Golden Dániel – ELTE Bölcsészettudományi Kar
Tóth Tünde – MTA Irodalomtudományi Intézet
Turi László – MTA Filozófiai Intézet

Virtuális örökkévalóság: objektumok a digitális könyvtárban*

A tanulmány a digitális könyvtár rögzítési, illetve archiválási módszereinek elméleti és gyakorlati kérdéseit vizsgálja nemzetközi tapasztalatok alapján. Elsősorban a szöveges dokumentumokkal foglalkozik, külön kitérve a magyar írásjelkészletből adódó sajátos nehézségekre. Emellett röviden áttekinti a kották, térképek, képek és audiovizuális anyagok digitális archiválásának gyakorlatát is, noha ezek kódolása még sokkal kevésbé szabványosított, mint a szövegeké. A digitális rögzítési szabványok közül elsősorban az SGML-t, illetve ennek alkalmazásait igyekszik bemutatni.

1. Előzetes megjegyzések

Tanulmányunk szempontjából digitális könyvtárnak nevezünk minden olyan szervezetet, amely digitalizált formában gyűjt, őriz, katalogizál és az olvasók számára hozzáférhetővé tesz publikált vagy kéziratos műveket, függetlenül attól, hogy azokat eredetileg is digitális formában készítették-e. A művek köre éppen olyan tág lehet, mint a hagyományos könyvtárakban: a folyóiratoktól a monografikus műveken át a hangzó és képi anyagokig bármi. A digitalizálás formája és módja is rendkívül sokféle, bár a gyakorlat egyre inkább egységesül, kialakulnak a szabványos, illetve konvencionális eljárások.

A digitális könyvtár tárgyait – *William Y. Arms* nyomán[1] – *digitális objektum*oknak nevezzük, melyekben megkülönböztetjük az adatot és az azt leíró metaadatot. E megkülönböztetésre azért van szükség, mert a műfajok, nyelvek és kultúrák, valamint a hardver–szoftver eszközök állandó fejlődése következtében a digitalizált anyagok napjainkra rendkívül nagy változatosságot mutatnak. A szabványosított formátumú metaadatok feladata a rendteremtés e sokféleségben.

Bár a digitális objektumok rögzítési szabványai is lehetővé teszik a leíró jellegű katalógusadatok rögzítését, úgy véljük, célszerű megőrizni a hagyományos könyvtárakból ismert, a tárgyaktól fizikailag elkülönítve kezelt, önálló katalógusadatbázisok rendszerét. Azaz szabványos formátumú bibliográfiai leíró adatbázisokat, és ugyancsak szabványos formában rögzített digitális objektumokat kell létrehozni.

A digitális objektumokkal kapcsolatos metaadatok szabványosításának helyzete azonban majdnem olyan változatos képet mutat, mint maguk az adatformátumok. A *Berkeley Digitális Könyvtár*[2] kutatói nyolc jelentősebb szabványt, illetve szabványjavaslatot különböztetnek meg, bár *csoportosításuk*[3] vitatható. Az alábbiakban a Berkeley-féle lista segítségével – ám nem pontosan követve azt – áttekintjük az egyes szabványokat. E szabványok többségének kizárólag a digitális objektum bibliográfiai leírása, katalogizálása a célja – vagyis nem az adattartalom rögzítése –, ezért ezekkel itt csak említésszerűen foglalkozunk.

A digitális objektumok rögzítési szabványainak fejlettsége meglehetősen eltérő az egyes típusok között. A legkidolgozottabb a szövegek rögzítése, a leginkább változó, fejlődő az időalapú művek (audio- és video-) rögzítése. Ha nincs gyártótól független rögzítési szabvány, a digitális könyvtár is kénytelen a legnépszerűbb gyártók formátumait alkalmazni. A következőkben elsősorban gyártóktól független szabványokat ismertetünk. A metaadatszabványok leírására szolgáló legelterjedtebb szabvány, vagyis a szabványok szabványa a később részletesen ismertetendő *SGML*, amely elsősorban szövegszerű formában kódolható digitális objektumok rögzítésére használható.

Éppen úgy, ahogy a hagyományos könyvtár is csak a legjobb minőségű papírra nyomott, legtartósabb fedélbe kötött könyv megvásárlását engedheti meg magának, a digitális könyvtár objektumainak rögzítése is csak információvesztés nélkül

* A tanulmány a Neumann-ház, a Magyar Elektronikus Könyvtár és a KFKI számára készült a Nemzeti Kulturális Alap támogatásával.

történhet. Csak olyan kódolási eljárást szabad használni, ami alkalmas az összes rendelkezésre álló információ rögzítésére, függetlenül attól, hogy erre az olvasónak éppen szüksége van-e. Nyilvánvaló, hogy az Internet mai adatátviteli kapacitása, a rendszerek inkompatibilitása még szükségszerűen határt szab az olvasó igényeinek, de ez nem szabad, hogy befolyásolja a digitális könyvtár rögzítési eljárását. A maximális hűséggel rögzített eredetiből kell előállítani az olvasó aktuális igényeinek, lehetőségeinek megfelelő gyengébb minőségű vagy egyszerűbb változatot.

1.1 Bibliográfiai szabványokról röviden

Tanulmányunk elsősorban a digitális objektumok rögzítési szabványaival foglalkozik. A digitális objektumok katalogizálásához használható szabványokról csak a teljesség kedvéért adunk rövid, utalásszerű áttekintést.

a) Nem SGML-alapúak

MARC (Machine Readable Cataloging)[4]

A számítógépes könyvtári katalógusok közismert és elterjedt formátuma, az első metaadat-szabvány. Noha a későbbiekben hivatkozunk még rá, ismertetése nem lehet e tanulmány tárgya. (Van már *SGML-alapú változata*[5] is.)

Z39.50[6]

Bibliográfiai információk lekérdezésére, a MARC alapján kidolgozott amerikai szabvány. Mivel nincs közvetlen összefüggésben a digitális objektumok tárolásával, e tanulmányban részletesen nem foglalkozunk vele.

ANSI/NISO Z39.56-199X[7]

Sorozatban megjelent művek szabványos azonosítója, tetszőleges médiumhoz.

URC (Uniform Resource Characteristics)[8]

Az Interneten elérhető digitális objektumok azonosító rendszere.

b) SGML-alapúak

EAD (Encoded Archival Description)[9]

Állományleíró levéltári, múzeumi és kézirattári segédletek, mutatók rögzítésére, illetve intézmények közötti cseréjére kidolgozott SGML DTD. A projekt 1993-ban indult a kaliforniai Berkeley Egyetemen.

DOI (Digital Object Identifier)[10]

Weben publikált művek azonosító rendszere, amely az ISBN-hez hasonlóan egyetlen kötött szerkezetű kódszámmal azonosítja a kiadót és a művet. A DOI rendszer magja egy központi adatbázis.

Dublini alapmetaadatok[11]

Mindössze tizenöt elemből álló jelkészlet, amely a HTML formátumú webfájlok leírására szolgál. A HTML formátumú fájl fejlécében elhelyezhető „META" kódok használatát szabályozza a javaslat.

