

E-service quality tanulmányok

Kollaboratív virtuális iroda egyszerű modellje és technológiája V0.91

Kulcsszavak:

e-service quality, UX, HCI-standard, Dublin Core, Topic Maps, ontology

Nem szokványosan értelmezett terminológiák:

Tartalom (content)= portál (portal)= e-service (Stiláris szempontok döntenek el, hogy mikor melyik terminológiát használjuk.)

Objektum (object): minden, ami **a felhasználó számára önálló jelentéssel bír, a virtuális térben egyben és oszthatatlanul létezik, és a képernyőn megjelenik** (akár úgy, hogy a képernyőn fix helye van, mint a logo-nak, vagy egyes kezelői eszközöknek, akár ablakban mozgatható, mint a dokumentumok). Az informatika hagyományos fejezeteiben a fogalom nem szerepel. A fájl fogalmával hozható analógiába. Részletesebb definíciót ld. később.

Virtuális tér (virtual space): Ahol az objektumok a képzeletünkben elhelyezkednek.

Tartalomjegyzék (contents): az egész virtuális teret (a felhasználó által az adott pillanatban kezelt virtuális teret) összefoglalja.

Mappa (directory): a tartalomjegyzék része. Használatos az al-mappa (subdirectory). Két része van: a fej és a tételek.

Fejezet A mappa szinonimája (ha kiderül, hogy működik az elnevezés).

Objektum megjelenése: natív mód, rövid-mód, mappa-mód

Mappa-képző entitások: bináris relációk, metainformációk

Művelet (operation): amit az objektumokon végzünk, pl. create, modify, delete, stb.

Domén (domain, professionalty) szakterület, amivel az e-szolgáltatás foglalkozik. Nem hálózati domain.

osz – az "osztályozási szempont" rövidítése

Bevezetés	3
1. Összefoglalás	4
1.a. Technikailag	4
1.b. Célközönség	4
1.c. A „Small Data Project”	4
2. A tanulmány szópja	5
2.a. Jelenlegi helyzet.....	5
2.a.1. Az ontológia (szemantika) hiánya. Mi micsoda a képernyőn, és mire való?.....	5
2.a.2. A pragmatikai szempontok elhanyagolása. Most én jövök, vagy a szolgáltató?	5
2.a.3. A virtuális tér hiánya. Mi hogy néz ki, és hol van?	5
2.a.4. A szemantikához tartozik: a HCI-standard hiánya	5
2.b. Ez a megközelítés természetesen következményekkel jár	6
2.c. Mit oldunk meg az ontológiai (szemantikai) megközelítéssel?	6
2.c.1. A képernyőn megjelenő objektumok értelmezése	6
2.c.2. Az objektumok osztályozása és lekérdezése – tartalomjegyzék- és mappaképzés.....	7
2.c.3. A felhasználói szerepkörök pontos megfogalmazása.....	7
2.c.4. Az ontológia, mint szimbólumtábla	7

3.	Milyen az a kollaboratív virtuális iroda?	8
3.a.	Miért off-line Office?	8
3.b.	A könyvesbolt és a könyvtár, mint metafora	8
3.c.	Elmélet: hogyan lesz ebből e-szolgáltatás?	10
4.	Egronómia, virtuális tér	11
5.	Ontológia, az objektumok HCI-szemponú osztályozása	12
5.a.	A metainformációk osztályozása	13
5.a.1.	Technikailag	13
5.b.	A paraméter típusú metainformációk áttekintése	14
5.b.1.	A tartalomjegyzék-motor	14
5.b.2.	Az objektumok háromféle azonosítása	15
5.b.3.	Lay-out-információk: az objektumok megjelenése	15
5.c.	A reláció típusú metainformációk	16
5.c.2.	Tartalomjegyzék és mappa (nem ide való a fejezet, hagyd most ki)	16
5.d.	Mi van képernyőn végül is?	17
5.e.	Két fő attrakció: az osztályozás és a lekérdezés	17
5.e.1.	Az osztályozás (tartalomjegyzék-képzés)	17
5.e.2.	A lekérdezés (keresés)	18
6.	Függelék: Ontológiák kezdeti megfogalmazása	19
6.a.	Felhasználói szerepek	19
6.b.	Hogyan kell mindezt megjeleníteni és értelmezni?	20
6.c.	Fontos: doménfüggő és doménfüggetlen értelmezés	21
6.d.	A szolgáltatás életciklusa	22
6.e.	Publikálás státusa	22
6.f.	Dokumentum tartalmának jellege	22
6.g.	Felhasználók	23
6.h.	Műveletek a virtuális tér objektumain	23
7.	Tudásbázis kezdetei: néhány dokumentum és metainformációi	24
8.	Példák egyszerű tartalomjegyzékekre. A virtuális tér változatai.	29

Bevezetés

A kutatási program, amelyhez az itteni szoftverterv-vázlat tartozik, a nagyméretű, sok információt tartalmazó, komplikált, professzionális szolgáltatást nyújtó portálok, használati minőségével foglalkozik. A kutatás elsősorban nem szoftver készítését, hanem a 'jó' szolgáltatás, kritériumainak feltárását, rendszerbe foglalását, és a jó gyakorlatok felkutatását, népszerűsítését célozza.

Azonban mégis szükséges bizonyos fejlesztői munka demo kialakítására, mert a téma újszerűsége miatt e nélkül nem lehet a kívülállónak (a szponzornak) megmagyarázni az említett kritériumok - és az egész kutatás - értelmét, jelentőségét.

A használhatóság kritériumainak ilyen rendszerbe foglalásával, ilyen kutatással az irodalomban nem találkozunk: a használhatóságnak (a professzionális e-szolgáltatások használhatóságának) jelenleg nincs kialakult modellje, nincs tudománya.

Másrészt a 'jó' szolgáltatás számunkra fontos kritériumait az informatika szakmán kívül, az un. "szomszédos tudományok", (neighboring sciences) területén kell, keresnünk. Mégis megkíséreljük itt "informatikus ésszel" egy kiinduló kritériumrendszer összeállítását, hogy a szomszédos tudományok képviselői számára is megfoghatóvá tegyük ezt a számukra teljesen idegen, átláthatatlan területet, azaz tudjunk tőlük kérdezni, tudjunk közös kutatási témákat javasolni.

Ez a kiinduló kritériumrendszer implicit módon definiál egy **szoftver-réteget**, vagy funkcióhalmazt, amely minden olyan e-szolgáltatásra (portálra) közös, és minden olyan portál mélyén ott kell lennie, amelyek eleget akarnak tenni a kritériumoknak. Ezt a doménfüggetlen, jelenleg hipotetikus **szoftver-réteget** nevezzük **kollaboratív virtuális irodának**.

1. Összefoglalás

A dolgozat egy *e-szolgáltatási modellnek*, melynek kulcsszava a client's sovereignty, a szemantikai / ontológiai aspektusát vázolja fel. Az egyéb: ergonómiai, pragmatikai aspektusok máshol lesznek kifejtve.

1.a. Technikailag

A következőkben kifejtett modell működését [A könyvesbolt és a könyvtár, mint metafora](#) fejezet magyarázataiból lehet megérteni. Pontokba szedve, időrendben:

- (a) **Fejlesztés:** a tervező meghatározza a portál által kezelt objektumok metainformációit.
- (b) Ebből a tartalomjegyzék-motor meghatározza a lehetséges osztályozási szempontokat, azaz tartalomjegyzékeket. (Előszörre kézzel végezzük.)
- (c) **Fejlesztés:** a tervező meghatározza az egyes metainformációkhoz szükséges további (rész)ontológiákat.
- (d) Itt készen van a portál - a doménfüggő algoritmusok és tartalom, azaz tudásbázis nélkül. Így semmire sem jó, tehát túlzás, hogy készen volna, majd valami jobb szó kell ide.
- (e) **Testreszabás:** s content builder (főszerkesztő, stb.), feltölti objektumokkal (leginkább dokumentumokkal). Kitölti a dokumentumok tervező által előírt metainformációit a tervező által elképzelt (rész)ontológiákkal. Ez a tudásbázis.
- (f) **Felhasználás:** ettől kezdve a felhasználó osztályozási szempontot választhat, és lekérdezhet.

Erre épülnének később a doménfüggő algoritmusok, amelyek kihasználnák a fenti lehetőségeket, a fölöslegeseket pedig blokkolnák, eltakarnák.

1.b. Célközönség

Az említett e-szolgáltatási modell és szoftver-technológia hasznélvezői elsősorban a magánszemélyek és KKV-k, azaz a SOHO kategóriájú felhasználók, akik ügyeiket gördülékenyen, kommunikációs, terminológiai, stb. problémák nélkül akarják intézni az Interneten.¹ Természetesen ez a modell hasznos lehet nagy szervezetek, kormányzervek számára is, de abban a körben a használatba vételnek szemléleti, és üzleti akadályai lehetnek².

1.c. A „Small Data Project”

Az ember kognitív képességei behatárolják a munkaterében tartható és átlátható ügyek, dokumentumok számát. Ennek megfelelően egy portál megjeleníthető tartalomelemeinek (dokumentumainak) számossága viszonylag kicsi, néhány száz, vagy ezer. Megfelel egy hagyományos iroda munkaterében lévő dokumentumok számának, és elenyésző az IKT szakmában megszokott adatbázisok rekordszámához képest. Ezért ezt a kutatást a *big data* név mintájára és attól való megkülönböztetésül *small data* projektnek is nevezzük.

Azonkívül, ez a név könnyen megjegyezhető és figyelemfelkeltő.

¹ Itt azonnal fölvetődik, egyrészt mi az, hogy “ügy”, másrészt minden elképzelhető “ügyre” van szoftver megoldás, több is, itt a langyos vizet akarja valaki megint fölhalálni... Ezeket más dolgozatokban tárgyaljuk: ld. www.vitalyos.hu

² A szolgáltató gyakran maga is nagy szervezet a SOHO-hoz képest, tehát a modellt és technológiát el kell fogadnia, és be kell vezetnie.

2. A tanulmány szkópja

2.a. Jelenlegi helyzet

A portálok (e-szolgáltatások) tartalmi-funkcionális elemeit - itteni terminológiával *objektumait* -, (pl. dokumentumait, HTML-lapjait) a domén (a szakterület, professionality) és a megrendelői igények lényegében meghatározzák.

A *megjelenést* a web-designer-re szokták bízni, ez a szakma elfogadottá vált.

