

Alkalmazásprofilok készítése RDA-hoz: Kísérleti projekt alkalmazásprofilok módszertanának kidolgozásához

Az alkalmazásprofilok fejlesztése nem új keletű dolog a könyvtárügyben sem, az RDA-keretrendszerre való átdolgozásával pedig még most aktuálisabb, mint valaha. A 3R projekt utáni RDA már nem lesz használható alkalmazásprofil és szabályzatok nélkül. Az alkalmazásprofilok kidolgozásának igénye azonban megelőzte a kidolgozásukhoz szükséges módszertan kikristályosodását. Az OSZK Könyvtári Szabványosítási Irodáján futó pilotprojekt célja egyrészt, hogy létrejöjjön egy általános RDA-alkalmazásprofil, másrészt, hogy feltérképezzék, milyen módszerek bizonyulnak a leghatékonyabbnak egy nagy elemkészletű keretrendszerhez készült alkalmazásprofil készítésénél.

Tárgyszavak: bibliográfia; katalogizálás; szabályzat; szabványalkotás

Bevezetés

A világon a legnagyobb strukturált metaadat-vagyonnal minden bizonnyal a könyvtárak rendelkeznek. Ennek ellenére a könyvtári metaadatok nem töltik be a metaadatok ökoszisztémájában azt a helyet, amelyet az adatvagyon minősége és mennyisége lehetővé tene. A könyvtári metaadatok nemcsak azért elszigeteltek a nem-könyvtári világ metaadataitól, mert más formátumot használunk, hanem azért is, mert eleve másképp gondolkodunk a metaadatainkról. „A technológia önmagában nem szükséges és nem elégséges, hogy lényegi változásokat hozzon, ha nem jár együtt szemléletbeli paradigmaváltással.”¹ Ez a gondolat a 2018-as IFLA-konferencián hangzott el a katalogizálási szekcióban. Mióta hallottam, minden releváns helyzetben idézem, mert véleményem szerint nagyon pontosan írja le a katalogizálás jelen helyzetét. A világon mindenhol látjuk, érezzük, tapasztaljuk, hogy a könyvtári katalógusnak változnia kell, mert a technikai fejlődés a használói igényeket is átalakította, és ha a könyvtárak relevánsak akarunk maradni, valamit tenni kell.

Na de minek is kéne lennie annak a bizonyos valaminek? Nem csak Peyard és Roche intenek óva attól, hogy egy „egyszerű” formátumváltásban lássuk a megoldást. Az IFLA-katalogizálást megújító „programjának” dokumentumaiban megjelenik ez az új szemlélet. Az FRBR¹ behozta a könyvtári köztudatba a funkcionális követelményeket, a

használó igényeit és feladatait középpontba helyező szemléletet. Bár talán inkább az entitásokat és a WEMI szintek² koncepcióját kötik inkább hozzá sokan. A 2016-os IFLA-kiadvány, a Nyilatkozat a nemzetközi katalogizálási alapelvekről már „figyelembe veszi a felhasználók új kategóriáit, a nyílt hozzáférésű környezetet, az adatok cserélhetőségét és elérhetőségét, a keresőeszközök sajátosságait, valamint általában a felhasználói viselkedés jelentős mértékű megváltozását.”³ Ezen felül a legszembetűnőbb terminológiai váltás, hogy már nem használja a rekord fogalmát, helyette bibliográfiai és authority adatról van benne szó.

Továbbvive az eddigieket, az IFLA 2017-ben közvételte az IFLA könyvtári referenciamodellt (LRM)⁴. Érdemes kiemelni az alcímét: A bibliográfiai információk elméleti modellje. Az LRM az FR-modell-család egyesítésével jött létre, összehangolva és továbbfejlesztve annak elemeit. Az LRM szellemiségét a felhasználói feladatok, entításalapú katalogizálás, illetve a rekordszintű adatrögzítés helyett az adatszintű adatrögzítés jellemzi.

Ezek a szemléletbeli változások nem azonnal kerültek át a gyakorlati munkába. Az FRBR első kiadása 1998-as, de a 2010-es évekig nem láthatunk FRBR szerinti feldolgozási gyakorlatokat. Ezek körül a mára leginkább elterjedt szabályzat az RDA (Resource Description & Access = Forrásleírás és –hozzáférés) első változatában nemcsak szellemiségében és fogalomhasználatában, de

struktúrájában is teljesen az FRBR-re épített. Ezért az FRBR-t meghaladó LRM közzétételekor szükségessé vált az RDA hozzáigazítása is az új modellhez. „Az RDA Toolkit újrastrukturálását és újratervezését célzó (RDA Toolkit Restructure and Redesign, 3R) projekt az RDA Toolkit szerkezetének és tartalmának néhány szükséges és kívánatos fejlesztését tartalmazza, beleértve egy standard dokumentummodell implementálását a tartalomkezelésre, az RDA integrációját az IFLA könyvtári referencia modellel (IFLA LRM), illetve a rendszer bővítését perszonalizációs funkciókkal.”⁵ Ez az újrastrukturálás és újratervezés a gyakorlatban azt jelentette, hogy az RDA-t nemcsak hozzáigazították az LRM-hez, de át is alakították a klasszikus katalogizálási szabványból bibliográfiai keretrendszerre. Ennek a szerkezetváltásnak a legfontosabb hozadéka, hogy a poszt-3R RDA-t a gyakorlati használat előtt szükséges „testre szabni”. Ezt alkalmazásprofilok és szabályzatok elkészítésével lehet megtenni. Az itt ismertetett projekt az előbbiről fog szólni.

Az alkalmazásprofilokkal kapcsolatban magyarul még nem jelentek meg írások. A csupán értelmező munkák írása mellett fontosnak tartjuk, hogy a primer források is elérhetők legyenek magyar nyelven is, így szándékunkban áll, a szerzők engedélyével persze, magyar fordításban közreadni a fontosabb alapidokumentumokat ebben a témában. Ezzel egyrészt az angolul nem beszélő kollégák információ-hozzáférést szeretnénk segíteni, másrészt szeretnénk már kezdettől fogva egy irányba tartani a kifejezések magyar változatát. Persze, hogy mi milyen formában terjed el, az a jövő zenéje.

Kutatásunkban nem állunk meg az elméletnél. Feladatunknak tekintjük, hogy a közösség részére kidolgozzuk a téma módszertanát. Saját tapasztalat szerzése érdekében egy pilotprojektet fogunk indítani. A projekt célja kettős: egy általános RDA alkalmazásprofil kidolgozása, illetve a kidolgozáshoz szükséges módszertan tesztelése.

Mi az alkalmazásprofil?

Definíciók

Az alkalmazásprofil mint fogalom nem új a könyvtári területen. Az elsőnek tekinthető definíció 2000-ből származik: „Az alkalmazásprofil egy vagy több névtérsémából származó adatelemek összekombinálása és egy bizonyos helyi alkalmazás számára optimalizálása. Az alkalmazásprofilok haszna,

hogy az implementáló kifejezheti vele, hogy miképp alkalmazza a standard sémákat.”⁶

Bár a fogalom már megjelent a szakirodalomban, sokáig nem került át a gyakorlatba, hiszen nem voltak olyan metaadatszabványaink, amelyek szerkezetileg túl nagy mozgásteret engedtek volna az alkalmazás során. Ezen változtatott a Dublin Core (DC) megjelenése és elterjedése. A DC fejlesztői kezdtek el először a gyakorlatban is foglalkozni az alkalmazásprofilok készítésével. Az ő definíciójuk szerint az alkalmazásprofil:

„Metaadat-elemek, szabályzatok és irányelvek csoportja, amelyet egy bizonyos alkalmazás kedvéért definiáltak. (...) Az alkalmazásprofil nem teljes dokumentáció nélkül. Ebben találhatóak az alkalmazásnak megfelelő szabályzatok és jó gyakorlatok”.⁷

A következő nagy változást az hozta meg, hogy az RDA katalogizálási szabványt átalakították bibliográfiai keretrendszerre. Az RDA jelenleg béta változatban lévő változata a mindennapi használatához megköveteli az alkalmazásprofilok és szabályzatok létrehozását. Így az alkalmazásprofilok áttekintettek a web lazább szabványokkal operáló világból a könyvtári világ tradicionálisan kötöttebb mintát követő szabványaihoz is.

„Az alkalmazásprofil specifikálja azokat az entitásokat, elemeket és szótárkódolási sémákat, amelyek szerepelhetnek olyan metaadatkészletekben, amelyek megfelelnek a metaadatot használó alkalmazás funkcióinak és követelményeinek.

Egy RDA alkalmazásprofil ezen felül kiterjedhet még:

- az elem rögzítéséhez használt rögzítési módra,
- az elem használatára vonatkozó opcionális instrukciókra,
- az elemre vonatkozó szabályzatra.”⁸

Az alkalmazásprofil készítése nem is áll annyira távol a könyvtári gyakorlattól, mint azt első hallásra gondolnánk. Tulajdonképpen a házi szabályzat is egy alkalmazásprofil, bár eléggé hiányosan dokumentált. Hiányzik belőle az a formalizáltság, amely elősegítheti egyrészt az újrafelhasználást másrészt a crosswalk⁹-ok készítését.

