témakörök, azonban érdemes figyelembe vennünk Tanrikorur néhány tanácsát is (Tanrikorur, 2007). Ő arra is felhívja a figyelmet, hogy érdemes csak egy cég termékét használni, mert ha több különböző szoftver kombinációját használjuk, sokkal nagyobb az esélye, hogy problémák lehetnek a kompatibilitással. Azt is figyelemmel kell követni, hogy ha más eszközökkel együtt szerezzük be a modellező szoftvert, akkor megfelelően vannak-e integrálva a különböző egységek. Érdemes arra is kitérni, hogy a folyamatok futtatását is elvárjuk-e az eszköztől, vagy megelégszünk a modellek kezelésének lehetőségével. Természetesen az egyik legfontosabb tényező nem más, mint a szoftver ára. Amennyiben pedig kiválasztottuk a megfelelő modellező eszközt, és meg is kezdtük a bevezetését, nem szabad megfeledkeznünk program felhasználóinak tréningjéről, képzéséről sem.

2.2.3.2 BOC Group ADONIS 5.1 és CE 2.0⁵

Az ADONIS egy belépő szintű folyamatmodellező eszköz. A professzionális és az üzleti verzió mellett létezik egy ingyenes verziója is, melynek neve Community Edition. Az ADONIS-on végzett változások először ebbe a verzióba kerülnek be, majd miután eldőlt, hogy hasznos-e az újítás, bekerül a fizetős verzióba is. Így a potenciális felhasználók jelentős mértékben be vannak vonva a fejlesztésekbe, a kialakult közösség segítségével pedig egy minden aspektusában kifinomult szoftver hozható létre (BOC Group, 2006).

Business Process Model (EPC)	Business Process Diagram (BPMN 2.0)
Swimlane	Pool vagy Pool (Collapsed) + Lane
Trigger	Intermediate event (Sequence Flow)
Process Start	Start Event
Sub-Process	Sub-Process
Activity	Task
Decision	Exclusive Gateway
Parallelity	Non-Exclusive Gateway
Merging	Non-Exclusive Gateway (Converging)
End	End Event
Cross-Reference	Cross-Reference
Note	Note
Aggregation	Aggregation
Subsequent	Subsequent
Has Cross-Reference	Has Cross-Reference
Has Note	Has Note
-	Intermediate event (Boundary) Data Object Message Group Text Annotation Relation Node Message Flow Association Data Association

4. táblázat: EPC és BPMN folyamatmodell megfeleltethetőség Adonis-ban

Az ADONIS sajátos jelöléseket használ folyamatmodellezésre, mely leginkább az EPC-re hasonlít, de mégsem feleltethető meg annak teljesen. A folyamatokon belül a tevékenységekhez nem két oldalról kell bekötni a bemeneteket és kimeneteket, illetve a végrehajtókat, hanem azokat külön modelleken kell létrehozni, majd a

⁵ Az Adonis CE a Community Edition rövidítése, míg a 5.1-es verzió a kliens-szerveres változatra utal. A kutatás során mindkettő verziót használtam, ugyanis bizonyos esetekben más funkcionalitással rendelkeznek.

folyamatmodellben csak hivatkozni kell rájuk. Így egy jóval kompaktabb nézet alakul ki.

Emellett immáron 1.0-s és 2.0-s BPMN modelleket is létrehozhatunk vele, támogatja a folyamatok elemzését és a szimulációt is (megadott attribútumok és véletlen szám generátorok segítségével). A BPMN 1.0-ás modellből BPEL, a 2.0-ás modellből XPDLL előállítására képes. Minden modellje kinyerhető XML formátumban.

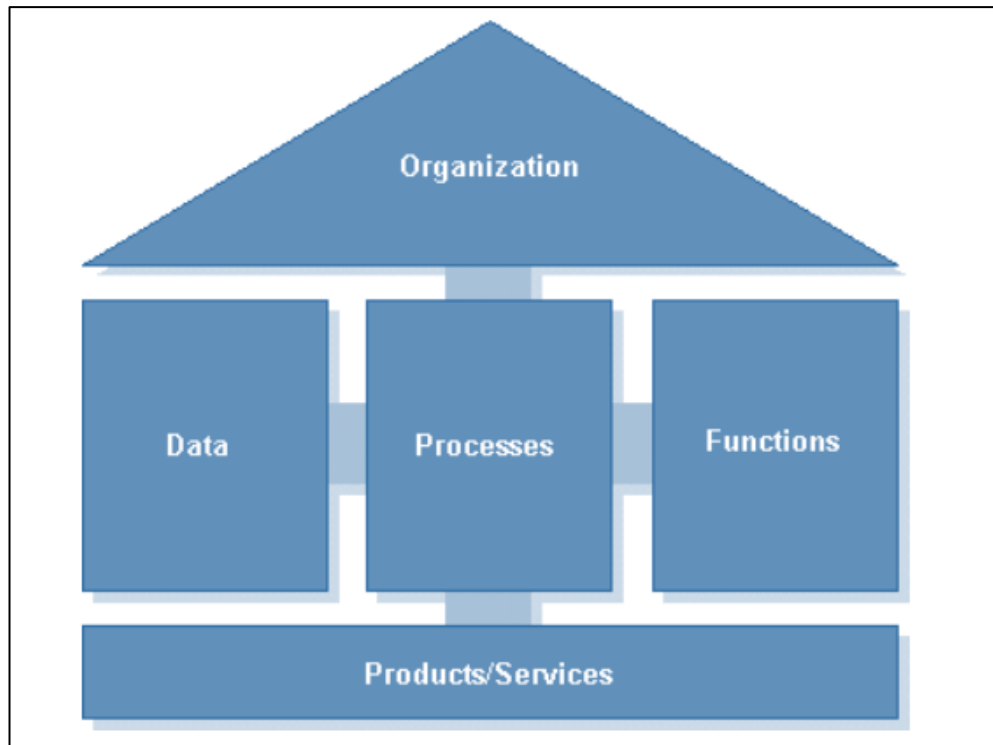
A 2. táblázat az EPC és BPMN összehasonlítását mutatja be az ADONIS modelljeinek objektum-elnevezésein keresztül. Jól látható, hogy a szoftveren belül törekedtek a modellezési módszertanok olyan szintű testre szabására, hogy azok a leginkább megfeleltethetők legyen egymásnak. Emiatt is kerülhetett be az ADONIS-ba az a funkció, amelyik EPC modellből egyetlen kattintással BPMN modell előállítására képes.

2.2.3.3 *Software AG ARIS v 7.2*

A Software AG tulajdonában lévő IDS Scheer folyamatmenedzsment eszköze az ARIS. Az ARIS termékcsalád a folyamatmenedzsment minden területének megfogására törekszik. Ennek érdekében egy négy platformból összeálló rendszer jött létre, melyek a folyamatmenedzsment rendszerhez illeszkednek: stratégia, tervezés, bevezetés és kontrolling.

A folyamatok és a hozzájuk kapcsolódó egyéb tényezők kezelésére jött létre a 12. ábrán látható, úgynevezett ARIS-ház (Scheer, 1997):

- **Szervezeti nézet** - Milyen szervezeti egységek léteznek?
- **Adat nézet** - Mik a fontos információk?
- **Funkció nézet** - Melyik funkciókat hajtják végre?
- **Termék/Szolgáltatás nézet** - Mik a fontos termékek/szolgáltatások?
- **Folyamat nézet** - Az adat, a funkciók és a szervezeti egységek közötti kapcsolatok



12. ábra: ARIS nézetek (Scheer, 1997 nyomán)

Az ARIS lehetőséget nyújt mind az EPC, mind a BPMN 2.0 módszertannal történő modellezésre, bár igazi erőssége az EPC terén mutatkozik meg, már csak azért is, mivel ezt a módszertant ők maguk fejlesztették ki. A szoftver jelentősen elterjedt az üzleti életben, szabványos interfészei vannak pl. a SAP R/3 felé, több módszertan referenciamodelljének leképezése történt meg segítségével (ITILv3, eTOM). Képes XML, BPEL, XPDL exportra.

2.2.3.4 *Intalio bpms Designer*

Az Intalio bpms Designer egy Eclipse környezetben működő eszköz. A BPMN szabvány szerint működik, és annak minden szabályát szigorúan próbálja be is tartatni a modellezést végző felhasználóval. Ez az erős kötöttség egyrészt jól vezeti a modellezőt, másrészt meg is köti a kezét, ha néhány szabályt figyelmen kívül szeretne hagyni. A szigorú ellenőrzés magyarázata az, hogy az eszköz funkciója többek között a szolgáltatások automatikus előállítására is, melyet csak a teljesség és szintaktikai megfelelés esetén tud biztosítani. (Intalio, 2010)

A szoftver érdekessége, hogy az Eseményeknél alapértelmezetten nem különbözteti meg a Catching és Throwing jellemzőt, csak egy egységes Intermediate

kategóriát kezel. A különböző Átjárók lefutási feltételeit külön adattérképező részben kell beállítani, ez is vizuálisan kezelhető. A szoftver a BPMN 1.1 verzióját támogatja.

2.2.3.5 MEGA Suite 2009 SP3

A MEGA Suite egy komplex eszköz, mely nem csak folyamatok kezelésére, feltérképezésére és ábrázolására képes, hanem gyakorlatilag egy vállalatot annak minden aspektusából meg tud jeleníteni, vagy át tud tekinteni. Ennek megfelelően a MEGA cég is főleg az architektúra-tervezéssel hirdeti termékét, amelyik emellett folyamatmenedzsment célokra is megfelelő.

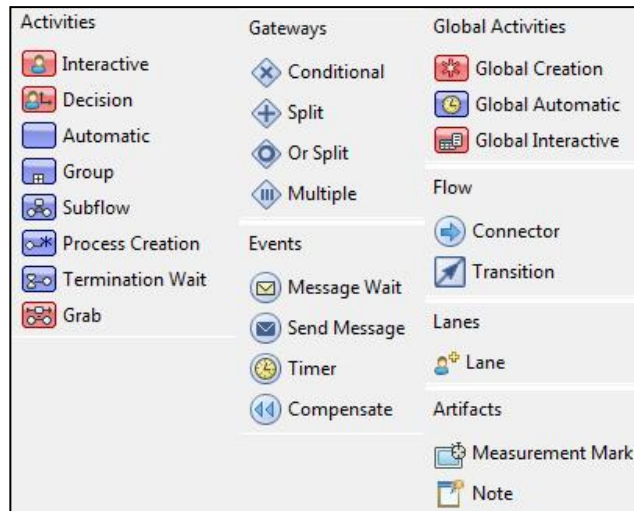
Az eszköz folyamatmodellezésre a BPMN jelölésrendszert használja, azzal az érdekességgel, hogy nem vízszintesen, hanem alapértelmezetten függőlegesen ábrázolja a folyamatokat. A szoftver a BPMN 1.2 verzióját támogatja.

2.2.3.6 Tibco Business Studio 3.2

Az Tibco Business Studio egy Eclipse környezetben működő eszköz. A BPMN szabvány szerint működik, és láthatóan ki is használja annak bővíthetőségét és testreszabhatóságát, ugyanis az összes objektum figyelemfelkeltő színeket kapott. (Tibco, 2009) Engedékenyebb a hibás modellezéssel szemben, mint az Intalio Designer, ezért élvezetesebb benne modellezni. Az elkészült ábrakon animációk is segítik a könnyebb eligazodást, például a bemenő dokumentumokat nem csak tevékenység felé mutató nyíl jelöli, hanem a dokumentum kijelölésekor a nyílon animált vonalak haladnak a tevékenység felé. A szoftver a BPMN 1.2 verzióját támogatja. (Tibco, 2009)

2.2.3.7 Oracle BPM Studio 10.3.1

Az Oracle megoldása egy Eclipse környezetbe helyezett eszköz. A BPMN szabványt veszi alapul, azonban nem felel meg annak. Az Oracle úttörő a SOA megoldásokban. Folyamatmenedzsment eszközében is inkább a szolgáltatások tervezésének támogatását tűzi ki célként.



13. ábra: Az Oracle BPM Studio modellezési objektumai

A 13. ábrán jól látható, hogy miként módosulnak a BPMN megismert objektumai az Oracle BPM Studio-ban. Szembetűnő a Tevékenységek nagy száma, illetve az Események foghíjas volta. Emellett az Átjárók is alternatív neveken szerepelnek, és új kategóriaként megjelentek a Globális Tevékenységek is. Szintén meglepő, hogy az Artifacts közül kikerültek az Adat objektumok, így a szoftver ennek kezelésére sem alkalmas.

A program nem csak Folyamatokat, hanem úgynevezett Screenflow-kat is képes megjeleníteni. Ezek a Folyamatok Úszósáv nélküli változatai.

2.2.3.8 Folyamatmodellező eszközök összehasonlítása

A folyamatmodellező eszközök összehasonlításánál elsődlegesen olyan szempontokat vettem figyelembe, melyek a kutatás végrehajtásához lényegesek. Ennek megfelelően, vizsgáltam, hogy a szoftver milyen módszertan szerinti modellezést támogat, és azt mennyire szabványosan. Mivel számomra a munkakörök elkülönített kezelése fontos, ezért vizsgáltam azt is, hogy milyen formában támogatják az eszközök a végrehajtók, vagy a RACI mátrix kezelését. Emellett az eszköz fő célja erőteljesen meghatározza annak funkcionalitását és használhatóságát, hasonlóan a modellezés testreszabhatóságához, ezért ezt a két szempontot is figyelembe vettem.

Az 5. táblázat mutatja be a folyamatmodellező eszközök összehasonlítását a fenti szempontok szerint.

	ADONIS 4.0 és CE 2.0	ARIS 7.2	Intalio bpms designer	MEGA Suite 2009 SP3	Tibco Business Studio 3.2	Oracle BPM Studio 10.3.1
Fő cél	Folyamat- modellezés	Folyamat- modellezés	Szolgáltató s-tervezés	Architektúr a-tervezés	Folyamat- modellezés	Szolgáltató s-tervezés
Használt módszer- tan	Átalakított EPC BPMN 1.0 és 2.0	karcsú és teljes EPC BPMN	BPMN 1.1	BPMN 1.2	BPMN 1.2	BPMN
Végrehajtók / RACI kezelése	külön modellen, úszósávosa n, RACI- ként	külön objektumon , úszósávosa n, RACI- ként	úszósávosa n	úszósávosa n	úszósávosa n	úszósávosa n
Testre- szab- hatóság	magas	magas	alacsony	alacsony	közepes	közepes

5. táblázat: A folyamatmodellező eszközök összehasonlítása

Az összehasonlításnak nincs „nyertese” vagy „vesztese”, de a munkakörök egyszerűbb és könnyebben kinyerhető kezelése miatt EPC jellegű modellezést követő eszközt, az Adonis folyamatmodellező szoftvert választottam a kutatás előkészítéséhez.

A folyamatmodellező eszközök kiválasztásában azonban még egy szempont nagyon fontos, mégpedig a transzport lehetőségek és formátumok, melyeket a következő fejezet ír le részletesebben.

2.2.4 Folyamatmodellek transzport formátumai

Mivel a kutatás célja folyamatmodellekből a szükséges tudáselemek kinyerése, ezért fontos vizsgálni a folyamatmodellek transzport formátumait. Ezek a megoldások is bizonyos mértékben a modellező eszköztől függenek, de alapvetően jól látható, hogy melyek azok a formátumok, melyek előállítására egy modern folyamatmodellező eszköznek képesnek kell lennie.

2.2.4.1 XML

Az XML (*eXtensible Markup Language*, Kiterjeszhető Jelölő Nyelv) a W3C által létrehozott nyelv, aminek a segítségével lehetővé válik az adatok rendezett tárolása, valamint átvitele is. Az XML strukturáltan tárolja az adatokat és platformfüggetlen, tehát elvileg képes lehet arra, hogy azonos folyamatmodellezési nyelvek esetén két,

különböző gyártótól származó modellező és/vagy értelmező rendszer között megteremtse az átjárhatóságot (W3C, 2008).

Az XML-nek jól formázottnak és érvényesnek kell lennie a megfelelő szintű feldolgozhatósághoz. Az XML dokumentum szintaxisára jellemző, hogy önleíró, valamint rendkívül egyszerű. A jól formázott XML dokumentumnak az alábbi követelményeknek kell megfelelnie (W3Schools, 2013):

- Minden XML dokumentumnak rendelkeznie kell egy gyökérellemmel.
- Minden elemnek rendelkeznie kell záró tag-gel.
- Az XML tag-ek érzékenyek a kis- és nagybetűkre.
- Az elemeket megfelelően kell beágyazni.
- Az attribútumokat idézőjelek közé kell tenni.

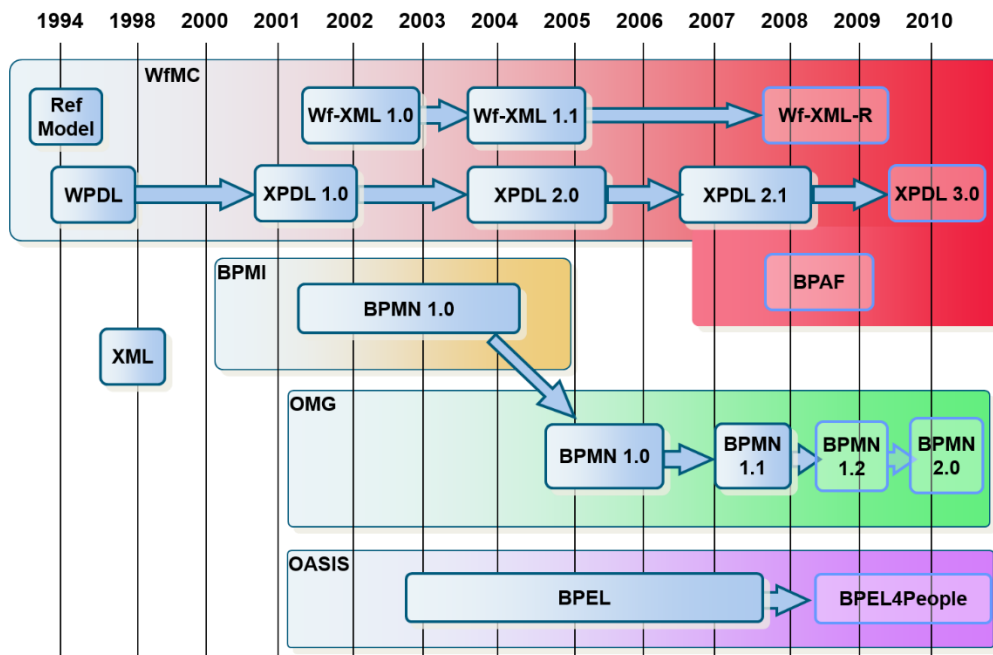
Az XML fontos jellemzője, hogy nincsenek előre definiált tag-jei, azokat a készítő saját maga definiálja. A folyamatmodellező szoftverek jellemzően saját XML sémát használnak, saját tag-ekkel, emiatt nehézkes az elvileg platformfüggetlenül exportált modellek átvitele egy teljesen másik modellező szoftverbe. Két, párhuzamosan fejlődő verziója létezik 1.0 és 1.1 számozással (W3C, 2006).

Az XSLT (*eXtensible Stylesheet Language Transformations*) fájlok segítségével az XML dokumentum transzformálására nyílik lehetőség (W3C, 2007), így egy XML fájlban változtatásokat hajthatunk végre, ezzel létrehozva egy új XML dokumentumot. Ennek a módszernek az előnye az, hogy nem lépünk ki az XML térből, nem veszünk igénybe semmilyen más átalakító réteget, csak az XML és XSLT lehetőségeit használjuk fel.

2.2.4.2 BPEL

A BPEL (*Business Process Execution Language*, illetve teljes nevén *Web Services Business Process Execution Language*) egy XML alapú folyamatleíró nyílt szabvány az OASIS gondozásában. Elsősorban üzleti folyamatok leírására használatos, de egységessége és elterjedtsége miatt sokszor alkalmazzák általános folyamat-integrációs, vagy munkafolyamatok (workflow) leírását igénylő feladatokban is (OASIS, 2007). Időbeli megjelenése miatt (14. ábra) a BPMN 1.0-ás szabványával szorosan

együttműködik, a BPMN 1.0 – 1.2-es nyelvben készült folyamatmodellek a legtöbb esetben gond nélkül átalakíthatóak BPEL formátumúra.



14. ábra: A különböző folyamatmenedzsmenthez kötődő szabványok és formátumok történelmi fejlődése

Maga a nyelv egy egyszerűbb programozási nyelv utasításkészletével rendelkezik, mely kiszolgálja a fent említett célokat: változók kezelése, adatműveletek, külső folyamat- és szolgáltatáshívások. Bár a nyelv végrehajtható, utasítás-készletének hiányosságai miatt önmagában nem alkalmas semmilyen konkrét feladat elvégzésére sem. Minden egyes BPEL utasítás egy külső, kiegészítő nyelven elkészített parancs meghívásával jár. A kiegészítő nyelv leggyakrabban Java, de lehet más programnyelv is.

BPEL-ben az XML központi szerepet játszik. XML maga a program- és modellezési nyelv, amely a folyamatokat írja le, XML-ben jelennek meg a benne lévő változók, melyeket XSLT-vel is transzformálhatunk, XML a kommunikáció nyelve a folyamat és a szolgáltatások vagy más folyamatok között.

2.2.4.3 XPDL

Az XPDL (*XML Process Definition Language*) formátum workflow-k folyamatleírásainak formátumaként jött létre, megalkotója a Workflow Management

Coalition. Az XPDL-ben egy folyamatleíró XML séma található, ez által az XML-nél már ismertetett előnyökkel jár a használata.

Az XPDL előnye az, hogy képes egy BPMN diagram minden attribútumának átvitelére, beleértve a grafikus és szemantikus elemeket is. Ez különbözteti meg erőteljesen a BPEL-től, amelyik csak a folyamatmodell futtatható részének megőrzésére koncentrál (WfMC, 2012). Végző soron az XPDL a BPMN „XML-esített” verziójának tekinthető.

A 14. ábrán látható, hogy az XPDL már harmadik generációjában jár. A BPMN 2.0 megjelenésekor az XPDL már kellően érett volt ahhoz, hogy a BPEL-t leváltsa, és így a BPMN 2.0-ás modellek esetén már azokat XPDL-be célszerű alakítani, nem BPEL-be.

2.2.4.4 Folyamatmodellező szoftverek export formátumai

A legtöbb folyamatmodellező eszköz képes egy saját formátumba is menteni a folyamatmodelleket. Az Adonis esetén ez az ADL, ami gyakorlatilag egy egyszerű szövegfájl, olyan, mintha a folyamatmodell minden egyes attribútumát egyesével kimásolnánk egy fájlba. Ennek megfelelően csak az Adonis képes megfelelően értelmezni, hiszen a szoftveren belüli elnevezéseket használja, ezáltal nem tekinthető szabványos transzport formátumnak.

Mivel az ADL egy egyszerű szövegfájl (mint ez a 15. ábrán is látható), ezért megismerése után könnyen építhetőek rá lekérdezések, illetve könnyedén manipulálható. Az ADL a XPDL-hez hasonlóan az objektumok elhelyezkedését is megőrzi, viszont nem XML alapú, nem használ *tag*-eket, ehelyett gyakorlatilag minden egyes attribútumát leírja egy objektumnak és modellnek.

```
BUSINESS PROCESS MODEL <Process model name> : <ADONIS:CE BPMS BP
Library 2.0>
VERSION <>
TYPE <Business process model>
  ATTRIBUTE <Author>
  VALUE "Admin"
  ATTRIBUTE <Creation date>
  VALUE "27.09.2012, 18:36"
  ATTRIBUTE <Date last changed>
  VALUE "27.09.2012, 19:37:40"
```

15. ábra: Az ADONIS ADL szintaktikája (részlet)

A folyamatmodellező eszközök transzport formátumainak vizsgálata során bemutattam, milyen sokféle mód nyílik a különböző eszközök közötti átvitelre. A

leginkább használható megoldások a szabványos XML alapú megoldások, hiszen ezek jól strukturáltak, részletes dokumentáció érhető el hozzájuk és ennek megfelelően elterjedtek is.

A továbbiakban a kutatási alapvető célját: a folyamatfejlesztést tekintem át szakirodalmi feldolgozáson keresztül.

2.3 Folyamatfejlesztés

Kutatásom célja a folyamatmodellekből kinyerhető kompetenciák, és a feladatok elvégzéséhez szükséges tudásterületek azonosítása után a folyamatok javításának tudásorientált lehetőségeit feltárni. A saját kutatás előkészítéseként áttekintem a folyamatfejlesztés tradicionális megoldásait.

2.3.1 A folyamatfejlesztés tradicionális módjai

A folyamatok fejlesztésére különböző okok miatt lehet szükség, mint például a stratégiai cél megvalósítása, amely további alapot jelenthet a fejlesztésekhez. Fontos szerepe van a vállalati felvásárlásoknál/összeolvadásoknál is, hiszen szükség van a különböző folyamatok egységesítésére, illetve egységes standard létrehozására/bevezetésére is (Horváth et. al., 2006). A folyamatfejlesztés célja „jobb, hatékonyabb, olcsóbb, rugalmasabb stb. folyamatok révén ügyfél-orientált, hatékonyabb és eredményesebb, a stratégiai céloknak jobban megfelelő működés elérése” (Szabó, 2011, 27. dia).

