MARC szintaktikai ellenőrző program - MARCEL 1.1

Kocsis Ferenc – Völfinger Réka

feri@oszk.hu vreka@oszk.hu

Országos Széchényi Könyvtár

Az előadás célja a Kocsis Ferenc által fejlesztett MARC szintaktikai ellenőrző program bemutatása, ami a MARCEL fantázianevet kapta. A következőkben szó lesz a MARCEL -t életre hívó igényekről, a program tervezésekor fellépő nehézségekről és a felhasznált mintákról, eszközökről. Röviden ismertetésre kerül a program működési elve, majd felhasználói nézőpontból is megvizsgáljuk, kitérve a szabálygyűjtemény természetére. Végül tájékoztatást nyújtunk a fejlesztések irányáról.

Néhány mondat a program létrehozásának körülményeiről. MARC állományt bogarászni, hibák után vizslatni, esetleg kézzel belejavítani emberpróbáló feladat. Ez nem is csoda, hiszen egy „géppel olvasható" formátumról van szó. Korábban már születtek alkalmazások, amelyek képesek szekvenciálisról strukturált formára alakítani a rekordokat és vissza. Olyan szoftvereket is ismerünk, amelyek szintaktikai ellenőrzést végeznek MARC21 szabvány szerint, pl. a Marc Report. A projekt beindításához az utolsó lökést az OSZK állományán végzendő retrospektív konverzió tervezése adta, melynek során nyilvánvalóvá vált, hogy nem nélkülözhetjük tovább az említett feladatokat ellátó HUNMARC, illetve ahogy a betöltéshez szükséges előírásainkkal kiegészítettük MIGRAMARC specifikus eszköztárat. Ekkor kapta meg Kocsis Ferenc a megbízást egy felhasználóbarát, grafikus keretbe illeszkedő program megírására, amit MARCEL névre kereszteltünk.

Már a tervezés során felmerült néhány alapvető kérdés, amiket nehéz pontosan megválaszolni:

- Mi a MARC szabvány? Milyen változatai vannak? Hogyan változnak a változatok... Mit is kell tulajdonképpen ellenőrizni? Az ISO 2709 Információ és dokumentáció. Információs csereformátum című szabvány írja le a MARC rekord általános szerkezetét, teret engedve az egyéni megvalósításoknak, például a lehetséges indikátorok számában vagy az almezőhatárjel használatában. A rekordfej elemzéséből nyilvánvaló, hogy milyen megvalósítással van dolgunk, ehhez viszont feltételeznünk kell a rekordfej érvényességét, ami nem lehet az alapja egy ellenőrzésre hivatott program működésének. Különféle MARC -ok léteznek nemzeti, nyelvi, hálózati vagy intézményi szempontok szerint kialakítva, amelyek változásai a legtöbb adatbázisban jól nyomon követhetők a különböző korszakokban létrehozott rekordokban. Milyen szabályrendszer alapján kellene ezeket ellenőrizni?
- Egy MARC szabályzat hány szabályból, előírásból áll? Például, a 100 hívójelű mezőben az első indikátor 4, a második 1 és az almezőazonosító 7 értéket vehet föl. Ez 1, 4, 7 vagy 28 szabály?
- Mennyi és milyen összefüggések vannak egy állományon belül két rekord között, illetve egy rekordon belül annak elemei között. Nemcsak a szintaktikát, hanem a belső logikát is ellenőrizni kellene. Ez katalogizáló rendszerenként, szabványonként változhat. Erre jó példa a "099" –es mező, ami az OSZK –ban használt Amicus betöltője miatt kell.
- Mitől lesz egy file MARC állomány? Ha beolvasol egy XML file-t, akkor most az egy nagyon-nagyon hibás MARC állománynak számít vagy nem? Az ISO 2709 szabvány mágnesszalagos tároláshoz készült, csak annyit mond: a rekord rekordhatárjellel zárul. A beolvasás szekvenciálisan történik, karakterről karakterre haladunk a file -ban. Mikor mondhatjuk azt, hogy

ez már biztos nem MARC állomány, meddig keressük a rekordhatárjelet?