Alkalmazásprofil a szakirodalomban

Szingapúri keretrendszer

Saját definíciója szerint: „A Szingapúri keretrendszer Dublin Core™ Alkalmazásprofilok számára

egy keretrendszer olyan metaadat-alkalmazások tervezéséhez, amelyek a lehetséges legnagyobb interoperabilitással bírnak, és ezen alkalmazások dokumentálásához, a maximális újrafelhasználhatóság érdekében. A keretrendszer definiálja azokat a leíró komponenseket, amelyek szükségesek vagy hasznosak egy Alkalmazásprofil dokumentálásához, és leírja, hogy ezek a dokumentációs szabványok hogyan kapcsolódnak szabványos szakterületi modellekhez és a Szemantikus web alapját adó szabványokhoz. A keretrendszer alapot teremt annak is, hogy egy Alkalmazásprofil felülvizsgálhassunk a dokumentációja teljessége és a webarchitektúra alapelveinek megfelelése szempontjából.¹⁰

A Keretrendszer megfogalmazta az alkalmazásprofil-készítés 1. ábrán látható szakaszait, illetve vázolta ezek egymásra épülését, és a kapcsolatrendszerét a metaadatvilág egyéb részeivel. Úgy is fogalmazhatnánk, hogy a Szingapúri keretrendszer olyan szerepet tölt be az alkalmazásprofil-készítés módszertanának kidolgozásában, mint a Keret-

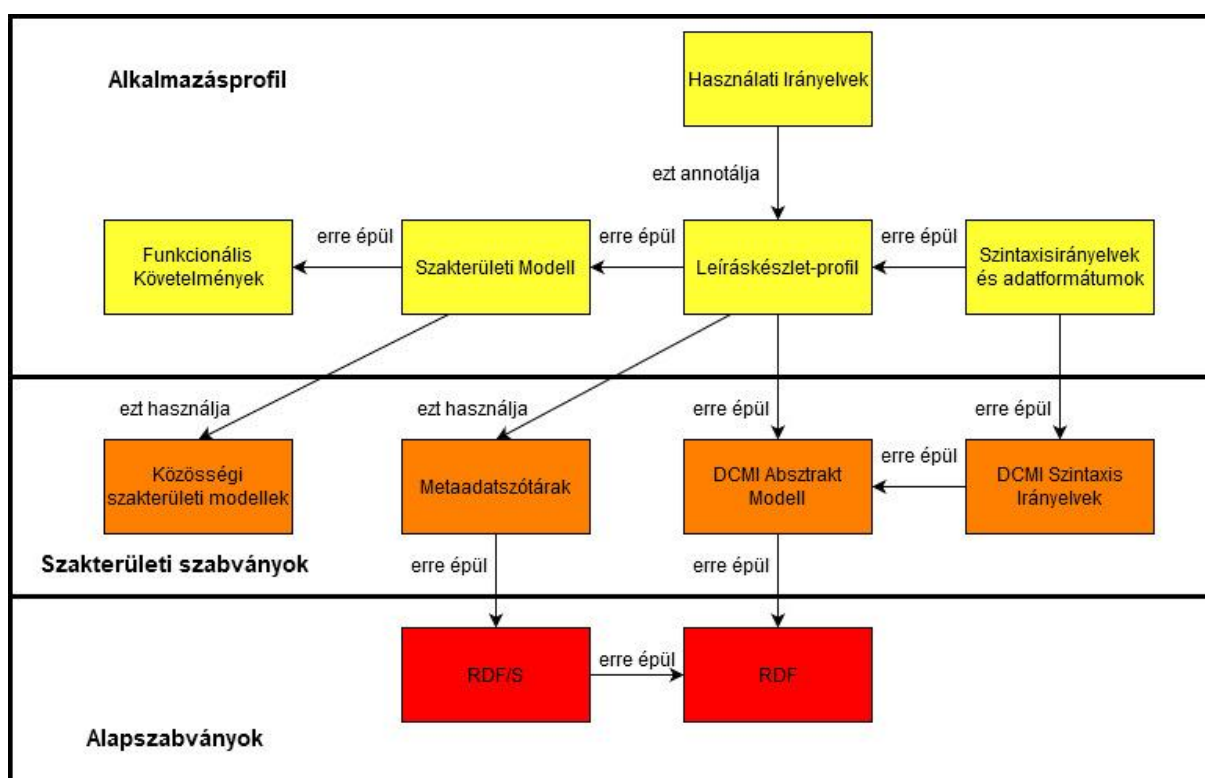
rendszerben a Szakterületi modell az alkalmazásprofil-készítési folyamatban.

DCMI-Irányelvek alkalmazásprofilokhoz¹¹

A 2009-ben íródott DCMI-Irányelvek ma is az alkalmazásprofilok könyvtári alkalmazásának kikerülhetetlen alapdokumentuma. A dokumentum a Szingapúri keretrendszer által felvázolt lépéseket részletesebben tárgyalja, illetve egy példaalkalmazásprofil készítésének folyamatán végig is visz. A példa terjedelmi okok miatt természetesen nem túl komplex, csak illusztrációs célokat szolgál.

A Szingapúri keretrendszer kiegészítései

A Szingapúri keretrendszer időtállóan bizonyult. Az alkalmazásprofilokkal foglalkozó frissebb szakirodalom nem vitatja a keretrendszer által felvázolt szakaszokat, hanem csak kiegészítéseket tesz hozzá. Az írók szerzői között vannak, akik csak a szakaszok egyik részére fókuszálnak, de olyan is van, aki az egészhez teszi hozzá a kiegészítéseit.



1. ábra Szingapúri keretrendszer

Ez utóbbi kategóriában tartozik a Me4DCAP¹². Ahogy a neve is mutatja (DCAP = Dublin Core Application Profile azaz Dublin Core alkalmazásprofil), nagyban épít a DCMI munkájára, elismerve, hogy ezek úttörő munkák voltak, másrészt arra is rávilágítva, hogy ezek az anyagok még önmagukban kevesek. Módszerük kidolgozásakor merítettek még a szoftverfejlesztési folyamatokból is. Ezeknek főleg a kezdeti szakaszaiból, ahol az adatmodell kidolgozása történik. A módszer a Szingapúri keretrendszer szakaszait helyezi a középpontba és egészíti ki extra szakaszokkal. Ezen extra szakaszok célja jellemzően az, hogy úgymond előkészítsék a terepet. A beiktatott szakaszokban ugyanis tulajdonképpen segédanyagokat készítünk a Szingapúri keretrendszer-szakaszokhoz, hogy azok könnyebben kivitelezhetőek legyenek.

A téma iránti érdeklődés pár évre leült, de 2019-ben több új megközelítésű kutatás is foglalkozott az alkalmazásprofil-készítés módszereivel. A legújabbnak számító kutatást, a Design for Simple Application Profiles-t (= Egyszerű alkalmazásprofilok tervezése) ugyanaz a szerzőpáros jegyzi, mint a DCMI-Irányelveket. A munka még folyamatban van, így nem tudunk rögtön ez alapján dolgozni, de figyelemmel fogjuk kísérni az eredményeiket. Maga a kutatás nem írja át a Szingapúri keretrendszert, szimplán a Leíráskészlet-profil elkészítéséhez szeretne kifejleszteni egy új eszközt. Ugyanis úgy látják, hogy a Leíráskészlet-profil számára kidolgozott megszorításokat megfogalmazó nyelv a DC-DSP azért nem lett általánosan elterjedt a metaadat-alkalmazók körében, mivel túlzottan kötődött az RDF-hez, ami akkoriban még éppen, hogy kezdett elterjedni. Azóta a W3C is felülvizsgálta a létező gyakorlatokat a megszorításokat megfogalmazó nyelvek tekintetében (köztük a DC-DSP-vel) és azóta végeztek is a Shape Expressions, azaz Formakifejezés (ShEx) és Shapes Constraint Language, azaz Formák megszorításokat megfogalmazó nyelve (SHACL) kidolgozásával, amelyek így a szemantikus web sémanyelvei lettek, és a gyakorlatban felváltották a jóval egyszerűbb DC-DSP-t. A kifejleszteni kívánt eszköz, a Simple Tabular Model for Application Profiles (AP-STM) azt szeretné megvalósítani, hogy lehetőség legyen táblázat alapján legenerálni a gépi felhasználásra szánt dokumentációt a megszorításokról.

Más megközelítésű kutatások is folytak/folynak. Ilyen például a: YAMA, vagyis Yet Another Metadata Application Profile (= Már megint egy újabb alkalmazásprofil). A japán szerzőcsoport kiinduló-

pontja ugyanaz, mint a DCMI-s kollegáiké: „Bár határozott igény mutatkozik alkalmazásprofil készítésére vonatkozó irányelvekre, főleg a gépi olvasásra szánt alkalmazásprofilok esetében, a nyilvánosan elérhető alkalmazásprofilok száma kevesebb, mint lennie kéne. Szintén nem érhető el géppel jól olvasható Leíráskészlet-profil. Ennek egyik oka, hogy nincsenek egyszerű munkafolyamatok és eszközök.”

„Egy másik fontos megoldandó probléma az alkalmazásprofilokkal a verziózás, változtatásmenedzsment és a géppel olvasható változtatásnaplók. Az alkalmazásprofilokat gyakran valamilyen vizualításra támaszkodó eszközzel (pl. táblázat) készítik, és inkább szolgálnak dokumentációs célokat.”

Bár a problémát ugyanabban látták, a megoldást már nem: „Kezdetben úgy gondoltuk, hogy táblázatos formában fogjuk definiálni az alkalmazásprofilot, de a táblázatos formának megvannak a maga korlátai. A fő gond, hogy nehéz leképezni a leíráskészlet-profil hierarchikus struktúráját egyetlen táblázatban.”¹³ Ehelyett megalkottak egy közvetítő formátumot, amelyből mind a gépi, mind az emberi felhasználásra szánt alkalmazásprofil előállítható és ezen felül még a verziózás és változtatásnaplók problémájára is választ ad.

Hogy képzeljük az RDA-alkalmazásprofil készítését?

Megvizsgálva a rendelkezésre álló szakirodalmat, arra a döntésre jutottunk, hogy a Me4DCAP módszert felhasználva indulunk el. A pilotprojekt során nemcsak az alkalmazásprofilunkat fogjuk folyamatosan értékelni, hanem a módszert magát is, és mindkettőn szükség szerint módosításokat végzünk. Így végül a projektnek két kimenete lesz: maga az alap-alkalmazásprofil és a módszer.

A módszert nem tartjuk ideálisnak. Egyrészt mivel alapja a DC, tehát egy meglehetősen kis elemszámú elemkészlet, amivel más dolgozni, mint az RDA hatalmas elemkészletével. Másrészt a tanulmány levezető részében megfogalmazott validációs szakasz és a Me4DCAP v.03. készítése elmaradt. Valamint a kora is a módszer ellen szól. A metaadatvilágban bekövetkezett újítások, készülő paradigmaváltás miatt 2013 messzebb van, mint 7 év.

Hogy miért választottuk ezt, annak oka prózaian egyszerű: jelenleg még nincs más. Illetve ponto-

sabban: jelenleg ez a legstabilabb irány, amerre el lehet indulni. A következő években el fog válni, hogy a vetélkedő alkalmazásprofil készítését segítő eszközökből melyik lesz a használhatóbb. Mivel a kutatások csak egy jól körülhatárolható szakasz, a Leíráskészlet-profil elkészítésével foglalkoznak, pilotprojektünk az alkalmazásprofil fejlesztési folyamat egészét tekintve releváns tapasztalatokkal fog szolgálni. Az új eszközöket pedig megjelenésük után tudjuk majd tesztelni. Alkalmazásprofilunk integritását nem fogja veszélyeztetni az eszközök változása, hiszen nem arról van szó, hogy a DC-DSP-vel ne lehetne megfelelő minőségű Leíráskészlet-profil alkotni, csak azt, hogy nem túl egyszerű vele dolgozni. A végcélunk valóban egy felhasználóbarát alkalmazásprofil-fejlesztési folyamat, de pilotprojektünk így is elég ambiciózus céllal rendelkezik, ezért a prioritásunk úgy néz ki, hogy először egy működő módszert szeretnénk látni, majd ezt követi ennek a felhasználóbarátta csiszolása.