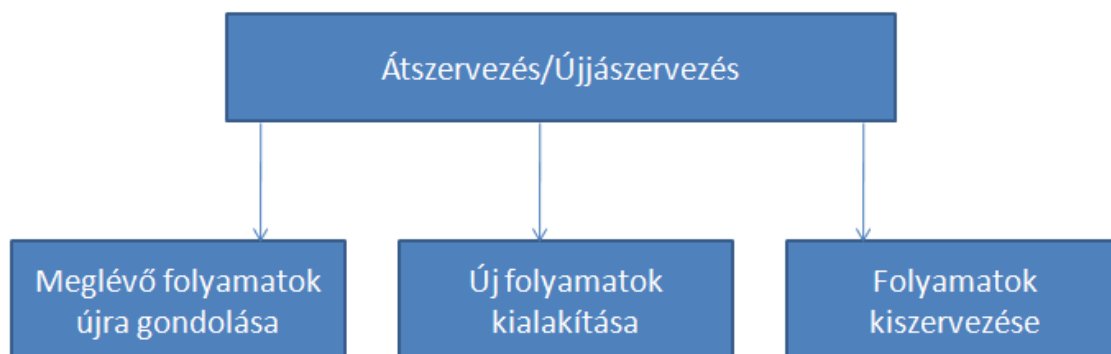
Folyamatok átalakítására két fő módszer létezik, az üzleti folyamatok újjáalakítása/újraszervezése (Business Process Reengineering - BPR) és a Folyamatos folyamatfejlesztés/optimalizálás (Continuous Process Improvement – CPI).

2.3.2 Üzleti folyamatok újjáalakítása/újraszervezése (BPR)

Az újjáalakítás és átszervezés a folyamatok jelentős átalakítását eredményezi, de míg az újjáalakítás valamennyi folyamat gyökeres átalakításával jár, addig az átszervezés inkább csak a szervezeti felépítést és a folyamatstruktúrát érinti.

A folyamatok újjáalakításának módszere a BPR, melynek alapjait Michael Hammer fektette le a kilencvenes években. Ez a folyamatok radikális, alapvető újragondolását és új folyamatok kialakítását jelenti a teljesítmény javítása érdekében (lásd 16. ábra). A

javulás olyan teljesítmény-mutatókban mutatkozik meg, mint idő, költség és minőség. Ez a legelterjedtebb és leggyakrabban használt módszer.



16. ábra: Folyamatok átszervezése (Horváth et. al., 2006, 31. old).

Mivel a BPR esetében az átalakítás top-down módon történik, nem szükséges a jelenlegi folyamatok lefutásának ismerete. Ilyenkor a vezetőség nem kéri ki a folyamatban résztvevők/érintettek véleményét az átszervezéssel kapcsolatban. Sok esetben az átalakítást kiszervezéssel és létszám-leépítéssel próbálják elérni, amiből következik, hogy nem lehetséges az alkalmazottak teljes körű bevonása. Az átszervezésnek az a célja, hogy olyan üzleti folyamatot kapjanak kézhez a folyamatban érintettek, amivel képesek a vevői igényeket teljes mértékben kielégíteni.

Ha a későbbiekben újabb igények merülnek fel, akkor megint újra kell szervezni a folyamatot, úgy, hogy biztosítsák a vállalat tartós versenyelőnyét. A tartós versenyelőny biztosításának eszközei a sikeres növekedés mind a régi, mind az új üzleti területeken, az alkalmazottak kompetenciáinak növelése, a költség és átfutási idő csökkentése, és a feladatok hatékonyabb elvégzése. Bár sok vállalatnak sikerült az újjászervezés révén költséget csökkentenie és versenyképességét javítania, nagyon sokan kudarcot vallottak, kezdeményezéseik megbuktak, elsősorban az alkalmazottak ellenállása miatt (Rok S. et. al., 2008).

Az üzleti folyamat sikeres újjászervezése a következő főbb lépésekből áll:

1. A fogyasztóra összpontosítva azonosítják a folyamat céljait. Ilyen célok lehetnek a hibák megszüntetése, költség és ciklus-idő csökkentése, stb.
2. A meglévő folyamatok feltérképezése és mérése, vagyis annak a megállapítása, hogy a jelenlegi folyamatok milyen költséggel, időátfutással és eredménnyel dolgoznak.
3. Ezek után a meglévő folyamatban lévő tevékenységeket elemezik és módosítják.

4. Összehasonlítják a folyamatokat az iparág legjobb gyakorlatával (Benchmarking), és alternatívákat keresnek vagy új innovatív ötletet használnak fel a folyamatok javításához.
5. Fentiek szerint újra szervezik a folyamatot.
6. Végezetül létrehozzák a megtervezett új folyamatot. Ilyenkor szükség van az alkalmazottak továbbképzésére, a dokumentáció elkészítésére és az eredmény mérésére.

Az átalakításhoz szükséges adatokat a belső működés eredményéből, külső, illetve benchmarkokkal való összehasonlításból szerezhetnek a vállalatok. Ehhez először egy átfogó esettanulmányt kell készíteni, amely kifejti az átalakítás terjedelmét, költségeit, a végrehajtáshoz szükséges időt, és a legnagyobb hiányosságokat is bemutatja. Meg kell határozni a projektet és a költség-megtakarítási célokat, és össze kell hangolni a keretfeltételeket és a projektszervezeteket. Tájékoztatni kell a vezetőket kell az eredményekről. Mindenekelőtt nagyon fontos, hogy kommunikáljuk az átszervezés szükségességét az alkalmazottaknak, hiszen a projekt sikeressége legnagyobb mértékben rajtuk múlik (Hammer, 2001).

A Business Process Reengineering szélsőséges módon a folyamatokat az alapjaitól kezdve építi újjá (Buda, 2003). Ebből kifolyólag a BPR egy egyszeri, de teljes körű tevékenység, amely úgy megy végbe, hogy először egy átfogó kép alakul ki az új folyamatok összességéről – beleértve azoknak a kapcsolódási pontjait is –, majd ebből a felületes képből lefűrva egy egyre részletesebb modell alakul ki.

Mivel a folyamatok – és olykor a szervezeti felépítés is – teljesen új formát ölt, ezért a BPR jelentős kockázati tényezőt rejt magában.

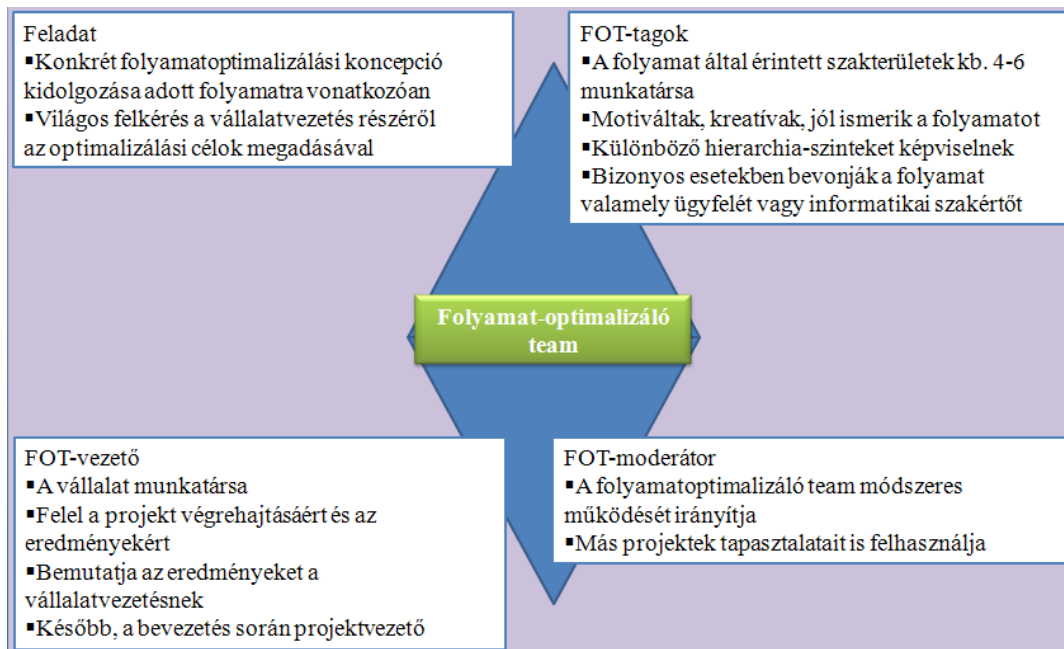
2.3.3 Folyamatos folyamatfejlesztés/optimalizálás (CPI)

Az üzleti folyamatok optimalizálása a meglévő folyamatok keretein belül történik, és – az átszervezéssel ellentétben – célja a gyenge pontok javítása, illetve a folyamat jellemzőinek hozzáigazítása az új követelményekhez, amikkel növelhetjük a hatékonyságot és a teljesítményt (Biren, 1995).

A sikeres folyamat-optimalizálás feltétele a minőség javítását szolgáló célok érthető megfogalmazása és megfelelő kommunikálása. Ennek feltétele egy világos felépítésű projektszervezet. Projektirányító bizottságot kell létrehozni, amely képes a gyors,

pontos és megalapozott döntések meghozatalára, a meghozott intézkedések elbírálására, így segítve a kritikus kérdések megválaszolását (Feng L., et. al., 2001).

Az optimalizálást a folyamat-optimalizáló team végzi (17. ábra), amelynek tagjai jól ismerik a folyamatot, motiváltak és kreatívak.



17. ábra: Folyamatoptimalizáló team (Horváth et. al., 2006, 117. old).

Első lépésként összehasonlítják a folyamatok lefutását adott iparági benchmark adatokkal, amelyek alapján képesek meghatározni a célokat, a fejlesztés irányát és a fejlesztés közben elérendő mérföldköveket is (Kamal M. - Al-S.-Harbi, 2000).

Az üzleti folyamatokat optimalizáló projektek céljait az elemzési szakaszban feltárt lehetőségekből vezetik le. A célok sikeres meghatározása nagymértékben múlik az optimalizálást végző team és a vállalatvezetés közötti kommunikáción, mivel a vállalatvezetés feladata az elérendő célok világos megfogalmazása. A vállalat vezetése dönti el azt is, hogy mely témák élveznek elsőbbséget az optimalizálás során. Ők határozzák meg a költségcsökkentés és a minőségjavítás keretfeltételeit. Figyelembe veszik a folyamatban résztvevők leterheltségét, a folyamatok mennyiségét és struktúráját. Mérlegelik a projekt terjedelmét, azaz hogy hány témával érdemes foglalkozni (Horváth et. al., 2006).

Ugyanakkor a kitűzött célokat az optimalizáló csapat hajtja végre. Az optimalizálásnak olyan általános elveket kell szem előtt tartani, mint a redundancia csökkentése, párhuzamosítás, delegálás, az adminisztráció csökkentése és az

automatizáltság növelése. Lényeges az előrehozott ellenőrzés, de még jobb az ellenőrzés helyett a megelőzés. Csökkenteni kell a folyamat-hurkokat (Kóczé, 2010).

A BPR-ral szemben a Continuous Process Improvement egy folyamatos folyamatfejlesztést jelent, amely alapul veszi a már meglévő folyamatokat, illetve az azokba foglalt tevékenységeket. Ez azt jelenti, hogy a CPI alulról építkezik, és sokszor minden előzetes tervezés nélkül jön létre a modell átfogó képe. A bottom-up megközelítés legfőbb előnye pedig esetünkben az, hogy több érintett bevonását is támogatja (IFUA, 2008). Ez azért fontos, mert így azoknak az előzetes tapasztalatait is be lehet vonni a tervezés folyamatába, akik a folyamat közvetlen felhasználói, így sok olyan problémát, nehézséget megláthatnak, amelyeket a szervezet vezetői, a menedzsment tagja nem feltétlenül vennének észre, vagy alulbecsülnék azok súlyosságát. A folyamat felhasználásával közvetlen kapcsolatban levő érintetteket azért is célszerű bevonni a tervezésbe, mert amennyiben az ő tapasztalataik is felhasználásra – de legalábbis meghallgatásra – kerülnek, akkor a folyamatok megváltozásának elfogadása részükről kisebb ellenállásba fog ütközni (Bakacsi, 2010).

A CPI-hez szorosan köthető a Raffai fogalomgyűjteményében szereplő folyamatos fejlődési modell, amely „olyan tevékenységsor, amellyel a jelenlegi folyamatok feltárását, elemzését követően, a célok elérését biztosító lépéseket megvalósítják, értékelik a teljesítményeket, az eredményeket, és szükség szerint ismét beavatkoznak a folyamatokba” (Raffai, 1999, p. 104).

A CPI a létező folyamatoknak – vagy azok egy bizonyos részének – hatékonyságát kiegészítésekkel, illetve módosításokkal igyekszik növelni, és sokkal kisebb kockázatot jelent, hiszen a CPI-hez köthető egy-egy módosítás általában gond nélkül visszaállítható eredeti állapotba. Ez már csak azért is így van, mert a Continuous Process Improvement a „meglévő szervezeti kereteken belül marad” (IFUA, 2008, p. 20), tehát figyelembe veszi a szervezet jellegzetességeit.

2.3.4 A BPR és CPI összehasonlítása

A BPR jellemzője, hogy egyszeri innovatív átalakulást jelent, vagyis nincs visszatérési állapot, ugyanakkor a CPI-nál állandó folyamatos inkrementális fejlesztést hajtanak végre. Míg a BPR a tevékenységek és folyamatok újradefiniálására helyezi a hangsúlyt, addig a CPI a meglévő folyamatokra koncentrál. A BPR alapelve a teljes

folyamat vizsgálata és fejlesztése, ezzel szemben a CPI a meglévő szervezeti felépítés átalakítására helyezi a hangsúlyt. Míg a BPR egy instabil átalakulási folyamat, addig a CPI viszonylag stabil, és kontrollált változtatással hajtható végre. A BPR felülről vezérelt top-down folyamat-átalakítás, ezzel szemben a CPI egy alulról kezdeményezett bottom-up folyamat (Hammer, 2001).

Az, hogy melyik módszert választja egy szervezet, nagymértékben függ az időtől, a versenytársaktól, a költségtől (mennyit fordít rá a vállalat, mennyit akar megspórolni) és a környezettől. Az idő fontos lehet például a vállalati felvásárlásoknál, amikor egy vállalatot azért vesznek meg, hogy megszerezzék, és minél előbb kihasználják a benne lévő alapképességeket. A költségre és környezetre jó példa a mai gazdasági környezet, hiszen sok vállalat került szorult helyzetbe. A kitörés érdekében fel kellett mérniük belső folyamataikat és azonosítaniuk kellett alapvető kompetenciáikat is. Így ahol lehetett, optimalizáltak, átszerveztek, és kiszervezték azokat a tevékenységeket, amik nem tartoznak alap-tevékenységeik körébe (például az informatikát cloud technológiával biztosítják, vagy akár outsourcing révén oldják meg számos cégnél).

	BPR	CPI
Fő cél	Tevékenységek és folyamatok újradefiniálása	Meglévő folyamatokra, tevékenységekre koncentráll
Jellege	Innovatív, nem visszatérő változtatási folyamat	Inkrementális, állandó folyamatfejlesztés
Teljesség	Alapelv a teljes folyamat vizsgálata, fejlesztése	Lehetséges egyes alfolyamatok vizsgálata, fejlesztése
Szervezeti struktúra	A folyamatokon alapuló szervezeti struktúra bevezetése	A meglévő szervezeti felépítés alakítása
Hangsúlyos elem	Hangsúly a folyamat hatékonyságon és az informatikai támogatás hatékonyságán	Minden szervezeti cél és hatékonysági kritérium együttes vizsgálata
Stabilitás	Instabil átalakulás	Viszonylagos stabilitás, kontrollált változtatással
Iránya	Top-down folyamat	Bottom-up folyamat

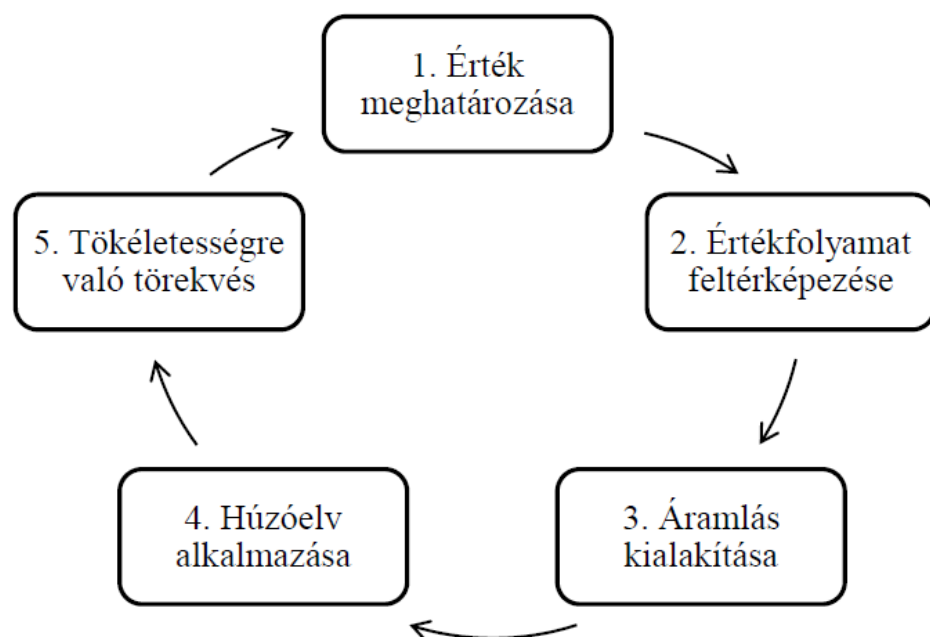
6. táblázat: A BPR és a CPI összehasonlítása

2.3.5 Lean menedzsment

A Lean menedzsment a Toyota termelési rendszerén alapul (TPS). A második világháború után a vállalat még gyerekcipőben járt, és a háború megviselte Japánt. A beszállítói bázis tönkrement, félni kellett a gyár bezártatásától, szűkös erőforrást bocsátott a vállalkozás rendelkezésére a cégalapító - mivel egy új iparágról volt szó, és a

tömegtermelés sem jöhetett szóba, mivel nem volt fizetőképes kereslet. Emiatt rövid átfutási időre volt szükség. Ez hívta életre a következő alapokat: kiváló minőség, gyors átfutási idő, rugalmasság, alacsony költség. (Liker, 2008). Az 1980-as évekre a termelési rendszer már az Amerikai Egyesült Államokban is virágkorát élte, és hot-topiknak számított a tevékenységmenedzsment szakembereinek körében. (Demeter et. al., 2011).

A lean menedzsment öt alapelve helyezi a termelési folyamatot, mellyel célja a hatékonyság maximalizálása és a veszteség minimalizálása. Első lépésként lehetőleg minél pontosabban definiáljuk, határozzuk meg az ügyfél igényeit, és olyan terméket alkossunk, melyek valóban értéket jelentenek a felhasználóknak. Utána térképezzük fel azokat a folyamatokat, melyek szükségesek a termék létrehozásához és a felhasználókhöz való eljuttatáshoz, és szűrjük ki azokat a fölösleges lépéseket, melyek nem teremtenek értéket. Majd ezeket a folyamatokat helyezük egymás mellé, és törekedjünk arra, hogy ezeken a folyamatokon keresztül minél előbb a felhasználóhoz jusson el a termék. Az áramlás kialakítása után optimalizáljuk a húzóelvet, azt határozzuk meg, hogy az egyes folyamatlépésekről miként léphetünk a következő fázisba. Végül folyamatosan mérjük a folyamatainkat és optimalizáljuk annak érdekében, hogy az értékáramlás minél gyorsabban menjen végbe, valamint ne termeljünk veszteséget (18. ábra). Erre az öt lépésre alapul a lean menedzsment.



18. ábra: A Lean öt fő alapelve

A lean menedzsment a hagyományos tömegtermelésből származó hátrányokra ad megoldást. A tömegtermelés során nagy mennyiségben, alacsony áron készültek a termékek, ehhez viszont óriási raktárkészletre volt szükség, hogy a folyamatos gyártáshoz biztosítsák a kellő nyersanyagokat. A termelést gyártóegységekre bontották, melyek egyes alkatrészek legyártását, a termékek összeszerelését vagy tesztelését végezték. Egy termék legyártása meglehetősen sok időt vett igénybe, emellett ha valamely alkatrésztől kiderült, hogy hibás, az csak a tesztelés során derült ki. Így nem csak a késztermékek vesznek kárba, hanem a folyamatban lévő termékek is.

Liker a Toyota-módszer könyvében a veszteségeknek 7+1 fő típusát gyűjtötte össze: Túltermelés, Várakozás, Felesleges szállítás, Túlzott vagy nem megfelelő feldolgozás, Túl sok készlet, Felesleges mozgás, Selejt, A munkatársak kihasználatlan kreativitása. Ezek a veszteségek első sorban az autógyártásra vonatkoznak, de átvitt értelemben a szoftverfejlesztési iparágra is igazak. (Liker, 2008)

A Lean használatának sok előnye van: növeli a produktivitást, profit és cash flow orientált, és nagyfokú ügyfélközpontúság jellemzi. Emellett nagy hangsúlyt fektet a minőségre és a hatékonyságra is.

Sajnos, ennek a módszertannak is vannak nem elhanyagolható hátrányai: sok az üresjárat, nagy az átfutási ideje, és költséges módszer.

2.3.6 Six Sigma

A Six Sigma jól meghatározott szabályokon, adatokon alapuló megközelítés a folyamatok javítására annak érdekében, hogy a folyamat csökkentse az ügyfél igényeknek való nem-megfelelését, pontosabban a folyamat szórását és a hibák számát. Célja a Six Sigma-nak a „gyökér” okok megszüntetése, ezáltal a minőség javítása, illetve a változékonyságból eredő hibák számának csökkentése és az értéknövelő lépések minőségének javítása. Ebből következik, hogy a Six Sigma NEM alternatívája a Lean -nek, hanem egymás kiegészítői, együttes alkalmazásukkal lehet jó eredményt elérni.

Ahogy Geoffrey Tennant megfogalmazza: a Six Sigma egy vízió, filozófia, szimbólum, metrika, cél, és metodológia is lehet egyben, de nem gyógyír minden betegségre, mérlegelni kell, hogy az adott problémára, az adott helyzetben hatékonyan alkalmazható e, vagy sem, és ha igen, akkor sem garancia a sikerre. A módszertan

továbbá nem egy egyszerű eszköz, még ha elsőre annak is tűnhet, valamint nem csak a gyártási jellegű iparban alkalmazható. (Tennant, 2001).

Véleményem szerint az egyszerűség látszata abból fakad, hogy amikor a Motorola megalkotta, még csak egy mérőrendszer volt, mely arra szolgált, hogy a selejtes termékek aránya alapján behatárolja a termelés folyamatának minőségét, az idő előrehaladtával azonban rengeteg egyéb eszköz, módszertan, tanulmány kapcsolódott hozzá, ezzel továbbfejlesztve és alkalmassá téve a gyárak határainak átlépést. Manapság már lényegében minden területen, legyen az termelő, vagy szolgáltató láthatunk példát a Six Sigma használatára, ahol folyamatok javításán keresztül célozzák a nagyobb hatékonyság elérését.

Bolya Árpád hasonlóan vélekedik a Six Sigmában rejlő lehetőségekről. Ő a következőket érti a fogalom alatt:

- Minőségi irányzat, melynek célja a legjobbnak lenni
- Filozófia és módszer, mivel egy rendszerezett, mindent átfogó megközelítés
- Mérőeszköz, mivel statisztikai módszerek segítségével alkalmas a folyamatok minőségének állapotfelmérésére
- Üzleti stratégia, mivel segít összehangolni a vállalat folyamatait a stratégiai céllal, egyúttal pedig a konkurenciával szemben versenyelőnyt biztosít. (Bolya, 2011)

Tóth Csaba László pedig ezek mellett még menedzsment rendszernek is tekinti, mely átfogja a vállalat teljes keresztmetszetét. (Tóth, 2007)

A Six Sigma tehát egy vállalati kultúrába beépíthető, annak minden területére hatással levő eszköz, mely statisztikai módszereken alapszik, és projektmenedzsment módszere segítségével konkrét iránymutatást ad a folyamatok javítására, a vevői elégedettség és versenyelőny növelésére.

A módszertan a folyamat minőségét szigmával jelölt szórás mértékegységében méri. A szigma értéke a hiba valószínűségének bekövetkezésével áll fordított arányban, azaz minél magasabb a szigma értéke, annál jobb a folyamat. Ha a folyamat hat szigmás, vagyis eléri a célt, amely a majdnem tökéletesség, akkor kevesebb, mint 3,4 hibás kimenet (termék) jut 1 millió kimenetre (Parts Per Million, PPM = 3,4) (Bolya, 2011).

2.3.7 A Lean menedzsment és a Six sigma összehasonlítása

A 7. táblázat összefoglalja a Lean menedzsment és a Six sigma fontosabb szempontok szerinti összehasonlítását. Ebből jól látható, hogy a Lean menedzsment célja kevésbé általános, míg a Six sigma alapja (a statisztika) megalapozottabbnak hat. A használhatóság terén bár a Lean menedzsmentet főleg a gyártás területére ajánlják, valójában mára mindkettő bármilyen folyamatok esetén használható. A Lean projektek gyorsabbak, a projektcsapatok kevésbé állandóak, míg a Six sigma projektek több időt vesznek igénybe, és a résztvevők dedikált erőforrásként kerülnek a projektre beosztásra.