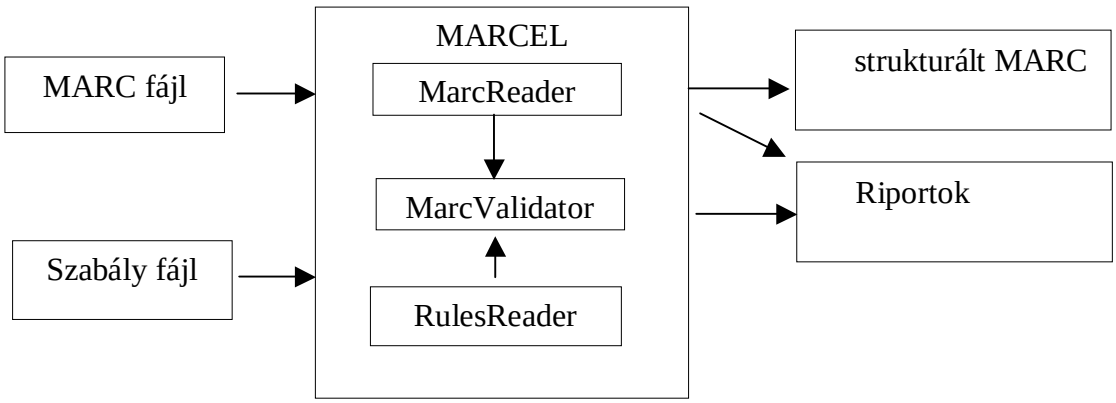
- Egy szabályt hányféle ravaszabbnál ravaszabb módon lehet megszegni? Például, tekintsük a következő szabályt! "A rekord utolsó adatmezője rekordhatárjel karakterrel (hex 1D) zárul." (Legyen a rekordhatárjel karakter röviden: RH) Elvileg bármi előfordulhat: az is hiba, ha nincs RH, az is, ha van, de rossz helyen, esetleg egy RH -t egy újabb követ.
- A legnagyobb probléma csak a végén derült ki. A szabványok helyi alkalmazása változatos formákat ölthet. Például a HUNMARC csak annyit mond, hogy "A rekordazonosító a bibliográfiai rekordot az adott adatállományon belül azonosító egyedi karaktersorozat." Így aztán találkozhatunk vele nullákkal vagy szóközökkel feltöltve, szöveges előtaggal, szóközökkel tagolva, csak az értékes számjegyeivel ábrázolva és ezek valamely keverékével egyaránt. Sőt, egy rekordon belül a különböző mezőkben sem egységes az előfordulásuk – gondoljunk csak a 001 -es és a rekordkapcsolati mezők $w almezőjére.

Így esett, hogy az első bátor nekirugaszkodás után, amikor a szabályrendszer általánosítása és teljes testreszabhatósága volt a cél – ez volt az 1.0 változat - jelenleg egy kevésbé rugalmas, de stabilan megvalósítható verzió kidolgozásánál tartunk.

A program megírását megkönnyítették az alábbi eszközök és minták.

- *MARC4J* projekt, ahol technikai eszközkészlet található a MARC – XML konverzióhoz Java API (Application Programming Interface) formájában (http://marc4j.tigris.org)
- *MARC Report* MARC21 szerinti ellenőrző program (http://www.marcofquality.com), amiből megvalósítási ötletek meríthetők.

A MARCEL program végül a következő működési elvvel valósult meg.



A program három modulból áll, amelyek funkcionálisan jól elkülöníthetők.