Ha nagyobb erőforrásokkal rendelkezni, akkor célszerű volna először a módszertanhoz egy másik pilotprojektet készíteni, oly módon, hogy a vonatkozó Magyar Szabványok (MSZ-ek) alapján elkészítjük egy adott házi szabályzat alkalmazásprofilját, mintegy rekonstruálva a munkát, de egy jóval módszeresebb, könnyebben újraalkotható formában. Maga a Me4DCAP is ezt látta problémának az alkalmazásprofilok tekintetében, ezért is használta fel a módszertan elkészítésében a tervezéstudomány módszereit.

A továbbiakban bemutatom a Me4DCAP elemeit. Először az egyes elemek definícióit ismertetem, majd ezután azokat a módszereket veszem számba, amelyeket a szakirodalom ehhez a szakaszhoz javasol. Míg végül arra térek ki, hogy a mi konkrét projektünkben, az RDA-elemkészlet általános alkalmazásprofilja elkészítése során, hogyan képzeljük az adott szakasz kivitelezését.

A Me4DCAP elemei

A Me4DCAP 10 blokkra osztja az alkalmazásprofil-fejlesztés lépéseit. A blokkokat a 2. ábrán láthatjuk. A számozottak egymás után következnek, míg a betűvel jelöltek több blokkon át párhuzamosan folynak a számozott blokkokkal. Az ábrán félkövérrel kiemelve a Szingapúri keretrendszer szakaszai. Az ábra a szerzők tanulmányában lévő ábra reprodukciója.

Érdekes ellentmondás, hogy a Szingapúri keretrendszer által 4. szakaszként definiált Használati Irányelvek a Me4DCAP-ban megelőzi a 3. szakaszt, vagyis a Leíráskészlet-profil.

Iteratív modellt választottak, hiszen a könyvtári metaadatok egy nagyon komplex rendszert képeznek, a tervezés még a legnagyobb odafigyeléssel se lesz hibátlan elsőre. Valamint, ha valami újat és kipróbálatlant fejlesztünk, mindig ott van az esély, hogy nem úgy működik, ahogy azt feltételezzük róla.

A Me4DCAP-projekt szereplői és az általuk betöltött szerepek

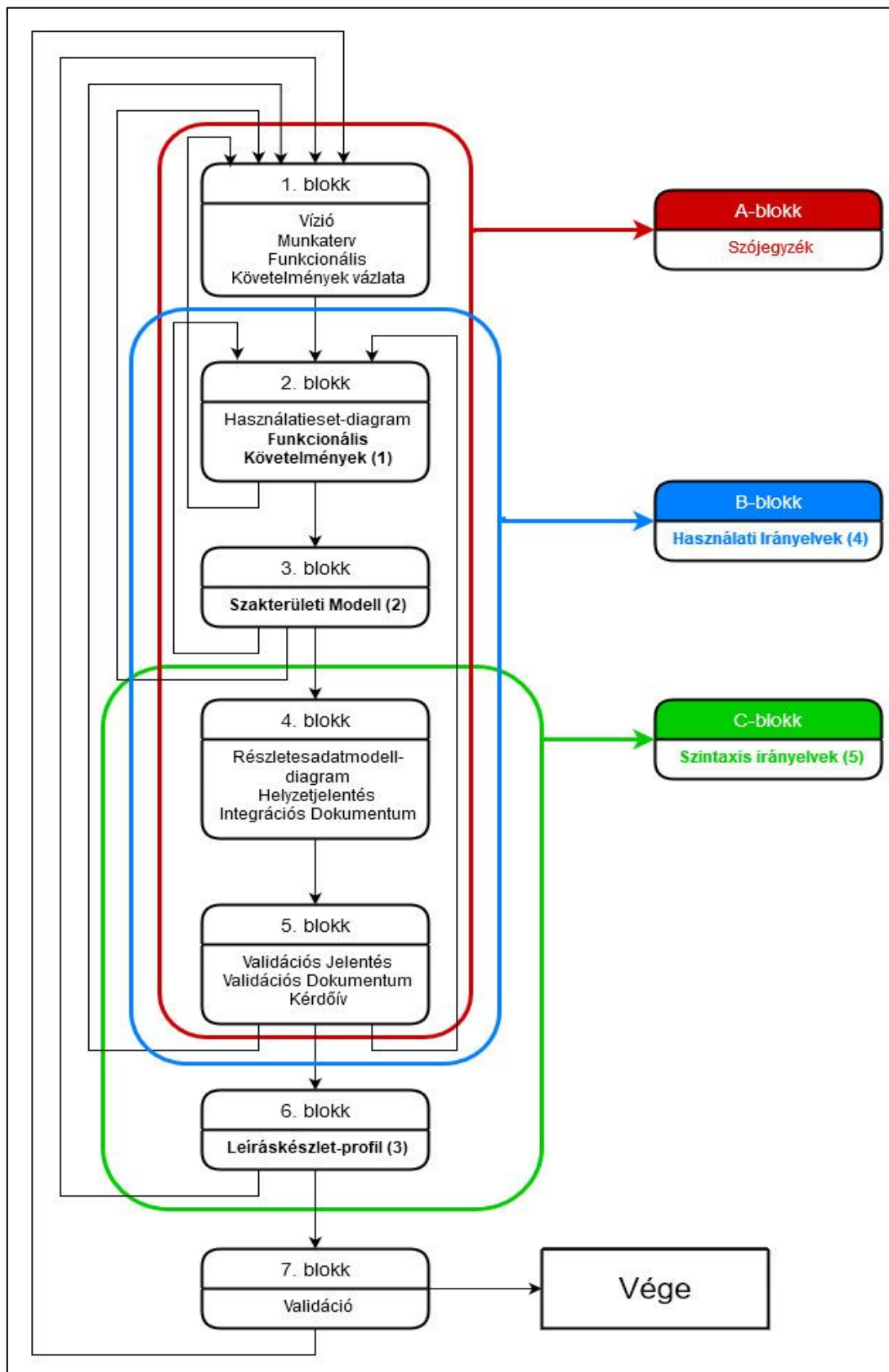
A Me4DCAP nemcsak kizárólag az alkalmazásprofil-fejlesztés lépéseit mutatja be a hozzájuk kapcsolódó módszerekkel, hanem magának az alkalmazásprofil-fejlesztési projekt bizonyos szervezési kérdéseire is kitér. A legtöbb lépésnél azt is ismerteti, hogy azt a csoport melyik specialistájának célszerű elvégeznie, vagy kinek kell vezetnie a közös munkát. A módszer a következő szerepeket különíti el:

Projektmenedzser: részletekbe menő interakciót folytat belső és külső résztvevőkkel, így alakul ki a globális rálátása a projekt egészére. Érdemes olyasvalakit választani a pozícióba, aki egy olyan intézménynek dolgozik, amely elkötelezett a Dublin Core Alkalmazásprofil fejlesztése iránt. Ennek a személynek tisztában kell lennie azzal, hogy mit akarnak az intézmények elérni a Dublin Core Alkalmazásprofittal.

Rendszerelemző: két tudásterületen kell jártasnak lennie: egyrészt rendelkeznie kell a szükséges technikai készségekkel a követelmények azonosításához és az adatmodellezéshez; másrészt rendelkeznie kell „alapvető vezetési ismeretekkel”, részleteiben is értenie kell az alkalmazás területét, és szükséges, hogy képes legyen megérteni a követelmények mögötti igazi motivációt és relevanciát.

Integrátor: az Integrátornak metaadat-tervezőnek, vagy -fejlesztőnek kell lennie, aki érti a Szemantikus web koncepcióját, ismeri a Leíráskészlet-profil (DSP) és az RDF-et.

Végfelhasználó: az a felhasználó, aki dolgozni fog azzal a rendszerrel, amely a Dublin Core Alkalmazásprofil fogja használni.



2. ábra Me4DCAP-életciklus

A munkacsoport tehát több emberből áll, akiknek különböző szerep jut. Egy vagy több ember szerepenként a munka mennyiségétől függően. Fontos megjegyezni, hogy a munkacsoport multidiszciplináris jellege nagyon fontos, egyben a sikeresség záloga.

A módszer jól elkülönített szerepekben gondolkodik. Míg ennek meglennének a maga tagadhatatlan előnyei, sajnos feltehetően emberierőforrás-problémák miatt valószínű, hogy a mi alkalmazásprofil-fejlesztésünkben vegyes szerepek lesznek. De a módszer alapvető értékét, a multidiszciplináris csoport eszméjét fontosnak tartanánk érvényesíteni, amennyire erre csak lehetőségünk lesz.

Vízió

A legelső lépés a vízió elkészítése. Ez egy dokumentum, amely megmutatja, hogy mit akarnak elérni a Dublin Core Alkalmazásprofittal a fejlesztői. Definiálja a profil érvényességi tartományát. Maximum 200 szóból álló sima szöveges dokumentum, amely leírja az Alkalmazásprofil használatának határait.

A Me4DCAP a brainstorming technikát javasolja a Vízió kifejlesztéséhez, ahol a csapat minden tagjának lehetősége van az ötleteit leírni (egy valós vagy virtuális táblára), majd ezt megbeszélés/vita követi. Végül a kiválasztott ötleteket egyszerűen megfogalmazott mondatokba rendezzük. A Projektmenedzser vezeti a csoportot ennek az anyagnak az elkészítése során. Minden csapattag részéről elvárt a részvétel.

Igyekszünk úgy megfogalmazni a víziókat az alap-alkalmazásprofil számára, hogy a fejlesztés során kiderüljön az is, hogy milyen további kutatásokat vagy pilotprojekteket kell véghezvinni a tényleges alkalmazás előtt. Már a tényleges olyan alkalmazás között, amely minden lényeges elemet implementál és kihasználja őket. Részleges implementációt is véghez lehet vinni, ehhez a jelenlegi rendszereink is alkalmasak. Ebben az esetben viszont nem hozza az RDA a várt hatást.

Munkaterv

A munkatervnek az a célja, hogy a projekttevékenységek időbeli tervét adja. Ebben nyomon követhető a projekt ütemezése és segítőként szolgál a munkacsoport tagjainak a projekt során. A munkaterv tartalmazza a minden egyes szakasz ütemezését, emellett hozzávetőleges kezdő és vége

dátumokat. Tartalmaznia kell ezen felül még minden „blokk” elkészültének esedékességét. Nem szükségszerű, hogy tartalmazza, de hasznos lehet, ha a Munkatervben feltüntetetik az adott fázisért vagy anyag elkészültéért felelős munkacsoporttagokat is.