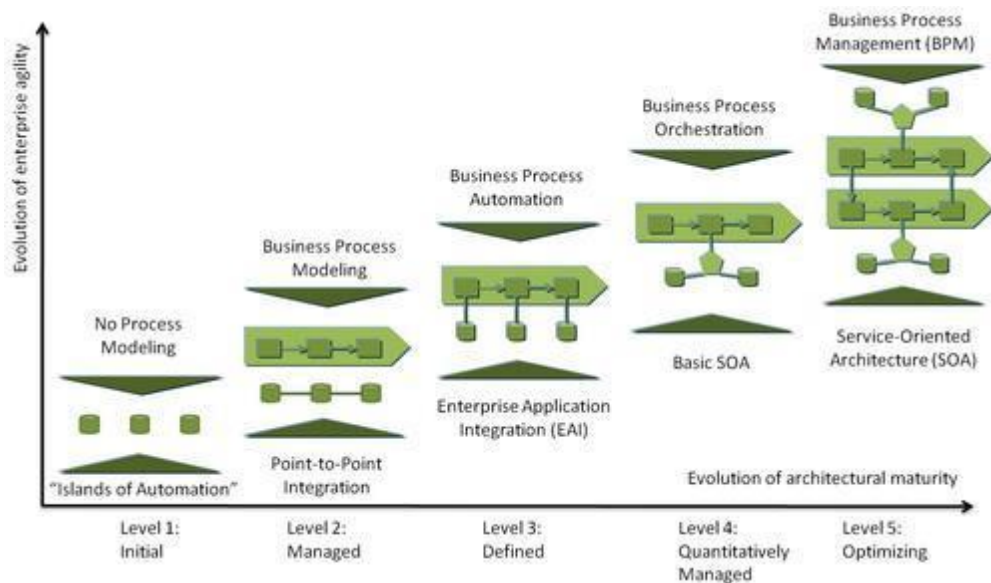
	Lean menedzsment	Six sigma
Cél	Folyamatos áramlás megteremtése, veszteségek csökkentése	Folyamatok javítása, szórás csökkentése
Célterület	Gyártás	Minden üzleti folyamat
Megközelítés	Elvek oktatás, best practice bevezetése	Statisztikai alapú problémamegoldó szemlélet
Fő eszköz	Értékáram-térkép alapján	Többféle megközelítéssel
Projektek időtartama	1 hét – 3 hónap	2 – 6 hónap
Projektcsapat	Szűk körben, változó csapattal	Dedikált erőforrással
Képzés	Learning by doing	Learning by doing

7. táblázat: A Lean menedzsment és a Six Sigma összehasonlítása (Koczor, 2009)

Véleményem szerint a két módszer együttesen nagyon jól használható, mert míg a Six Sigma lehetővé teszi a mélyebb problémák okainak feltárását, addig a Lean segítségével ki lehet javítani a hibákat. Ezek segítségével a folyamatok eredményesebbek és hatékonyabbak lesznek korábbi állapotukhoz képest.

2.3.8 Folyamat-érettségi modell

A folyamatelvű szemléletre való átállás a szervezet érdekében nehézkes és rendkívül energiaigényes folyamat. Ennek köszönhetően nem csak kétféle rendszert tudunk elkülöníteni, úgy mint, hagyományos és folyamatelvű szemlélet. A Business Process Management Maturity 5 eltérő CMMI szintet különböztet meg, a szervezetek folyamatokhoz való hozzáállása alapján.



19. ábra: CMMI szintek (Ebizq, 2011)

A CMMI szintek:

- **INITIAL** (ellátatlan vagy éretlen): Ezen a szinten a szervezet nem ismeri fel a folyamait, nincsenek pontosan definiálva, se dokumentálva. Az alapvető munkafeladatokat ellátják, viszont kimenetelük előre nem megjósolható, ezáltal a folyamat közben, mérésekre alkalmatlanok. Erre a szintre az informálisan végzett feladatok jellemzőek.
- **REPEATABLE:** Jellemzően a dokumentált jelzöt használhatjuk, hiszen az ezen a szinten álló vállalatok már ismerik a főbb folyamataikat, amelyeket újra és újra végrehajtva, meg tudják becsülni, hogy az adott tevékenységek milyen eredménnyel fognak zárulni. A tevékenységek olyan folyamatok mentén haladnak, amelyeknek már a felelősségi köreik is tisztázva vannak, így mindenhol megjelennek az alapszinten kezdetlegesen működő folyamatgazdák. Bár még a legtöbb folyamat nincs megfelelő kontroll és ellenőrizhetőség alatt.
- **DEFINED:** látható, hogy valamennyi alapfolyamat már dokumentálásra került, ezáltal némelyik már teljesen, de mindegyik legalább részlegesen kontrollálható. Alapvető folyamatmenedzsment eszközök is már rendelkezésre állnak, adottak a mérőszámok, amelyek segítségével és használatával a folyamatok eredményei mérhetőek és ellenőrizhetőek.
- **MANAGED:** már közel áll a folyamatelvű szemlélet teljes megvalósításához, bár még hiányosságokkal rendelkezik azzal szemben. Itt nevéből adódóan a

folyamatok menedzselése már előtérbe kerül, jóval nagyobb hangsúlyt kap, mint az előzőekben említett szinteken. Megjelenik a hierarchia a folyamatok között, olyan alfolyamatok létrehozása válik elterjedtté, amelyek különböző főfolyamatok céljait szolgálják ki, azok pedig közvetlenül segítik a szervezet céljainak elérését. Az adatgyűjtés és eredmények dokumentálása is végleges szintre ért, minden folyamat mérhető és ellenőrizhető. Ezáltal a kontroll elérte a maximális szintet, a célok teljesülésének mérése megvalósul.

- **OPTIMIZING:** Folyamatosan javuló szint. Az előzőleg említett Managed szinttől rendkívül kevéssé különbözik, itt már megjelenik a folyamatok folyamatos fejlesztése és javítása a jobb eredmények elérése érdekében. Elmondható, hogy a folyamatelvű munkavégzés a szervezet mindennapi kultúrájának alapjává válik.

Alapvetően ez az öt érettségi szint különböztethető meg a folyamatmenedzsment szempontjából. A nagyvállalatok napjainkra végigjárták ezeket a lépcsőket, ennek köszönhetően igazodni tudnak az új piaci igényekhez és rugalmasan tudják kezelni a változó környezet által megjelenő kihívásokat.

2.4 Kapcsolódó tudásmenedzsment fogalmak ismertetése

Kutatásom célja a folyamatmodellekből kinyerhető kompetenciák, és a feladatok elvégzéséhez szükséges tudásterületek azonosítása. Emiatt indokolt a dolgozatban áttekintenem néhány olyan fogalmat, melyek a tudásmenedzsment területére vezetnek. Ezek a fogalmak az ontológia és a kompetencia.

2.4.1 Ontológia

A mai napig igen sok meghatározás született arra, hogy mi is az ontológia, de még nem született olyan, amelyet a szakértők széles körben elfogadnának és alkalmaznának. Ráadásul ezek a definíciók időben is sokat változtak.

A szó görög eredetű – a “létező”+”tan” összetétellel keletkezett, filozófiai irányzatként került a köztudatba. A különböző ontológia definíciók különböző nézőpontokra világítanak rá. Egyesek az ontológia-építésből, mások magából a filozófiai fogalomból indulnak ki. Az azonban biztos, hogy az ontológiák a szemantikus web alappilléreinek tekinthetőek.

A szakirodalomban egy gyakran hivatkozott ontológia meghatározás Gruber nevéhez fűződik: "Az ontológia a fogalmi modell (fogalomalkotás) világos és részletes leírása." (Gruber 1993, pp. 199.), ahol a fogalmi modell, illetve a fogalomalkotás szélesebb értelemben véve egy fajta világnézet; egy adott szakterület gondolkodásmódját tükrözi.

Guarino a szakterületi ontológiára a következő meghatározást adja (Guarino, 1998): „A valóság egyfajta értelmezésének egy adott szóhasználattal történő leírása, explicit feltevések a valóság vélt értelmezéséről”. Szintén az értelmezésből indultak ki Wielinga és szerzőtársai: "Az ontológia egy elmélet arra nézve, hogy milyen entitások létezhetnek egy értelmes személy tudatában" (Wielinga et al. 1997).

Az ontológia a tudásreprezentáció egyik lehetséges módja. Tudás reprezentálására alkalmas és elterjedt a logikai állítások használata, ami könnyen algoritmizálható, vagy a teaurusz, amely a fogalmak közötti kapcsolatot írja le. Emellett egyéb megoldások is léteznek, ám az ontológia specialitása abban van, hogy nem csak a fogalmak közötti hierarchikus kapcsolatot lehet megjeleníteni, hanem a fogalmak közötti relációkat is.

Az ontológiák egyfajta kiterjesztett elméletek egy adott területtel kapcsolatos fogalmak fajtáiról, tulajdonságairól és a közöttük lévő kapcsolatokról. Igaz állításokat, kifejezéseket biztosítanak az adott területtel kapcsolatos tudásunk megragadására. Céljuk egy adott tárgyterülethez kapcsolódó tudás megragadása és az adott területnek egy széles körben elfogadott értelmezés nyújtása, ami alkalmazások és csoportok között újrahasznosítható és megosztható (Davies et al., 2003).

Az információrendszerek esetén az ontológiák használatával elsődleges célunk az, hogy egy szakterület, feladat, alkalmazás formális leírását adjuk meg. Az ontológiai megközelítés éppen ezért vált népszerűvé a tudásalapú rendszerek fejlesztésében (Kő, 2004).

Egy ontológia több különböző formában jelenhet meg, de mindenképpen tartalmaznia kell a tárgyterület szakkifejezéseit, terminológiáját és a jelentésük leírását (szemantika). Az ontológia gyakorlatilag mindig egy szakterület közös értelmezésének megjelenése, amely elősegíti a különböző érdekelt felek közötti kommunikációt. Egy ilyen közös alap hozzájárul a pontos és eredményes információcseréhez, amely

lehetőséget nyújt az újra felhasználhatóságra, a közös használatra és a közös üzemeltetésre.

Az ontológia fontos szerepet tölt be a tudás reprezentációjában is, így alapvető fontosságú a szervezeti tudás menedzsmentjében, a tudásalapú rendszerek kialakításában. Mivel kutatásomban a folyamatmodellekből szeretném kinyerni a folyamatok végrehajtásához szükséges tudáselemeket, logikus út a folyamatokból folyamatontológiát, a tudáselemekből pedig szakterületi ontológiát létrehozni. Az ehhez szükséges átalakítások feltérképezéséhez azonban meg kell vizsgálni az ontológia-leíró nyelveket.

2.4.2 Ontológia-leíró nyelvek

A weben található információk gépi feldolgozására, és strukturált tárolására jött létre a Resource Description Framework (RDF) (W3C, 2004a), melynek kiterjesztése a Web Ontology Language (OWL). Ennek két főbb verziója, a 2004-es OWL (W3C, 2004b, Guinness - Harmelen, 2004), és a 2009 óta folyamatosan fejlődő OWL 2 jelent meg (W3C, 2012).

Az OWL mindkét verzióján három-három különböző változata létezik, OWL esetén a Lite, DL és Full, melyek a modellezés mélységét adják meg. A Lite-től a Full felé haladva egyre kifejezőbbé válik a nyelv. Az OWL Full tekinthető az RDF kiterjesztésének. Az OWL 2-nél EL, QL és RL verziók léteznek. Az EL azon felhasználókat célozza, akik nagyon sok tulajdonságot vagy osztályt akarnak használni, a QL azokat, akiknél a lekérdezés (*querying*) kritikus fontosságú, míg az RL azokat, akiknél a következtetések (*reasoning*) sebessége fontos.

Az RDF és az OWL közötti különbség a következők szerint érzékeltethető:

- Az RDF objektumok és kapcsolatainak adatmodellje. Az adatmodell szemantikus leírása történik meg az RDF-ben, technikailag XML szintaxissal.
- Az RDF Schema az a nyelv, melynek segítségével az RDF objektumok tulajdonságai és osztályai leírhatóak általánosítások használatával, illetve hierarchikus rendszerben.

- Az OWL pedig további nyelvi elemeket vezet be a tulajdonságok és osztályok leírásához, mint az osztályok közötti kapcsolatok, számasságok kezelése, egyenlőség és hasonlóság kifejezése, bővebb tulajdonságkészlet, halmazműveletek bevezetése.

Összefoglalva, az OWL erőteljesebben képes a jelentés valós kifejezésére, mint az XML vagy az RDF, és ez által az OWL túlmutat a másik két formátumon a gépi értelmezésre való alkalmasságban (Jenz, 2003).

2.4.3 Ontológia-építés, *ontology learning*

Az ontológiák előállítása történhet manuálisan, tehát egy olyan szakértő által, aki ismeri az ontológiák felépítését és létrehozásának mikéntjét, azonban ez a megoldás nagyon időigényes és ennél fogva drága, hibák elkövetésére ad lehetőségeket, és erőteljesen a készítő világlátására támaszkodik (Gomez-Perez - Manzano-Macho, 2003; Shamsfard - Barforoush, 2003; Sabou et al., 2005).

A fentiek miatt az elmúlt évtizedben megjelent az úgynevezett *ontology learning* terület, melynek fő célkitűzése a legalább félig automatizált ontológiaépítési módszerek előállítása. A terület vizsgálatának tárgya az, hogy hogyan lehet a szöveges tartalmakat jól strukturált ontológiaelemekké transzformálni félig, vagy teljesen automatizált módszerekkel (Wong et al., 2012).

Az első *ontology learning* módszertanokat áttekintő elemzést az OntoWeb Consortium publikálta 2003-ban (Gomez-Perez - Manzano-Macho, 2003). A szerzők 36 ontológiaépítési megoldást vizsgáltak, és a következőkre jutottak: nincs megfelelő metodológia még az ontológiák előállítására szabad szövegekből, szakértők bevonására még majdnem minden módszer esetén szükséges, illetve nincs megfelelő módszertan a különböző megoldások hatékonyságának összehasonlítására sem.

Az utolsó pontban azonosított hiányosságot Shamsfard és Barforoush 2003-ban orvosolta, ugyanis egy keretrendszert állítottak fel az *ontology learning* módszerek értékelésére (Shamsfard - Barforoush, 2003), melyben olyan jellemzőket vizsgáltak, mint az ontológiába épített elemek jellege (fogalmak, relációk, osztályok, stb.), a tanulás jellege (felügyelt, nem felügyelt, statisztikai vagy nyelvtani alapon működő),

automatizáltság foka, vagy az elkészülő ontológia jellemzői. Kutatásukban a terület akkori főbb kihívásait is azonosították.

További kutatások (Buitelaar et al., 2005 és Zhou, 2007) azt hangoztatták, hogy a területen sokkal nagyobb hangsúlyt kell fektetni az axiómák kinyerésére, és megállapították, hogy az ontológiák előállítása során a humán szereplő sosem lesz elhagyható.

A területen a kiindulást tekintve megkülönböztetik a nem strukturált és a félig strukturált szövegek feldolgozását. A strukturált szövegek feldolgozása nem okoz nehézséget, hiszen gyakorlatilag ilyen szöveg előállítása a cél, ezért ez nem vizsgálat tárgya.

Az XML dokumentumok feldolgozása, és ontológia formába való alakítása az *ontology learning* egyik részterülete. Az ilyen szövegeket szövegbányászati módszerekkel (statisztikai alapú szövegfeldolgozás vagy szabályalapú feldolgozás), vagy NLP (természetes nyelvfeldolgozás – *Natural Language Processing*) módszerekkel lehet átalakítani.

Az XML dokumentumok ontológia formátumokba való átalakítását volt, amikor még teljesen lehetetlennek tartották (Decker et al., 2000), azonban ezt az állítást kutatások cáfolták. Már 1999-ben Melnik vizsgálta az XML RDF formába való képzésének lehetőségeit (Melnik, 1999), és Battle 2004-ben be is mutatott egy megoldást erre. Ő egy általános XML Schema létét feltételezte, melynek segítségével bármilyen XML dokumentum átalakítható (Battle, 2004). Bohring és Auer ezzel szemben úgy véli, hogy minden XML-hez egy külön XML Schema kialakítása szükséges a megfelelő ontológia leképezéshez (Bohring – Auer, 2005).

Az XML dokumentumok átalakítása során a legtöbb esetben valamilyen XSL/XSLT jellegű feldolgozó nyelv használata terjedt el, hiszen ez esetben nem kell kilépni az XML térből, és egyszerűen kivitelezhető. Az XSL stíluslapot megfelelően kialakítva az XML dokumentumot *tag*-enként feldolgozó átalakítások eredményeképp az elvárt ontológia struktúra jön létre. Ehhez természetesen ismerni kell az elvárt ontológia struktúrát, mely RDF és OWL esetén különböző lehet, de függ a kiinduló adatok jellegétől is az XML dokumentumban.

2.4.4 Folyamatontológia

A folyamatontológiának nincs egyértelmű meghatározása az irodalomban. Herborn és Wimmer egyszerűen egy folyamat fogalmi leíró keretrendszerének nevezik azt (Herborn – Wimmer, 2006). Az ő értelmezésük szerint a folyamatontológia absztrakt és általános. Az előzővel ellentétben az úgynevezett feladatontológiák egy feladatsorrendet állapítanak meg (Benjamins et al., 1996; Tate, 1998; Pease, 1998).

A mi értelmezésünkben a folyamatontológia a folyamatmodellekből képezhető, strukturált leírás, ami gépi eszközökkel feldolgozható módon tartalmazza a folyamatok végrehajtásához szükséges tevékenységeket, döntési pontokat és logikai kapcsolókat, azok sorrendjét és a szükséges erőforrásokat is. A mi értelmezésünkben a folyamatontológiának ugyan részei a folyamat végrehajtásához szükséges tudáselemek, de azokat strukturáltan egy szakterületi ontológiában tároljuk.

2.4.5 Kompetencia

A kompetencia fogalmának számos szakirodalmi értelmezése létezik. A legtöbb definíció valamilyen olyan tulajdonsággal azonosítja, ami egy személyre jellemző és megkülönböztető erejű, valamilyen tevékenység magas szintű elvégzését teszi lehetővé. Woodruffe szerint a kompetencia „viselkedésminták egy készlete, melyet a munkakör betöltőjének be kell vetnie ahhoz, hogy a munkaköri feladatokat és funkciókat megfelelően lássa el” (Woodruffe, 1993, pp. 29.). Klemp és McClelland a „kompetens” helyett már „hatékony” ellátást említ: „Az egyén olyan tulajdonsága, mely nélkülözhetetlen egy munkakörben vagy szerepben nyújtott hatékony teljesítményhez” (Klemp – McClelland, 1986, pp. 32.). Boyatzis pedig már a kiválóság feltételeként említi a kompetenciát: „az egyén hatékony és/vagy kiváló munkaköri teljesítményt eredményező személyiségjellemzője” (Boyatzis, 1982, pp. 21.). Ehhez képest Quinn definíciója semlegesebb: „egy bizonyos feladat vagy szerep teljesítéséhez szükséges tudás és képesség” (Tóthné Kiss, 2012).

A kompetencia a mi értelmezésünkben egy meta-fogalom és általánosságban ír le egy tudásterületet. Ilyen értelemben a Quinn-i definícióhoz áll közelebb a mi értelmezésünk. A kompetenciát szokás árnyaltabban tudáselemekre, készségekre, képességekre (*skill*) és attitűdre bontani. Mi a „*skill*”-t kéz- (vagy egyéb testrészes) ügyességre értjük. Vannak olyan értelmezések, amelyek a tudáselemeket is a „*skill*”-hez

sorolják, ennek annyiban lehet jogossága, hogy a kézügyesség és az internalizálódott tudás átfedheti egymást, függően attól, hogy a tudáskonverzió mely szakaszában van. Ha a „*skill*”-eket pedig megfeleltetjük a tacit tudásnak, akkor az átalakítható explicit tudássá is (externalizáció).

2.5 Szemantikus folyamatmenedzsment

A szemantikus folyamatmenedzsment azzal a céllal jött létre, hogy kiküszöbölje a folyamatmenedzsment nehézségeit, és ötvözze annak elveit a szemantikus technológiákkal, ami elsősorban ontológia-alapú fejlesztést jelent. Hepp szerzőtársaival, illetve Koschmider és Oberweis azt a kihívást azonosította a hagyományos folyamatmenedzsmentben, hogy az csupán az üzleti szakértők számára állít elő modelleket, és a technikai nézőpont csupán gyengén jelenik meg benne. Emiatt az elkészült folyamatmodellek nem dolgozhatóak fel gépi eszközökkel, így a folyamatmodellekből a működő szolgáltatások, csak további átalakítások után hozhatóak létre (Hepp et al., 2005, Koschmider – Oberweis, 2008).

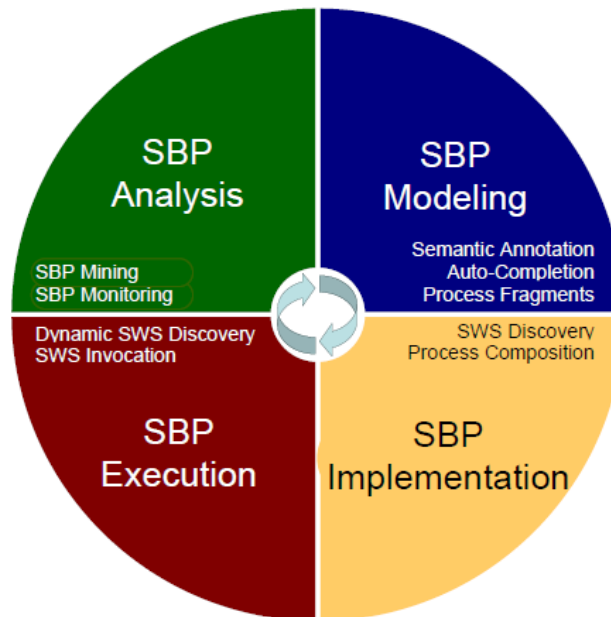
A szemantikus folyamatmenedzsment általános elképzelése tehát Hepp és társainak köszönhető. Az új terület fő célja az üzlet és IT területek közötti szakadék csökkentése a szemantikus technológiák alkalmazásával (mint ontológiák, következtető mechanizmusok, és szemantikus webszolgáltatások). Hepp és társai cikkükben nem mutattak be konkrét alkalmazásokat, csak egy elméleti keretrendszert.

A szemantikus folyamatmenedzsment területén több nagy projekt született már. A folyamatmodellek és az IT megoldások közötti egyértelmű és gyors megfeleltetést célozza a SUPER⁶ projekt. Ennek keretében kidolgozásra kerültek a szemantikus webszolgáltatások és a szemantikus folyamatmenedzsment alapjai (Belecheanu et al., 2007, Filipowska et al., 2007). Kifejlesztésre került a WSMO (*Web Service Modelling Ontology*, Fensel et al., 2006) és a szemantikus BPEL is.

⁶ SUPER: Semantics Utilised for Process Management within and between Enterprises – Integrated European Project

2.5.1 A szemantikus folyamatmenedzsment életciklusa és főbb területei

A szemantikus folyamatmenedzsment életciklusát a következő négy állomáson keresztül írják le Wetzstein és szerzőtársai (Wetzstein et al., 2007): folyamatmodellezés, folyamat implementáció, folyamat végrehajtás, folyamatelemzés. A szemantikus technológiák használatának segítségével az automatizálhatósága foka és a folyamatmenedzsment rendszer funkcióinak száma is növekszik (Oro – Ruffolo, 2012).



20. ábra: A szemantikus folyamatmenedzsment életciklusa (Wetzstein et al., 2007)

A 20. ábrán a különböző szemantikus folyamatmenedzsment rendszer elemekhez megoldásokat is kapcsoltak a szerzők. Ennek mentén számba veszem a terület különböző megoldásait, eszközeit.

2.5.2 A szemantikus folyamatmodellezés eszközei és megoldásai

A folyamatmodellezés során a folyamatokban rejlő tudás tacit formából explicit formába öntése történik meg. Ennél a fázisnál a szemantikus folyamatmenedzsment előnye nem csupán a dokumentálásban rejlik, hanem a modell olyan formában való elkészítésében, ami már gépi eszközökkel feldolgozható lesz. Ennek eredménye a legtöbb esetben a szemantikus annotált folyamatmodell (Lautenbacher et al., 2008). Az EPC-nek és a BPMN-nek is elkészült a szemantikus annotációra felkészített változata (Thomas - Fellmann, 2006, Filipowska et al., 2009, Abramowicz et al., 2007),

illetve ezek mellett a BPEL leképezését is célként tűzte ki Hepp és Roman, egy felsőszintű folyamatontológia használata mellett (Hepp – Roman, 2007).

A szemantikus annotálás célja a folyamatmodell minden jellemzőjének explicit megragadása és leképezése. Ezek körébe tartozik a tevékenységek és döntések funkcionalitása, az érintett szereplőkkel és egyéb erőforrásokkal együtt. A modellezés során az ontológiák támogató szerepe jellemző, ami olyan plusz funkciók elérését teszi lehetővé, mint az automatikus folyamatmodell-kiegészítés (Brockmans et al. 2006, Betz et al. 2006, Lautenbacher – Bauer, 2006), a folyamatmodell összevetése standard modellekkel, vagy épp a különböző folyamatdarabkák újrahasznosítása. Dimitrov (Dimitrov et al., 2007) kifejlesztett egy szemantikus folyamatmodellezési eszközt a WSMO Studio-ra alapozva, amelyben szemantikus BPMN-ben lehet a folyamatokat modellezni.