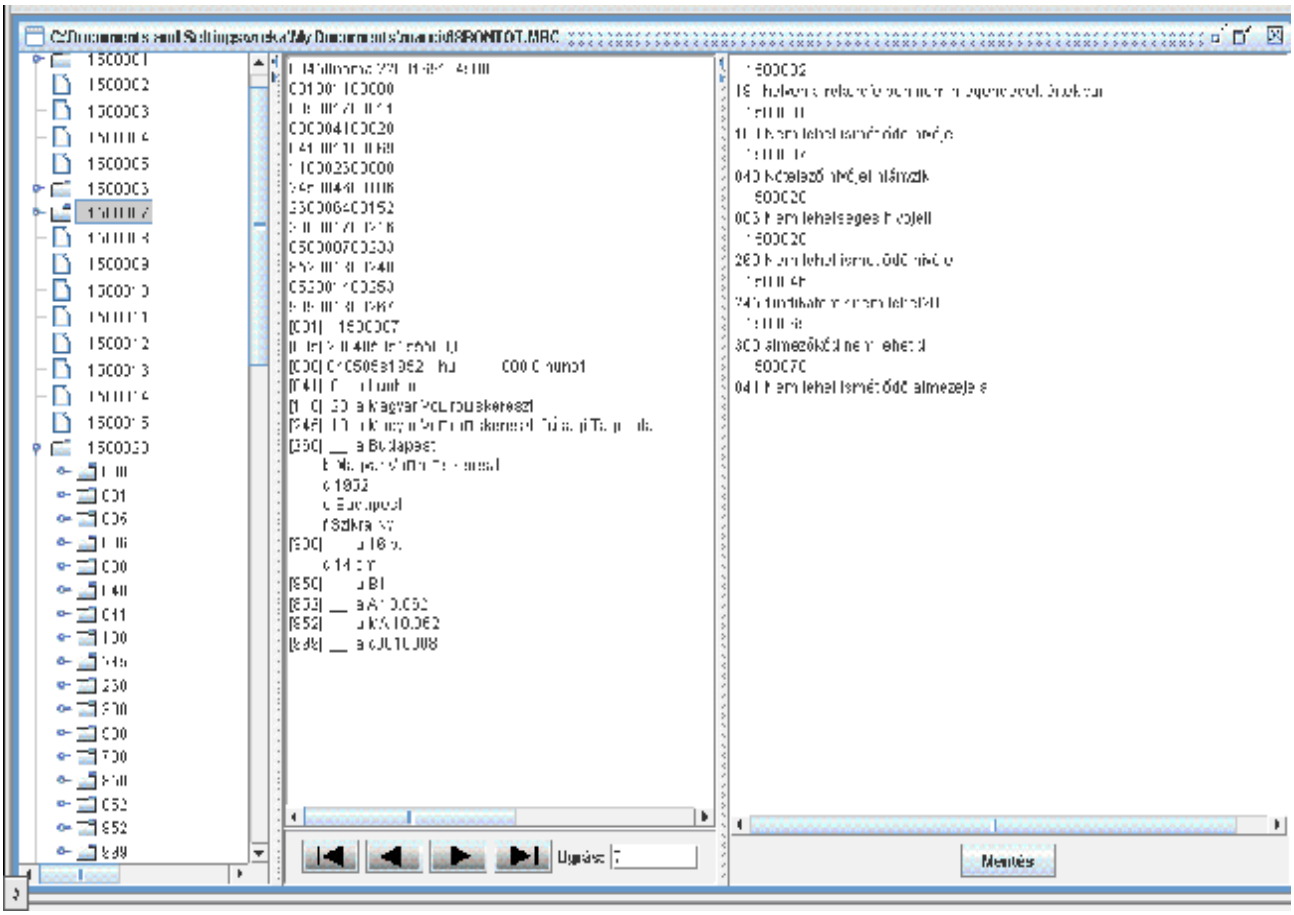
A *MarcReader* beolvassa a MARC formátumú állományt, értelmezi és alapvető formai ellenőrzést végez rajta. Felderíti a rekordok határait és szerkezeti elemeit egészen az almezők szintjéig. A feldolgozás eredményéről, ha szükséges hibariportot készít.

A *RulesReader* feladata az XML file -ben tárolt szabályok feldolgozása, értelmezése és előkészítése a validáló program számára. A szabályok egy része „bele van égetve” a programba, más része pedig ki-be kapcsolható. Később részletesebben tárgyaljuk a szabályrendszert.

A *MarcValidator* a MarcReader által beolvasott rekordok ellenőrzését végzi a RulesReader által megadott szabályok alapján és a megtalált hibákat egy riportban mutatja ki. Indítható parancssorból vagy a MARCEL rendszerből.

Tekintsünk most a programra felhasználói szemmel! Történjék ez egy példa ellenőrzésen keresztül! Készítettünk egy állományt, amiben elrontottunk néhány értéket. A program grafikus változatával beolvastattuk, és elvégeztettük rajta a szintaktikai ellenőrzést.