A Munkaterv lehet szöveges dokumentum, Gantt-diagram vagy bármilyen egyéb grafikus megoldás vagy séma, lényeg, hogy a munkacsoport megfelelőnek találja. A Munkatervet az összes munkacsoporttagnak együttesen kellene kidolgoznia, illetve alaposan megtárgyalnia, hogy tekintettel legyen a csoporttagok által rendelkezésre álló időre. A Projektmenedzser vezeti ezt a munkát. Elfogadott, hogy a Munkaterv változzon a projekt előrehaladtával.

Funkcionális Követelmények vázlat

Az első blokk utolsó lépése a Funkcionális Követelmények vázlat. Ez egy lista a munkacsoport által összegyűjtött funkcionális követelményekről. Nagyobb hangsúlyt kapnak a Végfelhasználó és a Projektmenedzser által megfogalmazott követelmények. Ez a dokumentum csak röviden fejtsse ki a követelményeket, követelményenként maximum 2 sorban. A Funkcionális Követelmények vázlatát a Használatieset-modell fogja tovább részletezni. A csoportot a Rendszerelemző vezeti és minden munkacsoporttag részt vesz benne.

Ennek az anyagnak az elkészítéséhez használhatjuk ugyanazt a technikát, amit a Víziónál.

Ez az első olyan lépcső az alkalmazásprofil fejlesztési folyamatban, ahol a projekt tárgya, az RDA, már bizonyos válaszokat ad az itt megválaszolandó kérdésekre. Az, hogy a funkcionális követelményekben gondolkodást be kellene vinni a katalógizálásba, már nem új keletű elképzelés. Az FRBR már a címében is (A bibliográfiai rekordok funkcionális követelményei) világossá teszi a legfőbb nézőpontbeli újítását. A szemlélet megmaradt az FR-családot leváltó LRM-ben is, de az új ICP (Nemzetközi katalógizálási alapelvek) a funkcionális követelmények és használói feladatok felől közelít a katalógushoz. Ez a szemlélet csalókéan egyszerűnek látszik, sőt, vélhetőleg lesznek olyanok is, akik ebben semmi újat nem látnak, hisz eddig is tudtuk, hogy van használó és eddig is fontosnak tartottuk a visszakeresést. Eddigi tapasztalataim alapján úgy gondolom, hogy ennek a szemléletnek meg kell érnie az ember fejében. Miközben dolgozunk ezekkel az új szemléletű alapvetésekkel, fokozatosan jövünk rá, hogy tulajdonképpen meny-

nyire nem is foglalkozunk a felhasználóval a feldolgozás során. Jelenleg a középpontban a dokumentum van, illetve a katalogizálási „ideáink”. Ezt követi az IKR-hez való alkalmazkodás, illetve a rendelkezésre álló erőforrások alapján a munkafolyamatok optimalizálása. A felhasználó, akiről persze jelenleg is tudjuk, hogy létezik, csak ez után következhet.

Használatieset-diagram

A Használati esetek módszeres és intuitív eszközei a funkcionális követelmények megragadásának. A Használati eseteket egyrészt arra fogjuk használni, hogy a Funkcionális Követelményeket kidolgozzuk, másrészt, hogy megértsük a vizsgált rendszer objektumait (és ismérveit). A Használatieset-modell a következőkből áll:

- *a rendszer funkcióit leíró UML Használatieset-diagram (ami tartalmazza a használati esetekben résztvevő aktorokat),*
- *az összes részletes használati eset.*

Ehhez a szakaszhoz a Me4DCAP az UML-t ajánlja módszernek. Az UML azaz Unified Modeling Language (= Egységes modellező nyelv) egy általános célú vizuális modellező nyelv, amely arra használható, hogy specifikáljuk, szemléltessük, megtervezzük és dokumentáljuk egy szoftver rendszer architektúráját.¹⁴ Az UML tehát egyértelműen a szoftverfejlesztéshez köthető, viszont az „általános célú” részt az alkotóinak sikerült terveztként jobban megvalósítani, így alkalmas metaadatosztályok tervezésére/szemléltetésére is. Másrészt amennyiben nem cédulakatalógusban gondolkodunk, a metaadatainkat valamilyen szoftver segítségével fogjuk rögzíteni, visszakeresni és megjeleníteni. Ilyen szempontból a metaadat-struktúra tervezése nem különül el szögesen a szoftverfejlesztéstől. Az UML pedig felvehet egyfajta közvetítő szerepet a szoftverfejlesztők és az alkalmazásprofil-fejlesztők között.

A használati esetek metaadatokra vetítve arról szól, hogy összeszedjük azokat a forgatókönyveket, hogy mit szeretnénk, hogy visszakereshető legyen, milyen szűrőket szeretnénk, mi az, amit elég csak megjeleníteni stb. Mivel a mi projektünk egy általános, IKR-független alkalmazásprofil megalkotását tűzte ki céljául, ezért úgy gondoljuk, hogy alkalmas lesz arra is, hogy feltérképezze, hogy milyen irányba kell fejlődnie az IKR-eknek ahhoz, hogy támogatni tudják az RDA-alapú katalogizálást. Amennyiben a fejlesztés már egy konk-

rét házi szabályzathoz történik, ez az a pont, ahol az IKR-ünk képességeit szedjük össze.

Funkcionális követelmények

A Funkcionális Követelmények „a sikeres alkalmazásprofil-fejlesztési folyamat alapvető alkotóeleme, amely irányt ad az alkalmazásprofil fejlesztésének azzal, hogy célokat és korlátokat határoz meg. Ez a fejlesztés gyakran egy tágabb közösség feladata és a résztvevők között találkozhatunk szolgáltatók vezetőivel, a használt anyagok szakértőivel, alkalmazásfejlesztőkkel és a szolgáltatás potenciális végfelhasználóival.”

A Funkcionális Követelmények fejlesztéséhez a munkacsoport a Funkcionális Követelmények vázlatát és a Használatieset-modellt használja annak érdekében, hogy azonosítsák a funkcionális követelményeket, amelyeket a használati esetek világossá tesznek.

A DCMI-Irányelvek ezt még kiegészíti a következővel: A Funkcionális Követelmények létrehozásában többféle módszertan is a segítségünkre lehet. Ilyen például az üzleti folyamatmodellezés vagy a követelmények vizualizációját segítő módszerek, mint például az UML. Az adott alkalmazás használati eseteinek és forgatókönyveinek definiálása segíthet azokat a funkcionális követelményeket is számba venni, amelyek másképp rejtve maradtak volna.

A DCMI-Irányelvek első szakaszként adja meg a Funkcionális Követelményeket, míg a Me4DCAP már eleve abban gondolkodik, hogy ennek kidolgozását két előzetes lépéssel meg kéne támogatni. Mivel még nem szoktunk hozzá, hogy Funkcionális Követelményekben és alkalmazásprofilokban gondolkodjunk a katalógusunkról, vélhetőleg segíteni fog, ha a munkát minél apróbb egységekre bontjuk. Az életciklusmodell szerint itt van az első visszacsatolási lehetőségünk a fejlesztési folyamat elejére.

Szakterületi modell

A DCMI-Irányelvek szerint „a szakterületi modell a leírása a dolgoknak, amiket a metaadataink leírnak, és a köztük lévő kapcsolatoknak. A Szakterületi Modell az alaprajz az alkalmazásprofil felépítéséhez.” Tulajdonképpen megmutatja a rendszer kontextusa szempontjából legfontosabb objektum-típusokat.

Kidolgozásához többféle módszer használható, de ezekből most egyet emelnék ki, mivel a könyvtári terület szakterületi modelljeiben eddig az „entity-relationship” modellt használták eszköznek. A módszer fordítása adatmodellezéssel foglalkozó szövegekben „egyed-kapcsolati” modell, viszont könyvtári kontextusban jellemzően „entitás-kapcsolati” modellként utalnak rá. A választás azal indokolható, hogy könyvtári szövegekörnyezetben speciálisabb jelentése van az entitásnak, és ezt a módszer nevének is tükröznie kell. Ilyen szempontból a magyar verzió az entitás-kapcsolati modellt az egyed-kapcsolati modell egy olyan altípusának tekinti, amelyben könyvtári értelemben vett entításokkal dolgozunk.

Mivel itt mi „hozott anyagból dolgozunk”, ezért a szakterületi modell már adott, ez az LRM, illetve az RDA által implementált LRM. Mi annyiban dönthetünk, hogy mi az itt felvázolt entítások és kapcsolatok közül melyikeket szeretnénk használni. Így ebben a fázisban nem fejlesztünk, hanem inkább törölünk.

A tervek szerint viszont az alapalkalmazásprofilunkban igyekszünk minél kevesebbet törölni, hogy megmutathassuk a magyarországi könyvtáros közösségnek, hogy mire képes az RDA „teljes fényében”. A minél teljesebb felől megközelítve könnyebb lesz azt is szétszálazni, hogy melyik entitás/adatelem elhagyása milyen hatással van a rendszer egészére.

Az RDA-elemek ugyanis nem egy szabadon választható lista, amelyről kiválasztjuk az éppen aktuálisan szükségeset. A komplexitása miatt az elemek között megfigyelhető egymásra épülés, illetve a különböző megközelítéseknek úgy ad terepet, hogy bizonyos adatelemek „nem használhatók együtt”. Ez nem egy formális tiltás, az RDA nem tartalmaz ilyeneket, hanem inkább redundáns volta miatt nem lényeges, vagy a kontextus teszi értelmetlenné az elem jelenlétét vagy tartalmát.

Részletesadatmodell-diagram

A következő blokk első lépésének, a *Részletesadatmodell-diagramnak*, a *Szakterületi Modellt* kell részletesebben bemutatnia, tehát az objektum tulajdonságainak definícióját, kötelező vagy opcionális voltát, ismételhetőségét, ontológiai forrását és a többnyelvűség igényét.

Az Me4DCAP a *Részletesadatmodell-diagram* készítéséhez az ORM-technikát javasolja. Más tech-

nikák is alkalmazhatók, mint például az egyed-kapcsolati diagram. A Rendszerelemző feladata ennek az anyagnak az elkészítése, majd az egész munkacsoportnak meg kell vitatnia és jóvá kell hagynia.

Az RDA-projektben itt is egy kicsit kötöttebb lesz ez a szakasz, mintha egy másik szabvány elemkészlete szerinti alkalmazásprofil készítenénk. Az RDA már tartalmaz önmagában számossági korlátozásokat. Illetve minden elem ontológiai forrása az RDA elemkészlete. A kötelező elemek tekintetében a poszt-3R RDA már jóval kevesebb megköltést tartalmaz elődjénél.