Azonban számos kérdés elsikkad, ezért sok hiányossága van a jelenkori e-szolgáltatásoknak. Ezeket 3 csoportba soroljuk:

2.a.1. Az ontológia (szemantika) hiánya. Mi micsoda a képernyőn, és mire való?

Az objektumok *csoportosítása, lekérdezése*, a közöttük lévő *összefüggések*, több szempontú, kényelmes megmutatása elhanyagolt kérdéskör. CMS-eszközök szemantikus technikai kezdetlegesek, az ontológiafejlesztés gyakorlata kiforratlan. A professzionális e-szolgáltatásokban ennek a fontossága nő, az ügyfelet egyre inkább érdekli. Más megfogalmazásban ezt a kérdést a szolgáltatás fogalmi pontosságának mondjuk.

Objektum ebben az értelemben minden, ami a képernyőn van, a kezelői felület minden eleme.

2.a.2. A pragmatikai szempontok elhanyagolása. Most én jövök, vagy a szolgáltató?

A pragmatika diszciplínája az emberi kommunikáció törvényszerűségeivel foglalkozik. Más megfogalmazásban itt a felhasználói biztonságérzetről (főleg a hitelesség érzetéről, a bizalomról) van szó, mert ez az érzet erősen függ a szolgáltató (e-szolgáltató) kommunikációjának korrektségétől.

A kollaborativitás kezelése is ide tartozik.

2.a.3. A virtuális tér hiánya. Mi hogy néz ki, és hol van?

A tartalmak (e-szolgáltatások) nem rendelkeznek konvencionálisan elfogadott átlátható struktúrával. Elvárásolt kastélyok, melyben élvezettel lehet bolyongani. Múzeumok vagy reklámfelületek, a játékautomaták utódai inkább, mint munkaeszközök. Legyen ehelyett a *munkaeszköznek szánt* tartalom struktúrája átlátható, legyen a szakmai kiadványok, dokumentumok elektronikus utóda legyen, és ne a játékautomatáké. Más megfogalmazásban ezt a szolgáltatások ergonómiájaként emlegetjük.

Ezt a témát az ergonómia címszó alatt is említjük.

A jelen kutatás a filozófiája az, hogy a Kollaboratív Virtuális Iroda megvalósításához a CMS-nek új technológiai komponensekre van szüksége, mert ezek a hiányok reális mértékű szoftvertervezői, -kivitelezői erőfeszítéssel nem oldhatók meg. A CMS-eknek a fenti 3 lábon kellene állniuk, amelyek sajnos szinte teljesen hiányzanak.

Ennek a hármas felosztásnak az eszméje Trinity Principle néven szerepel a dokumentumainkban.

2.a.4. A szemantikához tartozik: a HCI-standard hiánya

Ehhez jön még a HCI-standard hiányából adódó számtalan egyéb usability-probléma, amelyeket összefoglaló néven Babel-szindrómaként emlegetünk. Néhol a kulturális kompatibilitás hiányának mondják.

A könyvnek, mint sok száz éves kulturális ikonnak van egy standard formája. Lapokból áll, azok egyesével meg vannak számozva, fejezetek vannak, azoknak neve esetleg hierarchikus számozása van, van tartalomjegyzék, amely a fejezetek megtalálhatóságát segíti, esetleg tárgymutatója, fűlszövege van. Borítója, címe, megjelenése van, amely messziről felismerhetővé teszi.

Ezeket mindannyian értjük, és a könyvet, mint tárgyat használni tudjuk, anélkül, hogy a könyv tartalmát, akár nyelvét értenénk. A portáloknál, e-szolgáltatásoknál, mint elektronikus használati tárgyaknál mindez a kezdet kezdetén van. Hasznos konvenció pl. a Kapcsolat és Impresszum menüpont.

Úgy gondoljuk, hogy ez szabványosodási folyamat a piaci viszonyok között nem, vagy túl lassan megy végbe. Az e-szolgáltatások viharos terjedése viszont ezt nem várja meg, hanem elszennyezi a digitális ökoszisztémát rosszul használható zagyva szolgáltatásokkal. Ennek az eszmének, amelyik a kutatásunk fő motivációja, a Non-Evolution Principle nevet adtuk.

A Kollaboratív Virtuális Irodát azzal az igénnyel specifikáljuk, hogy – kiegészülve a szomszédos tudományok hozzájárulásával - alkalmassá váljon egy HCI-standard létrehozására.

A fejlesztés egyéb szempontjait és gondolati hátterét a referenciák tartalmazzák.

2.b. Ez a megközelítés természetesen következményekkel jár

A portál felépítésében nem engedünk meg olyan szabadságot, mint a jelenlegi CMS-eszközök. Ennek következtében (valószínűleg) lemondunk bizonyos szolgáltatások, pl. web-ruházak, vagy internetes sajtó/média vizsgálatáról, mert ezekre viszonylag jól megoldott céleszközöket használ a szolgáltatók többsége.

2.c. Mit oldunk meg az ontológiai (szemantikai) megközelítéssel?

Ebben a tanulmányban első lépésként az ontológiai-szemantikai kérdéskör kezelésére adunk megoldási javaslatot és demo-szintű specifikációt. A virtuális tér kérdését a szükséges mértékben érintjük, mert az objektumokat el kell tudni helyezni a térben.

A szemantikus hálóba rendezhető objektumok interaktív rendezése és keresése. Előnyei elsősorban a sok dokumentumot, ügyet, sok különböző fogalmat tartalmazó, több felhasználó által egyszerre használt portáloknál jelentkeznek.

(A legegyszerűbb szemantikus háló a fa-struktúra, ami a része-tartalmazza reláció szerint rendez, mint a fájlrendszer. Nyilván nem lehet benne sem hurok, sem kerülőút, összefolyás. A következő fokozat az, hogy ha többféle reláció szerint lehet fába rendezni az objektumokat. Ennek a továbbgondolása vezetett el az ontológiák fogalmához a 70-es 80-as években.)

A javaslat a következőkre ad megoldást:

2.c.1. A képernyőn megjelenő objektumok értelmezése

Mi minek a mije, és miért van ott? Mind az objektumok, mind a kezelői felület elemeinek típusa az ontológia adatbázisából - előre beépített technológiával - megjeleníthető. A jelenlegi „repülő”, vagy “buborékos” magyarázatok az esetlegességük és terminológiai kontrollálatlanságuk miatt nem megfelelőek. A típus vektor: URM minden szintjéhez elvileg tartozik egy érték. Itt egyelőre 2 értéket tárgyalunk: a HCI-

ontológiának és a domén-ontológiának megfelelő, az URM 3. és 4. szinthez tartozókat. Pl. egy objektum HCI szerint 'szöveg', a domén szerint pedig 'gyógyszerleírás'.

A usability szakmában "érthetőség" és "tanulhatóság" (understandability, learnability) neven szerepel ez a kérdéskör, de nem kapcsolják a kettőt össze. Itt viszont: amit könnyen meg tudunk akárhányszor újra nézni, az megtanuljuk. Amihez rejtvényt kell fejteni, azt egyszer talán megteesszük, de ha elfelejtettük, nincs türelmünk újra megfejteni, és lassan elfelejtjük.

2.c.2. Az objektumok osztályozása és lekérdezése – tartalomjegyzék- és mappaképzés

Elsősorban hagyományos dokumentumok, ügyek tekintetében hasznos: nemcsak a hagyományos módon – fájl-attribútumok szerint - rendezhetjük azokat, hanem lehetővé válik az ontológiát, azaz az objektumok szakmai szempontú rendezését követő mappastruktúra is. Emellett megmaradhat a hagyományos, pl. kártyarendezéssel tervezett és fixen programozott intuitív megoldás.

A usability-ben: megtalálhatóság (findability).

2.c.3. A felhasználói szerepkörök pontos megfogalmazása

A szerepkörök és kompetenciák pontos megfogalmazásának hiánya, ezek összekeverése sok nehézség forrása a szoftverek életciklusának minden fázisában. Közös gyökerük az, hogy a különböző szereplők – elsősorban a szolgáltatás tervezői és kivitelezői - nem tudnak eleget egymásról, egymás kompetenciáiról, és nem megfelelő szereplővel beszélnek meg a problémájukat, vagy nem megfelelő szereplőhöz juttatják el az üzeneteket.

Alappélda: egy üzemeltetőnek szóló információ az ügyfélnél jelenik meg, akit félrevezet, mert azt hiszi, hogy neki szól - és az üzemeltető nem is értesül. Vagy: az ügyfélnak órá nem tartozó információkat kell átlapoznia, és felfognia, hogy nem ezt keresi, mire eljut ahhoz, amit keres.

Megoldás az lehet, hogy a különböző szereplők számára különböző módon működhet ugyanaz a kezelői felület.

A usability nem tárgyalja külön ezt a kérdéskört.

2.c.4. Az ontológia, mint szimbólumtábla

Az űrlapok szimbólumtáblából választható mezőinek a szimbólumait a megfelelő ontológiából kell venni. Ez mindjárt magával hozná az egyes szimbólumok magyarázatát is, ami az ontológia része.

3. Milyen az a kollaboratív virtuális iroda?

3.a. Miért off-line Office?

A *dokumentum* szónál arra kell gondolni, hogy az e-szolgáltatások olyan anyagokkal foglalkoznak, amelyek általában nem maradnak meg az e-szolgáltatás portáljának hatáskörében, hanem kikerülnek más szolgáltatás területére, olykor papír formájában. Erre az *off-line* módon szerkesztett dokumentum alkalmasabb. Az on-line dokumentum pl. fórum anyagai nem tudnak kikerülni. Azonkívül, az on-line office eszközök használata kompatibilitási és elérhetőségi problémákat okoz. Ezek az eszközök csak egy homogén technológiát használó belterjes kör – akár egy nagyvállalat – számára adnak elfogadható megoldást.