A szemantikus folyamatmodellezés lehetőségeinek egyik legperspektivikusabb alkalmazása a vállalati folyamatmodellek különböző törvényi, best practice, vagy egyéb módon elvárt vagy ajánlott folyamatmodellekkel való összevetése (*compliance check*). Több forrás a külső kontrollokkal való összevetést tekinti a terület alapjának (Namiri - Stojanovic, 2007, Sadiq et al., 2007). Találhatóak ontológiák használata nélküli, meta-modellekre építő ellenőrző megoldások (Karagiannis, 2008), illetve ontológia-alapúak is a szakirodalomban (Schmidt et al., 2007). Ly pedig szemantikus megkötések használatát javasolja, és így biztosítja egy rendszer megfelelőségét az annak alapját képező folyamat változása esetén is (Ly et al., 2007, 2008, 2009).

El Kharbili és munkatársai meghatározták a folyamatok összevetéséhez szükséges keretrendszer nyolc fő tulajdonságát, és ismertetik is a különböző ellenőrzési megoldásokat, miközben a folyamatontológiák használatát hangsúlyozzák (El Kharbili et al., 2008, El Kharbili – Pulvermüller, 2009).

2.5.3 A szemantikus folyamatimplementáció eszközei és megoldásai

Az előző fázisban modellezett folyamatok még azonnal nem futtathatóak, hiszen bizonyos technikai információkat nem tartalmaznak. A szemantikus folyamatmenedzsmentnek azonban az is célja, hogy a folyamatmodellekből sokkal gyorsabban, automatizálhatóan legyenek futásképes programok. Valamiféle átalakításra szükség van, de a cél az, hogy ez az átalakítás legyen automatizálható, és ennek volt

első lépése az annotálás, ami a szükséges „építőkövek” felfedezését teszi majd lehetővé. A program összeállításának előbb említett építőkövei a szemantikus webszolgáltatások (SWS – Semantic Web Services), ezek megtalálását segíti az annotálás az SWS könyvtárakban (Weber, 2007).

A működő program webszolgáltatásokból áll össze, és szemantikus BPEL-re fordítható, ami már futtatható (Nitzsche et al., 2007).

2.5.4 A szemantikus folyamat-végrehajtás eszközei és megoldásai

A korábbiakban modellezett és futásra előkészített folyamat a végrehajtás fázisában kerül ténylegesen futtatásra. A folyamatmodell tehát már maga is szemantikus webszolgáltatássá válik, és kiajánlásra kerülhet a felhasználóknak. Innentől kezdve már ez a folyamat, illetve szolgáltatás is újrahasznosíthatóvá és beépíthetővé válik más szolgáltatásokba.

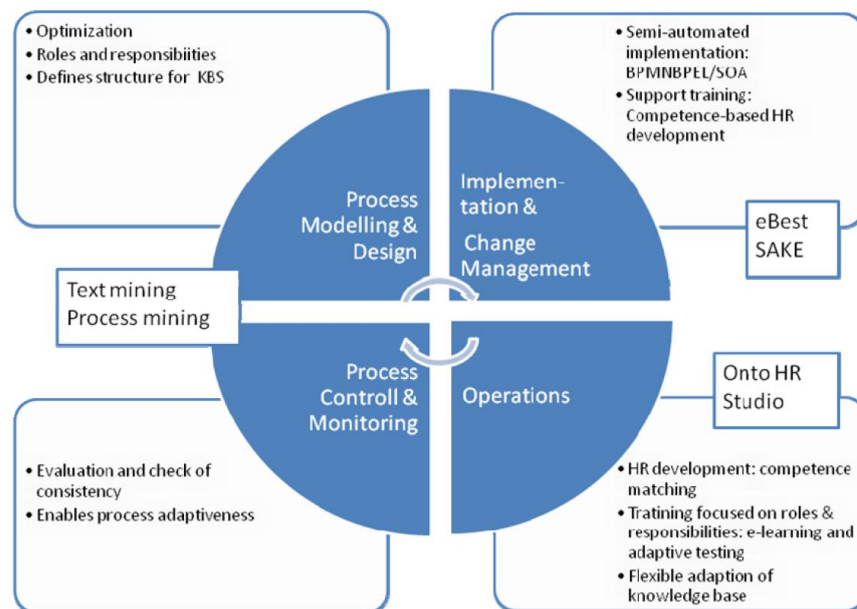
A szemantikus folyamatmenedzsment a hagyományos folyamatmenedzsmenthez képest dinamikusabb, ami többek között abban is megvalósul, hogy a folyamatmodell alapján webszolgáltatásokból összeállított program az őt alkotó webszolgáltatások elérhetetlensége, vagy nem megfelelő szolgáltatási szintje esetén dinamikusan új szolgáltatást tud keresni, és annak segítségével kiváltja a helyettesítendő részt (Wetzstein et al., 2007).

2.5.5 A szemantikus folyamatelemzés eszközei és megoldásai

A folyamatelemzési fázisban a működő szolgáltatások megfigyelése és elemzése történik meg. Az elemzés során működési adatok és fejlesztési lehetőségek nyerhetőek ki a már implementált szolgáltatásokból, és így az alapjukat képező folyamatokból is. A folyamatelemzés egyik fő területe a szemantikus folyamatbányászat, ami támaszkodik a szemantikus eszközök használatában rejlő lekérdezési és következtetési lehetőségekre (Alves de Medeiros - van der Aalst, 2007, Alves de Medeiros et al., 2007, Celino et al., 2007).

2.5.6 A kutatás és előzményeinek elhelyezése a szemantikus folyamatmenedzsment életciklusában

A 2.5.1 fejezetben ismertetett életciklus szemlélet mentén azonosított a szemantikus folyamatmenedzsment területén elérhető példákat Gábor és Szabó (Gábor – Szabó, 2013). Ők is négyes felosztás szerint kategorizálták a szemantikus folyamatmenedzsment területén elérhető kutatás-fejlesztési projekteket, azonban a rendszerük részei kissé eltérnek a fenti kategorizálástól. A modellezési és implementációs rész itt is megtalálható, majd a végrehajtás helyett itt üzemeltetés szerepel, végül az elemzés helyett a tágabban értelmezhető irányítás és megfigyelés szerepel. A felosztást a 21. ábra mutatja be.



21. ábra: A szemantikus folyamatmenedzsment területén megvalósult kutatás-fejlesztési projektek (Gábor - Szabó, 2013)

Az 1.6 alfejezetben azonosított kutatási előzmények ennek megfelelően egyértelműen köthetőek a szemantikus folyamatmenedzsment életciklusának szakaszaihoz. Az eBest⁷ projekt automatizáltan, folyamatmodellből előálló workflow

⁷ eBest: "Empowering Business Ecosystems of Small Service Enterprises to face" (FP7 Capacities/SME 243554)

megoldása és a SAKE⁸ rendszer az implementációhoz, az oktatáshoz kötődő STUDIO⁹ rendszer és OntoHR¹⁰ projekt pedig a működtetéshez köthető.

Jelen kutatás a folyamatmodellekből kinyerhető tudáselemeken alapuló folyamatfejlesztést vizsgálja, így a folyamatelemzési és modellezési fázisokat is érinti, ugyanis a folyamatmodellek elemzésén, bányászatán keresztül azonosítja a szükséges tudáselemeket.

⁸ SAKE: "Semantic enabled Agile Knowledge-based E-government" (FP6 IST 027128)

⁹ STUDIO: <http://corvinno.com/web.nsf/do?open&lang=en&page=proj-studio>

¹⁰ OntoHR: "Ontology based competency matching" (504151 - LLP -1 - 2009-1-HU-LEONARDO-LMP)

3 Gyakorlati megoldás felvázolása

Kutatásomban a folyamatmenedzsment és a tudásmenedzsment területeit kapcsolom össze, célom ugyanis a folyamatmodellekből indulva feltárni a munkakörök betöltéséhez szükséges tudáselemeket, illetve ezek kinyerésének módszereit, majd ezekből ontológia építésével a tudástranszfer megvalósításához konkrét megoldást adni.

Az 1.3-as alfejezetben azonosítottam azokat a kutatási kérdéseket, melyekre a gyakorlati részben választ keresek. Ebben a fejezetben részletesen bemutatom a gyakorlati feladatok megoldásának menetét, az elért eredményeket, és azok jelentőségét.

3.1 A gyakorlati megoldás áttekintése, a megoldás során felhasznált projektek és esetek ismertetése

A kutatás empirikus részében a szervezeti tudásvagyon folyamatmodellekből való kinyerését vizsgálom a hasznosítási lehetőségekkel.

A létrehozandó rendszer a vállalatok folyamatait leíró folyamatmodelleket veszi alapul. Ezekből a folyamatmodellekből kerülnek kinyerésre mindazon tudáselemek, melyek bizonyos munkakörökhöz szükségesek. A tudáselemek kinyerése a folyamatmodellekből nem triviális feladat, hiszen a folyamatmodellekben a tudáselemek tárolására dedikált objektum vagy attribútum nem található.

A folyamatmodelleket először át kell alakítani egy jól strukturált, legalább félig-automatizálhatóan feldolgozható dokumentummá, amire jó eszköz lehet a folyamatontológia. A folyamatontológia egyesíti magában a folyamatmodellek adattartalmát az ontológia strukturáltságával és jó feldolgozhatóságával.

A folyamatokból kinyert tudáselemek felhasználhatóak például a konkrét munkakör betöltéséhez szükséges tudás meglétének ellenőrzéséhez, például munkaerő-felvétel során. Ehhez a tudáselemeket egy szakterületi ontológiába kell leképezni, majd ehhez az ontológiához már illeszthető egy adaptív tesztrendszer (a korábbiakban már ismertetésre került STUDIO rendszer).

A kutatás során felvázolt megoldás működését három valós eset feldolgozásával igazolom.

3.1.1 Felsőoktatási intézmény folyamatainak összevetése a felsőoktatás működésének referencia folyamataival

Ebben az alfejezetben a „Hallgatói kérvénykezelés” folyamatán keresztül mutatom be a felvázolt megoldás lehetőségeit. A cél a jelenlegi folyamat összevetése a normatív folyamattal. Ennek során a két folyamatmodellt folyamatontológiává alakítjuk, majd az ontológia-összehasonlítás módszerével összevetjük azokat.

3.1.1.1 Az eset előzményének ismertetése

A magyar felsőoktatásnak a munkaerőpiac elvárásainak megfelelő kimeneti kompetenciákkal kell kibocsátani a hallgatókat. Ennek megfelelően a felsőoktatási reform az oktatást a kereslet-kínálat elveinek megfelelően kell, hogy átalakítsa (SZKTerv, 2012).

Ezzel együtt a költségvetési elvonások nem teszik lehetővé, hogy a felsőoktatási intézmények a munkaerőpiac elvárásainak megfelelően alakítsák át képzési kínálatukat, hiszen a költségkímélő megoldások keresése és megvalósítása magasabb priritást élvez, mint a munkaerő-piaci igényekre való összpontosítás. A Budapesti Corvinus Egyetem is végigvitt több olyan projektet, amely a költségcsökkentési lehetőségeket kereste, ám a direkt költségcsökkentő intézkedések mellett a hatékonyabb folyamatok megléte is segítheti a hatékonyabb erőforrás-felhasználást.

2004-ben, az akkori Oktatási Minisztérium támogatásával megalkotásra került a felsőoktatás felső szintű folyamatrendszer. 2005-2006-ban két konzorcium fejlesztette tovább ezt a magas szintű modellt. A HEFOP-3.3.1-P-2004.09.0134/1.0 (Structural and Organizational Development of Higher Education¹¹) projekt célja egy teljes körű normatív folyamat-, szervezet- és információs architektúra modell létrehozása volt. Erre alapozva kialakíthatóak a megfelelően ellenőrzött és kézben tartott operatív folyamatok, ami segíti az átláthatóságot és elszámoltathatóságot. A létrejövő modellek a támogató IT rendszerek specifikációjának alapjaként szolgálhat. A teljes projekt tehát a magas minőségű működést és oktatást célozta.

A projekt során két konzorcium mérte fel a felsőoktatásra jellemző folyamatokat, megalkotta a folyamatmodelleket, elvégezte a folyamatfejlesztést, majd erre alapozva meghatározták a referenciafolyamatokat. Tizenkét egyetem kollégái létrehoztak egy

¹¹ HEFOP normatív folyamatmodellek: <http://informatika.bke.hu/hefop>

megfelelően részletezett folyamatmodellt, így próbálva megteremteni az egységes és átlátható felsőoktatás alapjait (Gábor-Szabó, 2009).

A konzervatív akadémiai világban a szervezeti struktúra erőteljesen hierarchizált és a tudástranszfer emiatt erőteljesen töredezett. Ezzel szemben a környezet elvárása pont ellentétes: erős igény mutatkozik az együttműködésre, csapatmunkára, tanulásra és tudásátadásra képes hallgatókra és szakemberekre. Az elmúlt években az európai országok erős fejlődésen és változáson mentek keresztül, melynek során az alkalmazkodóképesség és az információs társadalomnak való megfelelés alapvető elvárássá váltak. A legújabb üzleti módszerek és megoldások ismeretét követeli meg a vállalati szféra a felsőoktatásból kibocsájtott hallgatóktól, és így az őket oktató oktatóktól is.

A HEFOP projekt referenciafolyamatait mind a mai napig használhatjuk az aktuális folyamatok minősítése során. A megfelelőségi vizsgálat során a projektben létrehozott folyamatok lesznek a referenciafolyamatok, melyekkel az aktuális folyamatok összevethetőek.

3.1.2 Poszt-operatív kórházi ellenőrzési folyamat összevetése a nemzetközi standardokkal

3.1.2.1 A Med-Assess projekt ismertetése

A Med-Assess¹² projekt egy innovációs projekt a LEONARDO DA VINCI „innovation transfer” felhívásában, az EU élethosszig tartó tanulás programjában. A korábbi OntoHR projekt alapjai kerülnek átültetésre az egészségügy szakterületére, többnyelvű környezetben.

A Med-Assess által adott megoldás segíti a nővéreket abban, hogy felmérjék a feladataik ellátásához szükséges tudás- és kompetenciaszintjüket, és javaslatokat kapjanak arra nézve, hogy miben kell fejlődniük vagy magasabb szintre hozni a tudásukat. Emellett a Med-Assess a feletteseket is támogatja a toborzásban, és az új munkaerő képességének felmérésében. A projekt terméke a szakterület tudáseleminek oktatásához szükséges oktatóanyag is, hiszen ennek segítségével a felmért tudás önállóan fejleszthetővé is válik. A hasznosítás egyik életszerű területe a különböző országokból érkező munkavállalók tudásszintjének összemérése, a birtokolt képesítések

¹² Med-Assess: Adaptive Medical Profession Assessor, Leonardo da Vinci project

egyenrangúságának biztosításához. Ezáltal a munkavállalók bizonyítványaiktól függetlenül értékelhetőek, a valós tudásszintjük alapján, mégis a helyi elvárásoknak megfelelő tudás kerül ellenőrzésre.

A projekt során felmértük egy gerinc betegségekre specializálódott magánklinika nővéreinek folyamatait, majd ezt modellező eszközben ábrázoltuk.

3.1.2.2 A ProKEx projekt áttekintése

A ProKEx¹³ projekt célja, hogy létrehozzon egy olyan integrált platformot, amely a vállalati tudásmenedzsmentet a szemantikus folyamatmenedzsment eszközeivel támogatja. A projekt olyan megoldást hoz létre, melynek segítségével kinyerhető, rendszerezhető és átadható a szervezeti folyamatmodellekben tárolt tudás, ezáltal a szervezeti tudásvagyon ellenőrzött módon gyarapítható. A munkavállalók könnyebben azonosíthatják a feladataik elvégzéséhez szükséges tudáselemeket és fejleszthetik tudásukat.

3.2 Folyamatmodellező nyelvek, módszertanok vizsgálata

Az irodalmi áttekintésben bemutattam a főbb folyamatmodellezési módszertanokat, és a folyamatmodellező eszközöket. A gyakorlati részben azonosítani fogom, melyek azok az elvárások és konvenciók, melyeket be kell tartani ahhoz, hogy az elkészülő folyamatmodellekből kinyerhetőek legyenek majd a szükséges tudáselemek, kompetenciák.

A folyamatmodellezés során a modellezési módszertan már kötöttségeket jelent, azonban a módszertanok bizonyos mértékben testre szabhatóak. Ezt a folyamatmodellező eszköz fejlesztője is sok esetben megteszi, ezért elképzelhető, hogy az Adonis, Tibco, Aris, IBM Websphere, vagy egyéb modellező szoftverek által készített BPMN modell máshogy néz ki, sőt, akár más-más objektumokat is tartalmaz. Mint már korábban is írtam, a BPMN ebből a szempontból kötöttebb, mint az EPC, melynek használata esetén például egy Adonis-ban készült modell szinte össze sem hasonlítható egy Aris-ban készülttel.

A fentiek következménye az, hogy kutatásomban nem elegendő csupán a modellezési módszertanokat vizsgálnom, be kell vonnom a vizsgálatba a főbb

¹³ ProKEx: Integrated Platform for Process-based Knowledge Extraction, EUREKA project

folyamatmodellező eszközöket is, hiszen a folyamatmodellek hasznosításához teljes körű információkkal kell rendelkezni a rendelkezésre álló modellekről. Ennek megfelelően elő fognak állni különböző modellezési konvenciók, melyek:

1. a folyamatmodellezési módszertanból következő konvenciók
2. a folyamatmodellező eszközökből adódó konvenciók
3. a kutatás céljából levezethető elvárások

Dolgozatomban mindhárom szempontot együttesen kezelem, és ennek alapján fogom kiválasztani, hogy melyik modellezés módszertanban, és milyen eszközzel fogom elkészíteni a kiindulást jelentő folyamatmodelleket.

3.2.1 A folyamatmodellek szükséges és elégséges adattartalma

Egy folyamatmodell összeállításakor nagyon sok paraméter megadására van lehetőségünk. A folyamat lefutását jelző tevékenységek, döntési pontok vagy logikai kapuk, események elhelyezésével gyorsan kialakítható a folyamatnak a váza, ám ez még csak a kezdet.

Egy folyamatmodell vertikális részletezettsége megadja, hogy mennyire részletesen modellezzük azt: működési területeket adunk meg, vagy folyamatterületeket, folyamatokat, alfolyamatokat, tevékenységeket, vagy pedig algoritmizált műveleteket.

Egy folyamatmodell horizontális tagoltsága azt adja meg, hogy mennyi plusz információt rögzítünk a modellen, mint például szervezeti vonatkozás (RACI mátrix feltöltése), erőforrások hozzárendelése (input, output, termékek), IT támogatás, mérőszámok és kockázatok.

Azt, hogy ezek közül mit használunk ki, illetve mindek a megadását követeljük meg a folyamatmodelleket készítő munkatársaktól, minden esetben a felhasználás függvénye. Teljesnek nevezhető egy folyamatmodell akkor, ha részletezettsége megfelelő a felhasználásához. Pontosán emiatt indul minden folyamatmodellezési feladatot magába foglaló projekt az elvárások összeállításával és a konvenciók összeállításával. A létrejövő konvenciók kézikönyvnek mindenki számára ismertnek és elfogadottnak kell lennie. Ez teszi lehetővé azt, hogy azonos részletezettséggel modellezzék a szervezet minden tagja, és egy folyamatmodell garantáltan ugyanazt jelentse minden értelmező számára.

A vizsgált projektben folyamatmodellek kerülnek elemzésre és összevetésre szabályzatok, törvények, policy-k leírásával. A cél tehát az összevethetőség, a matching lehetőségének megteremtése. Ehhez – Adonis modellező szoftver esetén – a következő paraméterek megadása szükséges:

- A folyamat logikai lefutását jellemző modell a szokásos, csak folyamatmodellekre jellemző objektumokkal, betartva a modellező szoftver által elvárt logikát
- A folyamat végrehajtásához szükséges szervezeti struktúra ábrázolása Working environment model-en, betartva a modellező szoftver által elvárt logikát
- A folyamat során bemenetként vagy kimenetként felmerülő elemek Document model-en ábrázolva, betartva a modellező szoftver által elvárt logikát
- A folyamat végrehajtásakor szerepet kapó IT elemek, IT model-en ábrázolva, betartva a modellező szoftver által elvárt logikát
- A tevékenységek elnevezése
- A tevékenységek description mezőjének kitöltése
- A tevékenységek RACI mátrixának feltöltése, amelyen belül a Responsible for execution és az Accountable for making decisions kötelező, míg az In cooperation with és a To inform megadása csak akkor elvárt, amennyiben értelmezhető
- A tevékenységek RACI mátrixában csak Role-ok szerepelhetnek, Organizational unit még ott sem adható meg, ahol a szoftver erre lehetőséget adna
- A tevékenységeknél a szükséges input, output és IT információk megadása

Felmerül a kérdés, hogy miként érhető el, hogy a folyamatmodellező munkatársak ezeket a konvenciókat kövessék. A folyamatmodellezés, és a folyamatmodellezéssel járó projektek ugyanis a különböző szakterületeken dolgozó munkatársak számára mindenképp kihívást jelentenek, amit ki-ki másként él meg. A megoldás több oldalról közelíthető meg.

Első pontként a megállapított konvenciókat a munkatársakkal meg kell ismertetni, oktatást kell nekik tartani, gyakoroltatni kell velük a modellezést. Ezt követően a

konvenciókat kötelezővé kell tenni. Ezután várható csak el, hogy azokat betartsák, és ezután ellenőrizhető maga a betartás ténye is.

Ha a konvenciók betartása előírás, akkor ennek automatikus kikényszerítése is megoldható, de ez nyilvánvalóan a használt folyamatmodellező eszköz továbbfejlesztésével jár.

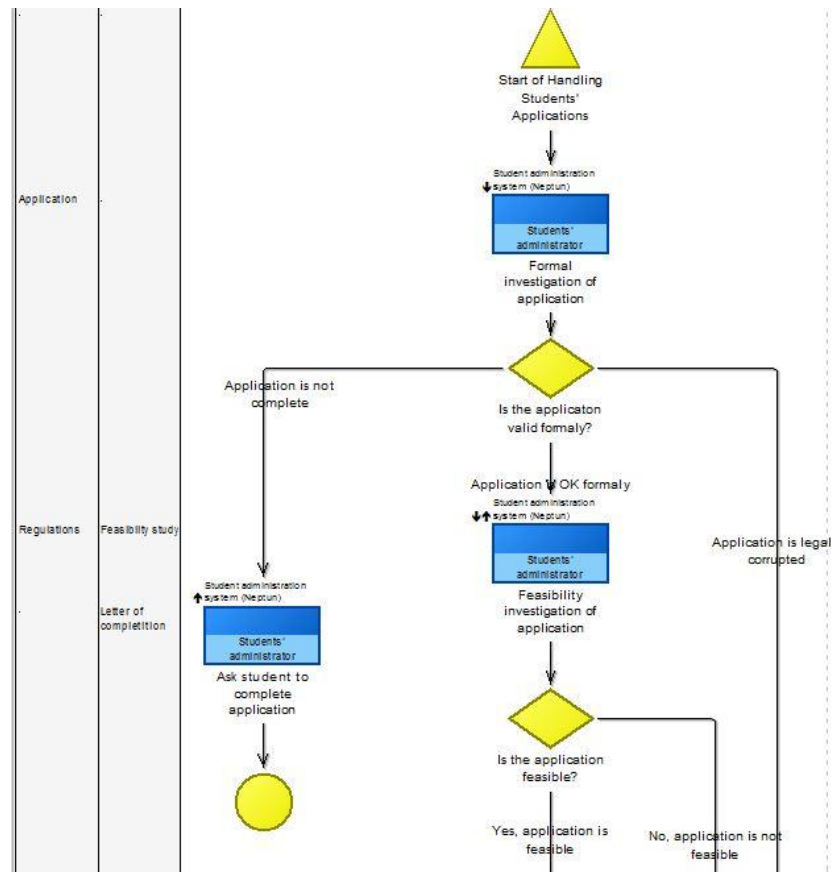
Az előírásként megjelenő konvenciók betartását ellenőrizni is lehet, ami történhet manuálisan, vagy automatikusan. Manuális esetben egy független munkatárs – tehát nem a modellező – szemrevételezi az elkészült modellt, és nem fogadja el addig helyesnek, amíg minden konvenció nem teljesül. Automatikus esetben egy szkript teszi meg ugyanezt, ami természetesen fejlesztéssel jár, és kevésbé képes kezelni a kivételeket.

Ha megtörténik az ellenőrzés, akkor a klasszikus HR-es motivációs elmélet is használható már:

- Pozitív viselkedésért járó jutalmazás: A folyamatmodelleket a konvenciók szerint létrehozó munkatársak elismerése, honorálása.
- Negatív viselkedésért járó büntetés: A folyamatmodelleket hiányosan, nem a konvenciók szerint létrehozó munkatársak büntetése, például bónusz megvonásával.

3.2.2 Folyamatmodellezés a Hallgatói kérvénykezelést vizsgáló eset során

A feladat első lépéseként létrehoztuk a jelenlegi folyamat modelljét az Adonis folyamatmodellező eszközben. A modellezés során a már definiált konvenciók szerint került a folyamatmodell összeállításra. A „Hallgatói kérvénykezelés” folyamat egy része látható a **22.** ábra ábrán.