2. Digitális objektumok

2.1 Történeti kitekintés

A szóbeliség korában a szöveg a lehető legszorosabban kötve volt elhangzásának teréhez és idejéhez. Az oralitás alkotásmódjából következően a közvetítés – amikor másvalaki mondta el a szöveget újra – mindig újraszerzés is volt, a közvetítő, illetve közvetítők sorának szellemi terméke.

A kéziratos korban a szöveg kevésbé volt újraformálható, a szerző sajátja volt, amelyet a mediátorok híven vagy hibásan másoltak újra. Ekkor jön létre a *textológia* tudománya, amelynek *célja a hibák kijavítása*, és a *voluntas auctoris*, a szerző szándéka szerinti szöveg megállapítása és továbbadása. A szövegkritika gondoskodik arról, hogy az olvasók a legjobb, a leginkább hiteles szöveget kapják. A másolók egyszerre csak egy szöveget, egy – esetleg csak az ideák szintjén létező – kritikai főszöveget másolnak.

A nyomtatás felfedezése lehetővé tette, hogy egy adott lenyomat (egyetlen másolat) több száz, ezer vagy millió példányban jelenjék meg; ez elvben a szöveg élettartamát is meghosszabbította, hiszen egyetlen példány megsemmisülése nem számított, nem számít pótolhatatlan veszteségnek.

A nyomtatás nem módosítja a szöveghagyományozás eljárását: egy vagy több forrásból állítanak elő egy következőt. Ebben a korszakban lehetővé válik az úgynevezett *fakszimilék* kiadása. A fakszimile elvben pontos mása az eredetijének. Vannak fényképes hasonmások, és vannak kvázifakszimilék, amelyek a nyomtatott szövegek újraszedésével készülnek. De még ezek a forrásuk pontos másának tűnő kiadványok sem tudják cáfolni a textológia alaptörvényét, mely szerint a másolat mindig különbözik az eredetijétől. Hol összecserélik a lapokat, hol elcsúsznak a többszínnyomással, vagy retusálják a pacákat, amelyekből a vérbeli textológus *Sherlock Holmes* elméjéhez illő következtetésekre képes jutni. No és természetesen nem lehet a vízjeleket, a papír állagát, az ívfüzetek terjedelmét soha, semmiféle fakszimilében híven visszaadni, s ezek megint csak olyan jellegzetességek, amelyekből messzemenő következtetéseket lehet levonni. A fakszimile kiadás tehát éppolyan szövegkiadás, mint bármely másik, csak annyiban különbözik tőle, hogy összemérhetetlenül jobban őrzi a forrás írás- és szedésképét.

Valószínű, hogy a digitális korszak alkotásmódja is más lesz, mint az előzőeké. Innen nézve a hagyományos értelemben vett „...irodalom egy olyan speciális eset, ahol a szöveghez nem kapcsolódnak kép-, hang-, illetve mozgóképállományok" [1]. A hardver- és szoftverfejlődési tendenciák is az olyan integrált „office-komplexumok" irányába mutatnak, amelyekkel szemben alapvető követelmény lesz a különböző médiumok összehangolt kezelése, egyre bonyolultabb hipermédiadokumentumok lejátszása, illetve létrehozása. A speciálisan digitális műfajok sajátos tulajdonságai még csak mostanában körvonalazódnak.

2.2 Az objektumok fajtái

2.2.1 Digitális és digitalizált

Első lépésben meg kell különböztetnünk a nem digitális (alapvetően: nyomtatott) médiumok átírásával keletkező („digitalizált") objektumokat, illetve az eleve digitális formában születőket. Az első esetben a digitális könyvtárbeli archiválás szükségszerűen átírást is jelent (lásd részletesen később), a másodikban azonban ez korántsem magától értetődő. Hiszen egy objektum legtökéletesebb archiválása magának az eredetinek változatlan formában való megőrzése.

Egy multimédia CD-ROM komplex struktúrájának visszaadására valószínűleg nehéz az eredeti kódolásnál megfelelőbbet találni, arról nem is szólva, hogy a médium alapvetően grafikus jellege miatt az ilyen típusú kiadványokról minden bizonnyal csak „fakszimiléket" lenne érdemes kiadni, minden más forma lényegi információvesztéssel járna. De *Papp Tibor: Disztichon-generátor* [2] című számítógépes költészeti alkotásának az eredeti Macintosh rendszertől eltérő környezetben való reprodukálása is tulajdonképpen a mű újraírásával volna egyenértékű.

A digitális dokumentumok archiválásakor tehát egymástól elválaszthatatlannak látszik az információ, illetve az információt fizikailag hordozó eszköz megőrzése és tárolása. Nincs ez másként a hagyományos médiumok, a könyv, a film, a hanglemez esetében sem, ezek éppúgy fizikai természetűek, mint a CD-ROM vagy a merevlemez. A különbség csak annyi, hogy a számítógépen készített dokumentumok esetében igénybe vett technikai segédlet jóval bonyolultabb, és hamarabb elavuló. Ahogy a hanglemezre karcolt zenét csak a megfelelő lejátszó segítségével tudjuk élvezhetővé tenni, a digitális dokumentumhoz is hozzátartozik az a program, amely képes a szöveget, képet, hangot, mozgóképet kezelni és megjeleníteni. Ám a program csak egy adott operációs rendszer alatt, az operációs rendszer pedig kizárólag egy adott hardverkonfiguráción futtatható.

A digitális könyvtárnak tehát valószínűleg az eleve digitális formában születő objektumok *eredetijének* megőrzése és szolgáltatása volna az egyik feladata – az ehhez szükséges hardverfeltételek biztosításával együtt. Ilyen értelemben a digitális könyvtárnak digitális múzeummá (is) kell válnia. Talán még nincs késő, hogy az utolsó HT-k, Commodore-ok és Spectrumok, a padlásokon rejtőző ChiWriterek és WordStarok begyűjtése megtörténjék!

Digitalizált és digitális közt félúton helyezkednek el az ún. kiadói fájlok: olyan elektronikusan kódolt dokumentumok, amelyek létrehozásában azonban a majdani nyomtatott változat játszik meghatározó szerepet. Nehéz eldönteni, vajon a digitális objektumot vagy inkább az ennek nyomán előállított nyomtatottat tekintsük az archiválás alapjának. Előbbi esetben (a nyomdai kódok értelmezésének nem ismeretében) lemondunk a dokumentum 'valódi' képéről, utóbbiban nyugodt szívvel vethetjük alá szisztematizáló átírásnak a digitális objektumot.

2.2.2 Két- és háromdimenziós médiumok

A digitálisan tárolt információ nem minden esetben tudja pótolni a nem digitálisat. Az időbeli művészetek, mint a film és a zene esetében a digitális technika jobbnak bizonyult a nem digitális rögzítéseknél. A térbelieknél, mint az építészet, a szobrászat, a festészet stb. a digitális formátum egyelőre inkább csak a tudományos kutatás, illetve az ismeretterjesztés segédeszköze lehet. A térbeli tárgyak digitális megjelenítésére ma ismert eljárások még nem adnak olyan mértékben valósághű másolatot, mint amennyire például a digitalizált zene az akusztikus zene reprodukciója lehet.