3.b. A könyvesbolt és a könyvtár, mint metafora

A következő táblázat összehasonlítja az e-szolgáltatások jelenlegi és szerintünk kívánatos tulajdonságait:

		A hagyományos portál	Az új technológiájú portál
1.	Cél	A tulajdonos reklámoz, elad, tájékoztat, szolgáltat, hatósági feladatot lát el.	A felhasználó(k) az ügyei(ke)t intézi(k)
2.	Minőség prioritásai	Ergonómia, biztonságérzés, egyértelműség	Egyértelműség, biztonságérzés, ergonómia
3.	Minőség kritériuma: mi legyen könnyen átlátható, megtalálható?	A portál készítője által elképzelt funkciók, információk	Az idővel egyre szaporodó dolgaink: dokumentumaink, ügyeink, azaz <i>objektumaink</i> .
4.	Mit kell csinálnia, megtanulnia "néhány kattintással" a felhasználónak?	Eligazodni a portál készítője által elképzelt navigációban, felismerni a tartalomjegyzék/menü funkcióit	Az <i>objektumaokat</i> számára célszerű módon <i>csoportosítani, keresni</i> és megjeleníteni. Tehát nincs fix tartalomjegyzék/menü. Csoportosítási eszközök vannak.
5.	Mi különbözteti meg a dolgokat, az ügyeket?	A tartalomjegyzék/menü pontjai.	A metainformációk (más, rokon jelentésű szó: <i>tudásbázis</i>)
6.	Mi a jószág fő technikai eszköze.	Egyértelmű, jól látható navigáció, tartalomjegyzék/menü A funkciók jól látató elnevezése.	Jól használható, megtanulható eszköz 1) az <i>objektumok</i> metainformációinak a beállítására (más terminológiával, a <i>tudásbázis</i> létrehozására), 2) valamint a 4. pont tevékenységére.
7.	Mi a tervezői tudás fókuszja?	A hagyományos 'design for usability' megfelelő alkalmazása.	A metainformációk lehetséges halmazának, és megengedett kapcsolatainak - röviden az <i>ontológiának</i> , pontosabban a portál által kezelt ügyek ontológiájának – a megfelelő meghatározása.
8.	Mi a tervezői tevékenység fókuszja	Funkciók, algoritmusok tisztázása.	Fogalmak tisztázása.
9.	Mi a kivitelezői tevékenység fókuszja	Szoftver készítése.	Ontológia készítése
10.	Kollaborativitás	Egyre inkább van, gyakran a gyorsan fejlődő social network kultúrájára épülve, pl. like az FB-n, LinkedIn, stb.	Ki kell munkálni. A social network kultúrája valószínűleg nem a jó irány itt. (A virtuális terek közötti (halmaz)műveletek, pl. mount-dismout tűnnek perspektivikusnak.)

11.	Az “ügy” absztrakt és konkrét fogalmának bevezetése		Az ontológiának az “ügy” fogalmait is le kell írnia.
12.	Domain-specifikus algoritmusok	-	Az ügyeknek lehetnek speciális igényeik, amelyeket meg kell programozni, mint a hagyományos portálnál.

Megjegyzések, magyarázatok:

1 (cél) Természetesen a hagyományos portálon is intézi az ügyfél az ügyeit, de azok a portálok nem az itt tárgyalt, bemutatott szempontok alapján készültek. Pl. nem ismerik az “ügy” fogalmát, ehelyett dokumentumokkal foglalkoznak; nincs kulturális kompatibilitás, mindegyik saját leleményű HCI-vel dolgozik, nem képezhető saját virtuális iroda, ahol a különböző portállal kapcsolatos ügyeim összerendezhetőek lennének, stb.

2 (biztonságérzés) Hasonló, mint a mantrává vált ‘biztonság’, de nem azonos azzal. Pl. mitől érzi az ügyfél hitelesnek, amit a virtuális térben lát? A pszichológia, mint “szomszédos” tudomány foglalkozik vele, tárgyalása külön tanulmányt igényel.

4 (objektumok csoportosítása). A csoportosításon kívül számos *művelet* létezik: mindaz, amit egy irodában, a *valós világban*, a papírokkal, ügyekkel szoktak tenni. A Függelék a Műveletek ontológiája fejezetben felsorolja a közismert műveleteket. (Egyelőre csak felsorolja)

NB: ezek a *műveletek* a hagyományos portálokon (és minden informatikai alkalmazásban, különösen az iratkezelő alkalmazásokban) többé-kevésbé meg vannak valósítva, amennyire ezt a funkcionalitás igényli – tehát a bal oszlop 8-9 sorába tartoznak. Az új portál-filozófiának az egyik alapvetése, hogy ezeknek a *műveletek*nek nem a portál funkcióihoz kell kötődniük, és esetleg minden portálnál (és minden alkalmazásnál) külön elképzelve, és megvalósítva lenniük. Ehelyett a kezelői felülethez, a HCI-hez kell kötődniük, a táblázat 4. sorában van a helyük, és így, amennyire lehet, szolgáltatási standarddá kell válniuk. Ez az alapvetés a Bábel-szindróma elkerülésnek egyik lehetséges útja.

NB: a copy, move, send *műveletek* nincsenek a felsorolásban. Nem léteznek, mert a virtuális térben az objektumokat nem másoljuk, és nem mozgatjuk: ilyet a jól tervezett szolgáltatás nem kíván meg. Ezek azon kevés dolgok közé tartoznak, amelyek a *valós világban* megszokottak, de a virtuális világban nem kell leképezni őket. A grant, publish, revoke *műveletek*kel tesszük mások számára elérhetővé, kezelhetővé az objektumokat. A synchronize az off line tevékenység miatt szükséges, szerepét ki kell dolgozni.

4,6,7-9 (metainformációk kezelése). 3-szinten történik:

- **7-9: fejlesztés, az ontológia létrehozása.** 2 lépésből áll: a metainformációk megtervezéséből, erre példa [A paraméter típusú metainformációk áttekintése](#) fejezet táblázata, valamint az [Ontológiák kezdeti megfogalmazása](#) lépésből, az ilyen című fejezetben. Az ontológia jelöli ki a virtuális teret, amiről valamilyen geometriai képzetünk lehet, és amit legprimitívebb formában a tartalomjegyzék jelenítenek meg, ld. példaként [A tartalomjegyzék-motor](#) fejezetet.
- **6: testreszabás, set up, a tudásbázis létrehozása.** Objektumok feltöltése és metainformációinak beállítása üzemeltetői, vezető felhasználói, stb. jogosultságok birtokában. Példák a [Tudásbázis kezdetei: néhány dokumentum és metainformációi](#) fejezetben vannak. Ez a művelet helyezi el az objektumokat a virtuális térben. A tér különböző elrendezéseit a [Példák egyszerű tartalomjegyzékekre. A virtuális tér változatai.](#) Fejezetben vannak.

- **4: megjelenítés.** Felhasználás. Az objektumok megjelenítése leginkább tartalomjegyzékek különféle szempontú képzésével, és rendezéssel. A virtuális tér különböző nézeteit állíthatja elő a felhasználó.

1-9 (a virtuális iroda definíciója). Ez a dolgozat a jobb oldali oszlop 1-9 pontjaival foglalkozik. Ami így kialakul, azt nevezzük virtuális irodának. Ez a konstrukció jó közelítéssel minden e-szolgáltatásnál ugyanaz lehet. Azaz, a jó minőségű e-szolgáltatás mélyén egy standard virtuális iroda technológiája van.

10 (kollaborativitás és szabványosság). Az ugyanazt a virtuálisiroda-technológiát – mondhatjuk így: standard technológiát – használó szolgáltatásokra lehet szintén szabványos együttműködési metódusrendszert kidolgozni. Egy ügyfélnek több szolgáltatónál is lehetnek ügyei, egy szolgáltatónál lehet több ügyfélnek közös ügye, a felhasználóknak lehetnek a szolgáltatóktól független közös ügyei¹. Kívánatos, hogy ezek az egyes felhasználók számára egyetlen integrált irodaként jelenjenek meg.

A portál 1-9 alatt leírt funkcionalitását úgy kell elképzelni, hogy a portálba behajigálunk dokumentumokat, ráragasztva mindegyikre a jól átgondolt metainformációját, és a fenti funkcionalitást megvalósító *untartalomjegyzék-motor* majd valahogy szemantikusán rendszerezi azokat.

Így működik a *könyvtár* is: beszereznek egy könyvet, metainformációkat rendelnek hozzá (ilyen pl. az ETO osztályozás, elektronikusan a DC), azaz katalogizálják, és a könyvtár rendszerező-kereső mechanizmusával később meg lehet találni.

Ezzel ellentétben a jelenlegi portálok a *könyvesboltok*hoz hasonlítanak, ahol az ügyfél vásárol. Az üzemeltető igyekszik fix kategóriák feliratait a polcokon jól láthatóan, vonzóan elhelyezni. A polcok egyúttal reklámfelületek is. Erre reklám- (usability-) szakembert alkalmaznak, ha egyáltalán fontosnak tartják.

3.c. Elmélet: hogyan lesz ebből e-szolgáltatás?

A jól működő e-szolgáltatás mélyén tehát egy virtuális irodának kell lennie. A virtuális irodának ez a könyvtárat utánozó mechanizmus az alsó szintje.

Az osztályozási szempont változtatásával annyiféle tartalomjegyzék képződhet, ahányféle erre alkalmas metainformáció-típus van, azaz amilyen az ontológia. Alkalmas pl. a létrehozás ideje, mint implicit metainformáció. (Implicit abban az értelemben, hogy nem kell kézzel megadni.) Ha **osz**-nak kijelöljük a létrehozás napját (vagy évét, vagy hónapját, stb.) akkor egy napok (évek vagy hónapok) szerinti csoportosítású mappák képződnek.

Ennek az osztályozási és tartalomjegyzék-képzési technikának a neve több szempontú osztályozás, faceted classification.

Erre épül a kollaborativitás mechanizmusa, majd az ügyek ontológiája és a domén-specifikus algoritmusok kezelése a táblázat 10-12 sorai szerint. Az 'ügy' magasabb kategória, mint a dokumentum, egy e-szolgáltatás ügyekkel, és nem dokumentumokkal foglalkozik. Ezeket később tárgyaljuk.

¹ Ezt továbbgondolva: a virtuális irodának működnie kell egyetlen felhasználó számára, hálózattól izolált munkaállomáson is. Ennek a távoli őse, előképe, amikor az Intézővel a fájljainkat rendezgetjük a gépünkön.

4. Egronómia, virtuális tér

Elfelejtjük a weblap fogalmát. Helyette a virtuális tér fogalmával dolgozunk, amelynek különféle nézeteit látjuk a képernyőn. A virtuális tér több entitásból áll, a két legfontosabb: Az *objektumok*, és azok *kapcsolatai*.

Ezenkívül áll még *eszközökből*, amelyekkel az előbbi kettőn műveleteket végzünk, és *aktorokból* – ezek felhasználók vagy agent-ek – akik a műveleteket végzik. Vannak még speciális entítások, pl. az előbbi *eszközök* setup-jai, stb.