A grafikus programokban megszokott ablakos – menüsoros szerkezet fogadja a felhasználót, ahonnan megnyitható az ellenőrizendő MARC állomány és kiválasztható a kívánt szabálygyűjtemény illetve ellenőrzési szempont. A marc állomány strukturált formában jelenik meg, ha nem nagyobb,



mint 10000 rekord. Erre a korlátra a memória használat és a megjelenítési idő kézbentarthatósága miatt volt szükség. A rekordok illetve mezőik között gördítő sáv segítségével fa szerkezetben is tallózhatunk, de nagyobb lépésekre kényelmesebb az „ugrás” funkciót használni.

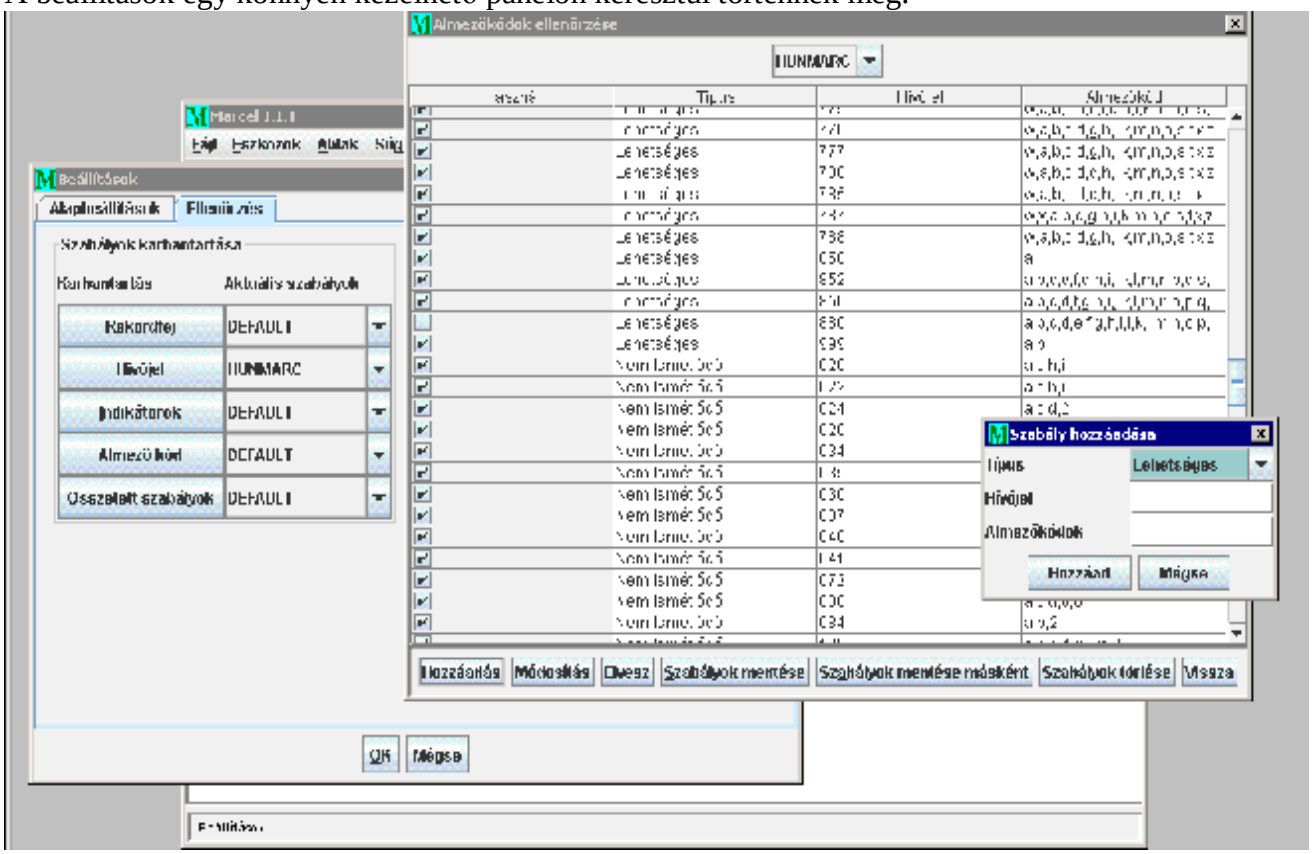
A vizsgálandó file megnyitása utána meghatározhatjuk az ellenőrzés módját: mit és milyen szempontok alapján szeretnénk tesztelni. Példánkban teljes HUNMARC szerinti kontrollt kértem.
A végrehajtás után újabb ablakrész jelenik meg a képernyőn, amely a hibaüzeneteket tartalmazza rekordazonosítóhoz rendelve, mint például
„ 1500007
040 kötelező hívójel hiányzik” .