22. ábra: A "Hallgatói kérvénykezelés" folyamatának kezdete

A referencia folyamat ARIS modellező eszközben készült. Az ARIS és Adonis XML struktúra nem feleltethető meg egymásnak, természetesen olyan szkript, amely képes őket átalakítani, megalkotható, ehelyett azonban a kutatás során az eredeti folyamat modellezésre került Adonisban, így már a két folyamatból azonos átalakításokkal lehetett folyamatontológiát létrehozni, melyet a következő fejezet mutat be.

3.2.3 Folyamatmodellezés a Poszt-operatív ápolást vizsgáló eset során

A poszt-operatív nővérek a műtétek után ébredő betegeknek nyújtanak intenzív segítséget. A nővérek általában jelentős tapasztalattal rendelkeznek a környezetükkel és az életmenő gyógyszerekkel kapcsolatosan, így gyorsan képesek észlelni a veszélyt jelző jeleket a betegeken, és reagálni is nagyon gyorsan tudnak azokra. Mivel a feladataikat napi szinten végzik, úgy tűnhet, hogy a munkájuk rutinszerű, azonban ez koránt sincs így, hiszen nagyon magas szintű tudást igényel annak végrehajtása.

A nővérek által végzett rendszeres ellenőrzések a poszt-operatív időszakban a beteg állapotáról festenek teljes képet az orvosok és a hozzátartozók számára. Az ellenőrzések

során tapasztalt jellemzőket összevetik a műtét előtti jelekkel és az elvárt értékekkel, így tudják megítélni a beteg előrehaladását, és értékelni az állapotát (Royle-Walsh, 1992).

A betegek poszt-operatív megfigyelésére két módszer létezik: a klinikai monitorozás illetve az általános megfigyelés (Alexander et al., 1994). A poszt-operatív ellátás az ábresztő szobában kezdődik, és a teljes felépülési időszak alatt tart. A főbb vizsgálati területek a légzőszervek és légutak, fájdalomérzet, mentális jelek ellenőrzése, vérnyomás és szívritmus megfigyelése, trombózis-elkerülés.

A Med-Assess projekt során a poszt-operatív ellenőrzés folyamatát a nővérekkel készített interjúk során ismertük meg. Az interjúk során megismertették velünk a feladataikat, azok céljait, és a hozzájuk társítható elvárt ismereteket. Megtudtuk, mi alapján hozzák meg döntéseiket bizonyos szituációkban. Az interjúk anyagából folyamatmodell készítettünk, amit az Adonis modellező eszközben ábrázoltunk.

A poszt-operatív ellenőrzés során a nővérek ellenőrzik a betegek vérnyomását, testhőmérsékletét és szívritmusát. El kell látniuk a műtéti sebeket, és fel kell mérniük a betegek fájdalomszintjét. Emellett el kell végezniük minden olyan további ellenőrzést, melyet a konkrét műtét indokol. Ha bármilyen olyan teszt elvégzésére van szükség, amelyre a beteget elő kell készíteni, az is az ő feladatuk. Bizonyos esetekben a laboratóriumi ellenőrzéseket is ők maguk végzik el. Az eredmények ellenőrzésével dönteniük kell arról, hogy be kell-e vonni egy orvost, vagy ők is intézkedhetnek a doktor nélkül. Végül pedig, mindent dokumentálniuk kell.

Ez az általunk megvizsgált magánkórház folyamata, de vajon ez egy „megfelelő”, vagy egy „helytelen” eljárásrend? Megfelel-e az egészségügyi sztenderdeknek, kompatibilis-e a best practice-ekkel? Erre kerestük a választ a szemantikus folyamatmenedzsment eszközeivel.

A poszt-operatív ellátásra nézve nem találtunk egy mindenek felett álló sztenderd eljárást, azonban sikerül azonosítani több best practice-t és cikket is a területen. Ez egyébként a legtöbb szakterületre igaz: a referencia folyamatok azonosítása nehézkes, nem egyértelmű. Az open data és a linked open data világában a megfelelő információforrások azonosítása mindenképp fontos kutatási kérdés, de jelen dolgozat keretein túlnyúlik.

A referenciaként állított poszt-operatív ellátási folyamat a következőképp néz ki: a nővernek ellenőriznie kell a vérnyomást, testhőmérsékletet és szívritmust; el kell látnia

a műtéti sebeket; kezelnie kell a fájdalmat és biztosítani kell a megfelelő légzést; biztosítani kell a megfelelő folyadékbevitelt és meg kell figyelnie az emésztési funkciókat, emellett a beteg mentális állapotára is ügyelnie kell. Ezután a kapott eredményeket össze kell vetnie az elvárt értékekkel, meg kell tennie a szükséges intézkedéseket, majd dokumentálnia kell mindent.

3.3 Folyamatmodellekből történő tudáskinyerés vizsgálata

Az irodalom-áttekintő részben megvizsgáltam a folyamatmodellek transzport formátumait. Szó esett az XML-ről és a folyamatmodelleknél gyakran használt BPEL-ről illetve XPDL-ről is. Az logikus, hogy a kutatás során egy jól strukturált, XML alapú export formátummal rendelkező modellező szoftvert kell használnom, de – mint azt az előző pontban leírtam – a szoftver maga is lehetőségeket is, kötöttségeket is jelent a modellezés során.

3.3.1 Az Adonis XML exportformátumának ismertetése

Az Adonis XML export készítésekor keletkezik egy .dtd fájl, amely leírja az XML-ben használt felépítést. Számunkra a legfontosabb, hogy az <ADOXML>, majd a <MODELS> tag-ekbe sorolja be a program az elkészített modelleket, ezeken belül <MODEL> tag-ek közé kerülnek maguk a modellek. <INSTANCE> tag-ek között kapnak helyet az objektumok, azok attribútumai pedig az <ATTRIBUTE> tag-ek között találhatóak. A hivatkozott objektumok az <INTERREF> tag-ek között kaptak helyet.

Az Adonis által előállított XML állomány értelmezése, és a kutatás szempontjából releváns részeinek ismertetése az 1. mellékletben található.

3.3.2 Adattranszformáció, szövegbányászati módszerek vizsgálata

A folyamatmodellekből kinyerhető tudáselemek és kompetenciák a folyamatmodellekben több helyen kerülhetnek tárolásra; szabad szöveges mezőkben, vagy egy erre dedikált objektumban.

Szükséges tudáselemeket tartalmazhat a tevékenységek valamelyik erre kijelölt szabad szöveges attribútum-mezője, ami az XML tekintetében egy *tag* lesz. Az ilyen szövegek elemzéséhez szövegbányászati eszközökre van szükség, melyeket vizsgálnom kell.

Valószínűleg nincs olyan folyamatmodellező eszköz, amelyben lenne dedikált kompetencia objektum a kompetenciák, tudáselemek tárolásához. Ha a modellezés során mégis kijelölünk erre egy objektumot, akkor a tudáselemek azonosítás jelentősen leegyszerűsödik. Amíg a modellező csak egy előre definiált objektumhalmazból választ, addig a nem kell vizsgálni azt, hogy létrehozott-e új tudáselemet, de ha engedjük a modellezőnek, hogy ő maga is létrehozzon új objektumot, akkor már szükséges annak vizsgálata, hogy ez valóban új tudáselem-e, és hogy hol kell elhelyezni a szakterületi ontológiában.

Hasonló vizsgálat szükséges akkor is, ha szabad szavas kifejezéseket írhat a modellező a folyamatmodell valamelyik attribútumába, ám ekkor a tudáselemek azonosítása is összetettebb szövegbányászati feladattá válik.

A szövegbányászati összetevő megalkotása a kutatás során Saira Gilani¹⁴ feladata lett. A teljesség és áttekinthetőség kedvéért ezt az összetevőt az ő munkája alapján idézem a továbbiakban (Gilani-Kő, 2014).

3.3.3 A Poszt-operatív ápolást vizsgáló eset tudáskinyerésének előkészítése

A ProKEx-ben alkalmazott tudáskinyerési és –átadási megoldás a szemantikus folyamatmenedzsment területén már ismert összehasonlítás alapú (compliance check) elemzéseket tudásalapú elemekkel egészíti ki, ezáltal a poszt-operatív ellátás folyamata nem csupán a folyamat szekvenciális elemzésén keresztül kerül ellenőrzésre és összevetésre a referencia folyamattal, hanem a feladatokhoz elvárt tudáselemek megismerésén keresztül a folyamat a szerepköröktől indulva, tudásalapúan kerül értékelésre.

Mivel a folyamatmodellt nem csak strukturális ellenőrzésnek vetjük alá, hanem a tevékenységekhez szükséges tudáselemeket is azonosítani akarjuk, a tevékenységek szöveges leírásán szövegbányászati elemzéseket végeztünk. A szövegbányászat eredménye adta a kiindulópontot a folyamatontológia és a szakterületi „Nővér” ontológia egymásra vonatkoztatása (mapping) során.

¹⁴ Saira Gilani a Gazdaságinformatika Doktori Iskola hallgatója.

3.4 Tudásreprezentáció

Mivel XML alapú exportba nyerjük ki a folyamatmodell tartalmát, ezért ennek feldolgozásával építhető fel a folyamatontológia. Ha nem akarjuk az XML teret elhagyni, akkor erre a legalkalmasabb az XSLT transzformáció használata, melynek során az XML fájlban egy XSLT stíluslap végez módosításokat, majd egy új fájl jön létre, ami a mi esetünkben egy RDF ontológia.

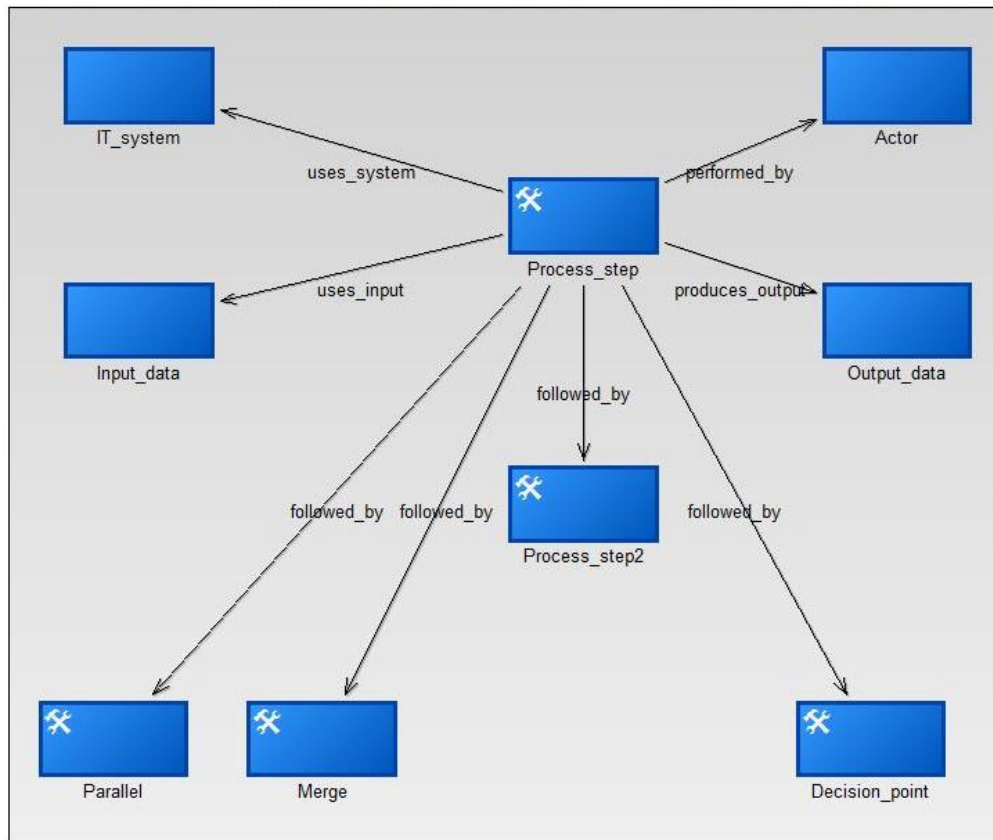
Másik lehetőség dedikált program létrehozása, amelyik az XML-ből bizonyos szabályok mentén tud kinyerni elemeket. Erre jó eszköz lehet a Java nyelv, melyben ezek a műveletek egyszerűen megoldhatóak, mint ahogy ez Borbásné dr. Szabó Ildikó disszertációjában (Borbásné Szabó, 2013) is jól látható.

3.4.1 A folyamatontológia metamodellje

A kutatásban felhasználok korábbi eredményeinket (Ternai et al., 2013), és az ott elállított ontológia szerkezetet fejleszttem tovább. Az kezdeti ontológiaszerkezet a következő (23. ábra):

- `Process_step`: osztály, mely a folyamatmodell tevékenységeit reprezentálja
- `Actor`: osztály, mely a tevékenységek végrehajtóit jelenti
- `IT_system`, osztály, mely a tevékenységeknél igénybevett IT rendszereket jeleníti meg
- `Input_data`, osztály, a tevékenységek bemeneti adatai
- `Output_data`, osztály, a tevékenységek kimeneti adatai
- `Parallel`, `Merge`, `Decision_point`: osztályok, a folyamatmodell tevékenységeken kívüli egyéb objektumai
- `followed_by`: kapcsolat, a `Process_step` osztály elemei között értelmezve, amely a folyamat egymást követő tevékenységeit köti össze
- `performed_by`: kapcsolat, a `Process_step` osztály elemeit köti össze az `Actor` osztályon belüli végrehajtóval
- `uses_system`: kapcsolat, a `Process_step` osztály elemeit köti össze az `IT_system` osztályon belüli igénybevett IT rendszerrel
- `uses_input`: kapcsolat, a `Process_step` osztály elemeit köti össze az `Input_data` osztályon belüli bemeneti adatokkal

- produces_output: kapcsolat, a Process_step osztály elemeit köti össze az Output_data osztályon belüli kimeneti adatokkal



23. ábra: A folyamatontológia kezdeti modellje

Ez az ontológia egy folyamathoz került kifejlesztésre, ezért nem képes kezelni komplex folyamatrendszereket. Bizonyos objektumokat nem alakít át ontológia osztályokká, így a logikai szál megszakadhat benne. Nincs felkészítve a dedikált kompetencia-leíró objektum kezelésére, de egy szabad szöveges mező tartalmát magában hordozza.

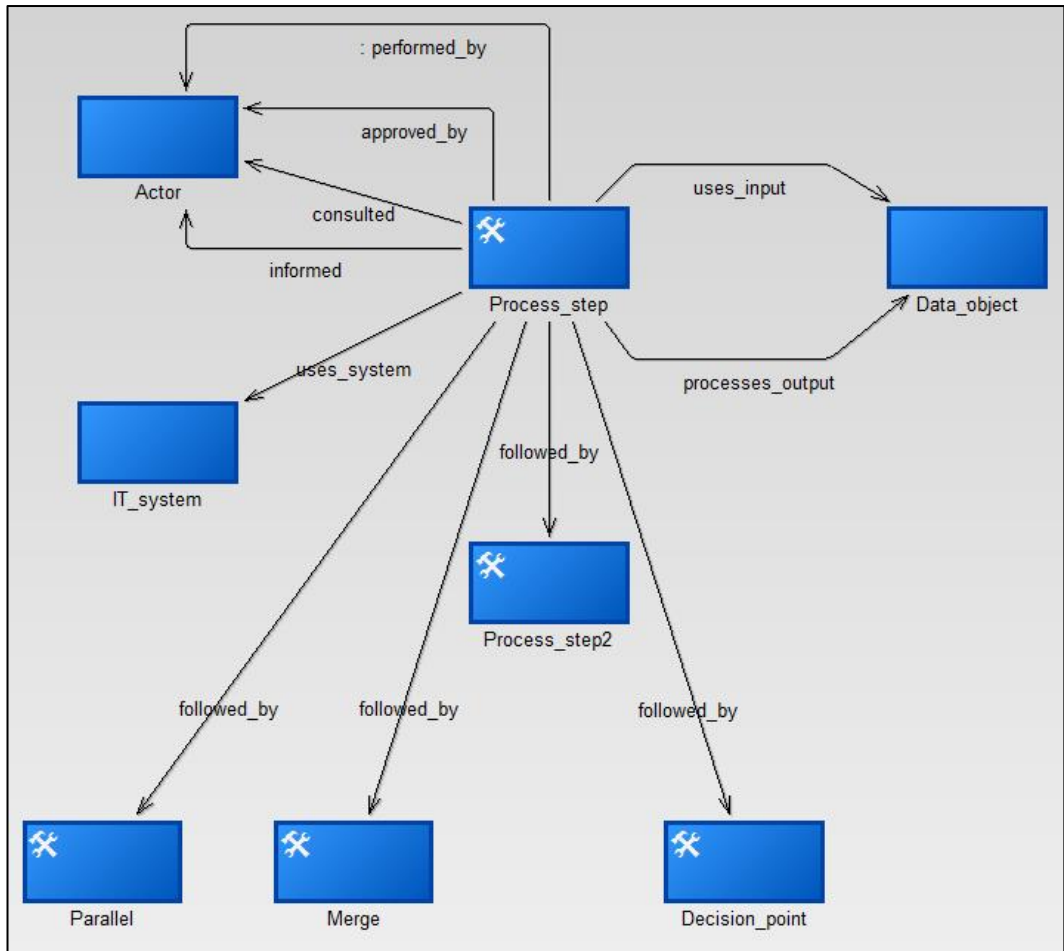
Jól látható tehát, hogy a fenti folyamatontológia kezdeti verzióinak megfelelő, de a kutatás céljainak és megoldandó feladatainak teljes körű megoldásához még fejlesztése szükséges.

3.4.2 A folyamatontológia fejlesztett verziójának metamodellje

Az eredeti kiegészül a következőkkel (24. ábra):

- Több folyamat kezelése

- Input_data és Output_data helyett Data_object osztály, amelyik uses_input illetve processes_output tulajdonságokkal kapcsolódik a Process_step osztályhoz
- Az Actor osztály a RACI-nak megfelelően négyféle kapcsolattal kötődhet a Process_step osztályhoz: performed_by; approved_by; consulted és informed.



24. ábra: A folyamatontológia kiegészített verziója

A változtatásokra azért került sor, mert a kutatás során kiderült, hogy az adat objektumok külön inputként és outputként való kezelése nem logikus, hiszen egy adott elem mindkét szerepet felveheti, de külön osztályként két példány jönne létre belőlük. Ugyanez a logika hívta életre az Aktor osztályt, amely négyféle kapcsolattal kötődhet a folyamatlépéshez.

A több folyamat kezelése pedig szintén a tapasztalatokon alapuló igény volt, hiszen a megoldásnak nem csak egyetlen folyamat elemzését kell támogatnia, hanem komplex folyamatrendszerét is.

3.4.3 Folyamatontológia létrehozása a Hallgatói kérvénykezelést vizsgáló eset során

A folyamatontológia létrehozása során a folyamatmodell elemeit kell ontológia elemekké alakítani. Erre jó lehetőséget kínál a meta-modell alkotás. Az átalakítás során így az egymásnak megfelelő objektumok pontos leképezése adható meg, ezáltal a megoldás könnyen használható azok számára is, akik nem rendelkeznek programozói háttérrel.

A folyamatontológia létrehozásához le kell tehát képezni a folyamatmodell objektumait és azok tulajdonságait ontológiaelemekre és azok tulajdonságaira. Az ontológia a modellező nyelv és a modell eleminek szemantikus leképezését is megadja (Kramler-Murzek, 2006). Több projekt foglalkozott folyamatontológia létrehozásával, például a SUPER projekt is, de a megoldásomban egy egyéni módszerrel létrehozott ontológiát használtam.

A folyamatontológia létrehozásának első lépéseként a folyamatmodell elemeit az ADONIS folyamatmodellező eszköz XML exportjaként nyertem ki. A folyamatmodell minden egyes objektuma „instance” lesz az XML állományban, az objektumok attribútumai „attribute” tag-ek közé kerülnek. A folyamatmodell tevékenységeihez tartozó egyéb objektumok (végrehajtók és a RACI egyéb elemei, bemenetek és kimenetek, IT rendszerek) az „interref” tag-ek közé kerülnek. A létrejött XML állomány egy részlete látható a 25. ábrán.

```

<INSTANCE id="obj.13793" class="Activity" name="Making the decision">
<ATTRIBUTE name="Description" type="STRING"></ATTRIBUTE>
<INTERREF name="Responsible role">
<IREF type="objectreference" tmodeltype="Working environment model" tmodelname="Roles"
tmodelver="" tclassname="Role" tobjname="Dean"></IREF>
</INTERREF>
<INTERREF name="Input">
<IREF type="objectreference" tmodeltype="Document model" tmodelname="Input / Output"
tmodelver="" tclassname="Document" tobjname="Application"></IREF>
</INTERREF>
<INTERREF name="Output">
<IREF type="objectreference" tmodeltype="Document model" tmodelname="Input / Output"
tmodelver="" tclassname="Document" tobjname="Decision"></IREF>
</INTERREF>
<INTERREF name="Referenced IT system elements"></INTERREF>

```

25. ábra: Adonis XML export egy részlete

Az így létrejövő XML állományt egy XSLT szkript alakítja át folyamatontológiává meta-szinten. Az átalakítás, és a folyamatontológia használatának értelme az, hogy megkapjuk a folyamatmodell szemantikus leképezését. A folyamatmodell minden objektuma osztályként jön létre az ontológiában, az objektumok tulajdonságai pedig az ontológiaosztályok attribútumai lesznek. Az XSLT szkript használatának az XML állomány esetén előnye az, hogy nem lépünk ki az XML térből, egy gyorsan végrehajtható, és jól dokumentált eszközzel alakíthatjuk át az XML állományt, nem szükséges bonyolultabb programozás, és jelentős külső eszköz az átalakításhoz. A létrejövő ontológia RDF kiterjesztésű lesz, melyet könnyedén OWL-lé alakíthatunk.

Az XSLT szkript egy részlete, amely egy tevékenység „Végrehajtó” tulajdonságát képezi le az ontológiába látható az 26. ábrán.

```

<xsl:if test="INTERREF[@name='Responsible role']/IREF">
<ObjectAllValuesFrom>
  <ObjectProperty IRI="#performed_by"/>
  <xsl:for-each select="INTERREF[@name='Responsible role']/IREF">
    <Class >
      <xsl:attribute name="IRI">#<xsl:value-of select=
"functx:words-to-camel-case(@tobjname)" /></xsl:attribute>
    </Class>
  </xsl:for-each>
</ObjectAllValuesFrom>
</xsl:if>

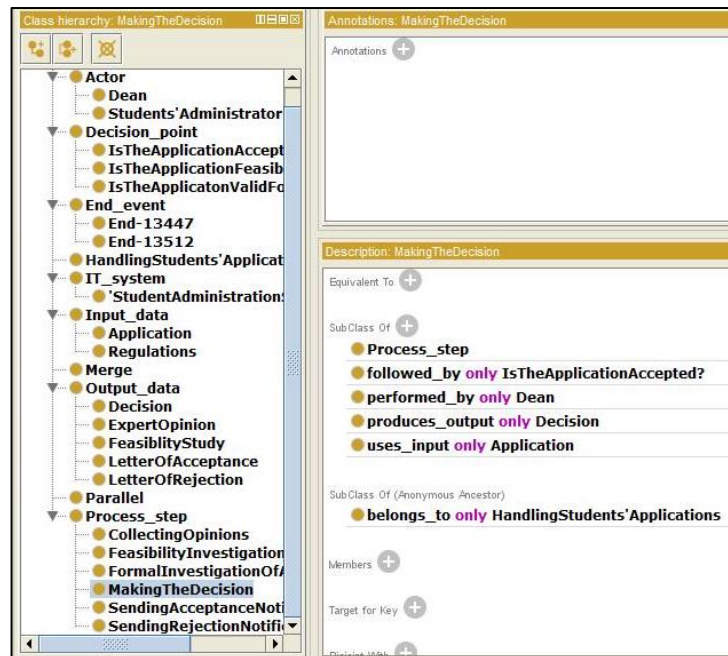
```

26. ábra: Az átalakító XSLT szkript egy részlete

Az ADONIS modell szemantikus leképezése két síkon megy végbe az ontológiában: először a modell „nyelvi elemeit” kell leképezni (ez történik meg meta-szinten, amikor a modellező nyelv alapvető objektumainak megfelelő ontológia elemek kerülnek definiálásra és leképezésre); majd magát a modell konkrét elemeit is létre kell hozni a

példányosított ontológia osztályokon és tulajdonságokon keresztül. A tulajdonságok és kapcsolatok használata teremti meg a folyamatmodell szemantikus leképezését a folyamatontológiában.

Az ontológia Protege programban a 27. ábrán látható.