A virtuális teret úgy kell elképzelni, mint egy gráfot, az objektumok gubancát, ahol a gráf pontjait az objektumok, éleit az objektumok kapcsolatai adják. A honlaptervezés (weblaptervezés) helyébe a virtuális tér tervezése és átlátható berendezése lép. Az objektumokat nagy részben meghatározza a szakmai-üzleti tartalom (a domén), míg a virtuális tér tervezése a kapcsolatok megfelelő kialakítását, és a gráf megfelelő vizualizálását jelenti.

A virtuális teret különböző nézetben és felbontással láthatjuk. Például, ha a virtuális tér egészét nézzük, akkor a tartalomjegyzéknek csak a legfelső szintje szerepel a képernyőn, az objektumoknak pedig csak az ikonja, vagy a miniatúrája, vagy a neve, vagy valamilyen felismerhető reprezentációja jelenik meg.

A virtuális térrel kapcsolatos elv az, hogy a tér és **az azt benépesítő objektumok nem viselkedhetnek lényegesen másképpen, mint a valóságos fizikai tér objektumai**. Pl. egy objektum csak egy helyen lehet egyszerre, a linkeknek - a tükör virtuális megfelelőinek – a száma korlátos legyen, stb. Az elv megsértése kognitív és tájékozódási problémákat okoz¹

Műveletek (operations) vs. communications: A Függelék Ontológiával foglalkozó fejezete felsorolja a Műveletek közismert halmazát. **Itt szét kell választani a művelet és a kommunikáció fogalmát, ahogyan ez fizikai világban is van**. A kommunikáció üzenetváltás. Felhasználó/ügyfél nem a géppel kommunikál, hanem a szolgáltatóval, annak ügyintézőjével, döntéshozójával, sw-agent-jeivel, a virtuális irodájában pedig tesz-vesz, azaz műveleteket végez². Virtuális tér az, amin a felsorolt műveleteket tudjuk végezni más szereplők bevonása nélkül³.

¹ Vitalyos: *Object Permanency Principle in the Usability*, 2011

<http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=5999484&url=http%3A%2F%2Fieeexplore.ieee.org%2Fstamp%2Fstamp.jsp%3Ftp%3D%26arnumber%3D5999484>

² Ha a könyvespolcra be akarunk tenni egy könyvet, és nem fér be, nem üzenetet kapunk a könyvespolctól, hanem érzékszerveinkkel tapasztaljuk, hogy a művelet nem megy. Az automatizált raktáraknál sem megfelelő működés az, ha a konveorkocsi lelkesen elgurul a könyvekkel, majd messziről visszaüzen, vagy szomorúan visszadöcög, hogy nincs hely. Ezt a virtuális tér (a raktárt mutató virtuális tér) megtekintéséből kell előre látni.

³ Természetesen, ez sem egyszerű: lehetnek műveletek, amelyek technikai okokból hosszú percekig tartanak, ami zavaró, mert az már nem a műveletek, hanem az üzenetek időtartománya – a virtuális tér megfelelő megtervezése egy szakma.

5. Ontológia, az objektumok HCI-szempon t u osztályozása

Objektum az, ami a képernyőn (a virtuális térben) látszik.

Egy objektum 3 *al-entitásból* (egyszerűbben: 3 valamiből) áll, hasonlóan a fájlokhoz:

Title, (Cím, Név). Ami azonosítja az *ember* számára. Ez látszik leggyakrabban. Emberi olvasásra való, mint a könyvben a fejezetcím.

Tartalom. Az igazi információ.

Meta-információk. Pl. az objektum típusa vagy kapcsolatai más objektumokkal. Csak szükség esetén látszanak.

Egyik fontos metainformáció a **típus**. Most pontosítjuk a szöveg elején lévő definíciót:

Egyszerű objektumok Egyszerű objektumok azok, amelyek belsejében nem lehet műveleteket végezni, pl. egy szót módosítani.

- *Dokumentum-típusú* objektumok¹
 - *Office-típusú* objektumok, azaz hagyományos dokumentum- vagy multimédia-fájlok, ezek is két részre oszthatók felhasználói szempontból:
 - *Off-line* módon szerkesztett fájlok². Professzionális e-szolgáltatás számára ezt tartjuk követendőnek.
 - *On-line* módon szerkesztett dokumentumok³, ezek leginkább hozzászólások, fórumbejegyzések, amelyek nem is válnak mindig fájlá. Használatuk nem javasolt.
 - *HTML-típusú* objektumok (jobb-nevet majd ki kell találni), amelyek a hagyományos HTML-lapok önállóan értelmes, összefüggő elemeinek felelnek meg.
 - Minden olyan, ami lehetne akár *dokumentum* is, de konvencionálisan nem az, a felhasználó hozza létre, és a képernyőn megjelenik
 - blog- és fórumbejegyzés
 - rendszer és az alkalmazások üzenetei, hibaüzenetek
 - ...
- *Panel-típusú (dash-típusú, vagy valós idejű)* objektumok, amelyek tartalma a felhasználótól függetlenül változhat. Itt pontosításra szorul a felhasználó fogalma: lehet élő személy, vagy szoftver agent. A panel-t nem ezek, hanem pl. mérőműszerek értékei, vagy adatbázisok változásai változtatják. Panel pl. az óra. (Jelenleg nem témánk.⁴)
 - Ide tartozik az olyan lap, amelyik adatbázisból veszi az adatait, vagy néhány adatot amelyek – akár böngészés közben - változnak.

Összetett objektumok Összetett objektummá válik egy egyszerű objektum, ha a belsejébe kezelői eszközöket, pl. egy dokumentumba kitölthető mezőt helyezünk el. (Itt is meg lehetne ismételni az egyszerű objektumok felosztását.)

- Az **úrlapok** – mind a sablon, mind a kitöltött úrlapok - office-típusúak, pontosabban az alkalmazásoknak úgy kellene kezelniük⁵. Az egy-két mezőből álló úrlapok, amelyeket nem érdemes, vagy nem szabad fájlba menteni, mint a pl. login vagy az egyszerű keresés panelja, valószínűleg HTML-típusúak lesznek.
- Tartalomjegyzék, mappa.
- A keresések eredményei: ezek speciális - célszerűen office-típusú, – dokumentumba kerülnek automatikusan, mutatókkal a keresett objektumokra. Kerülhetnek mappába is, ld. később.
- Fejezet. Az egy mappa alá tartozó objektumok összessége, mint egyetlen objektum
- ...

¹ A dokumentumra úgy kell gondolni, hogy ha megváltoztatjuk, akkor új verzió keletkezik, ami már új dokumentum. Az összetett objektumok esetén komplikált és nehezen kezelhető esetek vannak: pl. ha egy dokumentum úrlapot tartalmaz. Az ilyesmit célszerű egy technológiai ajánlásban kizárni.

² Az erre alapuló megoldások (virtuális iroda, portál, stb.) neve: *Object Based Collaboration*, ezt javasoljuk.

³ Az erre alapuló megoldások neve: *WEB Based Collaboration*, amit nem javasolunk.

⁴ A képernyő tartalmi konzisztenciájának biztosítása – pl. annak az elkerülése, hogy egy screen shot félig megváltozott képernyőt rögzíthessen - nehéz. Az adatbáziskezelők *rollback* és *commit* funkcióinak megfelelőjét kell kidolgozni és beépíteni a HCI-be.

⁵ A kitöltött úrlap nettó információja, amit CSV, semicolon vagy XML-formátumban tarthat az alkalmazás, nem objektum, mert a felhasználó számára nem jelenik meg.

5.a. A metainformációk osztályozása

A metainformáció nem objektum, a tulajdonságai tehát nem metainformációk. A következő osztályozások teljesek, azaz pl. minden metainformáció beletartozik a címke, paraméter, reláció **technikai** csoportok valamelyikébe, de csak egybe:

Technikailag: címke, paraméter, reláció

Funkció szerint: az objektum megjelenését előíró, tartalomjegyzék-képző, domain-függő

Változási mód szerint: fix, explicit kezelésű, implicit kezelésű

5.a.1. Technikailag

Címke: az objektum többi objektumtól független tulajdonsága. Az elterjedt címkefelhőt valósítja meg. Rögzítendő

- a címke megengedett hossza, kódkészlete, stb.
- egy objektumnak lehet-e kétszer ugyanolyan címkéje,
- rögzített címkefelhője van-e egy portálnak (ki rögzíti), vagy bővíthet, esetleg csökkenhet.

Paraméter: az objektum többi objektumtól független tulajdonsága. Van egy neve és értéke. Pl. az objektum létrehozásának ideje:

Time=hh:mm:ss dd.mm.yy

- A paraméterkészlet egy portálon belül rögzített, de a különböző objektumokhoz más-más paraméterek lehetnek kötelezőek, vagy nem használatosak.
- A paraméterek értékének formátumát, értékhatárát, stb. rögzíteni kell.
- A paraméterek nagy része *értéklistás*, értékeiket egy véges készletből kaphatják. A készlet is a portál ontológiájának része.
- Lehetnek *ismételhető* értékű paraméterek, pl. egy cikknek több szerzője lehet.

Legfontosabb paraméterek az objektum különféle szempontú kategóriái, mint a 2. fejezetbeli *értéklistás "típus"*.

A könyvári állományok metainformációit ajánlások rögzítik, pl a Dublin Core, 2002-2004 években. A könyvtárszakmában gyakran metaadat a nevük.

Reláció: az objektumok közötti kapcsolatok

Ezek *bináris relációk*, az egyes objektumok – általában egész objektumosztályok – közötti kapcsolatot írják le.

Többféle bináris reláció létezik egy tartalomban egyidejűleg, becslhető számuk 10-30. Fontosak az aszimmetrikus bináris relációk, amelyek ellentétpárokkal fogalmazhatók meg: pl. eleme-tartalmazza, oka-következménye, kérdés-válasz, összefoglalás-példa, megelőző-következő. (Eldöntendő, hogy ezek a reláció *típusai* legyenek-e és az azonos típusúakat majd a *név* azonosítja, vagy ezek a reláció *nevei*, és *típusuk* nincs is.)

A bináris relációk gráfot, az aszimmetrikus binárisok irányított gráfot jelölnek ki.

5.b. A paraméter típusú metainformációk áttekintése

A Dublin Core ajánlás (DC) 2002-ben készült a könyvtári keresés és adatcsere számára. 15 fő paramétert rögzített egy könyvtári – akkor még valóságos - dokumentum számára. Ezek egy része 'virtuális iroda' számára is hasznos.