2.2.3 Átmeneti műfajok

A hagyományos statikus nyomtatott szöveg számára a digitális médium számos megújulási lehetőséget kínál: a térbeli szöveg, a kinetikus szöveg, a multitext (= automatikus szöveggenerátor által előállított szöveg), a hipertext, a multimédia, vagy éppen a számítógépes szótár, lexikon, adatbázis formációit. Ezen új objektumfajták kezeléséhez egészen új szemléletre van szükség: a hagyományos értelemben vett szöveg vele (legalább) egyenrangú más információkkal (képi megjelenés, hipertextkapcsolatok, hierarchikus struktúra) egészül ki; az objektum archiválásának a szövegen túli összetevőkre is messzemenően figyelemmel kell lennie.

Számos kérdést vet fel a világhálón publikált hiperszöveges dokumentumok archiválása: Hogyan őrizhetők meg a csatolókban (link) hordozott információk? Hol húzhatók meg egy ilyen dokumentum határai, azaz archiválandók-e mindazok a

távoli hálózati anyagok is, amelyekre a forrás mutat? Mennyire tekintendők a hiperszöveg részének a benne felhasznált egyéb médiumok: egy ábra természetesen igen, de a háttér már nem? Hogyan tehető értelmessé az archiválási szándék a folyamatosan változó hálózati dokumentumok esetében, ahol lényegében két lehetőség közül választhatunk: vagy egyáltalán nincs múlt (az identitás megragadhatatlan), vagy túl sok múlt van (tízpercenként automatikus backup).

2.3 A kódolhatóság szintjei

A digitális információ nem más, mint meghatározott számú jel variálása, nyolcbites rendszerben például mindössze 256 elemből áll a jelkészlet. E variációk többé-kevésbé kötött mintákat követnek: szövegfájlok esetében a kötöttség nagyobb, képfájlok esetében viszont ez kevésbé jellemző.

Valójában a hagyományos információhordozók is elhelyezhetők egy képzeletbeli skálán, amelynek egyik végén a rendkívül kis számú elemből – az angol nyelv esetében például 26 írásjegyből – sokféle mintával építkező írott szöveg áll, a másik végén pedig az elméletileg végtelen számú jelet tartalmazó kép. Minél kevesebb a jelek száma, annál könnyebb a számítógépes kódolás szabványosítása. Az európai kottaírás e képzeletbeli skálán a két véglet között, inkább a szöveghez közel kaphatna helyet, hiszen jelkészlete bővebb, de mégis megoldható kódolt, kereshető számítógépes rögzítése. A térkép azonban már inkább a másik véglet felé közelít, bár a korszerű digitális térképek és térinformatikai szabványok bizonysága szerint megoldható a véges számú elemből álló jelkészletre redukálás. Mindenesetre nem véletlen, hogy a térképi ábrázolás digitális forradalma jó néhány évet késett a szöveges információkéhoz képest. A skála másik végére helyezhető képi ábrázolás elemeinek szabványos digitalizálása pedig már szinte megoldhatatlan feladatnak tűnik, annak ellenére, hogy készülnek olyan művészettörténeti adatbázisok – például az *ICONCLASS*[12] alapján készülő *Marburgi Index*[13] –, amelyek egy-egy korszak képi emlékeit igyekeznek a közös tartalmi elemek alapján meghatározott elemszámú szöveges információvá redukálni, így lehetővé téve az adatbázisokban megszokott keresést. Ugyanakkor a képi ábrázolás jelkészletét sokkal inkább meghatározza az adott kultúra, mint az írását, így valamely képi jelkészlet tényleges leírása általában az adott korpusz teljes körű feldolgozásával lehetséges. (Itt is vannak persze kivételek, pl. a fraktálok ebből a szempontból különleges, belső mintázatú képnek tekinthetők.)

Minden információhordozóról elmondható, hogy jelkészlete időben változó. E változás általában szűkülést jelent, vagyis a jelkészlet egyre kisebb elemszámra redukálódik, amint azt később a magyar nyelv írásjelkészletének változásán is bemutatjuk. Feltehetően több médiumra is igaz, hogy jelkészletének szűkülésében, egységesedésben nem kis szerepe van a rögzítés technikai változásának, azaz a nyomtatott, újabban pedig digitális terjesztésnek.

A következőkben az egyes információhordozók rögzítésének kérdéseit a kódolhatóság szintjeit követve igyekszünk áttekinteni.

3. A digitalizált szöveg

3.1 Elméleti problémák

A nyomtatott forrásból származó szövegek digitalizálása számos elméleti problémát is felvet. Nem mindegy, hogy mit és hogyan rögzítünk. Az elektronikus szövegkiadás hasonló szakmai felkészültséget igényel, mint a nyomtatott médiumbeli. A magyar nyelvű szöveghagyomány kritikai igényű kiadása irodalomtörténészi-textológusi szakértelmet igénylő feladat. A textológus szemével nem létezik a szöveg, csak szövegek vannak [3][14] – az idők során az újabb és újabb kiadások mindig új, az eredetitől többé vagy kevésbé eltérő szövegváltozatokat hoznak létre: „... kiderült, hogy a megírás csak egy stádiuma a szöveg állandó keletkezésének, hiszen a mű a publikált változatokban is továbbfejlődik" [4]. Az elektronikus és nem elektronikus rögzítések során bekövetkező szövegromlások lehetséges fajtáira nézve lásd *Megjegyzések a Szépirodalmi polchoz 1-2 (Golden Dániel)*[15,16].

De már az sem egészen egyértelmű, hogy pontosan milyen információk rögzítendők egy nyomtatott szöveg esetében. Természetes célunk lehet, hogy egy minél intelligensebb digitális objektumot hozzunk létre, tehát minél több szemantikai jellegű információt (pl. szerző, cím, bekezdés, verssorhatár, szakaszhatár) kódoljunk. A szkeptikus megközelítés azonban nem hajlandó ilyen mértékben az aktuális szövegkódoló interpretációjára bízni magát: ő csak a szintaktikai információkra tart igényt, melyekről maga szeretné eldönteni, milyen jelentést hordoznak (pl. hogy a kurzív szedés kiemelést, idézetet vagy példát jelent-e). Az abszolút bizalmatlanság álláspontjának azonban ez is kevés (illetve sok), ő a szöveg grafikai képének pontos mását szeretné megkapni, s azt is maga kívánja eldönteni, mi minősül az adott kontextusban jelnek, s mi egyszerű papírhibának.

Különleges esetet jelenthetnek a vizuális költészeti alkotások, ahol az egyes szövegrészek pozíciói, sőt az alkalmazott betűtípus is információhordozó. (A betűtípusok valószínűleg önálló digitális objektumokként is archiválandók, s inkább a vekto-

ros felépítésű Postscript formátumban, mint TrueType-ban.) Korlátozottabb mértékben, de más szövegek esetében is szükség lehet a térbeli elhelyezkedés vagy a grafikai megjelenés bizonyos aspektusainak rögzítésére.