Az ellenőrzés eredménye egy szöveg file -ba is beíródik, ahonnan célszerű elmenteni, mert minden ellenőrzéskor fölülíródik. Tervezünk még a riport elejére egy rövid statisztikai kimutatást, ami tartalmazná a beolvasott rekordok számát, az ellenőrzés időtartamát, a hibák számát, esetleg típusonként összegezve.

A szabálygyűjtemény a HUNMARC előírásait képezi le logikai állításokká, oly módon, hogy viszonylag könnyen behelyettesíthetők legyenek a konkrét értékek. A szabályok öt csoportra oszthatók: rekordfej, hívójelek, indikátorok, almezőkódok értékeire vonatkozók és az egyéb szabályok, pl. ha..., akkor típusúak, mint „ha van a rekordban 1XX –as mező, akkor 245 –ös mező első indikátora 1”.
Egy- egy csoporton belül a szabályok különféle típusúak lehetnek, pl. Kötelező hívójelek, Lehetséges indikátorok, Nem ismétlődő almezőkódok. Az egyéb szabályok csoportjában a legkülönfélébb kikötések fogalmazódnak meg a rekordkapcsolatok ellenőrzésétől kezdve az azonosítók érvényességéig – ezért nincs is hozzájuk típus meghatározva.

Szabálycsoport	Szabály típus	Módosítható értékek	Szabály hozzáadása	Szabály törlése	Használat	Csoport mentése más néven	Csoport törlése
Rekordfej	Lehetséges	igen	nem	nem	igen	igen	igen
Hívójelek	Kötelező Lehetséges Nem ismétlődő Egymást kizáró	igen	nem	nem	igen	igen	igen
Indikátorok	Lehetséges	igen	igen	igen	igen	igen	igen
Almezőkód	Lehetséges Nem ismétlődő	igen	igen	igen	igen	igen	igen
Egyéb	-	nem	nem	nem	igen	igen	igen

A felhasználó kap bizonyos mozgásteret az előírások saját kezű módosításában, és változtatásait el is mentheti mint saját szabálycsoportját. A legszorosabb megkötés az egyéb szabályok esetén van: csak a használatuk ki- és bekapcsolására van lehetőség. Ennek az az oka, hogy ezek a programba „égetett” ellenőrzések. Azokat a szabályokat kellett ily módon kódolni, amelyek egyediek a szabálykészítés szempontjából, ezért nincs szükség a paraméterezésükre, pl. duplikált rekordazonosító vizsgálata vagy olyan bonyolultak, hogy nehézséget okozna a felhasználónak a változók beállítása. Ilyen például a rekordkapcsolatok ellenőrzése.
A beállítások egy könnyen kezelhető panelon keresztül történnek meg.



Beállítható továbbá a program feliratainak, üzeneteinek nyelve, a rekordmegjelenítés és a

kimenetek karakterkészlete.

Az 1.0 változat jó szolgálatot tett az 1950 – 1975 között az OSZK állományába került könyvek cédulakatalógusainak retrospektív konverziója nemrég befejeződött első szakaszában, amikor mintegy 400 000 rekord került be a számítógépes integrált rendszerünkbe. A régi és a bemutatott 1.1 verzió között a legnagyobb különbség a szabályok szerkeszthetőségében van: a felhasználó szabadságából valamennyit fel kellett áldozni a megbízható és kiszámítható működés érdekében. Néhány további funkciót tervezünk még kialakítani a programban, hogy valóban komplex MARC eszközkészletté válhasson. Ezek az állományok darabolása, keresés bennük és a szerkesztésük lennének. Mindenekelőtt, azonban a fő célkitűzésünk megfeleltetni MARCEL -t a HUNMARC bibliográfiai formátumnak, és közzé tenni szabad felhasználásra.