Szimbólum	Magyarázat	K&I	Érték
Title (Cím, Név)	DC, emberi olvasásra		
Symbol (Helyi azonosító)	Technikai azonosító a virtuális téren belül. Ember és szoftver is olvashatja: használható, mint fájlnev, mappanév, mezőnév, index, stb.	K	ASCII
Identifier (Globális azonosító)	DC (URL, URI, ISBN, ISSN, DOI..)	Ki	Az azonosítás-szolgálat adja
IntendedAudience (Célközönség)		KI	==user
PublicationState (Megjelenés státusa)			==PublicationState
ContentNature (A tartalom jellege)		KI	==ContentNature
Subject (tárgyszavak)	DC	KI	
Publisher (Kiadó)	DC		Kiadó neve
AccessRights (jogosultságok)	A RWED kategóriákra 4 user-részthalmaz	KI	==user
CreationDate (Létrehozás dátuma)	Ld DC is!		
	Lay-out információk		

magyarázat: DC – a Dublin Core-ban is szerepel, magyarázat ott olvasható

K&I: K-kötelező, I-ismételhető

érték: ==*ontológia* A paraméter a megnevezett *ontológiából* vehet föl értékeket.

CreationDate: A DC első megfogalmazása 10 különféle dátumot javasol: *creation, issuing, acceptance, ... last_metaadat_modification*. Ezek mind a DC-beli ismételhető **Date** paraméter értékei. Itt példaként egyet használunk.

Access: Az olvasási jog köre bővebb, mint a **célközönség**. A többi jog független a **célközönségtől**.

Ebből látszik, hogy a metainformációk egy részét ontológia alapján, más részét egyéb információk alapján kell kitölteni.

5.b.1. A tartalomjegyzék-motor

A táblázat alapján – például - a következő osztályozási szempontokat (**osz**-okat) lehet kiválasztani a kezelői felületen, magyar változatban:

Név szerint Célközönség szerint Megjelenés státusa szerint A tartalom jellege szerint Tárgyszavak szerint

Kiadó szerint Létrehozás dátuma szerint Kiadó és a Tartalom jellege szerint Tartalom jellege és a Kiadó szerint
--

Nem mindegyikük fog biztosan megjelenni, már itt sem szerepel mindegyik. Magyarázat a [Példák egyszerű tartalomjegyzékekre. A virtuális tér változatai](#). fejezetben. Az itt látottn kívül lehetnek összetett **osz**-ek is.

Az **osz** kiválasztását menedzselő és a tartalomjegyzéket előállító szoftver a *contents engine*.

Pontos specifikációja későbbi feladat. Addig kézzel rakunk össze egyszerű tartalomjegyzékeket.

5.b.2. Az objektumok háromféle azonosítása

- 1) **Title** (cím) emberi olvasásra. A virtuális téren (szolgáltatáson) belül egyedi. A szolgáltatás tervezője adja.
- 2) **Symbol** (technikai azonosító) emberi és gépi olvasásra. A virtuális téren (szolgáltatáson) belül egyedi. Rövid, alkalmasnak kell lennie technikai, gépi azonosításra, pl. rövid mappanévnek. A szolgáltatás tervezője adja.
- 3) **Identifier** (globális azonosító). Azonosítás-szolgálat adja.

Várható az, hogy a szolgáltatásnak (portálnak) is kell 3 egyedi azonosító, jelenleg a mi céljainkra nincs. Az url globális azonosítónak jó lenne, de a jogi szabályozás ezt nem tekinti annak jelenleg. Számunkra a kollaboratívításnál lesz fontos, ahol több virtuális irodában kell egyszerre dolgozni, és az objektumok neve ütközhet, 'Szolgáltatásazonosító.Symbol' formában lesz egyedi.

5.b.3. Lay-out-információk: az objektumok megjelenése

5.b.3.a. Natív mód paraméterei

Az objektum megjelenésével kapcsolatos tulajdonságok. Azt befolyásolják, hogy az objektum – pl. egy hosszabb szöveg – hogyan ismerhető föl hamar, ha a címét nem látjuk. Könnyen HTML-típusú objektumokra valósítható meg. Az office-típusúakra meg kell vizsgálni a lehetőséget.

- Ablak típusa, keretének színe, stb. (Ha lehetne, az ablak formáját is meg kellene tudni adni – szív, kör, sokszög alakú ablak pl.)
- A háttérszín, vagy vízjel.
- Pozicionálás típusa: scroll-mode, more-mode, ...

5.b.3.b. Rövid-mód paraméterei

- *Ikon*. Megfelel a fájlrendszer fájl típusaihoz tartozó ikonoknak.
- *Cím*. DC, ld. előbb.
- *Létrehozás dátuma*. DC, ld. előbb.
- *Lejárat dátuma*.
- *Tulajdonos, létrehozó*.
- ...

5.b.3.c. Mappa-mód paraméterei

Azt, hogy egy objektum mappaként viselkedjen, azaz mappa-módban jelenjen meg, az objektumok közötti *relációk* döntik el. Az itt leírt *paraméterek* azt határozzák meg, hogy ebben az esetben hogyan nézzen ki.

- *Logo*. fejezethez tartozhat, ha az egy önálló, reklámozható tartalmat hordoz.
- *Szlogen*. Egy-két szavas jellemzés az itt található dolgokról. A logo szöveges megfelelője.
- *Magyarázat*. Arról, hogy a lap vagy directory mit tartalmaz, ill. mit tudok ott elintézni, ha interaktív lapról van szó. Ide tartozik a publikációkban megszokott fejszöveg (vagy survey) is hagyományos dokumentumok estén.
- *Multimédia*. Az ikon-ra vagy a logo-ra kattintva elindul. Leginkább rövid reklám.

- *Dinamika.* Mutatja, hogy a *fejezet* statikus, vagy dinamikus-e, és ha dinamikus, hányas szintű ügyintézészt biztosít (ld. később).
- A következőkben megadott adatok a mappa megjelenésének részletet írják le. Öröklődnek a fa alja felé, ha az ott lévő mappák adatai nem mondanak mást
 - *fa-struktúra mód.* Az objektumok **rövid módon** jelennek meg. Ez a mód hagyományos object browser-t valósítja meg.
 - *lépcsős mód.* Az objektumok **natív módon** jelennek meg a fa mélysége szerint eltolva. Ezzel a móddal pl. fórum-gépet lehet csinálni.
 - *ömlesztett mód.* Az objektumok **rövid módon** jelennek meg, de nem fa struktúrában. A tartalmazó mappa is látható, ahogyan egyes keresők megjelenítik a megtalált elemek tartalomjegyzékét.
 - *vektor mód.* Azonos típusú és méretű objektumokat jelenít meg, 1,2,3, stb. dimenzióban. Egyetlen mélységet ábrázol a fából, vagy az *ömlesztett módot* ábrázolja. Több dimenzióba való tördelés szabályait ki kell munkálni. A vektor dimenzióit a képernyő jól láthatóan mutatja.
 - *dokkolt* Az alá tartozó objektumok dokkoltak, a virtuális térben – és így a képernyőn is – egymáshoz hézagmentesen illeszkedve fordulnak elő.
 - *Lista mód.* Az *ömlesztett mód* módosulata, ahol a lista a rövid módnál látott metainformációk szerint rendezve/szűrve/csoportosítva van.

Egy objektum különböző relációra nézve különböző tartalmú mappa is lehet. Tehát ezek az információk annyi példányban kellene egy objektumhoz, ahány reláció van a portálon.

Ide tartoznak a lay-out-sablonok is, ld. később.

5.c. A reláció típusú metainformációk

Ezek melyek leírják, hogy bizonyos relációra nézve az egyik objektum kapcsolatban van-e egy vagy több másikkal, azaz mappa-e. A mappáról azt képzeljük, hogy “tartalmazza” az alá tartozó dolgokat. A reláció itt nem csak “tartalmazás”, hanem pl. “következmény”, “szükséges feltétel” értelmű is lehet.

5.c.1.a. Felhasználóhoz kötődő adatok

Az objektumoknak vannak olyan adatai, amelyek felhasználónként másak. Úgy lehet tekinteni, hogy az objektum relációban van a ‘felhasználó’-t leíró objektummal (ami általában egy táblázat).

- *Megjegyzés.* Az egyes objektumokhoz megjegyzéseket fűzhet a felhasználó. Lehet közöttük sablon (pipa, fityisz, dátum, stb.) és szabad szöveg.
- *Látta.* Azt mutatja, hogy a lapot a felhasználó látta-e. Értékei: *new* – nem látta, *modified* – az utolsó verziót még nem látta, *deleted* – már nem láthatja, *moved to* - a fa más helyén láthatja. Ha a lapot már látta, a marker nem jelenik meg.
- *Jogosultságok.* Az objektum egyes tulajdonságát milyen szereplő módosíthatja.
- *Kollaboráció adatai.* (Jogosultságok, log, history, stb.)
- *Megjegyzés.* Az egyes objektumokhoz megjegyzéseket fűzhet a felhasználó, amelyek látszanak, vagy rendezési/csoportosítási adatként jelennek meg. Van közöttük előre definiált (pipa, fityisz, dátum, stb.) és szabad szöveg.
- *Látta.* Azt mutatja, hogy a lapot a felhasználó látta-e. Értékei: *new* – nem látta, *modified* – az utolsó verziót még nem látta, *deleted* – már nem láthatja, *moved to* - a fa más helyén láthatja. Ha a lapot már látta, a marker nem jelenik meg. (Ez korrektil csak a felhasználót azonosító tartalmaknál oldható meg.)
- *Jogosultságok.* Az objektum egyes tulajdonságát milyen szereplő módosíthatja.
- *Tulajdonos, létrehozó.*
- *Kollaboráció adatai.* (Jogosultságok, log, history, stb.)

5.c.1.b. Relációk megjelenítésének beállítása

A relációk grafikus megjelenítésnek számos szellemes módja van, nyilak, pipák, buborékok, stb. Ez a virtuális tér ergonómiájához tartozik, a piac értékelné a jó és szabványosítható megoldásokat. A paraméterek a megjelenítési módokat szabályozzák, beállításuk a content_builder felhasználó jogosultsága.

5.c.2. Tartalomjegyzék és mappa (nem ide való a fejezet, hagyd most ki)

Minden tartalomban kell lennie legalább egy speciális részben-rendezési relációnak, olyannak, ahol

- ❖ a gráf *fa*,
- ❖ az összes objektumra kiterjed
- ❖ és a reláció neve/típusa: "eleme-tartalmazza" értelmű.