A kortárs textológiának a szövegváltozatok megállíthatatlan burjánzásáról szóló tapasztalata a szövegek tárolásának kérdését is más megvilágításba helyezi. A teoretikus felismeréseknek és az ezeken alapuló gyakorlatnak [5] olyan adatmodell felel meg, amelyben az alapegység nem a *mű*, s nem is a *könyv*, hanem a kettő találkozásaként létező *szöveg* [6]. Ebben az eredendően bibliográfiai struktúrában természetesen elhelyezhetők maguk a szövegek (akár többfajta átírásban), s az adott változatok fakszimiléi is.

E struktúrával szemben is megfogalmazhatók bizonyos kételyek: az egyik a 'műfaj' globális kritikája, mely szerint elvileg lehetetlen egy, a jövőre nézve teljes megoldás kidolgozása; a jövő tartalmi/technikai újdonságai, illetve az ezek megkövetelte rögzítési és tárolási kívánalmak semmiféleképpen nem láthatók előre. A másik kifogás a feldolgozandó anyagok eltérő természetére hivatkozik: például ami a régi magyar vers zárt korpuszára bevált, nem biztos, hogy minden lényeges változtatás nélkül alkalmazható lesz a közelmúlt szövegeire, nem is szólva a digitális korról, amikor ahány mentés – annyi szövegváltozat, illetve csak egyetlenegy, de állandóan újraíródó.

3.2 Szövegstruktúra

3.2.1 Project Gutenberg[17]

Minden bizonnyal az első szövegdigitalizálási kezdeményezés, minden elektronikus és digitális könyvtárak ősapja. Az 1971-ben indult projekt filozófiája a következő: „A Gutenberg Project e-szövegeinek olyan könnyen kezelhetőeknek kell lenniük, hogy soha senkinek ne okozzon gondot, hogy miként használja, olvassa, idézze és keresse őket. ... Ez arra indított bennünket, hogy a Gutenberg Project e-szövegeit 'tiszta ASCII-ban' [Plain Vanilla ASCII] jelenítsük meg. ... Ennek egyszerű oka van: ez az egyetlen olyan szövegformátum, mely kényelmes a szemnek és a számítógépnek is."

Kényelmes, mondhatnánk, mindaddig, amíg nem akarunk bonyolultabb tartalmakat kifejezni, pl. táblázatot, grafikont, matematikai képleteket alkalmazni. De szabályozás híján már az egyszerű kiemeléssel is meggyűlik a bajunk: a projekt a dokumentum létrehozójára bízza, hogy milyen karaktereket kíván használni pl. a kurzív szövegrészek jelölésére.

Az egyszerűség csapdája két ellentétes oldalról is bezárul. Egyrészt nem teszi lehetővé a szükséges információmennyiség továbbítását (a „könnyen olvasható" itt tehát azt jelenti: „túl könnyen"), márpedig egy digitális nemzeti könyvtár célja csakis a lehető legjobb minőség biztosítása lehet – egy információban gazdag dokumentum mindig lebutítható lesz az éppen aktuális szerényebb igényeknek megfelelően, fordítva azonban ez nem tehető meg. Másrészt azt az illúziót táplálja, hogy létezik egy olyan legkisebb közös többszörös, mely afféle „common sense"-ként korokon, földrészeken, nyelveken stb. átívelően mindenki számára érthető marad. Holott a háttérben egy egyszerű szabvány, az ASCII áll, amely fölött eljárt az idő, arról nem is szólva, hogy a nem angol nyelvű felhasználók igényeit már születése pillanatában sem volt képes kielégíteni.

Rendben van, mondhatnánk, ha a tiszta ASCII nem felel meg minőségi igényeinknek, keressünk olyan szöveg-, illetve kiadványszerkesztő programot, amellyel legtitkosabb vágyainkat is megvalósíthatjuk. Számtalan ilyen van, válasszuk talán a legelterjedtebbet, a Microsoft Wordöt! Valóban, ebben már elég sok mindent meg tudunk csinálni (még különböző betűtípusok is vannak). Itt azonban a Gutenberg-hívő diadalittasan csap az asztalra: azt aztán megnézheted, hogy néhány év múlva mit tudsz kezdeni az így készített dokumentumoddal! És igaza is van, hiszen a Word sok esetben még önmagával (saját korábbi változataival) sem teljesen kompatíbilis, nemhogy más forgalomban lévő szövegszerkesztőkkel; dokumentumunk elszigetelődése az idő előrehaladtával egyenes arányban nőni fog.

Mi a megoldás? Egyfelől a legteljesebb bonyolultság lehetőségének, másfelől a lehető legteljesebb program- és rendszerfüggetlenségnek a megteremtése. Csak olyan rendszer felel meg egy hosszú távú digitális szövegrögzítés céljaira, amely képes tökéletesen követhető módon számot adni saját kódolási eljárásairól, s ezzel biztosítja a platformfüggetlenséget, az átjárhatóságot, szükség esetén az anyagok megbízható konvertálhatóságát.

A megoldás a két szempont összeegyeztetése egy olyan metanyelvben, amely lehetővé teszi számunkra, hogy tetszőleges bonyolultsági fokú szövegkódolási rendszert definiáljunk, pusztán a tiszta ASCII karakterek felhasználásával. Ez a rendszer az SGML.

3.2.2 SGML (Standard Generalized Markup Language – ISO 8879:1986)[18]

A metaadat használatának alapfeltétele, hogy egyértelműen elválasztható legyen magától az adattól. Az SGML szabvány alkalmazásaiban a

„<>" zárójelpár-karaktereket szokás használni erre a célra. Azt, hogy milyen jellegű metaadatot érdemes rögzíteni, a szövegrögzítés célja határozza meg. Ha a könnyű programozhatóság fontosabb, mint az adat rendszerfüggetlensége, a rendszerek közötti átjárhatóság, akkor a metaadatok feldolgozási utasítások lesznek. A hardver- és szoftverfüggetlen adat kulcsa azonban az, hogy a metaadatok általánosabb érvényű információt rögzítsenek: minősítsék azt az adatot, amelyre vonatkoznak. Vagyis nem magát a végrehajtandó utasítást kell metaadatként megadni, hanem a szövegszegmentum azon tulajdonságát, amely ezt rendszeresen kiváltja, feltételezi.