Helyzetjelentés

Helyzetjelentés készítése az – RDF-ben leírt – létező metaadatsémák felhasználásával annak kiderítésére, hogy a létező sémákban található mely tulajdonságok írják le az azonosított ismérveket. Ezt az anyagot az Integrátornak kell elkészítenie. Azokban az esetekben, ahol nincs megfelelő tulajdonság, újat szükséges alkotni. Ez is az Integrátornak a feladata.

Ebben a szakaszban tulajdonképpen a jelenlegi fogalmainkat fogjuk megfeleltetni az RDA fogalomkészletének. Nem számítunk adatelem szinten hiányzó tulajdonságra, bár természetesen ebben a szakaszban még kizárni se tudjuk ennek lehetőségét. Az viszont már most biztosnak tekinthető, hogy a szótárkódolási sémák¹⁵ tekintetében nem ilyen jó a helyzet. Bizonyos szótárkódolási sémákból hiányoznak azok a kifejezések, amelyek a magyar gyakorlat szerves részét képezik. Ezekben az esetekben kiemelt feladatunk lesz a magyar kifejezések „védelme”. Csak abban az esetben szabad egy szótárkódolási sémát alkalmazni, ha az a leírásunk érdekeit szolgálja. Az általunk talált hiányosságokat először is teljes mélységében szeretnénk felfedezni, majd a megfelelő fórumokon javaslatnak előterjeszteni és nemzetközi szinten megoldást találni rá ahelyett, hogy házon belül próbálnánk megoldani a túl szoros cipő problémáját.

Integrációs dokumentum

Ebben a dokumentumban látható (mátrixban, soronként) minden ismérv és azok megköltései a kiválasztott metaadat- és kódolási sémák jellemzőivel leírva. Ezt a munkát az Integrátornak kell elvégeznie.

A Me4DCAP külön mellékletként ajánl egy egyszerű sablont ehhez a feladathoz (3. ábra).

	A	B	C	D	E	F	G	H	I	J
1	Integrációs dokumentum									
2										
3	1. táblázat: Használt névterek definíciói									
4	Title		Teljes névtér URI			Prefix				
5										
6	szükség szerinti sort használva...									
7										
8	2. táblázat: Tulajdonságok difiníciói									
9	Címke	Osztály vagy Tulajdonság	Definíció	Értéktartomány	Értéktartomány típusa*	Kapcsolódó leírás	Minimum	Maximum		
10										
11	szükség szerinti sort használva...									
12										
13	*szükség szerinti oszlopot használva									
14										

3. ábra Integrációs dokumentum, sablon

Validációs jelentés

A Validációs jelentés a Validációs dokumentum és a Kérdőív együttesen fogja kiadni a Validációs dossziét. *A Validációs dossziét azért kell elvégezni, hogy ellenőrizzük, hogy amink eddig van, az megfelel-e a Víziónban megfogalmazottaknak. A munkacsoportnak közösen kell kiértékelnie ezt egy megbeszélés keretében. A munkacsoportnak össze kell állítania egy jelentést (szöveges dokumentum) – ez a Validációs jelentés – a megbeszélésen elhangzott megállapodásokból és javaslatokból.*

Ez a tulajdonképpeni első tesztfázis. Nagyon fontos pontja az alkalmazásprofil-készítés folyamatának, hiszen ebben a szakaszban jönnek az első gyakorlati visszajelzések. A dolgok természete miatt teljesen biztosra vehető, hogy a beérkezett

visszajelzések feldolgozása után vissza kell térnünk az 1. vagy a 2. blokkhoz, illetve lehetséges, hogy ki kell egészítenünk a Szójegyzéket és a Használati Irányelveket is.

Validációs dokumentum

A Validációs dokumentum úgy keletkezik, hogy a *Dublin Core Alkalmazásprofil*t kipróbáljuk egy konkrét mintán. *A munkacsoportnak azonosítani kell a források egy csoportját, amit megbízható mintának tekintenek az alkalmazási terület szempontjából. Innentől a Végfelhasználó (az Integrátor segítségével) feladata, hogy kitöltse a Validációs Dokumentumot minden forrás adatával.*

Az Me4DCAP külön mellékletként ajánl egy egyszerű sablont erre (4. ábra).

	A	B	C
1	Validációs dokumentum		
2	A forrás nevének forrása		
3	Címke	Érték	
4	<i>Osztály neve</i>	<i>Osztály linkje</i>	
5	Tulajdonság neve	tulajdonság értéke	
6	szükség szerinti sort használva...		
7			
8			

4. ábra Validációs dokumentum, sablon

Tehát tulajdonképpen formátumfüggetlen próbaleírásokat készítünk ebben a lépésben. A formátumfüggetlenség kiemelten fontos ebben a szakaszban, mivel így biztosítható, hogy tisztán az alkalmazásprofil hiányosságait tudjuk feltérképezni. A formátumoknak is megvannak a maguk megszorításai és korlátai, ezért az alkalmazásprofilkészítésben, illetve általában az RDA-val foglalkozó anyagainkban igyekszünk az adatelemek tartalmára vonatkozó szabályzatot külön kezelni a formátumra vonatkozó szabályzattól. Ebben tulajdonképpen semmi újdonság nincs, az MSZ-ek eddig is külön voltak a HUNMARC-tól, az RDA-t is külön kezeljük a formátumoktól. Viszont az eddig kapott visszajelzések azt a benyomást keltik, hogy a kollégák az RDA-val ismerkedve már egy konkrét MARC-ban leképezett házi szabályzatot szeretnének látni a jelenleg is létező tartalom/formátum szétválasztása helyett. Erre a problémára megoldást találni nem feltétlenül a projekt feladata, de mivel a probléma erősen befolyásolja mind a projektről, mind az RDA-ról való kommunikációnk sikerességét, ezért itt (is) foglalkoznunk kell vele.

Kérdőív

A Validációs dosszié utolsó darabja. Mindenkinek, aki a Validációs Dokumentumon dolgozott, ki kell töltenie egy kérdőívet, amelyben értékeli a validációs folyamat nehézségeit. A cél, hogy kiderüljön, van-e olyan adat, amire az Alkalmazásprofil nem kínál leírási lehetőséget, vagy hogy vannak-e olyan elemek az Alkalmazásprofilban, amelyeket kötelezőként definiáltak, de nem lehet kitölteni őket az adott forrásban rendelkezésre álló információk alapján, vagy fölmerül-e bármilyen más típusú nehézség vagy félreérthetőség.

Szójegyzék

A Szójegyzék egy szöveges dokumentum. A Dublin Core Alkalmazásprofilban használt kulcsszavakból és azok leírásából áll. Definiálnia kell a fontosabb szavakat, amelyeket a munkacsoport használni fog a profil kidolgozása során. Ezt lehetőség szerint a fejlesztési folyamat kezdetétől építjük.

Már el is kezdtük ennek az alapját lerakni egy háziszótárral. Definíciók még nincsenek benne, de könnyedén ki tudjuk vele egészíteni. Mivel az RDA szövege meglehetősen nehéz, ezért az is felmerült bennünk, hogy a fontosabb fogalmakat közzéteszünk könnyebben érthető megfogalmazásban is.

Erre már történtek törekvések (lásd, Adattisztítás Tartalomtípus generálásának előkészítéséhez¹⁶). A közös munka során folyamatosan kérni fogjuk a kollégák visszajelzését a definíciók és az elnevezések érthetőségéről. Az így keletkezett Szójegyzéket valamilyen formában közzé is szeretnénk tenni. A Toolkit hivatalos fordításába nem fog beépülni, mivel ennél szöveghűbb fordítás az elvárt, viszont értelmező, módszertani munkák alapvető része kell, hogy legyen.

Használati Irányelvek

A Leíráskészlet-profil definiálja a „mi”-t az alkalmazásprofilban, a Használati Irányelvek pedig a „hogyan”-t és a „miért”-et. A Használati Irányelvek a metaadatrekordok készítőinek adnak instrukciókat. Ideális esetben minden tulajdonságra kiterjednek és számolnak az összes döntési helyzettel, amelylyel a rekord elkészítése közben szembesülni fogunk.

A Használati Irányelveket többnyire a Rendszer-elemzőnek kell elkészítenie. Az ismérvek leírása és az objektumosztályok olyan típusú információk, amelyekkel csak az alkalmazási terület szakértői tudnak szolgálni, így a Projektmenedzsernek és a Végfelhasználónak is részt kell vennie a folyamatban.

A Használati Irányelvek a Szingapúri Keretrendszer első opcionális szakasza. Ha viszont egy RDA-alkalmazásprofil készítését, ezt is kötelező szakasznak kell vennünk, hiszen ez tulajdonképpen a szabályzat megírását takarja. Jelenleg az RDA közel 6000 olyan döntési helyzetet tartalmaz, amelyre kell valamilyen döntést megfogalmazni. A ténylegesen megírt döntési helyzetek száma viszont jóval több lesz ennél, hiszen a poszt-3R RDA egy keretrendszer. A napi munkához alkalmazható szabályzat összeállításához (itt természetesen minden dokumentumtípus leírását egybe-számoljuk) sokkal több döntési helyzetet és ezek elvárt megoldásait kell felvázolnunk, mint amennyit az RDA megfogalmaz.

Elképzelésünk szerint itt a különböző jelenlegi magyar szabványokban és házi szabályzatokban leírt döntéshelyzetre támaszkodnánk, „feldúsítva” ezt a projekt fogalmai szerint Végfelhasználónak nevezett specialisták (akik a részterületen előforduló legkritikább speciális eseteket is ismerik) kiegészítéseivel. Első körben ezeknek a megfelelőit próbáljuk feltérképezni. Ennek időigénye és eredménye egyelőre megjósolhatatlan. Elindult az MSZ

3424/1-78 és az RDA összevetése, amelynek célja, hogy képet kapjunk róla, hogy milyen módszerrel a legcélszerűbb dolgozni, mi a munka várható időkerete, illetve hogy mennyiben fedt le az RDA a jelenlegi gyakorlatunkat. Ebben a „tapogatózó körben” még nem gondoltuk a le nem fedett eseteket mélyebben elemezni. Lehetséges, hogy ezek az esetek könnyen orvosolhatók lesznek, de az is lehet, hogy oly mértékű probléma lesz, amelyet mindenképpen meg kell majd osztani az RDA fejlesztői felé. Céljuk ugyanis, hogy az RDA a világ minden táján adaptálható legyen, erre ismét felhívta a figyelmünket az RDA Board European RDA Interest Group (EURIG) területi képviselője a legutóbbi éves ülésük alkalmával tartott előadásában¹⁷. Természetesen nem lehet célunk az MSZ és a jelen gyakorlat egy-az-egybeni átmentése RDA-környezetbe, viszont fontosnak tartjuk, hogy a magyar sajtóságok kifejezhetőek maradhassanak.