Az ilyen relációk szerint képződnek **automatikusan** a *mappák* az objektumok *metainformációiból*. És *neveiből*. A legfelső szintű mappát **tartalomjegyzéknek**¹ hívjuk. Ha több alkalmas reláció van, több tartalomjegyzék, azaz mappa-hierarchia létezik. Ilyenkor felhasználó választhat közülük. A mappák tételei hagyományos értelemben vett linkek.

Mappa tehát minden olyan objektum, amely a kiválasztott reláció tekintetében más objektumokat tartalmaz. Ebből adódik, hogy egy objektum valamely relációra nézve lehet mappa, esetleg tartalomjegyzék, más relációra nézve egyszerű objektum.

A tartalomjegyzék és mappa megjelenítése sarkalatos kérdés, az objektum arculata (lay-out-ja) határozza meg. Ezek a hagyományos HTML-lap utódai.

5.d. Mi van képernyőn végül is?

Elfelejtjük a site-map és a menü fogalmát. Helyette a **mappa (directory)** fogalmát vezetjük be, ill. rehabilitáljuk. A legfelső szintű mappa a **tartalomjegyzék (contents)**, ez mutatja a portál egészét. Pontos definíció később.

A képernyőn kétféle dolgot láthatunk: **mappát**, melyben az objektumok címe, és egyéb, mappában elérő metainformációi vannak, és magukat az **objektumokat**. A mappa tulajdonképpen a virtuális tér vagy annak részlete valamilyen felbontással. A mappa összetett objektum.

Objektum tehát megjelenhet többféle módon, attól függően, hogy az *aktuális rendezési relációra vonatkozó mappaként vagy egyszerű objektumként* viselkedik:

- **natív mód** amikor nem mappa. Hagományos megjelenés egy pozícionálható ablakban, mint egy dokumentum, vagy a képernyő fix helyén, mint egy logo.
- **rövid-mód** amikor nem mappa, hanem valamely mappa egy tételeként jelenik meg. Leginkább az ikonja és a neve jelenik meg, az Intézőnél a dátuma, stb. (Itt lehetnek változatok: miniatűr, mozaik-, ikon-mód, mint az Intézőnél.)
- **mappa-mód**. Ha az objektum mappaként viselkedik. Mutatja az alá tartozó objektumoknak - általában – a rövid módját.

5.e. Két fő attrakció: az osztályozás és a lekérdezés

Mindkettő elsősorban a metainformációk alapján történik, lehetséges szempont még a Cím, és – amennyire meg lehet oldani – a tartalom².

5.e.1. Az osztályozás (tartalomjegyzék-képzés)

Az osztályozás egy tartalomjegyzéket állít elő az **osz** szerint. Alapeltv, hogy a tartalomjegyzékben az összes objektum szerepel¹, és mindegyik csak egyszer, ahogyan ezt egy jól nevelt tartalomjegyzéktől több száz éve

¹ Eszerint itt névütközés van: tartalomjegyzék az egész strukturált lista is, és az egész listát tartalmazó legfelső szintű objektum is. Ezt még tisztázni kell.

² A tartalomban való kereséshez különböző szabadszoftver-modulokat használnak a web-fejlesztők.

megszoktuk. Más szóval a virtuális térből nem tűnnek el, és benne nem jelennek meg objektumok felhasználói akció nélkül. Ez is a mellékhatás-mentesség elvének egy következménye.

Ha a dokumentumok halmaza változik, vagy a metainformációik változnak pl. valamely ügy intézése során, a tartalomjegyzék elavul. Célszerű, ha a tartalomjegyzéket a motor automatikusan megújítja.

5.e.2. A lekérdezés (keresés)

A lekérdezés eredménye egy linkgyűjtemény, amelyik a találatokra mutat – ez általában most is így van. Amivel itt többet tudunk, az a szemantikus szempontok szerinti keresés: pl. csak azokat a szövegeket keressük, amelyek definiálják a keresett fogalmat, másokat, akik beszélnek róla, nem.

A linkgyűjtemény többféleképp jeleníthető meg:

- mint egy speciális dokumentum, amelyik a virtuális tér valami erre kijelölt zugában (mappájában) van. A linkre kattintva megjelenik a talált dokumentum, ebből a szempontból mappaként viselkedik, de nem az. Természetesen a találatok szövegkörnyezetét illik mutatni, amire számtalan jó szabadszoftveres megoldás van az interneten. (Ez olyan, amit kritika nélkül át kell venni – ha technikailag lehet.)
- Megjeleníthető úgy is, hogy a tartalomjegyzék egyes tételei mellé jelet teszünk, a jelnek mutatnia kell, hogy melyik keresési feltétel találta meg, stb.

A keresések eredményét el kell tenni az erre kijelölt mappába.

Ha a dokumentumok halmaza változik, vagy a metainformációik változnak pl. valamely ügy intézése során, akkor a keresés eredménye elavul. Ezt kezelni kell valahogy.

Ha 4. és 5. fejezetbeli dolgokat rendesen megoldjuk, tkp. egy ontológia-motor, és egy adat-vizualizáció réteg egymásra épülése. A motor nagyon hasonlítana a Topic Maps típusú ontológiamotorokra:

http://en.wikipedia.org/wiki/Topic_Maps

Érdemes keresni ebben gyakorlott szakembert, és egy szabadon hozzáférhető motort és fejlesztői technológiát, amire építkezni lehet. Akkor az eddig leírtak nem pontosan, hanem csak lényegében ilyenek lesznek, mert hasonulnak a technológiához.

Sajnos az elnevezések is mások lesznek akkor, ez nem kerülhető el.

¹ Ebbe a fix objektumok, mint a portál címe vagy a logo, nem számítanak bele, ezek csak a content-builder szerepű felhasználó számára jelennek meg, mint tartomelemek, mert ő módosíthatja, más szereplő számára nem is lenne érthető, hogy mit lát. Ennek az ideológiáját még ki kell találni.

6. Függelék: Ontológiák kezdeti megfogalmazása

Ebben a fejezetben a paraméter típusú ontológiákkal foglalkozunk.

Az ontológia jelen kezdeti, kísérleti megfogalmazásában fogalmak neveinek egyszerű felsorolása, pontosabban a felsorolások fa-struktúrája, a hozzájuk fűzött magyarázatokkal együtt. A használható szintre kiépült ontológia néhány ezer fogalomból áll majd, és a struktúra bonyolultabb lesz.

Egyik iránya a bonyolultságnak, hogy a címszavakra egyszerre több fa-struktúra épül majd – ezt legjobban a tezaszók valósítják meg. Másik megoldás lehet, hogy ez az ontológia a kétféle objektumtípus – pl. a 'felhasználó' típus és 'tulajdonság' típusok - közötti kapcsolatokról álló mátrix. Sőt, lehet, hogy egyes 'felhasználók' más 'tulajdonságúak' a szolgáltatás különböző részeivel (moduljaival) kapcsolatban, akkor az ontológia a 'felhasználók', 'tulajdonságok' és a 'modulok' közötti 3-elemű kapcsolatokról álló 3d-s adathalmaz. Ilyen feladatokra a Topic Maps típusú ontológiák alkalmasak.

A magyarázatok rendszere, ami követi a fogalmak struktúráját, a szolgáltatás része. **A magyarázatok didaktikus és korrekt megfogalmazása a szolgáltatás minőségének fontos eleme, és a szolgáltatás építésében jelentős költség.**

6.a. Felhasználói szerepek

User

Felhasználó mindenki, aki bármilyen okból a szolgáltatást interaktív módon használja. Lehetnek nem interaktív felhasználók is, akik feljogosítanak más felhasználót (pl. látássérültek az ügysegédet) ügyeik intézésére, ezt most nem tárgyaljuk. Lehetnek gépi felhasználók is, nem tárgyaljuk itt.

Célszerű, ha a felhasználói szerepek rendszere *teljes*, tehát átfedés- és hézagmentes. Az, hogy egy felhasználó milyen szerepben dolgozik éppen, az a bejelentkezéskor, a szolgáltatás jogosultsági rendszeréből derül ki. Egy felhasználónak lehet egy időben több szerepe, azaz több jogosultsága is, itt lehetséges átfedés.

User.sp (service provider, szolgáltató) A szolgáltató üzleti vállalkozás, nonprofit szervezet, közösség, kormányzatszerv, stb.

User.sp.decision_maker Aki a szolgáltató szervezetében a szolgáltatással kapcsolatos döntéseket hozza. A döntés vonatkozhat a **technical support**, a **crm**, a **process owner** hatáskörébe tartozó dolgokra is, ő ezeknek a főnöke. De vonatkozhat a szolgáltatás fejlesztésére, vagy leselejtezésére is, ő az **sb** megrendelője. A felhasználói szerepek ontológiája tehát valamilyen mértékben leképezi a szolgáltató szervezeti felépítését.

User.sp.technical_support - aki a szolgáltatás technológiai működtetéséért felel.

User.sp.technical_support.system_manager - rendszergazda

User.sp.crm (ügyfélkapcsolat-kezelő)

User.sp.crm.call_center

User.sp.process_owner Az üzleti vagy munkafolyamat felelőse a szolgáltatónál. Általában középvezető

User.sp.domain – a domainspecifikus felhasználók tartoznak ide, hitelbíráló, diszpécser, stb. Ezek mélysége is lehet nagy.

User.sb (service builder, szolgáltatásépítő) Ide tartoznak leginkább a fejlesztők.

User.sb.decision_maker Hagyományosan fejlesztésvezető, vagy CIO.

User.sb.business_analyst (folyamatelemző, -szervező) Aki a szolgáltatás üzleti folyamatait tára föl, ill. tervezi.

User.sb.workflow_analyst (folyamatelemző, -szervező) A nonprofit a közösségi és kormányzati szolgáltatóknál - ugyanaz, mint a Business Analyst

User.sb.content_builder (tartalomépítő) Aki a szolgáltatás hagyományos – 3G,4G - programozási eszközök/ismeretek nélkül építhető részeit építi: dokumentumokat, ontológiákat készít, a szolgáltatást paraméterezi, stb.

User.client (ügyfél) A szolgáltatás használója

User.client.decision_maker (döntéshozó) Ha az ügyfél szervezet, akkor nála is lehet ilyen.

User.client.data_owner (adatgazda) – A szolgáltatás által kezelt adatok tulajdonosa. Általában a szolgáltatás ügyfele, és általában a személyéhez köthető adatokról van szó.