Egy példával: ha az alcím kiemelését például a kövér betűkre vonatkozó utasítással jelezzük, az más rendszerben csak akkor lesz értelmezhető, ha a kövér betűk ott is ugyolyan kódot kapnak. Ezért célszerűbb azt rögzíteni, hogy az adott szövegrészlet alcímként funkcionál. Szöveges adatok esetében például olyan jellemző tulajdonságot kell rögzíteni, amely képes kifejezni az adott szegmentum viszonyát a szöveg többi szegmentumához. E szempontoknak megfelelő tulajdonság a szövegszegmentumok hierarchikus elrendezettsége. *J. M. Lotman* [7] szavaival:

„A szöveg hierarchikus jellege, azaz rendszerének szétesése alrendszerek bonyolult konstrukciójára, oda vezet, hogy a belső struktúrába tartozó számos elem a különféle típusú alrendszerekben határjellegűnek bizonyul (a fejezetek, strófák, verssorok, félverssorok határai). A határ, amely jelzi az olvasónak, hogy szöveggel van dolga, és tudatában felidézi a megfelelő művészi kódok egész rendszerét, strukturális jellegű, erős helyzetben. Tekintve, hogy egyes elemek egy bizonyos határ jelei, mások viszont néhány, a szövegben elfoglalt általános helyzetük szerint egybeeső több határ jelei (egy fejezet vége egyben a könyv vége); tekintve, hogy a hierarchia szintjei alapján beszélhetünk egyik vagy másik határ domináns helyzetéről (a fejezethatárok hierarchikusan magasabban állnak, mint a strófahatárok, a regény határa magasabban, mint a fejezet határa), strukturálisan összemérhetővé válik az elhatárolás ilyen vagy amolyan jelének szerepe. ... A fenti tételek alapján hasznos szabályok adódnak. Először: a szöveg leírásának nyelve – hierarchia." (58–61)

Az SGML szabvány elsődleges feladata az, hogy szintaktikai szabályokat biztosítson a szöveg hierarchikusan rendeződő elemeinek formális leírásához. Tehát a szabvány nem azt határozza meg, hogy az egyes szövegtípusokban milyen szegmentumrendszer feltételezhető, hanem azt, hogy ez a rendszer hogyan definiálható. A gyakorlat tanúsága szerint a rögzítés, illetve felhasználás céljától, koncepciójától függően egyszerre többféle hierarchia is megjelenhet egyetlen dokumentumban. Ezeket az SGML szabvány alapján egymással azonos értékű, ún. konkurens struktúrákként lehet kódolni, vagyis a szöveget nem kell kizárólagosan hozzárendelni például az oldalszámok vagy a műfaji egységek struktúrájához, azaz párhuzamosan, az összekeverés veszélye nélkül jelölhetők pl. az oldal-, illetve strófahatárok.

Nem határozza meg a szabvány azt sem, hogy az így definiált szegmentumok milyen számítógépes eljárással azonosíthatók vagy dolgozhatók föl a szövegben, ezek a szövegfeldolgozó programrendszerek feladatai. A szabvány feltételezi – de nem írja elő – egy olyan számítógépes programnak a használatát, amely kapcsolatot teremt a szövegben jelölt szegmentumok, a szegmentumok definíciója, valamint a szöveg lehetséges felhasználása között. E program (a szabványban: „parser") ellenőrzi, hogy a szövegben jelölt szegmentumok azonosak-e a definiáltakkal, és emellett esetleg e szegmentumokat át is tudja szervezni.

A program működéséhez a következők szükségesek:

- A szintaxis bizonyos alapvető jellegzetességeinek a definiálása, például a különböző speciális karaktereknek, a kódok hosszának vagy a szintaxisban megengedett variációknak a meghatározása. Bár a hasonlat nem tökéletes, mondhatni, hogy ez a szövegleíró nyelv alkalmazott nyelvjárásának leírása. Ennek az adatcsoportnak a szabványban használt neve: „SGML Declaration".
- A szövegben feltételezhető hierarchikus szegmentumrendszernek és jelölésének leírása. A szabványban ennek a leírásnak a neve „Document Type Definiton" (DTD).
- Természetesen a feldolgozandó szöveg is szükséges a program működéséhez. A szabványban az ilyen számítógépen rögzített, és a DTD-ben meghatározott szegmentumok jelölésével ellátott természetes nyelvi szöveg neve: „document instance". Egy „SGML Declaration" egy vagy több DTD-re vonatkozhat, és egy DTD egy vagy több „document instance" szegmentumrendszerét írhatja le. Azokat a szövegeket, amelyeknek azonos a szegmentumrendszere, a szabvány szövegtípusoknak tekinti.

A hierarchikusan rendeződő szövegszegmentumoknak, azaz a hierarchia alkotóelemeinek a szabványban használatos elnevezése: „element" (magyar megnevezése a továbbiakban: alkotóelem). A rendszerben az azonos szerepet betöltő alkotóelemek azonos nevet kapnak, amely nem fejez ki semmi mást, csak az adott szegmentum viszonyát a szöveg többi szegmentumához.

Azt mondhatjuk, hogy az a szövegleíró nyelv, amelyik utasításokat ad a számítógépnek (például: „innentől aláhúzandó!"), csak igéket használ, az SGML pedig csak névszókat: a szövegszegmentumok nevét, és az azt egyedítő, a hasonló nevű szegmentumoktól megkülönböztető tulajdonságo-

kat. Ugyanis a szabvány szerint a szegmentumot nyitó azonosító jel – a zárójelpár között – nemcsak a szegmentum nevét tartalmazhatja, de egy vagy több erre vonatkozó „attribute"-ot is (a továbbiakban: tulajdonság). Mivel egyetlen szegmentum többféle szempontból, azaz többféle tulajdonsággal is jellemezhető, ezért a tulajdonság is két részből áll: a tulajdonság megnevezéséből és az adott szegmentumra jellemző ún. értékéből.

Az „entity" is a szöveg valamely szegmentumára való hivatkozás eszköze, akárcsak az „element" fogalma. De ellentétben az „element" fogalmával, amely az adott szegmentumra annak szegmentumrendszerbeli pozíciójával utal, az „entity" a szegmentumot egy mindenféle hierarchiától független egységnek tekinti. Ez az egység a szegmentumok hierarchiájának bármelyik szintjén előfordulhat, tetszőleges méretű, strukturált és strukturálatlan is lehet. Az „entity" fogalma azt hangsúlyozza, hogy a szöveg kisebb és nagyobb – egymástól független – szegmentumokból áll: a teljes dokumentum is egy „entity", a legnagyobb. Ezzel szemben az „element" fogalmának lényege a szegmentumok hierarchiába rendeződő kapcsolata, függősége.

A szabványban e két szemlélet nem ellentétben áll, hanem kiegészíti egymást: az „entity" fogalma amellett, hogy lehetővé teszi a szövegen kívüli bármely egységre való egyértelmű hivatkozást, számos más feladatra is alkalmas.

Az „entity" alkalmazása igen egyszerű. A dokumentumot megelőző DTD-ben kell – a szabványban meghatározott szintaxissal – azonosítani az egyed tartalmát annak tetszőlegesen választott nevével, majd a dokumentumban ezzel a névvel lehet az egyedre hivatkozni, az egyed nevét az elején „&" és a végén „;" karakterrel választva el a szövegtől. A szabvány szerint működő elemzőprogram behelyettesíti az egyed nevét annak tartalmával. Az egyed olyan – legalább egy karakterből álló – szövegegység helyettesítésére, azonosítására szolgál, amelyet a számítógép közvetlenül nem tud feldolgozni, megjeleníteni vagy továbbítani, illetve amelyet a felhasználó valamilyen okból nem kíván közvetlenül kezelni – például azért, mert túl hosszú.