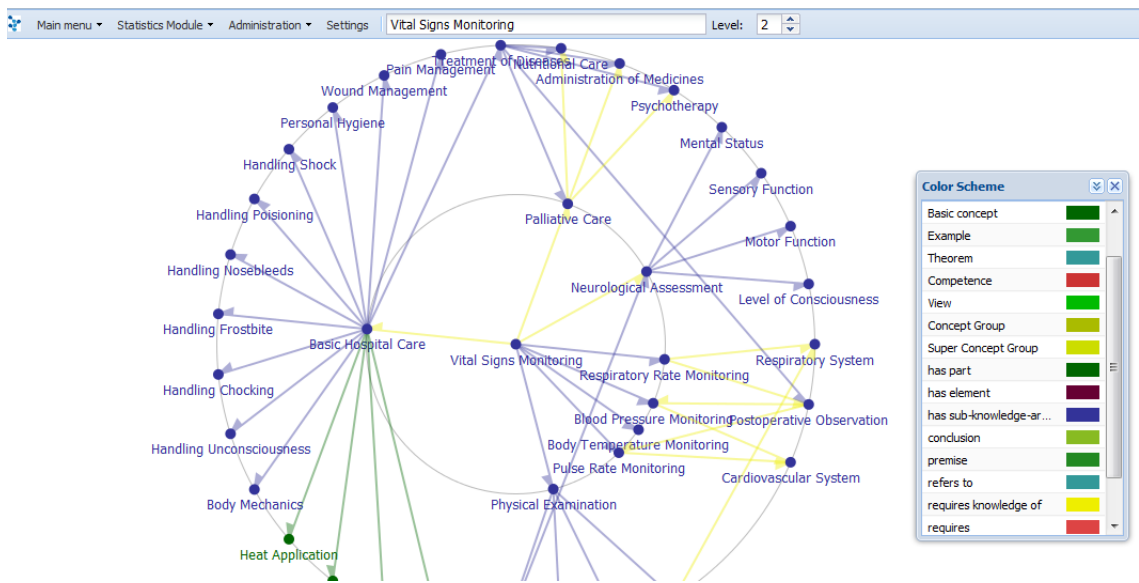
27. ábra: A folyamatontológia Protege programban megjelenítve

3.4.4 Folyamatontológia létrehozása a Poszt-operatív ápolást vizsgáló eset során

3.4.4.1 Szakterületi ontológia a STUDIO rendszerben

A Studio rendszer egy kompetencia-alapú e-learning megoldás, amely a hiányzó tudásterületek feltárásában nyújt segítséget a felhasználóknak. A rendszer alapja az oktatási ontológia, ellenőrző kérdéseken keresztül fűr le az ontológiában, így ellenőrizve az egyre inkább részletekbe menő tudást. A tudást <Tudásterület>, <Alapfogalom>, <Elmélet> és <Példa> osztályokba sorolja be. A tudásterületek közötti hierarchiát a “része” és az “alterülete” tulajdonságok használatán keresztül építi fel. A tudásterületek között kognitív láncot is létrehoz a “ismeretét_követeli”, “feltétel”, “következtetés” and “vonatkozik” tulajdonságokon keresztül (Vas et al., 2009).

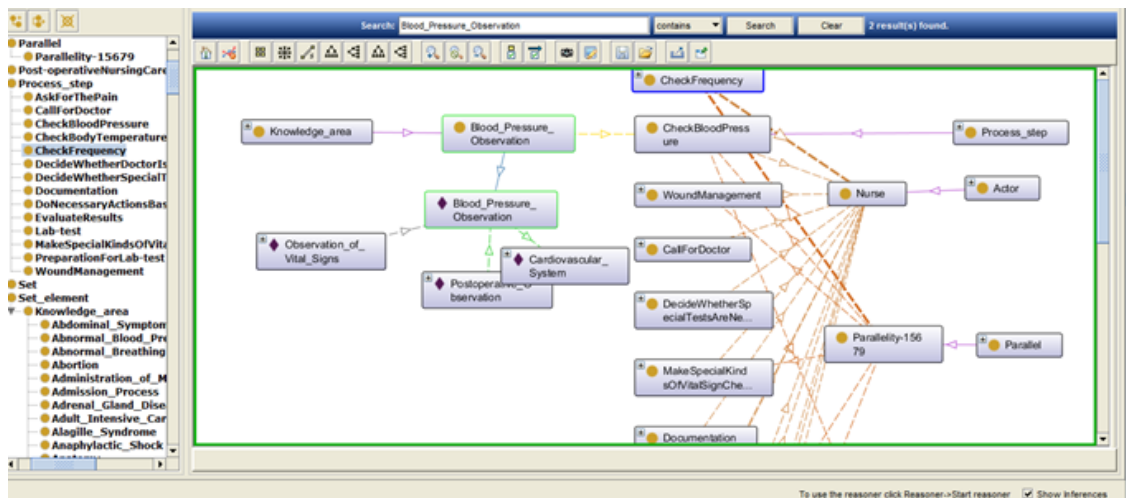
A Studio rendszer már több projektben szerepet kapott, ennek megfelelően az alapjául szolgáló ontológia folyamatosan bővül. A „Nővér” ontológia a Med-Assess projekt keretében került létrehozásra (28. ábra).



28. ábra: A "Nővér" ontológia a STUDIO rendszerben megjelenítve

3.4.4.2 A folyamatontológia és a szakterületi ontológia összerendelése

A nővér ontológia RDF formátumban került exportálásra a Studio rendszerből, és a Protege rendszerben vontuk össze a folyamatontológiával. A kialakult ontológia a 29. ábrán látható.



29. ábra: Szakterületi és folyamatontológia együtt a Protege rendszerben

3.5 Tudástranszfer és folyamat-összehasonlításon alapuló fejlesztés

A hasznosítás egyik területe az emberi erőforrás-gazdálkodás, vállalati folyamatokra épülő kiválasztó és képzőrendszer hozható létre. A létrejövő ontológia adaptív tesztrendszerhez illesztve egyaránt alkalmas egy munkakör meghirdetések a jelentkezők szűrésére, valamint előrelépéskor, karriertervezéskor a hiányzó

tudásterületek felderítésére és azok megtanítására is. Cél a szervezeti tudás artikulálása (feltárása) és hosszú-távú megőrzése (intellektuális vagyon kezelése).

Az emberi erőforrás-menedzsmentben az így létrejövő rendszer tehát az állásajánlatra jelentkezők tudásának felmérésben, vagy a vállalaton belüli karrierút tervezésben is hasznos lehet. Megfelelően modellezett folyamatok esetén az egyes munkakörökhöz pontosan és egyszerűen azonosíthatóak a szükséges kompetenciák, és ezek megléte felmérhető, majd a tudásszint személyre szabott tananyagokkal fejleszthető.

A kutatás során előálló többi eredmény, például a folyamatontológia előállításához szükséges modellezési feltételek azonosítása szintén hasznosítható a továbbiakban, hiszen egy folyamatontológiára alapozva például a „compliance check” területén a vállalati folyamatok összevethetőek előírásokkal, best practice-ekkel, vagy referencia folyamatokkal, még ha ezek más folyamatmodellezési módszertannal is kerültek modellezésre, hiszen a közös folyamatontológia-formátum garantálja az összevethetőséget.

A kutatás során előálló keretrendszer felhasználható több szakterületen is. A különböző területeken való alkalmazáshoz, a következő előfeltételek teljesülése kell, hogy biztosított legyen:

- Folyamatmodellezési konvenciók betartása
- Folyamatmodellezési szoftver megfelelősége
- Szakterületi ontológia megléte

3.5.1 Az ontológiák összehasonlítása a Hallgatói kérvénykezelést vizsgáló eset során

Az ontológiák összehasonlításához a Protege beépített megoldásait használtuk.

Az OWL DIFF (OWLDiff, 2008) egy Java-alapú Protégé plugin, amelyik ontológiák egyesítésére és összehasonlítására használható. Segítségével a folyamatontológiák összehasonlítása leginkább strukturálisan végezhető el. Az eszköz két ontológia axiomatikus megfeleltetését vizsgálja. A különbségeket egy-egy fában jeleníti meg, a két ontológiára külön-külön.

A „Compare Ontologies” (ontológiák összehasonlítása) a Protege egyik beépített funkciója. Különböző névterekből származó ontológiák összehasonlítására is képes, amire az OWL DIFF nem. Ez a megoldás is axiomatikus különbségeket keres, de az eredményeit táblázatos formában jeleníti meg. Háromféle lehetőséggel végezhet az összehasonlítás: létrehozás, törlés, módosítás – az eredményeket is ezek szerint csoportosítja a funkció.

Az összehasonlítás során olyan kérdéseket vizsgáltunk, melyeket egy folyamatgazda is feltenne a folyamatok vizsgálata során:

- A tevékenységek azonos sorrendet követnek-e mindkét folyamatban?
- Mely tevékenységek hiányoznak a jelenlegi folyamatból, illetve melyek kerültek bele feleslegesen a referencia folyamathoz képest?
- Szükség van-e tevékenységek beszúrására a jelenlegi folyamatba a referencia folyamat alapján?
- Azonos feladatokért felelősek az azonos szereplők?
- Ad-e valamilyen extra információt az elvártakról a referencia folyamat? Milyen javítási lehetőségek fogalmazhatóak meg mind a folyamat strukturális felépítésében, mind a támogató erőforrásokban?

A feltett kérdésekre egy olyan jelentés adhat választ, amelyik a különbségeket minél strukturáltabban adja meg. A „Compare ontologies” táblázatos formátuma, a létrehozás, törlés, módosítás akciókkal jó alapot adhat egy folyamatgazdának a folyamatok vizsgálatakor.

3.5.1.1 Az ontológiák összehasonlításának eredménye

A „Compare ontologies” funkció valójában egy ontológia két különböző verziójának összehasonlítására szolgál. Ennek megfelelően az általunk vizsgált esetet ehhez kellett igazítani: a jelenlegi folyamatot tekintettük az ontológia alapverziójának, míg a HEFOP projektből származó referenciafolyamatot a fejlesztett, újabb verziónak. Így az összehasonlítás során látni fogjuk, miben különbözik, miben „fejlettebb” a referenciafolyamat az aktuális folyamattól.

Az első összehasonlítás során azt vizsgáltuk, „SubClass of Process_step” kifejezésekre milyen eltéréseket tár fel az eszköz. Ezzel azt vizsgáltuk meg, hogy milyen olyan tevékenységek találhatóak a referenciafolyamatban, melyek az eredeti folyamatban nem voltak megtalálhatóak. Eredményül az

„AskStudentToCompleteApplication” és az „OfferingPossibilityToReview” új tevékenységeket kaptuk, vagyis a rendszer megtalálta fejlesztési lehetőségként a hiánypótlási és a fellebbezési lehetőségeket, melyek az alapverzióban nem szerepeltek. A tevékenységekhez az összehasonlítás megadta az elvárt végrehajtót, input és output adatot, illetve támogató IT rendszert is (8. táblázat).

Block	Created: AskStudentToCompleteApplication	
Action	Original version	Updated version
Added		Class: AskStudentToCompleteApplication
Added		AskStudentToCompleteApplication SubClassOf Process_step
Added		AskStudentToCompleteApplication SubClassOf followed_by only End-13424
Added		AskStudentToCompleteApplication SubClassOf performed_by only Students'Administrator
Added		AskStudentToCompleteApplication SubClassOf produces_output only LetterOfCompletion
Added		AskStudentToCompleteApplication SubClassOf uses_system only StudentAdministrationSystem(Neptun)
Block	Created: OfferingPossibilityToReview	
Action	Original version	Updated version
Added		Class: OfferingPossibilityToReview
Added		OfferingPossibilityToReview SubClassOf Process_step
Added		OfferingPossibilityToReview SubClassOf followed_by only End-13447
Added		OfferingPossibilityToReview SubClassOf performed_by only Rector'sOffice

8. táblázat: A HEFOP projekt referenciaciklusa alapján létrehozott új tevékenységek

Azt, hogy az új folyamatlépések hova tartoznak a folyamaton belül, a „followed_by AskStudentToCompleteApplication” és a „followed_by OfferingPossibilityToReview” keresésekkel találhatjuk meg. Ennek hatására az „IsTheApplicatonValidFormaly?” döntési pontot és a „SendingRejectionNotification” tevékenységet találja az eszköz. Mivel ezeket a folyamatlelemeket a megváltoztatott elemek között sorolja fel a funkció, ez alapján tudható, hogy ezek már szerepeltek az aktuális folyamatban. Mint a 9. táblázatból látható a „SendingRejectionNotification” tevékenység (vagyis az elutasításról szóló értesítés kiküldése) input adatot kell kapjon, és ezt kell, hogy kövesse a fellebbezés lehetősége, vagyis az „OfferingPossibilityToReview” step.

Block:	Modified: IsTheApplicatonValidFormaly?	
Action	Original version	Updated version
Added		IsTheApplicatonValidFormaly? SubClassOf followed_by only AskStudentToCompleteApplication
Block:	Modified: SendingRejectionNotification	
Action	Original version	Updated version
Added		SendingRejectionNotification SubClassOf followed_by only OfferingPossibilityToReview
Added		SendingRejectionNotification SubClassOf uses_input only Decision
Deleted	SendingRejectionNotification SubClassOf followed_by only End-13447	

9. táblázat: A HEFOP projekt referencifolyamata alapján megváltoztatott tevékenységek

A megváltozott folyamatban az elutasításról szóló értesítés küldése után egy „End” objektum szerepelt, ami a folyamat végét jelzi, de az új verzióban már a fellebbezés szerepel ennek helyén, ezért a korábbi „End” a jelentésben törölt elemként jelenik meg.

Hasonló módon a különböző erőforrásokról is gyűjthetünk adatokat, a 10. táblázatban például egy tevékenység végrehajtójának változtatására tesz javaslatot a program.

Block:	Modified: MakingTheDecision	
Action	Original version	Updated version
Superclass changed	MakingTheDecision SubClassOf performed_by only Dean	MakingTheDecision SubClassOf performed_by only Committee

10. táblázat: A HEFOP projekt referencifolyamata alapján megváltoztatott végrehajtó

Összefoglalva, ez a folyamatokat vizsgáló összehasonlítás ontológiákra alapozva történt meg, keresési kifejezésekkel paraméterezve a beépített összehasonlító motort.

A SubClassOf <<class>> kapcsolat a Létrehozott elemeket bemutató táblázatban tehát azt jelenti, hogy az aktuális folyamatba a referenciát alapul véve egy új elemet kellene beilleszteni. A megváltoztatott vagy törölt elemeket összegző táblázat pedig a folyamatmodell objektumainak megváltoztatását vagy törlését javasolja – a referencifolyamat alapján.

A létrehozott vagy megváltoztatott objektumok esetén célszerű a keresést folytatni a hozzájuk tartozó kapcsolatok vizsgálatával, így tudható meg például az, hogy egy új elemet egész pontosan hova kell beszúrni a folyamatba, vagy, hogy egy új szereplő melyik tevékenységek végrehajtója lesz.

A két folyamat összehasonlítása alapján a valóban megvalósuló változtatásokat a folyamatgazdának kell meghatároznia. Nem minden eltérés a referenciaművelethez képest jelent negatív torzulást. A szervezeti méret, a felelőségek összevonása, a folyamatok eltérő kialakítása tisztán gépi eszközökkel nem értékelhető. A felvázolt megoldás azonban képes arra, hogy jelezze az eltéréseket és javaslatot tegyen a változtatásokra.

3.5.2 Ontológia-alapú folyamat-összehasonlítás a Poszt-operatív ápolási folyamat esetén

A létrehozott ontológiákat a következő kérdések megválaszolása céljából vetjük össze:

1. Mennyiben fed át a jelenlegi és a referencia folyamat, beleértve a tevékenységeket, adatelemeket, szereplőket és a támogató IT rendszereket?
Az összehasonlítás egyik nehézsége, hogy az aktuális folyamat konkrét megvalósulásokat tartalmazhat (pl. Omron® 7) az általános megnevezéseket tartalmazó referencia folyamattal szemben (vérnyomás-mérő eszköz). A szakterületi ontológia és egyéb szemantikus technológiák használata azonban ilyen esetekben is megfelelő lehet.
2. Mennyiben fednek át a jelenlegi folyamat és a referencia folyamat szerepkörei a szerepkörökhöz tartozó tevékenységek által megkövetelt tudáselemek alapján? Ennek több szintje is vizsgálatra érdemes:
 - a. Az elvárt tudáselemek azonosok-e tevékenységről-tevékenységre a jelenlegi és a referencia folyamatban?
 - b. Az elvárt tudáselemek azonosok-e megegyező munkakörök esetén a teljes jelenlegi és referencia folyamatban?
 - c. Az elvárt tudáselemek azonosok-e megegyező munkakörök esetén a jelenlegi és a referencia folyamatcsoportban?

Ezek a vizsgálati kérdések egy szintekre bontható lekérdezés halmazt definiálnak, hiszen először a tevékenységeket, majd a folyamatokat, végül a teljes folyamatcsoportot vizsgáljuk. Ezáltal a folyamatgazda egy jól felépített riportot kaphat az elvárt tudás alapján a fenti szintek szerint.

A jelentés a következőket mutatja meg a folyamatgazdának:

Tevékenységek

1. Milyen tudáselemek kerültek azonosításra a jelenlegi folyamatban;
2. Az azonosított tudáselemek közül melyek részei a referencia folyamatnak;
3. Az azonosított tudáselemek közül melyek nem részei a referencia folyamatnak.

Folyamatok

1. Milyen tudáselemek kerültek azonosításra a jelenlegi folyamatban;
2. Az azonosított tudáselemek közül melyek részei a referencia folyamatnak;
3. Az azonosított tudáselemek közül melyek nem részei a referencia folyamatnak.

Munkakörök

1. Milyen tudáselemek kerültek azonosításra a jelenlegi folyamatban;
2. Az azonosított tudáselemek közül melyek részei a referencia folyamatnak;
3. Az azonosított tudáselemek közül melyek nem részei a referencia folyamatnak.

A munkakör-tevékenység-tudás hármastól nem elválasztható, ezért is jó eszköz a szakterületi ontológiával kombinált folyamatontológia a vizsgálatokhoz.

3.5.2.1 Eszköz a szemantikus összehasonlítás elvégzéséhez

Az összehasonlítást a Protege ontológia-szerkesztő és elemző programmal végeztük, amely egy ingyenes, JAVA alapú, open source eszköz, amely széles támogatottságnak és elismertségnek örvend.

Az ontológia megfelelő elemeinek leválogatásához az úgynevezett DL query-eket használtuk, amelyek több lépéses ontológia-elemzést tesznek lehetővé a Protege-ben. A használt DL query-eket, és azok jelentését a 11. táblázat foglalja össze.

Leírás	DL Query
Munkakörhöz szükséges tudás	required_by some (performed_by only <Role>)
Folyamat végrehajtásához szükséges tudáselem	required_by some (belongs_to only <Process>)
Adott tevékenység végrehajtásához szükséges tudáselem	required_by some (<Process_step>)

11. táblázat: Ontológia lekérdezésre használt DL query-k

A DL Query-k használatával az ontológiának egy meghatározott részét tudjuk tovább vizsgálni. A részontológiák elemit elláttuk egy „type” tulajdonsággal (data property), amely „actual” vagy „ref” értéket vehet fel, így utalva a jelenlegi és a referencia folyamatra. Ezután a „Compare ontologies” beépített funkciót használtuk a megegyező és a különböző tudáselemek kigyűjtésére. A „Compare ontologies” alapvetően egy ontológia két verziójának összehasonlítására képes, így a mi esetünkben a jelenlegi folyamat volt az eredeti verzió, míg a referencia folyamat a továbbfejlesztett.

A jelentés három részre tagolódik:

- A létrehozott elemek olyan elemek, melyek a referencia folyamatnak részei, de a jelenlegi modellnek nem.
- A törölt elemek olyan elemek, melyek a jelenlegi folyamatnak részei ugyan, de a referencia modellnek nem.
- A módosított és/vagy átnevezett elemek olyan elemek, melyek ugyan mindkét folyamatnak elemei, de az axiomatikus elemzés különbséget talált közöttük (például eltérő tulajdonságok vagy kapcsolatok miatt).

A „Nővér” ontológia minden eleme a módosított elemek közé került, aminek az az oka, hogy a szakterületi ontológia azonos elemei került hozzáadásra a két folyamatontológiához, de a jelenlegi folyamatban ezek az elemek „actual” tulajdonsággal, míg a referencia folyamatban „ref” tulajdonsággal kerültek megjelölésre.

```

1 ...
2 -----
3 Created RespirationAndAirwayManagement
4 -----
5 Added: Class: RespirationAndAirwayManagement
6 Added: RespirationAndAirwayManagement SubClassOf Process_step
7 Added: RespirationAndAirwayManagement SubClassOf followed_by only Merging-15682
8 Added: RespirationAndAirwayManagement SubClassOf performed_by only Nurse
9 -----
10 ...
11 -----
12 Deleted AskForThePain
13 -----
14 Deleted: Class: AskForThePain
15 Deleted: AskForThePain SubClassOf Process_step
16 Deleted: AskForThePain SubClassOf followed_by only Merging-15682
17 Deleted: AskForThePain SubClassOf performed_by only Nurse
18 -----
19 Modified Body_Temperature_Observation -> Body_Temperature_Observation
20 -----
21 Added: Body_Temperature_Observation type "ref"^^string
22 Deleted: Body_Temperature_Observation type "actual"^^string

```

30. ábra: Az ontológia-összehasonlítás eredménye

A fenti riportot tovább alakítva, a következő ábrán látható részletesebb jelentést kapjuk.

```

***** Match By Id *****
Algorithm Match By Id completed.
Match By Id step took 172 ms.
1100 owl entities matched up.
4205 axioms matched up.
*****
Took 328ms
Calculating presentation
Took 62ms
Total time = 1794ms
Semantic Compliance Checking Report
Process name: Post operative nursing care process
Number of tasks: 14
Role: Nurse
Date: 15-7-2014 12:56
Knowledge that reference role possess, but your role doesn't
abnormal breathing
alveoli
apnea
asbestosis
asthma
bronchi
bronchial disease
bronchiectasis
bronchitis
chest pain
chest symptom
chronic cough
cough
dry cough
dysphagia
dyspnea
hemoptysis
interstitial lung disease
lower respiratory tract
lower respiratory tract disease
lung
lung disease
nasal congestion
observation of vital signs
....
Knowledge that reference role doesn't require, but your role possess
Knowledge that reference role and your role possess
abnormal blood pressure
anemia
aneurysm disease
angiodyplasia
aortic disease
arrhythmia
arterial occlusive disease
arteriosclerosis
bacteremia
blood pressure observation
blood vessel
bloodstream symptom
body temperature observation
bradycardia
Reference process name Post operati
Number of tasks: 12
Role: Nurse

```

31. ábra: Az összehasonlító jelentés

A jelentés arról tájékoztatja a folyamatgazdát, hogy a jelenlegi folyamat részletesebb, legalábbis több tevékenységet tartalmaz. Azonban a „Nővér” szerepkör tudása hiányos. A jelentés értelmezése minden esetben a folyamatgazda feladata. Különbséget kell tenni ugyanis aközött, hogy a feltárt eltérések hibákra, vagy csupán a vizsgált szervezet referenciától ugyan eltérő, de indokolható különbségeire utal. Egy kisméretű szervezetben, ahol a felelőségek szétválasztása nem olyan jelentős, mint egy nagyobb vállalatnál, a jelentés azt mutatja majd, hogy egy szerepben több tudáselem sűrűsödik. De ettől még a folyamat, vagy a kisebb vállalat működése nem „rossz”, csupán méretéből adódóan más.

A kidolgozott megoldás egy eszközt ad a folyamatgazda kezébe, hogy a folyamatokat ne csak szekvenciálisan vizsgálja meg, hanem vegyen figyelembe egyéb területeket is, amelyek jellemezhetik a folyamatokat. Erre jó megoldás a tudásintenzív folyamatok vizsgálata, és ezen folyamatok összevetése a sztenderdekkel, best practice-ekkel.

A folyamatok összevetésére, majd a jelentés készítésére alkalmas eszköz Java nyelven készült, felhasználva az OWL API-t, a DLQueryExample.java-t és a Compare Ontologies funkció SVN repository-ját.

3.6 A változások kezelésének vizsgálata

A kutatási feladatban több körülmény is változhat. Az egyik a kiindulási alapot jelentő folyamatmodell. Ha a folyamatmodell változik, akkor meg kell azt vizsgálni, hogy ez hogyan érintett a vizsgált szerepkört. Mivel a STUDIO szakterületi ontológiájában a munkakörhöz tartozó feladatok alapján *sub-concept-group* (az azonosított tudáselemek által kifeszített fa az ontológiában) segítségével kerül lehatárolásra mindazon ontológiaelemek halmaza, melyeket a munkakörhöz tartozónak fogadunk el, vizsgálni kell, hogy érinti-e a folyamatmodell megváltozása a *sub-concept-group*-ot. A folyamatmodell változásai a következők lehetnek:

- Folyamatok vagy tevékenységek törlése, módosítása, hozzáadása
- Végrehajtók törlése, módosítása, hozzáadása
- Tevékenységek leírásának módosítása
- *Information* objektumok törlése, módosítása, hozzáadása
- *Information* objektum tevékenységhez rendelésének törlése, módosítása, hozzáadása

Mindezen változásokra fel kell készülni, hiszen a dinamikus változásokra felkészített rendszerben ezeket ideális esetben automatizáltan, de legalább félig automatikusan kellene tudni kezelni.

A fentiek mellett vizsgálni kell még a szervezeten belüli, tehát a munkakörökben, vagy az elvárt tudáselemekben bekövetkező változások kezelését is. Bár ezen változásoknak – a konvenciók betartása esetén – meg kell jelenniük a folyamatmodellben is, a vizsgálatuk mindenképp fontos.

4 Összefoglalás

A disszertációban leírt kutatás a vállalatokban meglévő munkakörök betöltéséhez szükséges tudás és kompetencia feltérképezésének újszerű lehetőségeit vizsgálja. Célja olyan informatikai és tudásmenedzsment eszköz kialakítása, amely képes a szervezeti tudásvagyon a szervezet folyamatait leképező folyamatmodellekből kinyerni, azt ontológiába leképezni, majd - hasznosításként – a folyamatot egy referencia folyamattal tudásalapon összehasonlítani.