User.client.data_processor (adatifeldolgozó) – Aki a szolgáltatás (az ügy, és nem a szolgáltató) adatait bármely okból technikailag kezeli. Általában a szolgáltató alvállalkozója, PaaS-, SaaS-, vagy felhőszolgáltató. (A jog még nem tisztázta, hogy az ISP, akinek a hálózati berendezése az interaktív információ titkosítva áthalad, adatifeldolgozó-e.)

User.qs (quality supplier) A szolgáltatás minőségét biztosító szervezeti egység

User.qs.sec (security office) A szolgáltatás biztonságát biztosító szervezeti egység. Lehet, hogy szervezetileg a cégbiztonság alá tartozik.

User.qs.sec.decision_maker (hagyományosan: security_chief_officer)

User.qs.sec.operator (Biztonsági operátor)

User.qs.sec.controller (belső ellenőr, "fék a rolleren")

User.qs.sec.auditor (biztonsági audit)

User.gest - Olyan felhasználó, aki nem azonosítja magát, azaz nem jelentkezik be.

.....

6.b. Hogyan kell mindezt megjeleníteni és értelmezni?

Az ontológia ilyen primitív változatát viszonylag könnyű átlátható szöveggé formázni, több módon is. A bonyolultabb ontológiáknál ez nincs így, ez a körülmény az egyik akadálya az ontológiák terjedésének. Az előbbi formázásban a "pont" azt fejezi ki, hogy az alsóbb kategóriába tartozó dolog részalmazza a felsőnek: a **User.qs.sec.auditor** részalmazza tartozó szerep egyúttal a **User.qs.sec**-be is tartozik, stb.

Az egymás mellett lévő részeknek nincs egymással kapcsolata.

A fenti *folyó szöveges* megjelenítésnél jobban olvasható, de technikailag nehezebben kezelhető a *strukturált megjelenítés*. Képernyőre is az való, ld. lentebb, de ki kell találni, hogy a "pont" hol legyen.

NB: ez az ontológia nem a felhasználókat, hanem a felhasználói tevékenységeket, a szerepeket osztályozza. Egy felhasználó egyszerre több szerepben is dolgozhat. Az összeférhetlenségek szintén kifejezhetők komplikáltabb ontológiákkal.

NB: osztályozás nem csak halmaz-részalmaz viszonyt fejezhet ki. Például a dokumentumok között kérdés-válasz, ok-okozat, tankönyv - specifikáció - összefoglalás, koncepcióterv - jóváhagyás - kiviteli terv - megvalósíthatósági tanulmány, és egyéb komplikáltabb összetartozási viszony is lehet. Ezeket mind ki lehet fejezni az ontológiákkal, erre sok példa van az irodalomban.

Amire kevés a példa, és ami ennek a kutatásnak a lényege, hogy ezeket a viszonyokat valahogyan átlátható, ergonómikus módon meg kell jeleníteni a virtuális térben. Bizonyos mélységig esetleg papíron is. Ezt a kérdéskört hívjuk *adat-vizualizációnak*. Ennek hiánya a szolgáltatások használhatóságának egyik akadálya.

User

sp (service provider, szolgáltató)

decision_maker (üzemeltetésvezető)
technical_support (üzemeltető munkatárs)
system_manager (rendszergazda)
crm (ügyfélkapcsolat-kezelő)
call_center
process_owner (folyamatgazda)

sb (service builder, szolgáltatásépítő)

decision_maker (fejlesztésvezető, CIO)
business_analyst (folyamatelemző, -szervező)
workflow_analyst (folyamatelemző, -szervező)
content_builder (tartalomépítő)
software_developer (fejlesztő)
architect (architekt)
UX designer (használhatósági tervező)

client (ügyfél, a szolgáltatás használója)

decision_maker (üzletágvezető)
data_owner (adatgazda)
data_processor (adatfeldolgozó)

qs (quality supplier)

sec (security office)
decision_maker (hagyományosan: security_chief_officer)
operator (biztonsági operátor)
controller (belső biztonsági ellenőr, "fék a rolleren")
auditor (biztonsági audit)
ux (felhasználói élmény tesztelője)
auditor (használhatósági elemző)

guest (vendég. Olyan felhasználó, aki nem azonosítja magát, azaz nem jelentkezik be.)

6.c. Fontos: doménfüggő és doménfüggetlen értelmezés

A **User** kategória – és minden alkategóriája - kétféle embercsoportot jelenthet. Egyrészt jelentheti annak az e-szolgáltatásnak a felhasználóját – és minden alkategóriáját, pl. a chief_security_officer-jét - amelyik ezzel az ontológiával épült. Ezt fogjuk *doménfüggő értelmezésnek* hívni. Ennek a fejlesztésnek a végső célja nyilvánvalóan ez: ilyen e-szolgáltatások épüljenek.

Másrészt jelentheti mindazokat, akik magukat általában felhasználónak, ezen belül pl. valamely e-szolgáltatás chief_security_officer-jének tartják - ez nyilván bővebb csoport.

Ez a két értelmezés az ontológiák mindegyikében létezik. A projektnek ebben a megalapozó fázisában egyelőre doménfüggetlen szoftver-technológiával foglalkozunk, tehát a második, a *doménfüggetlen értelmezést* használjuk.

Az itteni példák szintaxisában ezt az értelmezést egyelőre nem jelöljük.

6.d. A szolgáltatás életciklusa

Életciklusa nemcsak a szolgáltatás egészének, hanem, moduljainak külön-külön van. Tehát ennél az ontológiánál is azonnal fölmerül, hogy használható formájában valószínűleg a 'modul' és az 'life_cycle' típusok közötti kapcsolatokat leíró mátrix formájú lesz.

NB: az egymás után írt fő részek valójában időbeli, sorrendi kapcsolatban állnak egymással. Ez az ontológia tehát más természetű, mint a szerepeké. A *folyó szöveges* megjelenítésben a "pont" helyett mást kell kitalálni (de csak a baloldali "pont" helyett).

Life_cycle

- concept_planning,
 - functional plan
 - ontology plan
 - environment plan
- technical_planning,
 - system planning
 - ontology plan
 - usability plan
- implementing,
 - architecture
 - programming
- content_building,
 - testing,
 -
- turning_to_live,
 - using,
 - maintenance

6.e. Publikálás státusa

PublicationState

- manuscript (kézirat)
- in print (megelenés alatt)
- published (megelent)

6.f. Dokumentum tartalmának jellege

ContentNature

Scientific Paper (tudományos dolgozat)
Conference Paper (konferenciaelőadás)
Textbook (tankönyv)
Manual (kézikönyv)
Tutorial (bevezető tankönyv)
Essay (esszé)
R&D proposal (k+f javaslat)
Journalism (publicisztika)
Case study (esettanulmány)
Appeal (felhívás)
Call For Paper (conferencia-felhívás)
Staff (szakértőink)
Portal evaluation (portálértékelés)
Technical plan (technikai projektterv)
Financial plan (pénzügyi projektterv)
Call for application (pályázati felhívás)
Uncertified Audit (minősítés nélküli audit)
Certified Audit (audit bizonyítvány)

6.g. Felhasználók

A felhasználók ontológiája nem más, mint a jogosultsági rendszerbeli táblázat, ahol a regisztrált felhasználók nyilván vannak tartva. A felhasználó a mi terminológiánk szerint nem objektum, hanem aktor, a jogosultsági táblázat őrá vonatkozó lapja az objektum (vagy az egész táblázat maga összetett objektum, ha látható táblázatként egyben létezik a virtuális térben).

6.h. Műveletek a virtuális tér objektumain

create, annotate, read, verify, modify, encrypt, decrypt, sign, compare, update, delete, synchronize, endorse, file, group, ungroup, classify, authorize, grant, publish, revoke, sort, merge with, split, concatenate, stb.

Ezek tehát nem „kommunikációk a géppel”, hanem *műveletek az objektumokon (Operations on the objects)*. A kommunikáció a pragmatika tárgykörébe tartozik és az aktorok (felhasználók, sw-agent-ek) közötti üzenetváltásról szól.

7. Tudásbázis kezdetei: néhány dokumentum és metainformációi

A portálba a következő dokumentumokat tesszük föl kezdetnek. A dokumentumok metainformációit a [A paraméter típusú metainformációk áttekintése](#) fejezet táblázata szerint töltjük ki.

Nem foglalkozunk most a logo-val, meg képernyő egyéb olyan elemével, amelyet egyelőre fixnek tekintünk.

1. Felhivas_E-szolgáltatások_v54.pdf

Paraméter	Paraméter értéke	Értéke magyarul (lokalizáció)
Title (Cím)	Appeal	Felhívás
Symbol (Helyi azonosító)	Felhivas_E-szolgáltatások_v54.pdf	
Identifier (Globális azonosító)	-	-
IntendedAudience (Célközönség)	User.sp.decision_maker User.sp.process_owner User.sb.decision_maker User.sb.content_builder	(üzemeltetésvezető) (folyamatgazda) (fejlesztésvezető, CIO) (tartalomépítő)
PublicationState (Megjelenés státusa)	PublicationState.manuscript	Kézirat
ContentNature (A tartalom jellege)	ContentNature.appeal	Felhívás
Subject (tárgyszavak)	UX, ontology, virtual space, e-service, service quality	Felhasználói élmény, ontológia, e-szolgáltatás, virtuális tér, szolgáltatásminőség
Publisher (Kiadó)		
AccessRights (jogosultságok)	R: User.guest	R: vendég
CreationDate (Létrehozás dátuma)	2014.11.01	2014.11.01

2. E-szolgáltatások Minősége Szakmai Közösség 2015 03 27.xls

Paraméter	Paraméter értéke	Értéke magyarul (lokalizáció)
Title (Cím)	Staff	Tagjaink
Symbol (Helyi azonosító)	Ide a fájlnev jön...	
Identifier (Globális azonosító)	-	-
IntendedAudience (Célközönség)	User.sp.decision_maker User.sp.process_owner User.sb.decision_maker User.sb.content_builder	(üzemeltetésvezető) (folyamatgazda) (fejlesztésvezető, CIO) (tartalomépítő)
PublicationState (Megjelenés státusa)	PublicationState.manuscript	Kézirat
ContentNature (A tartalom jellege)	ContentNature.staff	Szakértőink
Subject (tárgyszavak)	Accessibility	elérhetőség
Publisher (Kiadó)	-	-
AccessRights (jogosultságok)	?Doménfüggő, még ki kell dolgozni.	?