Az SGML szabvány a szintaktikai szabályokon kívül olyan listákat is tartalmaz, amelyekben számos közvetlenül föl nem dolgozható karakter – például a mai magyar nyelv összes ékezetes magánhangzója –, valamint matematikai és egyéb szimbólum külön-külön egyedként van azonosítva. Ezekben a listákban (a szabványban: „public entity set") minden egyed meghatározásához hozzátartozik a hivatkozott karakter formájának rövid, angol nyelvű leírása. Az egyedek neve – tehát az a rövidítés, amellyel a szövegben hivatkozni kell rá – természetesen csak az angol ábécé karaktereit tartalmazza. A listák csak egyezményes hivatkozási alapként használhatók, semmiféle utalást nem tartalmaznak a karakter nyomtatón vagy képernyőn való megjelenítésére – hiszen a szabvány alapelve az, hogy csak az adat tárolását határozza meg, felhasználását nem.

3.2.3 HTML (Hypertext Markup Language)[19]

A web közismert adatformátuma, bemutatására nincs szükség. Fontos azonban megjegyezni, hogy a HTML is egy SGML-alkalmazás, vagyis egy DTD, amelyet a World Wide Web Consortium definiált. A webböngészők valójában olyan SGML-olvasók, melyek csak egyetlen – viszonylag egyszerű – DTD feldolgozására alkalmasak. A HTML DTD elsősorban olyan alkotóelemeket tartalmaz, amelyek a képernyő-megjelenítést szabályozzák, vagyis minimális mértékben határozza csak meg az adat logikai-szemantikai szerkezetét, hierarchiáját. Bár tartalmaz strukturált szemantikai leíráshoz használható kódokat – például a H1, H2 és H3 címeket vagy az „address" alkotóelemet –, ezek rendeltetésszerű használata azonban nem terjedt el. Ebből következően a HTML formátum kevéssé alkalmas jól visszakereshető, strukturált digitális objektumok rögzítésére.

3.2.4 TEI (Text Encoding Initiative)[20]

Három számítógépes nyelvészeti és irodalmi kutatásokkal foglalkozó angolszász tudományos társaság indította a projektet 1987-ben. Az akkor meghatározott cél kiállta az idők próbáját: olyan szöveg-, illetve adatrögzítési útmutatót készíteni, amely egyaránt szoftver-, hardver- és alkalmazásfüggetlen, és alkalmas bármilyen nyelvű és korú szöveg rögzítésére. Rendkívül fontos, hogy nem szabványt, hanem útmutatót kívántak alkotni a kezdeményezők. Az első ilyen dokumentum 1990-ben készült el, majd a szakmai vita alapján 1992-re a második, végül 1994-re a harmadik, immár véglegesnek tekintett ajánlás is megjelent. Az egyes változatok között egyre szélesebb kört vontak be az értékelésbe, ez persze az eredeti elképzelésekhez képest szükségszerűen egyre több kompromisszumos elemet is hozott. Jelenleg folyik a végső szöveg kisebb hibáinak javítása, illetve a projekt további sorsának a vizsgálata. Az ajánlás azonban lényegében késznek tekinthető.

A TEI-ajánlás egyaránt igyekszik állást foglalni a metaadat-rögzítés két fő kérdésében: mit és hogyan? Az utóbbira a válasz egyszerű: a TEI a metaadatokat az SGML szabványnak megfelelően rögzíti. A TEI tekinthető az SGML szintaxison alapuló szemantikai rendszernek. Az előbbi kérdésre ugyanakkor az ajánlás igyekszik egy minél szélesebb kör számára elfogadható megoldást kínálni, melyet az Oxford és Chicago központtal működő

projekt vezetői gyakran hasonlítanak a chicagói pizzához: a vendég előbb eldönti, hogy vékonyabb, ropogós vagy vastagabb, kenyérszerű tésztát kér, majd kiválasztja a feltéteket. A szövegrögzítésre vonatkoztatva ez azt jelenti, hogy az ajánlás meghatározza (1) a minden szövegre érvényes alapkódkészletet *(core tags)*, (2) a főbb szövegtípusok szerint különböző hat fő kódkészletet *(base tag sets)*, és (3) az elsősorban a szövegrögzítés céljától függő kiegészítő kódkészleteket *(additional tag sets)*, amelyekből tizet különböztet meg az ajánlás.

1. Az *alapkódkészlet* nemcsak a legelemibb, általánosan érvényes szövegelemek metakódját tartalmazza (bekezdés, sor, dátum stb.), de a szöveg egészére vonatkozó bibliográfia jellegű információkat tartalmazó ún. fejlécet *(header)* is. Az ajánlás meglehetősen sokféle információ rögzítését javasolja, bár nem zárja ki a pusztán azonosításra szolgáló, minimális fejléc alkalmazását sem.
2. Az ajánlásban megkülönböztetett főbb *szövegtípusok:*
   - próza,
   - vers,
   - dráma,
   - lejegyzett beszéd,
   - nyomtatott szótárak,
   - terminológiai adatbázis.
3. *Kiegészítő kódkészletek:*
   - hipertextkapcsolatok, mutatók jelölése;
   - analitikus információk kódolása;
   - strukturális nyelvészeti és más elemzések eredményének kódjai;
   - a szöveg értelmezésekor, rögzítésekor felmerülő bizonytalanságok jelölése;
   - kéziratos források átírásánál használatos különleges jelek;
   - kritikai szövegrögzítés;
   - nevek és dátumok kódolása;
   - gráfok, fák és hálózatok ábrázolása;
   - táblázatok és képletek;
   - nyelvi korpuszok.

Az ajánlás megkülönböztet egy negyedik csoportot is, a *járulékos dokumentumok*at *(auxiliary document types)*, amelyek közül a legfontosabbnak az ún. írásrendszer-definíció tűnik *(writing system declaration)*. Az írásrendszer-definíció célja, hogy meghatározza a kérdéses nyelv, lejegyzésének módja (ábécé, szótagírás stb.), és a lejegyzéshez használt írásjelkészlet közötti összefüggéseket. E technikai megoldás tehát lehetővé teszi a magyar karakterkészlet pontos rögzítését.

Az ajánlás jelenlegi szövegét többévi tudományos vita és konszenzuskeresés előzte meg, amit a bevezetőben így összegeznek a szerkesztők:

„Az ajánlás nem kíván különbséget tenni a szövegre vonatkozó »objektív« és »szubjektív« információ, illetve a szöveg »megjelenítése« és »interpretációja« között. E megkülönböztetések – noha szűkebb, jobban meghatározható összefüggésben gyakran hasznosnak bizonyulhatnak – itt leginkább úgy jelentkeznek, mint olyan kérdések, melyekben lehetséges a tudományos konszenzus, és olyanok, melyekben nem. Kétségtelen, hogy e konszenzus tartalma megváltozhat. A TEI-ajánlás nem ajánlja és nem teszi kötelezővé semmiféle metaadat rögzítését. ... A szövegrögzítés pontosságáról, illetve az interpretáció helyességéről mindig magának a felhasználónak kell döntenie. Az ajánlás csak eszközt biztosít magának a szövegrögzítésnek a dokumentálására, így maga az eljárás, illetve a mögöttes értelmezői döntések átláthatóvá válnak a szöveg felhasználója számára."