4.1 A disszertáció eredményei

Kutatásomban a folyamatmenedzsment és a tudásmenedzsment területeit kapcsoltam össze, célom ugyanis a folyamatmodellekből indulva feltárni a munkakörök betöltéséhez szükséges tudáselemeket, illetve ezek kinyerésének módszereit, majd ezekből ontológia építésével a tudástranszfer megvalósításához konkrét megoldást adni.

Kutatásom problémamegoldó, feltáró jellege miatt nem fogalmaztam meg statisztikai módszerekkel igazolható hipotéziseket. A disszertáció életciklus szemléletben megoldandó feladatok sorozatát adja meg, melyeken keresztül elérhető az alapvető kutatási cél.

A megoldott feladatok, vizsgált kutatási területek a következők voltak:

1. **Folyamatmodellező nyelvek, módszertanok vizsgálata:** A feladatot az irodalom-áttekintés során részben megoldottam. Áttekintettem a folyamatmenedzsment, folyamatmodellezés, folyamatmodellező módszertanok és szoftverek irodalmát, illetve – nem hagyományos irodalom-áttekintésként – az elérhető szoftverek piacát. Mivel a mi értelmezésünkben a tudásterületeket a tevékenységekhez kötjük, melyeket különböző munkakörökhöz, majd ezeken keresztül szerepkörökhöz kötünk, ezért olyan folyamatmodellező eszközt kerestem, amely alkalmas ezen objektumok és tulajdonságok kezelésére. Végző választásom a BOC Group Adonis folyamatmodellező eszközére esett.
2. **Tudáskinyerés folyamatmodellekből:** A feladat elméleti háttérét az irodalom-áttekintés során megvizsgáltam. A kutatás célja és a kiválasztott folyamatmodellezési módszertan és szoftver együttese határozza meg a végleges információkinyerési metódust. Az Adonis folyamatmodellező eszköz esetén a

folyamatmodell kinyerhető strukturált módon, XML-ben, ami jó alapot biztosít a további tudáskinyeréshez.

3. **Tudásreprezentáció:** Erre a célra az ontológiát választottam, melynek elméleti alapjait áttekintettem az első részben. Folyamat- és szakterületi (*domain*) ontológia került megkülönböztetésre a kutatásban. A folyamatontológia létrehozására választott megoldás az Adonis XML export-jának XSLT szkripttel való átalakítása, illetve ezzel párhuzamosan a tevékenységek szöveges leírásán végzett szövegbányászat a tudáselemek kinyerése céljából.

A folyamat és szakterületi ontológia közötti kapcsolat létrehozása az Adonis-ban elérhető *information object*-en keresztül történt, azonban erről Török Mátyás írt részletesebben disszertációjában.

4. **Tudástranszfer és hasznosítási lehetőségek:** A hasznosítási lehetőségeket a gyakorlati eseten keresztül mutattam be.

- A hasznosítás egyik területe az emberi erőforrás-gazdálkodás, hiszen vállalati folyamatokra épülő kiválasztó és képzőrendszer hozható létre. A létrejövő ontológia adaptív tesztrendszerhez illesztve egyaránt alkalmas egy munkakör meghirdetésekor a jelentkezők szűrésére, és előrelépéskor, karriertervezéskor a hiányzó tudásterületek felderítésére és azok megtanítására is.
- Cél, de az előzőből kiadódó nyereségként is felfogható a szervezeti tudás artikulálása (feltárása) és hosszú-távú megőrzése (intellektuális vagyon kezelése).
- A hasznosítás további területe – és a dolgozatban ez kapott nagyobb hangsúlyt – a folyamatok fejlesztése, azok strukturális vizsgálatán túlmutató tudásalapú összehasonlító elemzésekre támaszkodva. A folyamatmodellek statikusan ugyan, de tartalmazzák a szervezet folyamatainak aktuális verzióit és a végrehajtásukhoz szükséges tudáselemeket – ezeket referencifolyamatokkal és sztenderdekkel összehasonlítva a szervezeti folyamatok fejleszthetőek.

5. **A változások kezelésének vizsgálata:** Felvázoltam a változások kezelésének lehetséges megoldását, az *information object* használatán keresztül. A megoldás ezáltal egy olyan eszközzé válik, amely képes a folyamatmodellbe visszacsatolást adni, a releváns tudáselemek elhelyezésén keresztül.

A disszertáció gyakorlati része során két eseten keresztül mutattam be a megoldást, melyet ilyen módon validáltam.

Bemutattam a megoldás használatát a felsőoktatás egyik folyamatának fejlesztési céllal történő elemzésén keresztül. Ennek során egy aktuális folyamat összehasonlítása történt meg a HEFOP projekt során létrehozott referencia folyamattal. Az eset során a megoldás még csak a folyamat struktúráját vizsgálta, de már azt is folyamatontológiák összehasonlításával. Az eset jól demonstrálta a kidolgozott megoldásból a folyamatmodell folyamatontológiává alakítását, majd az ontológia-összehasonlítás eszköztárának használhatóságát. Már ez az eset is bemutatta az összehasonlítás eredményeként létrejövő riportot, melynek értelmezése a folyamatgazda feladata.

A másik esetben a poszt-operatív kórházi ápolás folyamatát hasonlítottam össze egy sztenderd folyamattal. Ennek során, a korábbi esetben már ismertetett megoldást fejlesztettem tovább, melynek segítségével a folyamatontológia a szakterületi ontológia elemeivel dúsításra került, majd az elemzések is egy háromrétegű, tudásalapú vizsgálaton keresztül jellemezték mind a folyamatot, mind annak szereplőit.

A kutatás illeszkedik az Információrendszerek Tanszéken zajló kutatások sorába, fő jelentősége abban rejlik, hogy az eddigi, főként statikusként használt folyamatmodelleket dinamikussá teszi, hiszen a kutatásban felvázolt rendszer a folyamatmodellezésben a dinamikát hangsúlyozza azáltal, hogy a folyamatmodellezés célját kiterjeszti. A folyamatmodell az alapját képezi a rendszernek, ezért a folyamatmodell változásával a rá épülő rendszer is változni képes. Emellett, mivel a fő cél a szervezeti tudásvagyon azonosítása a folyamatmodellekben, ezért a vállalat változásait a folyamatmodelleken keresztül azonosítva dinamikusan tudja szabályozni a megfelelő munkaerő kiválasztását.

Az ontológia használatával a felvázolt rendszer bármilyen szakterületen hasznosítható, hiszen a folyamatmodell „cserélhető”, de a funkcionalitása ugyanúgy elérhető marad.

4.2 Továbbfejlesztési lehetőségek

Technológiai szinten a folyamatmodell átalakítása folyamatontológiává mindenképpen fejlesztésre szorul. A folyamatmodellekben lévő beágyazott folyamatok kezelése jelenleg nem megoldott. A folyamatláncok automatikus felismerése és

kezelése (cross-reference objektumokon keresztül) szintén további fejlesztés eredménye lehet.

Az összehasonlítás során erőteljesen támaszkodtunk a Protege beépített eszközszerére, majd ezen túllépve egy önálló Java alapú programmal végeztettük az ellenőrzést. Ennek a programnak több paramétere „beégetett”, ennek megfelelően hordozhatóság, és más folyamatontológiáknál való hasznosíthatósága programozással jár. Ennek kiküszöbölése szintén további feladat.

A folyamatontológia és a szakterületi ontológia egymásra vetítésére (mapping) egy ontológia-építő megoldás fejlesztése javasolt.

A kutatás utóéletének fontos pontja további valós körülmények közötti teszt esetek vizsgálata. Emellett, indokolt lenne folyamatgazdákkal további megbeszéléseket tartani, hogy számukra releváns folyamatfejlesztési kérdésekre vonatkozó vizsgálatokra készítsük fel a felvázolt megoldást.

Mindenképp fontos továbbfejlesztési területe a megoldásnak és a kutatásnak is, a referencia folyamatok, sztenderdek, best practice-ek bekerülésének és leképezésének vizsgálata. Közhely, hogy az interneten bármi elérhető, de a megbízható, releváns és jól strukturált, ezáltal jól feldolgozható információforrások megtalálása nem triviális. Az open data és linked open data olyan vizsgálati területeket nyitnak meg, melyek ezen a dolgozaton még túlmutattak, de a jövőbeli kutatásokhoz nagyon jó terepet adnak.

5 Forrásjegyzék

1. **van der Aalst, W. M. P.; ter Hofstede, A. H. M.; Weske, M. (2003):** Business process management: A survey, In Proceedings of Business Process Management: International Conference, BPM 2003, June 26-27, Eindhoven, the Netherlands, ser. LNCS. Springer, 2003, pp. 1–12.
http://dx.doi.org/10.1007/3-540-44895-0_1
2. **Abramowicz, W.; Filipowska, A.; Kaczmarek, M.; Kaczmarek, T. (2007):** Semantically enhanced Business Process Modelling Notation, In Workshop on Semantic Business Process and Product Lifecycle Management (SBPM), volume 251 of CEUR Workshop Proceedings, pages 88–91, Innsbruck, Austria, June 2007.
3. **Alexander, M. Fawcett, J., Runciman, P. (1994):** Nursing Practice: Hospital and home. The adult. London: Churchill Livingstone
4. **Alves de Medeiros, A.K.; van der Aalst, W.M.P. (2009):** Process Mining Towards Semantics, in Advances in Web Semantics I, Lecture Notes in Computer Science Volume 4891, pp 35-80
http://dx.doi.org/10.1007/978-3-540-89784-2_3
5. **Alves de Medeiros, A. K.; Pedrinaci, C.; van der Aalst, W. M. P.; Domingue, J.; Song, M.; Rozinat, A.; Norton, B.; Cabral, L. (2007):** An outlook on semantic business process mining and monitoring, In On the Move to Meaningful Internet Systems 2007: OTM 2007 Workshops, volume 4806 of LNCS, pages 1244–1255, Vilamoura, Portugal, November 2007.
6. **APQC (2012):** Process Classification Framework (PCF) 6.0.0.
<http://www.appc.org/knowledge-base/documents/apqc-process-classification-framework-pcf-cross-industry-excel-version-600>
Letöltve: 2013. március 1.
7. **Arkin, A.; Askary, S.; Bloch, B.; Curbera, F.; Golland, Y.; Kartha, N.; Liu, C. K.; Thatte, S.; Yendluri, P.; Yiu, A. (2005):** Web services business process execution language version 2.0. wsbpel-specification-draft-01, OASIS, September 2005.
8. **Babbie, E. (2003):** A társadalomtudományi kutatás gyakorlata, 6. átd. kiad., Balassi Kiadó, Budapest
9. **Bácsfalvi, M.; Bordáné dr. Rabózki, M.; Csengői, Cs.; Gyökér, I.; Horváth, E.; Kirchknopf, Gy.; Kiss, F.; Kocsis, J.; Koczor, Z.; Maczó, K.; Németh, B.; Pákozdi, S.; Papp, O.; Pataki, B.; Romhányi, G.; Sándor, L.; Véry, Z. (2001):** Controlling a gyakorlatban
<http://www.tankonyvtar.hu/gazdasagtudomany/controlling-gyakorlatban-080904-404>
Letöltve: 2013. február 12.
10. **Balaton, K.; Dobák, M. (1991):** Mennyiségi és minőségi módszerek az empirikus szervezatkutatásban, In: Antal-Mokos, Z.; Drótos, Gy.; Kovács, S.

(szerk.): Módszertani gyűjtemény a vezetés és szervezés tárgyhoz, Aula Kiadó, Budapest

11. **Battle, S. (2004):** Round-tripping between XML and RDF, in: International Semantic Web Conference (ISWC), Hiroshima, Japan, November 2004. Springer
12. **Belecheanu, R.; Cabral, L.; Domingue, J.; Gaaloul, W.; Hepp, M.; Filipowska, A.; Kaczmarek, M.; Kaczmarek, T.; Nitzsche, J.; Norton, B.; Pedrinaci, C.; Roman, D.; Stollberg, M.; Stein, S. (2007):** Business Process Ontology Framework. SUPER research project deliverable D1.1. 2007.
<http://www.ip-super.org/res/Deliverables/M12/D1.1.pdf>
Letöltve: 2013. február 12.
13. **Benbasat, I.; Goldstein, D. K.; Mead, M. (1987):** The Case Research Strategy in Studies of Information Research, MIS Quarterly, 11(3), September, pp.369-386.
14. **Benjamins, V. R.; Nunes de Barros, L.; Valente, A. (1996):** Constructing Planners through Problem-Solving Methods, in Proceedings of KAW'96 (Banff), pp. 14.1-14.20, 1996
15. **Betz, S.; Kling, S.; Koschmider, A.; Oberweis, A. (2006):** Automatic User Support for Business Process Modeling. In: Hinkelmann, K.; Karagiannis, D.; Stojanovic, N.; Wagner, G. (eds.): Proceeding of the Workshop on Semantics for Business Process Management at the 3rd European Semantic Web Conference 2006, Budva, Montenegro, June 2006
16. **Bichler, P.; Preuner, G.; Schrefl, M. (1997):** Workflow Transparency, in Advanced Information Systems Engineering, in Proceedings of the 9th International Conference (CAiSE '97), Barcelona, Spain, pp. 423 – 436, Springer Verlag, June 1997
17. **Boda, I. (2006):** A menedzsment fogalma.
http://www.inf.unideb.hu/~bodai/menedzs/menedzsment_fogalma.html
Letöltve: 2012. december 29.
18. **Bohring, H., Auer, S. (2005):** Mapping XML to OWL Ontologies. Leipziger Informatik-Tage, volume 72 of LNI, pp. 147-156
19. **Boyatzis, R. E. (1982):** The Competent Manager: A model for effective performance. New York: Wiley
20. **Brauer, W.; Reisig, W. (1986):** Petri Nets: Central Models and Their Properties, Lecture Notes in Computer Science, Vol. 254: Advances in Petri Nets 1986, Part I, proceedings of an Advanced Course, Bad Honnef, Springer-Verlag, September 1986
21. **Brockmans, S.; Ehrig, M.; Koschmider, A.; Oberweis, A.; Studer, R. (2006):** Semantic Alignment of Business Processes, ICEIS, Paphos, Cyprus, May.
22. **Buitelaar, P.; Cimiano, P.; Magnini, B. (2005):** Ontology learning from text: An overview, in: Ontology Learning from Text: Methods, Evaluation and Applications, P. Buitelaar, P. Cimimiano, and B. Magnini eds.; IOS Press, Amsterdam
23. **Business Process Management initiative [BPMI] (2004):** Business Process Modeling Notation (BPMN) Version 1.0

http://www.omg.org/bpmn/Documents/BPMN_V1-0_May_3_2004.pdf

Letöltve: 2013. április 6.

24. **Celino, I.; Alves de Medeiros, A. K.; Zeissler, G.; Oppitz, M.; Facca, F.; Zöller, S. (2007):** Semantic Business Process Analysis. In Workshop on Semantic Business Process and Product Lifecycle Management (SBPM), volume 251 of CEUR Workshop Proceedings, pages 44–47, Innsbruck, Austria, June 2007.
25. **Coleman, P. (2007):** How Do I Choose the Best Business Process Modeling Software?
<http://www.wisegeek.com/how-do-i-choose-the-best-business-process-modeling-software.htm>
Letöltve: 2013. február 26.
26. **Davenport, T. H. (1993):** Process Innovation: Reengineering Work Through Information Technology, Harvard Business School Press, Cambridge, MA
27. **Davies, J.; Fensel, D.; Van Harmelen, F. (2003):** In Towards the Semantic Web Ontology-driven Knowledge Management. Sussex: John Wiley and Sons Ltd., 2003.
28. **Decker, S.; Melnik, S.; van Harmelen, F.; Fensel, D.; Klein, M. C. A.; Broekstra, J.; Erdmann, M.; Horrocks, I. (2000):** The Semantic Web: The Roles of XML and RDF, in IEEE Internet Computing, 4(5):63–74.
29. **Dimitrov, M.; Simov, A.; Stein, S.; Konstantinov, M. (2007):** A BPMO Based Semantic Business Process Modelling Environment. In M. Hepp, K. Hinkelmann, D. Karagiannis, R. Klein, and N. Stojanovic, editors, Workshop on Semantic Business Process and Product Lifecycle Management (SBPM), volume 251 of CEUR Workshop Proceedings, pages 101–104, Innsbruck, Austria, June 2007.
30. **Eisenhardt, K. M. (1989):** Building Theories from Case Study Research, Academy of Management Review, Vol. 14, No. 4, pp. 532-550.
31. **El Kharbili, M.; Stein, S.; Markovic, I.; Pulvermüller, E. (2008):** Towards a Framework for Semantic Business Process Compliance Management. In Sadiq, S.; Indulska, M. & zur Muehlen, M. (ed.) Proceedings of the workshop on Governance, Risk and Compliance for Information Systems (GRCIS 2008), CEUR Workshop Proceedings, Montpellier, France, June 2008, 339, pp. 1-15.
32. **El Kharbili, M.; Pulvermüller, E. (2009):** A Semantic Framework for Compliance Management in Business Process Management, In Business Process and Services Computing (BPSC 2009), Leipzig, Germany, pp. 60-80.
33. **Fehér, P. (2004a):** Munkafolyamat (Workflow) Menedzsment. BCE Információrendszerek Tanszék
34. **Fehér, P. (2004b):** Tudásmenedzsmentet támogató tényezők szerepe szoftverfejlesztő szervezetekben. Doktori (PhD) értekezés, Budapesti Corvinus Egyetem, Gazdálkodástani Doktori Iskola.
35. **Fensel, D.; Lausen, H.; Polleres, A.; de Bruijn, J.; Stollberg, M.; Roman, D.; Domingue, J. (2006):** Enabling Semantic Web Services: The Web Service Modeling Ontology. Springer, 2006.

36. **Filipowska, A.; Haller, A.; Kaczmarek, M.; van Lessen, T.; Nitzsche, J.; Norton, B. (2007):** Process ontology language and operational semantics for semantic business processes. Technical report, Integrated project SUPER, 2007.
37. **Filipowska, A.; Kaczmarek, M.; Stein, S. (2009):** Semantically Annotated EPC within Semantic Business Process Management
38. **Gábor, A.; Szabó, Z. (2009):** Knowledge based institutional capacity building in the Hungarian Higher Education. In: Berdai, A., Sekhari, A. (eds.) The Proceeding of the International Conference on Software, Knowledge and Information Management and Applications (SKIMA 2009), Fez, Morocco, October 21-23, pp. 78–85 (2009) ISBN: 9781851432516
39. **Gábor, A.; Szabó, Z. (2013):** Semantic Technologies in Business Process Management, in: Integration of Practice-Oriented Knowledge Technology: Trends and Perspectives 2013, pp 17-28, Springer
40. **Galliers, R. (1992):** Information Systems Research. Issues, Methodes and Practical Guidelines. Alfred Waller Ltd., Henley-on-Thames.
41. **Gartner (2010):** Magic Quadrant for Business Process Management Suites
<http://www.images.adobe.com/www.adobe.com/content/dam/Adobe/en/enterprise/pdfs/magic-quadrant-for-business-process-management-suites.pdf>
 Letöltve: 2013. június 26.
42. **Gartner (2012):** Business Process Management Definition
<http://www.gartner.com/it-glossary/business-process-management-bpm/>
 Letöltve: 2013. február 26.
43. **Gillani, S.; Kó, A (2014):** Process-based Knowledge Extraction in a Public Authority: A Text Mining Approach, in A. Kó and E. Francesconi (Eds.): EGOVIS 2014, LNCS 8650, pp. 91–103, Springer International Publishing Switzerland 2014.
44. **Gómez-Pérez, A.; Manzano-Macho, D. (2003):** A survey of ontology learning methods and techniques (1.5), Technical report, OntoWeb Consortium
45. **Gruber, T. R. (1993):** A Translation Approach to Portable Ontology Specifications, in Knowledge Acquisition, Vol. 5(2), pp. 199-220, Academic Press, Stanford University, USA, 1993
46. **Guarino N. (1998):** Formal ontology and information systems. In Nicola Guarino, editor, Formal Ontology in Information Systems - Proceedings of FOIS'98, Trento, Italy, pages 3-15. IOS Press, Amsterdam, 1998.
47. **Guinness, D.; Harmelen, F. (2004):** OWL Web Ontology Language - Overview, February 2004, W3C Recommendation.
48. **Hammer, M.; Champy, J. (2000):** Vállalatok újrászervezése. Panem Könyvkiadó, Budapest
49. **Hepp, M.; Leymann, F.; Domingue, J.; Wahler, A.; Fensel, D. (2005):** Semantic business process management: A vision towards using semantic web services for business process management. In Proceedings of IEEE Intl. Conf. on e-Business Engineering (ICEBE 2005), pages 535-540. IEEE Computer Society

50. **Hepp, M.; Roman, D. (2007):** An ontology framework for semantic business process management, In Proceedings of Wirtschaftsinformatik 2007, Karlsruhe, Germany, pp. 1-18.
51. **Herborn, T.; Wimmer, M. A. (2006):** Process Ontologies Facilitating Interoperability in eGovernment - A Methodological Framework. In: Hinkelmann, K.; Karagiannis, D.; Stojanovic, N.; Wagner, G. (eds.): Proceeding of the Workshop on Semantics for Business Process Management at the 3rd European Semantic Web Conference 2006, Budva, Montenegro, June 2006, pp. 76–89
52. **Jenz, D. E. (2003):** Business Process Ontologies: Frequently Asked Questions
<http://www.bptrends.com/publicationfiles/09-03%20WP%20BP%20Ontologies%20FAQ%20.pdf>
 Letöltve: 2013. május 26.
53. **Karagiannis, D. (2008):** A Business process Based Modelling Extension for Regulatory Compliance. Proceedings of the Multikonferenz Wirtschaftsinformatik 2008, Munich.
54. **Kismihók, G.; Kovács, B.; Vas, R.F. (2009):** Integrating Ontology-based Content Management into a Mobilized Learning Environment. In: Caballé, S., Xhafa, F., Daradoumis, T., Juan, A.A. (eds.) Architectures for Distributed and Complex MLearning Systems: Applying Intelligent Technologies, pp. 192–210. IGI Global, Hershey
55. **Kismihók, G.; Mol, S. (2011):** The OntoHR project: Bridging the gap between vocational education and the workplace. In: Book of Abstracts, EAWOP 2011, pp. 194–195. University of Maastricht
56. **Kismihók, G. (2012):** Rugalmas tanulás, rugalmas munkavégzés. Az ontológia alapú tartalommenedzsment lehetőségeinek kiaknázása, Doktori (PhD) értekezés, Budapesti Corvinus Egyetem, Gazdálkodástani Doktori Iskola.
57. **Klemp, G. O.; McClelland D. C. (1986):** What characterizes intelligent functioning among senior managers? Practical Intelligence. Cambridge: University Press
58. **Klimkó, G. (2001):** A szervezeti tudás feltérképezése. Doktori (PhD) értekezés, Budapesti Közgazdaságtudományi és Államigazgatási Egyetem, Gazdálkodástani Ph.D program
59. **Koschmider, A.; Oberweis, A. (2008):** Modeling semantic business process models, In Rittgen, P. (Ed.), Handbook of Ontologies for Business Interaction, Idea Group, Harpenden.
60. **Kovács, B. (2008):** Improving Content Management - a Semantic Approach. Acta Cybernetica 4, 579–593
61. **Kovács, B. (2010):** Az információ túlterhelés csökkentése szervezeti munkafolyamat-rendszerekben, Doktori (PhD) értekezés, Budapesti Corvinus Egyetem, Gazdálkodástani Doktori Iskola.
62. **Kó, A. (2004):** Az információtechnológia szerepe és lehetőségei a tudásmenedzsmentben: Az ontológiaépítés, mint a tudásmenedzsment eszköze. Doktori (PhD) értekezés, Budapesti Corvinus Egyetem, Gazdálkodástani Ph.D program

63. **Kő, A.; Gábor, A.; Kovács, B. (2011):** Agile Knowledge-Based E-Government Supported By Sake System. *Journal of Cases on Information Technology (JCIT)* 3, 1–20
64. **Kő, A.; Ternai, K. (2011):** A Development Method for Ontology Based Business Processes. In: *eChallenges e-2011 Conference*, October 26-28, 2011
65. **Kramler, G., Murzek, M. (2006):** Business Process Model Transformation Issues
66. **Lautenbacher, F.; Bauer, B. (2006):** Semantic Reference- and Business Process Modeling enables an Automatic Synthesis. In: Hinkelmann, K.; Karagiannis, D.; Stojanovic, N.; Wagner, G. (eds.): *Proceeding of the Workshop on Semantics for Business Process Management at the 3rd European Semantic Web Conference 2006*, Budva, Montenegro, June 2006, pp. 89–100
67. **Lautenbacher, F.; Bauer, B.; Seitz, C. (2008):** Semantic Business Process Modeling - Benefits and Capability. In *Proceedings of AAAI Spring Symposium: AI Meets Business Rules and Process Management*. 2008, 71-76.
68. **Lee, A. S. (1989):** A Scientific Methodology for MIS Case Studies, In: *MIS Quarterly* 13(1), March, pp.33-52.
69. **Liker J. K. (2008):** A Toyota módszer - 14 vállalatirányítási alapelv. Budapest, HVG Kiadó Zrt.
70. **Ly, L. T.; Rinderle-Ma, S.; Dadam, P. (2007):** Integration and verification of semantic constraints in adaptive process management systems. *Data & Knowledge Engineering*, 64, 3–23.
71. **Ly, L. T.; Göser, K.; Rinderle-Ma, S.; Dadam, P. (2008):** Compliance of Semantic Constraints – A Requirements Analysis for Process Management Systems. In Sadiq, S.; Indulska, M. & zur Muehlen, M. (ed.). *Proceedings of the workshop on Governance, Risk and Compliance for Information Systems (GRCIS 2008)*, CEUR Workshop Proceedings, Montpellier, France, June 2008, 339, 31-45.
72. **Ly, L. T.; Rinderle-Ma, S.; Göser, K.; Dadam, P. (2009):** On Enabling Integrated Process Compliance with Semantic Constraints in Process Management Systems. *Information Systems Frontiers*. pp. 1-25. ISSN 1387-3326
73. **Melnik, S. (1999):** Bridging the gap between XML and RDF
<http://www.db.stanford.edu/melnik/rdf/fusion.html>
Letöltve: 2013. március 9.
74. **Namiri, K.; Stojanovic., N. (2007):** A Formal Approach for Internal Controls Compliance in Business Processes. 8th Workshop on Business Process Modeling, Development, and Support (BPMDS07). Trondheim, Norway.
75. **Németh, B. (2008):** Folyamatmenedzsment megvalósítása a vállalati gyakorlatban. *Minőség és megbízhatóság*, 42. évf. 1.sz. 27-31. oldal
76. **Nitzsche, J.; Wutke, D.; van Lessen, T. (2007):** An Ontology for Executable Business Processes. In *Workshop on Semantic Business Process and Product Lifecycle Management (SBPM)*, volume 251 of *CEUR Workshop Proceedings*, pages 52–63, Innsbruck, Austria, June 2007.