Leíráskészlet-profil

Amint az Integrációs dossziét véglegesítettük, lehetőségessé válik továbblépni a Leíráskészlet-profil elkészítéséhez, ami kötelező elem. Az Integrátornak kell részleteznie az Alkalmazásprofil a Leíráskészlet-profil keretrendszer (vagyis Mikael Nilsson 2008-as Leíráskészlet-profil: Megszorításokat megfogalmazó nyelv Dublin Core alkalmazásprofilokhoz című munkáját¹⁸) használva, az Integrációs dossziét segítségül véve.

A Leíráskészlet-profil a leíráskészlethez tartozó strukturális megszorítások leírásának egy módja. Korlátozza a források körét, amelyek a leírás tárgyát képezhetik a leíráskészletben, a használható tulajdonságokat és az adatértékek megadásának módját.

Egy Leíráskészlet-profil különböző célokra használható, például:

- egy Dublin Core™ Alkalmazásprofil korlátozásainak formális reprezentációjához;
 - adatbázisok konfigurálásához;
 - metaadat-szerkesztő eszköz konfigurálásához.
- Egy Leíráskészlet-profil nem tér ki a következőkre:*
- emberek számára olvasható dokumentáció;
 - szótárdefiníciók;
 - verziók kezelése.

Ez tulajdonképpen a Szingapúri keretrendszer gerince. Az eddigi szakaszok azért kellettek, hogy előkészítsék a terepet a Leíráskészlet-profil meg-

alkotásához, míg az utána következők erre építve pontosítják, tulajdonképpen élővé teszik a profilt.

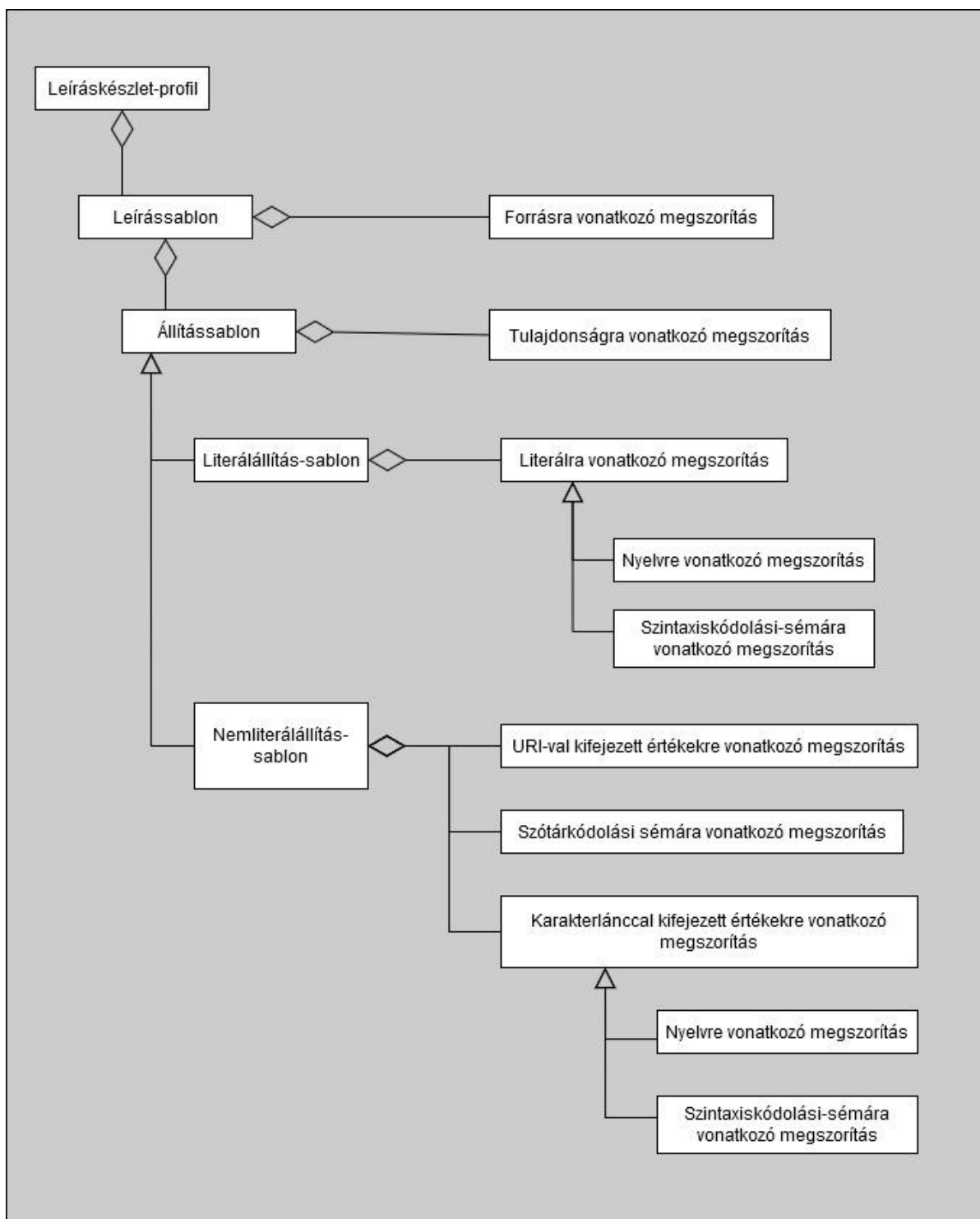
A DCMI közreadott egy külön megszorításokat megfogalmazó nyelvet is ehhez a szakaszhoz. (Ez a fent említett Nilsson-féle keretrendszer, angol rövidítéssel DC-DSP.) Bár, mint ahogy azt már említettem a szakirodalmi kitekintésben, azóta ezt egyszerűsége miatt már elavultnak tekinthetjük, az alapvető struktúráját érdemes ismertetni, mivel ez tulajdonképpen tartalmazza a megszorítások típusait és egymásba kapcsolódásukat. Tehát ezek azok a megszorítástípusok, amelyeket ki kell fejelnünk, a jelenleg futó kutatások csak ennek a kifejezésnek a hogyanjára vonatkoznak.

Maguk a megszorítástípusok nem fognak túl sok újat behozni a könyvtári világba. Eddig is használtuk ezeknek a megszorítástípusoknak a nagy részét a szabályzatainkban. Az újdonság inkább a struktúra lerajzolása. A probléma ugyanis az, hogy a szabályzatainkban kifejezett megszorítások esetlegesen, gyakran keverten, esetleg csak implicit módon jelennek meg. A megszorítások formális kifejezése egyrészt lehetővé teszi, hogy gépi felhasználásra is alkalmas alkalmazásprofilot fejlesszünk, másrészt az alkalmazásprofilok könnyebben összehasonlíthatók lesznek. Ez természetesen nem azt jelenti, hogy a katalogizálóknak szánt szabályzatok is ezen a formális nyelven lesznek megfogalmazva, csak azt, hogy ez lesz a háttérben. Tehát a katalogizálóknak nem kell „beszélniük ezt a nyelvet”, de a szabályzatíróknak mindenképpen szükséges lenne.

A Leíráskészlet-profil struktúráját osztálydiagramon kifejezve az 5. ábrán láthatjuk. Összefoglalva: egy ilyen profil a leíráskészlet struktúráját írja le sablonok és megszorítások segítségével. A sablonok kifejezik, a megszorítások pedig lehatárolják a struktúrát.

Az ábrán található jelek értelme:

- a csúcsára állított „üres” rombusz gyenge tartalmazás kapcsolatot vagy másképpen aggregációt jelent, a rombusz az egésznél lévő vonalvégen található,
- a „üres” fejú nyíl pedig az általánosítást jelzi. Ez az objektumok speciális viszonya, gyermek-szülő kapcsolat, amelyben a fölérendelt elem az általános, az alárendelt a specializált. Osztályszerű elemek közötti strukturális kapcsolat. A nyíl iránya jelzi az általánosítás irányát.¹⁹



5. ábra Leíráskészlet-profil, megszorítások hierarchiája

A Leíráskészlet-profilban a sablonoknak két szintje van:

- **Állítás sablonok**, amelyek tartalmazzák az összes tulajdonságra vonatkozó megszorításokat (karakterlánc-értékek, szótárkódolási sémák

stb.), amelyek egy bizonyos fajta állításra vonatkoznak.

- **Leírás sablonok**, amelyek tartalmazzák azokat az állítás sablonokat, amelyek egy bizonyos fajta

leírásra vonatkoznak, valamint a leírt forrásra vonatkozó megszorításokat.

Az állítássablonnak két lehetséges speciális esete van: a literálállítás-sablon és a nemliterálállítás-sablon. A literálállítás-sablon csak literál(érték)ra vonatkozó megszorításokat tartalmazhat. Ezek vagy nyelvre vonatkozó vagy szintaxiskódolási sémára vonatkozó megszorítások lehetnek. A nemliterálállítás-sablon karakterlánccal kifejezett értékekre vonatkozó megszorításoknak ugyanezek az alosztai lehetnek. Ám a sablon maga még tartalmazhat URI-val kifejezett értékre vonatkozó megszorítást és szótárkódolási sémákra vonatkozó megszorítást is.

Amikor az RDA-t keretrendszerre alakították, vélhetőleg szándékosan törekedtek arra, hogy az alkalmazásprofil-készítés irodalmában lefektetett fogalmakat használják. Így, bár most még idegennek hat a szabályozott szótár helyett a szótárkódolási séma fogalma, de a terminológiai egységesség és a metaadatok általános felosztásába való jobb beilleszthetőség miatt jó választás az elnevezés.

Mivel ez a szakasz az, ahol most még folynak kutatások, ezért nincsenek köbe vésett elképzeléseink arról, hogy milyen módszert fogunk használni. A projekt feltérképező volta miatt, ahol csak lehet, szándékunkban áll minden lehetséges módszert kipróbálni, hogy gyakorlati tapasztalatot szerezzünk a használhatóságukról.

Szintaxis Irányelvek

A DCMI Irányelvek definícióját használva a *Szintaxis Irányelvek tartalmazzák az „alkalmazásprofil-specifikus szintaxisokat és/vagy Szintaxis Irányelveket, ha vannak ilyenek”*. Ahogy a feltételes mód is utal rá, ez egy opcionális szakasza a Szingapúri keretrendszernek. *A Szintaxis Irányelveket az Integrátornak ajánlatos elkészítenie, mivel egy eléggé műszaki jellegű dokumentum*. Ez az a szakasz, ahol a választott formátum sajátosságaival kiegészítjük az alkalmazásprofilunkat.