A TEI-ajánlást számos bírálat éri, általános jellegű és lényegi egyaránt. Az 1200 oldalas dokumentáció természetesen nem könnyű olvasmány, a teljes rendszer megismerése az SGML ismerete mellett is komoly elmélyülést igényel. Kezdő TEI-alkalmazóknak nagy segítség lehet a *TEI Lite DTD*[21], amely egyszerűsített, ám a teljes változattal kompatibilis kódkészletet kínál.

A felhasználótól megkívánt erőfeszítések mellett a TEI jelentős gépi erőforrást is igényel, ugyanis a TEI DTD meglehetősen bonyolult SGML-alkalmazásnak számít. Az első változat elkészültekor, a 90-es évek elején nem volt olyan PC-kompatíbilis szoftver, amely a teljes DTD feldolgozására képes lett volna, a közismert DOS-os memóriakorlátok miatt erre akkoriban csak Unix- és Macintosh-alapú számítógépek voltak képesek. Ezek az akadályok azóta elhárultak, de a kezdeti nehézségek nyomát őrzi az, hogy a TEI-kódolók által legelterjedtebben használt SGML-editor továbbra is a Macintosh-alapú Author/Editor (SoftQuad). Köszönhetően az SGML egyre nagyobb népszerűségének az ipari dokumentáció és a könyvkiadás (elsősorban szótár- és lexikonkiadás) területén, napjainkban már rendkívül széles az SGML-kompatíbilis shareware és kereskedelmi szoftverek választéka.

A technikai és tartalmi nehézségek ellenére a TEI-ajánlás akadálytalanul terjed, ugyanis gyakorlatilag nincs alternatívája, vagyis jelenleg nincs más, ilyen mélységig kidolgozott metaadat-ajánlás. Az angolszász szövegtudományokban meghatározónak számító tudományos társaság, a Modern Language Association (MLA) 1997 augusztusában *elfogadta mint kötelező szövegrögzítési formát*[22]. *Robin Cover* SGML-oldala[23] 1998 elején közel ötven olyan jelentősebb projektről tud, amely a TEI-ajánlás alapján működik. Noha a rögzített szövegek többsége továbbra is angol nyelvű, ma már jelentős számban található közöttük francia, latin, olasz, német, holland, svéd, norvég, spanyol, japán, görög és héber nyelvű szöveg is.

Külön említést érdemel a *MULTEXT projekt*[24], illetve ennek kelet-európai változata, a *MULTEXT-*

East [25]. Az Európai Unió által támogatott MULTEXT projekt célja az, hogy a TEI-ajánlás alapján kialakított ún. *Corpus Encoding Specification* DTD-nek megfelelő többnyelvű mintakorpuszokat hozzon létre, ezzel tesztelve a többnyelvű szövegek számítógépes feldolgozásának lehetőségeit, illetve a TEI alkalmazhatóságát nem angol nyelvű szövegekre. A MULTEXT-East projekt keretében egyetlen regényt (*Orwell* 1984 c. művét) rögzítettek tíz nyelven, az egyes változatokat egységes azonosító rendszerrel rendelve egymáshoz. A tíz nyelv: bolgár, cseh, észt, lett, litván, magyar, orosz, román, szerb-horvát és szlovén. (A magyar változatot *Tihanyi László*, a Morphologic, illetve *Oravecz Csaba*, az MTA Nyelvtudományi Intézet munkatársa készítették.)

Hasonló kísérleteket folytat a Copernicus együttműködési program keretében a *TELRI projekt* [26] is, *Platón* Állam című művét rögzítették 17 nyelven.

A MULTEXT-East szövegen kívül magyarországi TEI-alkalmazásokról nincs tudomásunk, bár SGML-alapú szövegkorpuszokról és projektekről igen: házi készítésű DTD-t használ a Nyelvtudományi Intézet a készülő nagyszótár szövegkorpuszához és az Akadémiai Kiadó is.

A részleteket érintő, lényegi bírálatok és kérdések számára a TEI-projekt Listserv-fórumot tart fenn, melyen a felhasználók észrevételeire a szerkesztők, illetve a tapasztaltabb felhasználók válaszolnak.

3.2.5 SGML és a web

A HTML sokat segített és sokat ártott is az SGML szabványnak. Segített, hiszen a World Wide Web nélkül az SGML a legtöbb számítógéphasználó számára talán még ma is az az obskúrus műszaki szabvány volna, ami 1987-ben, amikor a TEI projekt vezetői elhatározták, hogy ezt kell megtanítani a bölcsészeknek. Az egyszerű HTML-böngészők és -szerkesztők elterjedése óta az *element*, *entity* és *attribute* fogalma szinte trivialitásnak számít. Természetesen a DTD és a hierarchikus szövegstruktúra továbbra is ismeretlen maradt, hiszen – mint említettük – a HTML egyetlen, szinte kizárólag tipográfiai célú DTD-t használ.

A HTML-alapú web hihetetlenül gyors világméretű hódításának első éveiben az ortodox SGML-felhasználók fanyalogva beszéltek a HTML-ről, mint valami játékszerről. A szerver-kliens szerkezetű dinamikus webadatbázisok megjelenése azonban megváltoztatta a helyzetet. *Lou Burnard*, a TEI-ajánlás egyik szerkesztője – maga is ortodox SGML-felhasználó – *1996-ban már így ír* [27]:

„Mégis miért használjuk a HTML-t? A gazdasági, politikai és szociológiai érvek mellett van még egy eddig figyelmen kívül hagyott szempont: a web tartalmának jelentős része eredendően tiszavirág-életű. Ezek az anyagok csak »itt és most« kívánnak hatni, például terméket eladni vagy egyszerűen szenzációt kelteni. Ebből következően semmi értelme ezekre több energiát pazarolni, mint a hasonló papírbrosúrákra. A gondot inkább az okozza, hogy éppen úgy HTML-t kell használnunk, ha egy fontos kézikönyvet digitalizálunk, mint ha éppen egy üdítőitalt reklámoznánk.

Valójában azonban még az értékesebb művek rögzítésénél is csak akkor tűnik fel a HTML gyengesége, ha a szerző vagy a kiadó szempontjából vizsgáljuk a helyzetet. Ha a képernyőkép tetszetős, az olvasó számára végső soron mindegy, hogy az korszerű objektum-orientált adatbázis-kezelőből, postscript fájlból vagy pedig feketemágiával előállított HTML-fájlból származike. ... A HTML-nek mint szerver-formátumnak van néhány nyilvánvaló hátránya. Noha a kezdeti költségek alacsonyak, HTML-dokumentumokkal aligha tanácsos komolyabb, hosszabb távú szolgáltatást indítani. A hivatkozások konzisztenciájának megőrzése már egy csak viszonylag dinamikus állomány esetében is rendkívül sok fejfájást okozhat."

A megoldást, úgy tűnik, a tényleges SGML és a kurrens HTML-verzió ötvözése jelentheti, mindkettőt arra használva, amire való: valódi SGML formátumot használni a szerveroldalon, és HTML-t a kliensoldali megjelenítéshez. Íme néhány e hibrid megoldás előnyei közül Burnard idézett cikke alapján:

A szerveroldali SGML előnyei:

- A dokumentum rögzítése, kódolása illeszkedhet a szöveg, illetve felhasználásának sajátosságaihoz.
- A keresés kontextusérzékeny lehet, ezáltal sokkal pontosabb, használhatóbb eredményt ad.
- A dokumentum könnyen konvertálható, illetve adaptálható tetszőleges célokra.