CreationDate (Létrehozás dátuma)	2014.11.01	2014.11.01
--	------------	------------

3. Napirenden az elektronikus szolgáltatások minősége

<http://computerworld.hu/computerworld/napirenden-az-elektronikus-szolgáltatások-minosege.html>

Paraméter	Paraméter értéke	Értéke magyarul (lokalizáció)
Title (Cím)	-	Napirenden az elektronikus szolgáltatások minősége
Symbol (Helyi azonosító)	CWI_2005	CWI_2005
Identifier (Globális azonosító)	? meg kell kérdezni	?
IntendedAudience (Célközönség)	User.sp.decision_maker User.sp.process_owner User.sb.decision_maker User.sb.content_builder	(üzemeltetésvezető) (folyamatgazda) (fejlesztésvezető, CIO) (tartalomépítő)
PublicationState (Megjelenés státusa)	PublicationState.published	Megjelent
ContentNature (A tartalom jellege)	ContentNature.journalism	publicisztika
Subject (tárgyszavak)	UX, ontology, virtual space, e-service, service quality	Felhasználói élmény, ontológia, e-szolgáltatás, virtuális tér, szolgáltatásminőség
Publisher (Kiadó)	CWI	CWI
AccessRights (jogosultságok)	R: User	R: bárki
CreationDate (Létrehozás dátuma)	2015.07.28	2015.7.28

4. NJSzT_portal_megujitas_2015_V1.doc

Paraméter	Paraméter értéke	Értéke magyarul (lokalizáció)
Title (Cím)	NJSzT portal redesign	NJSzT portál megújítása
Symbol (Helyi azonosító)	Njszt.hu_redesign	Njszt.hu_megújítás
Identifier (Globális azonosító)	-	-
IntendedAudience (Célközönség)	User.sp.decision_maker User.sp.process_owner User.sb.decision_maker User.sb.content_builder	(üzemeltetésvezető) (folyamatgazda) (fejlesztésvezető, CIO) (tartalomépítő)

Paraméter	Paraméter értéke	Értéke magyarul (lokalizáció)
PublicationState (Megjelenés státusa)	PublicationState.manuscript	Kézirat
ContentNature (A tartalom jellege)	ContentNature.technical_plan	Technikai projektterv
Subject (tárgyszavak)	UX, virtual space, e-service, service quality, firm strategy	Felhasználói élmény, e-szolgáltatás, virtuális tér, szolgáltatásminőség, szervezeti stratégia
Publisher (Kiadó)		
AccessRights (jogosultságok)	R: User	R: bárki
CreationDate (Létrehozás dátuma)	2015.07.01	2014.11.01

5. Webtudor_evaluation_2015.3.17.doc

Paraméter	Paraméter értéke	Értéke magyarul (lokalizáció)
Title (Cím)	Webtudor_evaluation	Webtudor_portálértékelés
Symbol (Helyi azonosító)	A fájlnev...	Webtudor_portálértékelés
Identifier (Globális azonosító)	-	-
IntendedAudience (Célközönség)	User.sp.decision_maker User.sp.process_owner User.sb.decision_maker User.sb.content_builder User.sb.UX_designer User.qs.ux	(üzemeltetésvezető) (folyamatgazda) (fejlesztésvezető, CIO) (tartalomépítő) (használhatósági tervező) (felhasználói élmény tesztelője)
PublicationState (Megjelenés státusa)	PublicationState.manuscript	Kézirat
ContentNature (A tartalom jellege)	ContentNature.portal_evaluation	Portálértékelés
Subject (tárgyszavak)	UX, HCI, e-service, service quality, firm strategy	Felhasználói élmény, HCI, e-szolgáltatás, szolgáltatásminőség, szervezeti stratégia
Publisher (Kiadó)		
AccessRights (jogosultságok)	R: User	R: bárki
CreationDate (Létrehozás dátuma)	2015.03.17	2015.03.17

6. Eves_beszámoló-munkaterv_Szakmai 2014-2015_v3.docx

Paraméter	Paraméter értéke	Értéke magyarul (lokalizáció)
Title (Cím)	Szakosztály munkaterve 2015	Szakosztály munkaterve 2015
Symbol (Helyi azonosító)	A fájlnev...	
Identifier (Globális azonosító)	-	-
IntendedAudience (Célközönség)	NJSZT-tagság. Tehát doménfüggő, és még nincs meg a hivatkozható ontológia.	?
PublicationState (Megjelenés státusa)	PublicationState.manuscript	Kézirat
ContentNature (A tartalom jellege)	ContentNature.r&d proposal	K+F javaslat
Subject (tárgyszavak)	UX, HCI, e-service, service quality, firm strategy	Felhasználói élmény, HCI, e-szolgáltatás, szolgáltatásminőség, szervezeti stratégia
Publisher (Kiadó)		
AccessRights (jogosultságok)	R: User	R: bárki
CreationDate (Létrehozás dátuma)	2014.12.05	2014.12.05

7. Our_motivation_examples_V25.doc

Paraméter	Paraméter értéke	Értéke magyarul (lokalizáció)
Title (Cím)	Our motivation example	Mi akadályozza az internet-használókat
Symbol (Helyi azonosító)	A fájlnev...	
Identifier (Globális azonosító)	-	-
IntendedAudience (Célközönség)	User.sp.decision_maker User.sp.process_owner User.sb.decision_maker User.sb.content_builder User.sb.UX_designer User.qs.ux	(üzemeltetésvezető) (folyamatgazda) (fejlesztésvezető, CIO) (tartalomépítő) (használhatósági tervező) (felhasználói élmény tesztelője)
PublicationState (Megjelenés státusa)	PublicationState.manuscript	Kézirat
ContentNature (A tartalom jellege)	ContentNature.case_study	Esettanulmány
Subject (tárgyszavak)	UX, HCI, e-service, service quality	Felhasználói élmény, HCI, e-szolgáltatás, szolgáltatásminőség

Paraméter	Paraméter értéke	Értéke magyarul (lokalizáció)
Publisher (Kiadó)	-	-
AccessRights (jogosultságok)	R: User	R: bárki
CreationDate (Létrehozás dátuma)	2015.4.10	2015.4.10

8. Példák egyszerű tartalomjegyzékekre. A virtuális tér változatai.

Tartalomjegyzék Cím szerint

A Cím nem ontológiából veszi az értékét, de *nem ismételhető* paraméter, ezért a tartalomjegyzék egyszerű felsorolás. Rendezés és alaosztás pl. abc szerint később lehetséges. A “...” a cím mellett megjelenítendő metainformációkra utal, amit még tisztázni kell.

Felhívás.... Tagjaink... NJSzT portál megújítása... Webtudor_portálértékelés... Szakosztály munkaterve 2015 ... Napirenden az elektronikus szolgáltatások minősége...
--

Tartalomjegyzék Helyi azonosító szerint

Tartalomépítőknek lehet rá szüksége. Olyan, mint a Cím szerinti.

Tartalomjegyzék Célközönség szerint

Nem megy, mert van dokumentum, amelyik több kategóriába is – pl. a user.sb-be és a user.sp-be is - beleszne, az pedig tartalomjegyzékben nem lehet. Más felépítésű ontológia kellene. Pedig ez egy fontos kategória, valamit ki kell találni erre.

Tartalomjegyzék a Megjelenés státusa szerint

kézirat Felhívás.... Tagjaink... NJSzT portál megújítása... Webtudor_portálértékelés... Szakosztály munkaterve 2015 ... megjelent Napirenden az elektronikus szolgáltatások minősége...
--

Természetesen, ilyen stílusú is lehet a tartalomjegyzék, ez jelenleg még mindegy:

kézirat Felhívás.... kézirat Tagjaink... kézirat NJSzT portál megújítása... kézirat Webtudor_portálértékelés... kézirat Szakosztály munkaterve 2015 ... megjelent Napirenden az elektronikus szolgáltatások minősége...
--

Tartalomjegyzék A tartalom jellege szerint

Szakértőink Tagjaink ...

Publicisztika
Napirenden az elektronikus szolgáltatások minősége ...

Technikai projektterv
NJSzT portál megújítása ...

Portálértékelés
Webtudor_portálértékelés ...

K+F javaslat
Szakosztály munkaterve 2015 ...

esettanulmány
Mi akadályozza az internet-használókat ...

Tartalomjegyzék Címszavak szerint

A címszavak nem ontológiából veszik az értéküket, viszont *ismételhető paraméterek*. Ezért ez itt nem megy, mert egy dokumentum több címszó alá is tartozhatna. Erre valamit ki kell még találni.

Tartalomjegyzék a Kiadó szerint

CWI
Napirenden az elektronikus szolgáltatások minősége...
-ergység-
Felhívás....
Tagjaink...
NJSzT portál megújítása...
Webtudor_portálértékelés...
Szakosztály munkaterve 2015 ...

Összetett tartalomjegyzék a Kiadó és a Tartalom jellege szerint

CWI
Publicisztika
Napirenden az elektronikus szolgáltatások minősége...
-ergység-
Szakértőink
Tagjaink ...
Technikai projektterv
NJSzT portál megújítása ...
Portálértékelés
Webtudor_portálértékelés ...
K+F javaslat
Szakosztály munkaterve 2015 ...
Esettanulmány
Mi akadályozza az internet-használókat ...

Összetett tartalomjegyzék a Tartalom jellege és a Kiadó szerint

Szakértőink	
-egyéb-	
	Tagjaink ...
Publicisztika	
CWI	
	Napirenden az elektronikus szolgáltatások minősége ...
Technikai projektterv	
-egyéb-	
	NJSzT portál megújítása ...
Portálértékelés	
-egyéb-	
	Webtudor_portálértékelés ...
K+F javaslat	
-egyéb-	
	Szakosztály munkaterve 2015 ...
Esettanulmány	
-egyéb-	
	Mi akadályozza az internet-használókat ...

További elhelyezendő dokumentumok

Virtual_office_concept_demo_v08.doc

Call_for_papers_2015.doc

Popularity_vs_Professionality_in_the_user_behavior_v2_3.doc

Terminology_and_Slogans_v1_3.doc

Technological_bases_of_the_usability.doc

JINKA2.3final.pdf

CognInfoComm2013_To_learn_from_other_sciences_Vitalyos.pdf

Research_plan_short

http://www.vitalyos.hu/ICon_project/Research_plan_short.pdf

Pragmatics in the Usability discipline

Axiomatic Foundation of the HCI, II.

http://www.vitalyos.hu/ICon_project/CognInfoComm2012_Pragmatics_in_HCI_Vitalyos_reviewed_V1_2.pdf

[The Object Permanency Principle in the Usability discipline](#)