Validáció

Az Alkalmazásprofil gyakorlati validációját is el kell végezni. Ez a második tesztfázis. Ezt a validációs folyamatot végezhetjük naplózásos technikával vagy a Végfelhasználók munkájának megfigyelésével. Mindezt egy olyan rendszerben, ami már implementálta a kifejlesztett Alkalmazásprofilot. A

megfigyelést az Integrátor végezze! Az Integrátor gyűjtse össze a javasolt változtatásokat és időről-időre jelentsen róla a munkacsoportnak, hogy felül tudják vizsgálni az Alkalmazásprofil definícióit.

Ha új információt szeretnénk belevinni a Profilba, akkor az egész fejlesztési folyamatot újra kell kezdeni az 1. blokktól. Át kell nézni a már elkészített anyagainkat, hogy szükséges-e rajtuk változtatni az új információk tükrében. A folyamaton így végighaladva, a szükséges változtatásokat megtéve készítjük el az új Alkalmazásprofil-verziót.

A gyakorlati tesztelés azt jelenti, hogy az alkalmazásprofil a normál feldolgozó munkafolyamatba ágyazva tesztelik a Végfelhasználók. Ez azt is jelenti, hogy ebben a stádiumban már valamilyen szoftverrel kell dolgozniuk. Két eshetőséget fontolgatunk erre a célra, az egyik a RIMMF4, a másik pedig a Koha.

RIMMF4²⁰ vagyis az RDA in Many Metadata Formats (= RDA különböző metaadat-formátumokban) egy szoftver, amelyet RDA-s leírások szemléltetéséhez készítettek (6. ábra). A RIMMF-ben minden RDA-entitásnak saját adatlapja van, így alkalmas a szoftver az entitás alapú katalogizálás begyakorlására. A RIMMF korábbi, 3-as változatát több cikk ismerteti.^{21,22} A mostani, még béta változatban lévő RIMMF4-et már az LRM-re átdolgozott RDA-val lehet használni. Viszont nincsen meg benne (jelenleg legalábbis) sok olyan funkció, amely az előző verzióban még megvolt. Jelenleg nem ismeri a MARC-ot se import se export formátumnak. Illetve nincs összekötve a nagyobb névterekkel.

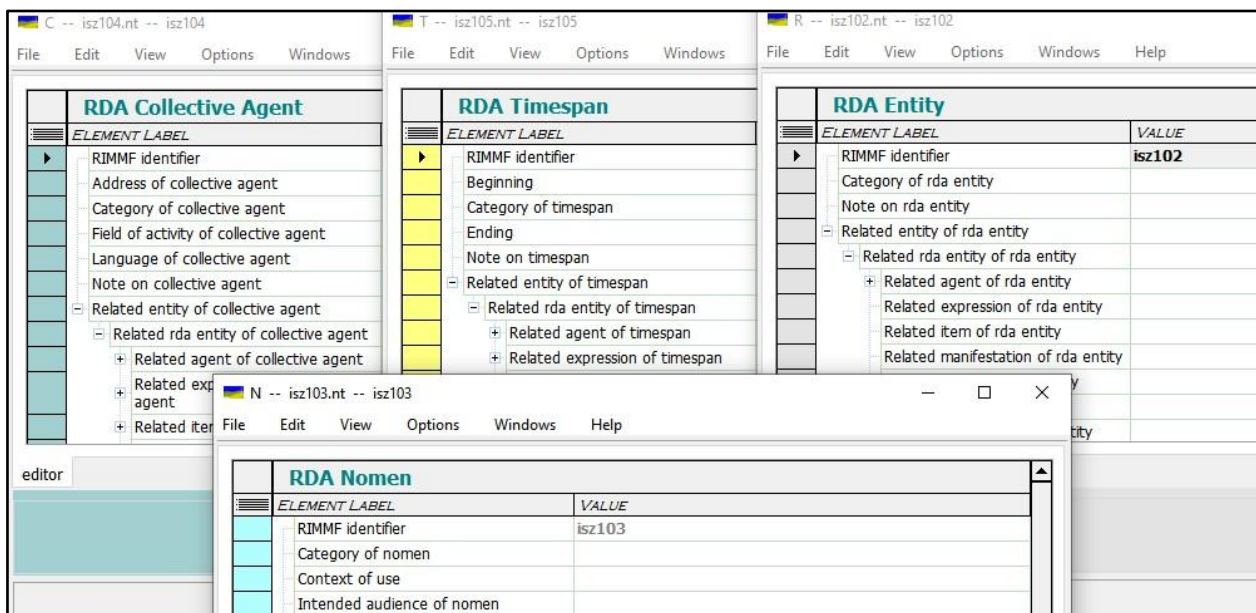
A másik lehetséges tesztkörnyezet a Koha (7. ábra), ami egy nyílt forráskódú integrált könyvtári rendszer. Mellette szól, hogy eddig is ezt használtuk „homokozónak”, például a szemantikus mini-projektünkben.²³ A Koha viszont a RIMMF4-gyel ellentétben csak a MARC 21 formátumot ismeri.

Ez a két opció nem zárja ki egymást, sőt, a kettős tesztelésnek az lenne az előnye, hogy a RIMMF4-ben csak az RDA alkalmazásprofilunkat teszteljük, míg a Kohában a MARC 21-es kódolását is. Csak a Kohában való teszteléskor nehezebb lenne elkülöníteni, hogy mi alkalmazásprofil-kérdés és mi kódolási kérdés. Jelenleg az RDA-t még egy ideig MARC-kal együtt fogjuk használni, így az is fontos, hogy MARC-kal hogyan működik együtt. Másrészt jelenleg zajlik a MARC átalakítása a poszt-3R RDA-hoz, így ezeket a változtatásokat is le tudjuk majd tesztelni és visszajelzést adni a MARC

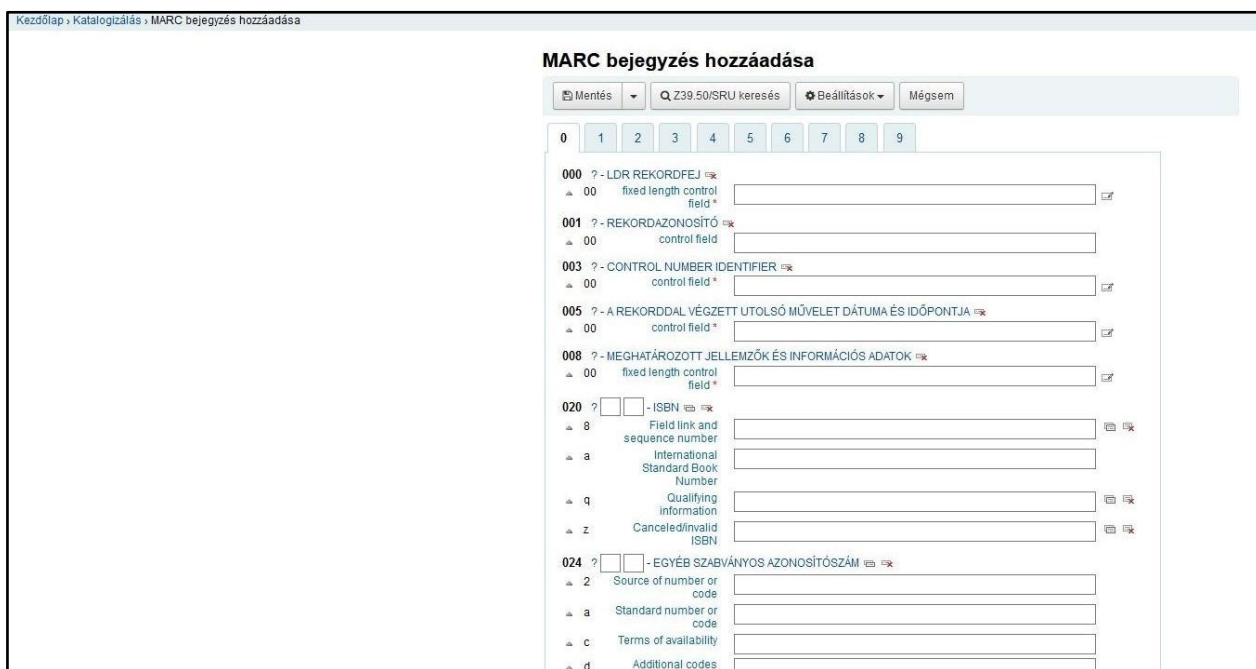
Ilácsa Sz.: Alkalmazásprofilok készítése RDA-hoz ...

Advisory Committee-nak. Mivel az átalakítás most kezdődött és MAC gyűlés félévente van, nem tudni, hogy mikor lesznek kész a munkával. Így lehet-

séges, hogy a MARC 21-es tesztet időben elcsúsztatva fogjuk végezni a RIMMF4-es tesztől.



6. ábra RIMMF4



7. ábra A Koha úrlapos katalogizáló felülete

Konklúzió

Az alkalmazásprofilok fejlesztése nem új keletű dolog még a könyvtárügyben sem. Viszont az RDA

keretrendszerre átdolgozása miatt most még aktuálisabb lett, mint valaha volt. A poszt-3R RDA már nem lesz használható alkalmazásprofil és szabályzatok nélkül. Viszont az alkalmazásprofilok kidol-

gozásának igénye megelőzte a kidolgozásukhoz szükséges módszertan kikristályosodását. Bár már léteznek alkalmazásprofil-fejlesztési módszertani anyagok, ezeket nem az RDA-hoz hasonló hatalmas elemkészletű metaadat-keretrendszerre optimalizálták. Épp ezért megfelelő segédeszközök sincsenek a fejlesztéshez, jelenleg még mindent „kézzel kell csinálni”.

A pilotprojektünk célja egyrészt, hogy létrehozzunk egy általános RDA-alkalmazásprofil, másrészt, hogy feltérképezzük, hogy milyen módszerek bizonyulnak a leghatékonyabbnak egy nagy elemkészletű keretrendszerhez készült alkalmazásprofil készítésénél.