77. **Object Management Group [OMG] (2010):** Unified Modeling Language (OMG UML), Infrastructure - Version 2.4
<http://www.omg.org/spec/UML/2.4/Infrastructure/Beta2/PDF/>
 Letöltve: 2013. április 20.
78. **Object Management Group [OMG] (2011):** Business Process Model and Notation (BPMN) Version 2.0
<http://www.omg.org/spec/BPMN/2.0/PDF/>
 Letöltve: 2013. április 6.
79. **Organization for the Advancement of Structured Information Standards [OASIS] (2007):** Web Services Business Process Execution Language Version 2.0 - OASIS Standard
<http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html>
 Letöltve: 2013. április 13.
80. **Oro, E.; Ruffolo, M. (2012):** A Knowledge Representation Formalism for Semantic Business Process Management, In Advances in Knowledge Representation
81. **OWLDiff (2008):** OWL Diff Documentation
<http://krizik.felk.cvut.cz/km/owldiff/documentation.html>
 Letöltve: 2014. április 13.
82. **Pease, A. (1998):** The Warplan: A Method Independent Plan Schema, in L. de Barros, R. Benjamins, Y. Shahar, A. Tate, A. Valente (eds), in Proceedings of the AIPS 1998, Workshop on Knowledge Engineering and Acquisition for Planning, AAAI Technical, Report WS-98-03, 1998
83. **Raffai, M. (1999):** BPR – Üzleti folyamatok újjászervezése. Novadat, Budapest
84. **Royle, J., Walsh, M. (1992):** Watson's Medical Surgical Nursing and Related Physiology. London: Bailliere Tindall
85. **Reisig, W.; Rozenberg, G. (1998):** Lectures on Petri Nets: Basic Models. Lecture Notes in Computer Science. Springer, 1 edition, 1998.
86. **Sabou, M.; Wroe, C.; Goble, C.; and Mishne, G. (2005):** Learning Domain Ontologies for Web Service Descriptions: an Experiment in Bioinformatics, in: Proceedings of the 14th International World Wide Web Conference (WWW2005), Chiba, Japan
87. **Sadiq S.; Governatori G.; Namiri K. (2007):** Modeling Control Objectives for Business Process Compliance. Proceedings of the 5th International Conference, BPM 2007 (pp.149-164). Brisbane, Springer.
88. **Scheer, A.-W. (1997):** ARIS – House of Business Engineering: Konzept zur Beschreibung und Ausführung von Referenzmodellen. In: Becker, J.; Rosemann, M.; Schütte, R. (eds.): Entwicklungsstand und Entwicklungsperspektiven der Referenzmodellierung : Proceedings zur Veranstaltung vom 10. März 1997. Münster : Institut für Wirtschaftsinformatik, Westfälische Wilhelms-Universität, 1997, pp. 3–15 (in German)

89. **Scheer, A.-W. (1999):** ARIS – business process modeling. 2nd ed. Berlin : Springer
90. **Scheer, A.-W.; Nüttgens, M. (2000):** ARIS Architecture and Reference Models for Business Process Management. In Business Process Management, Models, Techniques, and Empirical Studies, volume 1806, pages 376–389. Springer, 2000.
91. **Scheer, A.-W.; Abolhassan, F.; Jost, W.; Kirchmer M. (2002):** Business Process Excellence - ARIS in Practice, Springer
92. **Schmidt, R.; Bartsch, C.; Oberhauser, R. (2007):** Ontology-based representation of compliance requirements for service processes. Proceedings of the Workshop on Semantic Business Process and Product Lifecycle Management (SBPM 2007).
93. **Shamsfard, M.; Barforoush, A. A. (2003):** The state of the art in ontology learning: A framework for comparison, in: The Knowledge Engineering Review, Vol. 18 No.4 pp. 293- 316.
94. **Staud, J. (2001):** Geschäftsprozessanalyse: Ereignisgesteuerte Prozessketten und objektorientierte Geschäftsprozessmodellierung für Betriebswirtschaftliche Standardsoftware, Berlin, Springer, 2001, ISBN 3-540-41461-4
95. **Szabó, I. (2011):** Comparing The Competence Contents of Demand and Supply Sides on the Labour Market. In: 33rd International Conference on Information Technology Interfaces, Cavtat, Croatia
96. **Szabó, Z. (2000):** A szervezeti információfeldolgozás strukturális és technológiai tényezőinek összerendelése. Doktori (PhD) értekezés, Budapesti Közgazdaságtudományi és Államigazgatási Egyetem, Gazdálkodástani Ph.D program
97. **Szabó, Z. (2009):** Business Process Management, 28. dia, Előadás anyag
98. **Szabó, Z. (2010):** Business Process Management. Oktatási anyag a BCE Folyamatmenedzsment kurzusán, 2010. szeptember-december
99. **Szabó, Z. (2012):** Folyamatmenedzsment, BCE Információrendszerek tanszék Oktatási segédanyag, Budapest
100. **SZKTerv (2012):** A következő lépés Széll Kálmán terv 2.0
http://index.hu/assets/documents/belfold/szkt_2_0.pdf
Letöltve: 2014. április 6.
101. **Tanrikorur, T. (2007)** Business Process Management 101: The Basics of BPM and How to Choose the Right Suite
<http://www.informationweek.com/software/business-intelligence/business-process-management-101-the-basi/199204260>
Letöltve: 2013. március 26.
102. **Tate, A. (1998):** Roots of SPAR - Shared Planning and Activity Representation, The Knowledge Engineering Review, Vol. 13(1), pp. 121-128, Special Issue on "Putting Ontologies to Use" (eds. Uschold, M. and Tate, A.), Cambridge University Press, 1998.

103. **Tenner, A. R.; deToro, I. J. (1996):** Process Redesign: The Implementation Guide for Managers. Addison-Wesley
104. **Ternai, K. (2009):** Az ERP rendszerek metamorfózisa, Doktori (PhD) értekezés, Budapesti Corvinus Egyetem, Gazdálkodástani Ph.D program
105. **Ternai, K. (2011):** A New Approach in the Development of Ontology Based Workflow Architectures. In: 17th International Conference on Concurrent Enterprising, Aachen, Germany, June 20-22, 2011
106. **Ternai, K.; Szabó, I.; Varga, K. (2013):** Ontology-Based Compliance Checking on Higher Education Processes, in: Kő, A. et al. (Eds.): EGOVIS/EDEM 2013, LNCS 8061, pp. 58--71. Springer, Heidelberg
107. **Ternai, K.; Török, M. (2011):** Semantic modeling for automated workflow software generation – An open model. In: 5th International Conference on Software, Knowledge Information, Industrial Management and Applications (SKIMA 2011), Benevento, Italy, September 8-11, 2011
108. **Thomas, O.; Fellmann, M. (2006):** Semantic Event-driven Process Chains, In: Hinkelmann, K.; Karagiannis, D.; Stojanovic, N.; Wagner, G. (eds.): Proceeding of the Workshop on Semantics for Business Process Management at the 3rd European Semantic Web Conference 2006, Budva, Montenegro, June 2006
109. **Tóthné Kiss, A. (2012):** Competence-based Operation in Human Processes of Companies, in: Club of Economics in Miskolc, TMP Vol. 8., Nr. 1., pp. 96-100.
110. **Vári, A. (ism.):** Folyamatmenedzsment
<http://www.fitt.ifua.hu/?levelid=2&cikkid=10>
 Letöltve: 2013. április 6.
111. **Vas, R. F. (2007):** Tudásfelmérést támogató oktatási ontológia szerepe és alkalmazási lehetőségei. Doktori (PhD) értekezés, Budapesti Corvinus Egyetem, Gazdálkodástani Doktori Iskola
112. **Vas, R. F.; Kovács, B. (2008):** Ontology-based Content Management Systems in Public Administration. In: Schweighofer, E. (ed.) Legal Informatics. The LEFIS Series, vol. 2, pp. 45–62. Prensas Universitarias de Zaragoza, Zaragoza
113. **Vas, R. F.; Kovács, B.; Kismihók, G. (2009):** Ontology-based Mobile Learning and Knowledge Testing. International Journal of Mobile Learning and Organization 2, 128–147
114. **Weber, I. (2007):** Requirements for the Implementation of Business Process Models through Composition of Semantic Web Services, in: Proceedings of the 3rd International Conference on Interoperability for Enterprise Software and Applications (I-ESA) March 2007, Funchal, Portugal
115. **WebFinance, Inc. (2013):** What is a business process model? Definition and meaning
<http://www.businessdictionary.com/definition/Business-Process-Model.html>
 Letöltve: 2013. március 26.
116. **Wesner, J. W.; Hiatt, J. M.; Trimble, D. C. (1994):** Winning with Quality: Applying Quality Principles in Product Development (Engineering Process Improvement). Addison-Wesley

117. **White, S. A (2004):** Business Process Modeling Notation. Specification, BPMI.org, 2004.
118. **Wong, W.; Liu, W.; Bennamoun, M. (2012):** Ontology learning from text: A look back and into the future, in: ACM Comput. Surv. 44, 4, Article 20
119. **Woodruffe, C. (1993):** What is meant by a competency? Leadership and Organization.Development Journal, 14:1, 29-36
120. **Workflow Management Coalition [WfMC] (1999):** Terminology & Glossary, Document Number WFMCTC-1011, Document Status - Issue 3.0, Feb 99, http://www.wfmc.org/standards/docs/TC-1011_term_glossary_v3.pdf
Letöltve: 2012. december 29.
121. **Workflow Management Coalition [WfMC] (2012):** Process Definition Interface - XML Process Definition Language
<http://www.xpdl.org/standards/xpdl-2.2/XPDL%202.2%20%282012-08-30%29.pdf>
Letöltve: 2013. április 13.
122. **World Wide Web Consortium [W3C] (2004a):** Resource Description Framework (RDF)
<http://www.w3.org/RDF/>
Letöltve: 2013. április 7.
123. **World Wide Web Consortium [W3C] (2004b):** OWL Web Ontology Language - Overview
<http://www.w3.org/TR/owl-features/>
Letöltve: 2013. április 14.
124. **World Wide Web Consortium [W3C] (2006):** Extensible Markup Language (XML) 1.1 (Second Edition)
<http://www.w3.org/TR/xml11/>
Letöltve: 2013. április 7.
125. **World Wide Web Consortium [W3C] (2007):** XSL Transformations (XSLT) Version 2.0
<http://www.w3.org/TR/xslt20/>
Letöltve: 2013. április 7.
126. **World Wide Web Consortium [W3C] (2008):** Extensible Markup Language (XML) 1.0 (Fifth Edition)
<http://www.w3.org/TR/xml/>
Letöltve: 2013. április 7.
127. **World Wide Web Consortium [W3C] (2012):** OWL 2 Web Ontology Language - Document Overview (Second Edition)
<http://www.w3.org/TR/owl2-overview/>
Letöltve: 2013. április 14.

128. **Yin, R. K. (1994):** Case Study Research (2nd edition), Sage Publications, London, UK
129. **Zhou, L. (2007):** Ontology learning: state of the art and open issues, in: Information Technology and Management, 8(3), pp. 241–252.

6 Idegen szavak és rövidítések jegyzéke

Rövidítés	Angol megfelelő	Magyar megfelelő
APQC	American Productivity and Quality Center	Amerikai Teljesítmény- és Minőségügyi Központ
BPE	Business Process Excellence	Üzleti folyamatkiválóság
BPM	Business Process Management	Üzleti folyamatmenedzsment
BPMI	Business Process Management Initiative	Üzleti Folyamatmenedzsment Kezdeményezés
BPMN	Business Process Model and Notation	Üzleti folyamatmodell és jelölésrendszer
BPR	Business Process Reengineering / Redesign	Folyamatok radikális átalakítása
CPI	Continuous Process Improvement	Állandó folyamatfejlesztés
EPC	Event-driven Process Chain	Eseményvezérelt folyamatlánc
ERP	Enterprise Resource Planning	Integrált vállalatirányítási rendszer
eTOM	Enhanced Telecom Operations Map	Kiterjesztett telekom működési térkép
HRM	Human Resources' Management	Emberi erőforrás menedzsment
IDEF	ICAM Definition / Integration DEFinition	ICAM Definíció / Integrált Definíció
IT	Information Technology	információtechnológia
ITIL	IT Infrastructure Library	IT infrastruktúra menedzsment
KPI	Key Performance Indicators	Kulcs teljesítménymutatók
OMG	Object Management Group	Objektum Menedzsment Csoport
OWL	Web Ontology Language	Webes Ontológia Nyelv
PMLC	Process Management LifeCycle	Folyamatmenedzsment életciklus
RDF	Resource Description Framework	Erőforrás-leíró Keretrendszer
SOA	Service Oriented Architecture	Szolgáltatásorientált Architektúra
SWS	Semantic Web Service	Szemantikus webszolgáltatás
UML	Unified Modeling Language	Egységes Modellező Nyelv
WfMC	Workflow Management Coalition	Workflow Menedzsment Koalíció
WS-BPEL	Web Services Business Process Execution Language	Webszolgáltatások üzleti folyamatvégrehajtó nyelve
WSMO	Web Service Modeling Ontology	Webszolgáltatás-modellező Ontológia
XML	eXtensible Markup Language	Kiterjeszhető Jelölő Nyelv
XPDL	XML Process Definition Language	XML Folyamatleíró Nyelv
XSLT	eXtensible Stylesheet Language Transformations	Kiterjeszhető Stíluslap-átalakítások

7 Mellékletek

7.1 Az Adonis XML értelmezése, a számunkra releváns tag-ekkel

Folyamatmodell		
Name	folyamat	<MODEL id="mod.25204" name="folyamat" version="" modeltype="Business process model" libtype="bp" applib="ADONIS:CE BPMS BP Library 2.0">
Objektumok		
Trigger		
Name	Trigger	<INSTANCE id="obj.25206" class="Trigger" name="Trigger">
Process start		
Name	START	<INSTANCE id="obj.25209" class="Process start" name="START">
Activity		
Name	Tevékenység	<INSTANCE id="obj.25212" class="Activity" name="Tevékenység">
Description	A tevékenység leírása, itt lehet fontos dolog is. De ezt már erőteljesen szövegbányászni kell, mert nem struktúrált szöveg.	<ATTRIBUTE name="Description" type="STRING">A tevékenység leírása, itt lehet fontos dolog is. De ezt már erőteljesen szövegbányászni kell, mert nem struktúrált szöveg.</ATTRIBUTE>
Responsible role	Responsible User (Szervezet-ről)	<INTERREF name="Responsible role"><IREF type="objectreference" tmodeltype="Working environment model" tmodelName="Szerv" tmodelver="" tclassname="Role" tobjname="Responsible User"></IREF></INTERREF>
Display responsible role	Yes	<ATTRIBUTE name="Display responsible role" type="ENUMERATION">Yes</ATTRIBUTE>
Responsible for execution	Responsible User (Szerv-ről)	<INTERREF name="Responsible for execution"><IREF type="objectreference" tmodeltype="Working environment model" tmodelName="Szerv" tmodelver="" tclassname="Role" tobjname="Responsible User"></IREF></INTERREF>
Accountable for approving results	Accountable User (Szerv-ről)	<INTERREF name="Accountable for approving results"><IREF type="objectreference" tmodeltype="Working environment model" tmodelName="Szerv" tmodelver="" tclassname="Role" tobjname="Accountable User"></IREF></INTERREF>
Cooperation/participation	Cooperation User (Szerv-ről)	<INTERREF name="Cooperation/participation"><IREF type="objectreference" tmodeltype="Working environment model" tmodelName="Szerv" tmodelver="" tclassname="Role" tobjname="Cooperation User"></IREF></INTERREF>

To inform	Informed User (Szerv-ről)	<INTERREF name="To inform"><IREF type="objectreference" tmodeltype="Working environment model" tmodelname="Szerv" tmodelver="" tclassname="Role" tobjname="Informed User"></IREF></INTERREF>
Input	Input data (I/O-ról)	<INTERREF name="Input"><IREF type="objectreference" tmodeltype="Document model" tmodelname="I/O" tmodelver="" tclassname="Document" tobjname="Input" data"></IREF></INTERREF>
Output	Output data (I/O-ról)	<INTERREF name="Output"><IREF type="objectreference" tmodeltype="Document model" tmodelname="I/O" tmodelver="" tclassname="Document" tobjname="Output" data"></IREF></INTERREF>
Referenced IT system elements	IT application (IT-ról)	<INTERREF name="Referenced IT system elements"><IREF type="objectreference" tmodeltype="IT system model" tmodelname="IT" tmodelver="" tclassname="Application" tobjname="IT application"></IREF></INTERREF>
Visualise referenced IT system elements	Yes	<ATTRIBUTE name="Visualise referenced IT system elements" type="INTEGER">1</ATTRIBUTE>
End		
Name	VÉGE	<INSTANCE id="obj.25215" class="End" name="VÉGE">
Nyilak		
Subsequent	Nyíl a Triggertől a Process Start-ig	<CONNECTOR id="con.25218" class="Subsequent"><FROM instance="Trigger" class="Trigger"></FROM><TO instance="START" class="Process" start"></TO></CONNECTOR>
Subsequent	Nyíl A Process Start-tól az Activity-ig	<CONNECTOR id="con.25219" class="Subsequent"><FROM instance="START" class="Process" start"></FROM><TO instance="Tevékenység" class="Activity"></TO></CONNECTOR>
Subsequent	Nyíl az Activity-től az End-ig	<CONNECTOR id="con.25220" class="Subsequent"><FROM instance="Tevékenység" class="Activity"></FROM><TO instance="VÉGE" class="End"></TO></CONNECTOR>
Dokumentum modell		
Name	I/O	<MODEL id="mod.25237" name="I/O" version="" modeltype="Document model" libtype="bp" applib="ADONIS:CE BPMS BP Library 2.0">
Objektumok		

Document		
Name	Input data	<INSTANCE id="obj.25239" class="Document" name="Input data">
Document		
Name	Output data	<INSTANCE id="obj.25242" class="Document" name="Output data">
IT modell		
Name	IT	<MODEL id="mod.25246" name="IT" version="" modeltype="IT system model" libtype="bp" applib="ADONIS:CE BPMS BP Library 2.0">
Objektumok		
Application		
Name	IT application	<INSTANCE id="obj.25248" class="Application" name="IT application">
Szervezeti modell		
Name	Szerv	<MODEL id="mod.25222" name="Szerv" version="" modeltype="Working environment model" libtype="we" applib="ADONIS:CE BPMS WE Library 2.0">
Objektumok		
Role		
Name	Responsible User	<INSTANCE id="obj.25224" class="Role" name="Responsible User">
Role		
Name	Accountable User	<INSTANCE id="obj.25227" class="Role" name="Accountable User">
Role		
Name	Cooperation User	<INSTANCE id="obj.25230" class="Role" name="Cooperation User">
Role		
Name	Informed User	<INSTANCE id="obj.25233" class="Role" name="Informed User">

8 Publikációs lista

FOLYÓIRATCIKK

2014 Ildikó Szabó, **Krisztián Varga**

Knowledge-based compliance checking of business processes

Befogadva az ODBASE14 konferenciára, kiadás alatt az LNCS-ben

2014 Katalin Ternai, Mátyás Török, **Krisztián Varga**

Combining Knowledge Management and Business Process Management – A Solution for Information Extraction from Business Process Models Focusing on BPM Challenges

A. Kő and E. Francesconi (Eds.): EGOVIS 2014, LNCS 8650, pp. 104–117, Springer International Publishing Switzerland 2014

ISBN: 978-3-319-10177-4

2013 András Gábor, Andrea Kő, Ildikó Szabó, Katalin Ternai, **Krisztián Varga**

Compliance Check in Semantic Business Process Management

Y.T. Demey and H. Panetto (Eds.): OnTheMove Conferences & Workshops (OTM 2013), Graz, Austria, 9-13 September 2013, LNCS 8186, pp. 353–362, Springer-Verlag Berlin Heidelberg

ISBN: 978-3-642-41032-1

2013 Katalin Ternai, Ildikó Szabó, **Krisztián Varga**

Ontology-based compliance checking on higher education processes

Kő, A. et al. (Eds.): Technology-Enabled Innovation for Democracy, Government and Governance - Proceedings of the 2nd Joint International Conference on Electronic Government and the Information Systems Perspective and Electronic Democracy (EGOVIS/EDEM 2013), Prague, Czech Republic, August 2013, LNCS 8061, pp. 58--71. Springer, Heidelberg

ISBN: 978-3-642-40159-6

KONFERENCIAKÖZLEMÉNY

2011 Andrea Kő, Péter Fehér, **Krisztián Varga**

Role of ICT in Knowledge Networks among EU SMEs

Proceedings (CD) of the 5th IEEE International Conference on Software, Knowledge Information, Industrial Management and Applications (SKIMA 2011), 8-11 September, 2011, Benevento, Italy, edited by Savino, M. M., Chackpitak, N., Curran Associates, Inc, Red Hook, NY, USA, 7 oldal

IEEE Catalog Number: CFP1182R-PRT

ISBN: 978-1-4673-0247-0

2011 Andrea Kő, Péter Fehér, **Krisztián Varga**

Facilitating Knowledge Sharing in Virtual Networks

Proceedings of the 12th European Conference on Knowledge Management (ECKM 2011), 1-2 September, 2011, Passau, Germany, edited by Lehner, F. and Bredl, K., Academic Publishing Limited, Reading, UK, pp. 524 – 534

ISBN: 978-1-908272-10-2 CD

2008 **Krisztián Varga**

The Role of Innovation in Cross-Border Disaster Management

Proceedings and Abstracts of the International Innovation Conference for Co-operation Development (InCoDe), 16-18 October, 2008, Pécs, edited by Fojtik, J., Misprint Kft., Pécs, Hungary, pp. 201 – 211

ISBN: 978-963-642-248-6

2007 Péter Fehér, Balázs Járvás, **Krisztián Varga**

Developing Integrated Knowledge Management Solution based on Business Process Management

Proceedings of the 8th European Conference on Knowledge Management (ECKM 2007), 6-7 September, 2007, Consorci Escola Industrial de Barcelona, Spain, edited by Martin, B. and Remenyi, D., Academic Conferences Limited, Reading, UK, pp. 327 – 333

ISBN: 978-1-905305-53-7 CD

ABSZTRAKT

2013 Gábor András, Kő Andrea, Szabó Ildikó, Ternai Katalin, **Varga Krisztián**

Semantic Business Process Management Challenges and Opportunities in Compliance Checking

10. Országos Gazdaságinformaticai Konferencia (OGIK 2013), 6. International Symposium on Business Information Systems, 2013. november 8-9, Győr, szerk.: Raffai, M., Dobay, P., Radu, P. C., pp. 69-70, Győr, Széchenyi István Egyetem

2011 Andrea Kő, Péter Fehér, **Krisztián Varga**

Role and Support of Knowledge Networks among EU SMEs

Országos Gazdaságinformaticai Konferencia (OGIK'2011), 2011. november 11-12, Győr, szerk.: Raffai, M., Dobay, P., pp. 68 – 69, Palatia Nyomda és Kiadó Kft., Győr, Magyarország

2011 **Krisztián Varga**

IT in Crisis - Crisis in IT

IT Service Management for SMEs: Challenges and Opportunities (INNOTRAIN IT), 2011. október 27, Debrecen, szerk.: Mátyus, L., Homolay, K., pp. 8.

POSZTER

2011 Zoltán Szabó, Péter Fehér, **Krisztián Varga**

IT in Crisis - Crisis in IT

Budapesti Corvinus Egyetem, "Kari Tudományos Fórum – BCE
Gazdálkodástudományi Kar", 2011. szeptember 30.