Hivatkozások

- 1 A bibliográfiai tételek funkcionális követelményei <https://www.ifla.org/files/assets/cataloguing/frbr/frbr-hu.pdf> (Utolsó hozzáférés dátuma: 2020.05.27.)
- 2 A WEMI az első entitáscsoport angol elnevezéseinek kezdőbetűjéből alkotott betűszó. Work = mű, Expression = kifejezési forma, Manifestation = megjelenési forma, Item = példány
- 3 Nyilatkozat a nemzetközi katalogizálási alapelvekről https://www.ifla.org/files/assets/cataloguing/icp/icp_2016-hu.pdf (Utolsó hozzáférés dátuma: 2020.05.27.)
- 4 RIVA, Pat – LE BOEUF, Patrick – ŽUMER, Maja: IFLA könyvtári referenciamodell https://www.ifla.org/files/assets/cataloguing/frbr-irm/ifla-irm-august-2017_rev201712-hu.pdf (Utolsó hozzáférés dátuma: 2020.05.27.)
- 5 RDA FAQ– Gyakran feltett kérdések <http://www.rda-rsc.org/node/641> (Utolsó hozzáférés dátuma: 2020.05.27.)
- 6 HEERY, Rachel – PATEL, Manjula: Application profiles : mixing and matching metadata schemas <http://www.ariadne.ac.uk/issue/25/app-profiles/> (Utolsó hozzáférés dátuma: 2020.05.27.)
- 7 DCMI: Using Dublin Core / Glossary <https://www.dublincore.org/specifications/dublin-core/usageguide/2001-04-12/glossary/> (Utolsó hozzáférés dátuma: 2020.05.27.)
- 8 RDA Toolkit (beta) / Guidance / Application Profiles. https://beta.rdatoolkit.org/Guidance/Index?externalId=en-US_ala-591ca278-2807-399b-9530-6b44171e6ccc (Utolsó hozzáférés dátuma: 2020.05.27.)
- 9 A különböző sémákhoz tartozó metaadatok egymásnak való szemantikus megfeleltetése.
- 10 NILSSON, Mikael – BAKER, Tom – JOHNSTON, Pete: The Singapore Framework for Dublin Core™ Application Profiles <https://www.dublincore.org/specifications/dublin-core/singapore-framework/> (Utolsó hozzáférés dátuma: 2020.05.27.)
- 11 COYLE, Karen – BAKER, Tom: Guidelines for Dublin Core™ Application Profiles <https://www.dublincore.org/specifications/dublin-core/profile-guidelines/> (Utolsó hozzáférés dátuma: 2020.05.27.)
- 12 CURADO MALTA, Mariana – BAPTISTA, Ana Alice: A Method for the Development of Dublin Core Application Profiles (Me4DCAP V0.2): Detailed Description. In: DC-2013--The Lisbon Proceedings : Papers, Project Reports and Posters for DC-2013 in Lisbon, Portugal, 2-6 September 2013 <https://dcpapers.dublincore.org/pubs/article/view/3674/1897> (Utolsó hozzáférés dátuma: 2020.05.27.)
- 13 THALHATH, Nishad (et al.): Yet Another Metadata Application Profile (YAMA): Authoring, Versioning and Publishing of Application Profiles. In: DC-2019--The Seoul, South Korea Proceedings : Papers, Posters and Presentations for DC-2019 in Seoul, South Korea, 23-25 September 2019 <https://dcpapers.dublincore.org/pubs/article/view/4253/2447> (Utolsó hozzáférés dátuma: 2020.05.27.)
- 14 SZIRAY József – KOVÁCS Katalin: Az UML nyelv használata. Győr : Széchenyi István Egyetem, 2006. [elektronikus jegyzet] <http://jegyzet.sze.hu/letolt.php?dwn=2azumlnyelvhasz> (Utolsó hozzáférés dátuma: 2020.05.27.)
- 15 Más terminológia szerint: szabályozott szótár, érték-szótár. A poszt-3R RDA már következetesen a szótárkódolási séma kifejezést használja.
- 16 http://www.oszk.hu/sites/default/files/Adattisztitas_Tartalomtipus_generalasanak_elokeszitesi_esehez.pdf
- 17 JUNGER, Ulrike: RDA Board Update. Elhangzott: EURIG éves gyűlés 2020, online. Az előadás diái elérhetőek: http://www.rda-rsc.org/sites/all/files/RDA_Board_Europe_Rep_EURIG_2020_0.pdf (Utolsó hozzáférés dátuma: 2020.05.27.)
- 18 NILSSON, Mikael: Description Set Profiles: A constraint language for Dublin Core™ Application Profiles <https://dublincore.org/specifications/dublin-core/dc-dsp/> (Utolsó hozzáférés dátuma: 2020.05.27.)
- 19 SZEPESNÉ STIFTINGER Mária: Rendszertervezés 4., A rendszerfejlesztés eszközei (technikák, CASE, UML). Nyugat-magyarországi Egyetem : Digitális tankönyvtár, 2010

- https://regi.tankonyvtar.hu/hu/tartalom/tamop425/0027_RSZ4/ch01s02.html (Utolsó hozzáférés dátuma: 2020.05.27.)
- ²⁰ <http://www.marcofquality.com/wiki/rimmf4/doku.php> (Utolsó hozzáférés dátuma: 2020.05.27.)
- ²¹ DANCS Szabolcs: Hogyan fogunk katalogizálni a jövőben? FRBR-alapú bibliográfiai leírás a gyakorlatban. In: Könyv, könyvtár, könyvtáros, 26. évf. 2. sz. 2017. 10-16. p.
https://epa.oszk.hu/01300/01367/00287/pdf/EPA01367_3K_2017_02_010-016.pdf (Utolsó hozzáférés dátuma: 2020.05.27.)
- ²² DANCS Szabolcs: Digitális szurrogátumok RDA-alapú feldolgozása költséghatékonyan – az edinburghi módszer. In: Könyv, könyvtár, könyvtáros, 27. évf. 2. sz. 2017. 3-10 p.
https://epa.oszk.hu/01300/01367/00299/pdf/EPA01367_3K_2018_02_003-010.pdf (Utolsó hozzáférés dátuma: 2020.05.27.)
- ²³ DANCS Szabolcs – MOHAY Anikó – HUBAY Miklós: Az RDA és a BIBFRAME hazai implementálása. In: Tudományos és Műszaki Tájékoztató, 2019. 9. sz. 534 – 539 p.
http://epa.oszk.hu/03000/03071/00132/pdf/EPA03071_1_tmt_2019_09_534-539.pdf (Utolsó hozzáférés dátuma: 2020.05.27.)

Irodalom

A bibliográfiai tételek funkcionális követelményei

<https://www.ifla.org/files/assets/cataloguing/frbr/frbr-hu.pdf>

COYLE, Karen – BAKER, Tom: Guidelines for Dublin Core™ Application Profiles

<https://www.dublincore.org/specifications/dublin-core/profile-guidelines/>

CURADO MALTA, Mariana – BAPTISTA, Ana Alice: A Method for the Development of Dublin Core Application Profiles (Me4DCAP V0.2): Detailed Description. In: DC-2013--The Lisbon Proceedings : Papers, Project Reports and Posters for DC-2013 in Lisbon, Portugal, 2-6 September 2013

<https://dcpapers.dublincore.org/pubs/article/view/3674/1897>

DANCS Szabolcs: Hogyan fogunk katalogizálni a jövőben? FRBR-alapú bibliográfiai leírás a gyakorlatban. In: Könyv, könyvtár, könyvtáros, 26. évf. 2. sz. 2017. 10-16. p.

https://epa.oszk.hu/01300/01367/00287/pdf/EPA01367_3K_2017_02_010-016.pdf

DANCS Szabolcs: Digitális szurrogátumok RDA-alapú feldolgozása költséghatékonyan – az edinburghi módszer. In: Könyv, könyvtár, könyvtáros, 27. évf. 2. sz. 2017. 3-10 p.

https://epa.oszk.hu/01300/01367/00299/pdf/EPA01367_3K_2018_02_003-010.pdf

DANCS Szabolcs – MOHAY Anikó – HUBAY Miklós: Az RDA és a BIBFRAME hazai implementálása. In: Tudományos és Műszaki Tájékoztató, 2019. 9. sz. 534 – 539 p.

http://epa.oszk.hu/03000/03071/00132/pdf/EPA03071_tmt_2019_09_534-539.pdf

DCMI: Using Dublin Core / Glossary

<https://www.dublincore.org/specifications/dublin-core/usageguide/2001-04-12/glossary/>

HEERY, Rachel – PATEL, Manjula: Application profiles : mixing and matching metadata schemas

<http://www.ariadne.ac.uk/issue/25/app-profiles/>

JUNGER, Ulrike: RDA Board Update. Elhangzott: EURIG éves gyűlés 2020, online. Az előadás diái elérhetőek:

http://www.rda-rsc.org/sites/all/files/RDA_Board_Europe_Rep_EURIG_2020_0.pdf

NILSSON, Mikael: Description Set Profiles: A constraint language for Dublin Core™ Application Profiles

<https://dublincore.org/specifications/dublin-core/dc-dsp/>

NILSSON, Mikael – BAKER, Tom – JOHNSTON, Pete: The Singapore Framework for Dublin Core™ Application Profiles

<https://www.dublincore.org/specifications/dublin-core/singapore-framework/>

Nyilatkozat a nemzetközi katalogizálási alapelvekről

https://www.ifla.org/files/assets/cataloguing/icp/icp_2016-hu.pdf

PEYRARD, Sébastien – ROCHE, Mélanie: Still Waiting for That Funeral: the Challenges and Promises of a Next-Gen INTERMARC. Elhangzott: IFLA WLIC 2018 – Kuala Lumpur, Malaysia – Transform Libraries, Transform Societies in Session 141 – Cataloguing.

<http://library.ifla.org/2204/>

RDA FAQ – Gyakran feltett kérdések

<http://www.rda-rsc.org/node/641>

RDA Toolkit (beta) / Guidance / Application Profiles.

https://beta.rdatoolkit.org/Guidance/Index?externalId=en-US_ala-591ca278-2807-399b-9530-6b44171e6ccc

RIVA, Pat – LE BOEUF, Patrick – ŽUMER, Maja: IFLA könyvtári referenciamodell

https://www.ifla.org/files/assets/cataloguing/frbr-irm/ifla-irm-august-2017_rev201712-hu.pdf

SZIRAY József – KOVÁCS Katalin: Az UML nyelv használata. Győr : Széchenyi István Egyetem, 2006. [elektronikus jegyzet]

<http://jegyzet.sze.hu/letolt.php?dwn=2azuml nyelvhasz>

SZEPESNÉ STIFTINGER Mária: Rendszertervezés 4., A rendszerfejlesztés eszközei (technikák, CASE, UML).

TMT 67. évf. 2020. 9. sz.

Nyugat-magyarországi Egyetem : Digitális tankönyvtár, 2010

https://regi.tankonyvtar.hu/hu/tartalom/tamop425/0027_RSZ4/ch01s02.html

THALHATH, Nishad (et al.): Yet Another Metadata Application Profile (YAMA): Authoring, Versioning and Publishing of Application Profiles. In: DC-2019--The Seoul, South Korea Proceedings : Papers, Posters and Presentations for DC-2019 in Seoul, South Korea, 23-25 September 2019

<https://dcpapers.dublincore.org/pubs/article/view/4253/2447>

Beérkezett: 2020. VIII. 24-én.



Ilácsa Szabina

Metaadat-szakértő

Országos Széchényi Könyvtár

Könyvtári Szabványosítási Iroda.

E-mail: ilacsaszabina@oszk.hu