A kliensoldali HTML előnyei:

- A böngészők könnyen hozzáférhetők minden hardverplatformon.
- A letölthető böngésző plug-in programokkal sokféle kiegészítő funkció megoldható.
- A HTML stíluslapok terjedésével a megjelenítés is könnyebben befolyásolható.

Az első és legismertebb SGML-alapú WWW szerver az Oxford English Dictionaryhez használt PAT szövegkezelő kiegészítése, ezt használja a Michigani Egyetemen működő *Humanities Text Initiative* [28] projekt és a Virginiai Egyetemhez tartozó *Electronic Text Center* [29]. E megoldásban a felhasználótól kapott keresőkérdéseket a szerver SGML formátumúvá konvertálja, majd a megfelelő szövegrészeket SGML-ből HTML-be, és ezt küldi a felhasználónak vissza. Ez meglehetősen nagy teljesítményű szervert igényel. Egyszerűbb, olcsóbb megoldásnak tűnik az, ha a szerver csak SGML formátumú szöveget szolgáltat, a megje-

lenítés pedig egy kliensoldali kiegészítő program (plug-in) feladata. Erre jelenleg csak a SoftQuad[30] által készített, ingyen letölthető Panorama SGML-olvasó alkalmas.

E terület napjainkban rendkívül gyorsan fejlődik, különösen ígéretes lehetőségnek tűnik a World Wide Web Consortium által 1998. február elején elfogadott *XML 1.0 webszabvány*[31], melyet a TEI-felhasználók is üdvözöltek. Az XML az SGML egyszerűsített változata, vagyis többféle dokumentumtípus rögzítéséhez használható metaadat-szabvány (szemben a HTML-lel, amely csak egyféle dokumentumtípushoz használható). Az XML szabvány nem tartalmazza az SGML azon részeit, amelyek nehezen programozhatónak és ritkábban használtnak bizonyultak, ugyanakkor megőrzi az SGML flexibilitását. Bár az XML még rendkívül új fejlemény a web világában, a Microsoft már a szabvány bejelentése előtt elkészítette saját *Java-alapú XML-parser*[32] programját.

3.3 Karakterek kódolása

3.3.1 A betűtől a karakterig

A betűírás a kezdetektől fogva grafikai tevékenységet jelentett, a betűket rajzolni kellett, ezért aztán ahány kézírás, annyiféle *a*, *b* stb. létezett (eltekintve persze a normatív, kalligrafikus írásmódtól). Ebben a tekintetben nem hozott alapvető változást a nyomtatás megjelenése sem: a manufakturálisan előállított ólombetűk szintén elég nagy változatosságot mutattak.

Az így előállított grafikus jelek értelmezésekor ahhoz hasonlóan járunk el, mint amikor a különböző frekvenciájú hangokat élesen elkülönített fonémák rendszerévé alakítjuk. A különböző formákat adott tűréshatáron belül egy bizonyos *graféma* eltérő megjelenéseinek tekintjük. A graféma tehát az írásnak nevezett ponthalmaz legkisebb jelentéssel bíró egységeként definiálható.

Az egységesülés felé az első lépést a fénynyomás megjelenése jelentette: itt már valóban identikus betűkről beszélhetünk. Ez azonban egy olyan lényeges változással járt együtt, ami azután a számítógépek esetében csak még hangsúlyosabbá vált: a betűt mint grafémát felváltotta a betű mint számkód. Az új digitális technika alapja a kód és funkció közötti kölcsönösen egyértelmű megfeleltetés lett; jelen esetben egy adott számkódhoz egy adott betű megjelenítése tartozik. A sorrend felcserélődött: többé nem a sok különbözőből hozunk létre absztrakcióval egy ideális közöset, hanem az absztrakt általánosat igyekszünk további manipulációkkal (betűtípus stb.) minél egyedibbé tenni.

E két eltérő logika különbsége a következőképpen szemléltethető: míg az első szerint az *á* az egy *a* és egy ' együttese, addig a második szerint nem más, mint pl. '225', aminek lényegében semmi köze az *a*-hoz, ami viszont '97'.

3.3.2 Kódtáblák

A számítógép tehát karakterkódokban gondolkodik. A karakterek egy adott gyűjteményét karakterkészletnek (character set) nevezzük. Az egy készletbe tartozó karaktereknek bináris kódokhoz való egyértelmű hozzárendelésével egy kódtáblát (codepage) kapunk. (Korábban a technikai lehetőségek korlátai miatt, mivel minden karakter nem fért el egyszerre, alternatív kódtáblákat kellett kidolgozni, s így az elviekben akár egységesnek is tekinthető „latin betűs írásrendszerű nyelvek" karakterkészletét is alkészletekre kellett szétszabdalni; bővebben lásd később.) A humán felhasználó kedvéért azonban egy további lépésre is szükség van: a gép kénytelen a vizuális (képernyőn, illetve nyomtatón való) megjelenítésről gondoskodni. Amikor szövegszerkesztő programmal dolgozunk, akkor egy adott karakterkészlethez készült betűkészletet (font) használunk. Egy betűkészlet az egyes betűk és más jelek grafikus képeiből (glyphs) épül fel. Például a 245-ös kód az ISO Latin 2-es kódtábla szerint a magyar kis hosszú *ő*-t jelenti, ez azonban csak akkor fog valóban ilyenként megjelenni, ha egy, a kódokat ennek a kódtáblának megfelelően interpretáló betűkészletet alkalmazunk, pl. a 'régi' Windows-betűkészletek közül a „CE" jelűeket. A betű grafikai megjelenése pedig azon múlik, milyen betűtípus (Helvetica, Times stb.) alapján készült az általunk választott betűkészlet (lásd még *Character sets and codepages*[33]).

Tekintsük át röviden a kódtáblák (leegyszerűsített) történetét – magyar szemmel. (Az alkalmazott rövidítések: ANSI = American National Standards Institute[34]; ISO = International Organization for Standardization[35]; MSZH = Magyar Szabványügyi Hivatal.)

A személyi számítógépek első karakterszabványa az ASCII (American Standard Code for Information Interchange) volt. Ez a szabvány (ANSI X3.4-1986 [R1992], illetve ISO 646:1991) 7 bitet bocsátott rendelkezésre a kódok tárolásához, ennek megfelelően 128 különböző karakter (vagy egyéb jel) egyidejű használatát tette lehetővé. Ez tökéletesen elegendő is volt az angol nyelvű szövegek esetében, ám figyelmen kívül hagyta a más nyelveken kommunikálni kívánók igényeit.

A következő lépés a 7 bitről 8 bitre való áttérés, a lehetőségek megduplázása volt. Az így rendelkezésre álló 256 karakterhely – az új, 8 bites kódlapok első felét (0–127) továbbra is egységesen az

ASCII-ban meghatározott karaktereknek tartották fenn – már elégnek bizonyult a legtöbb nyugat-európai nyelv speciális karaktereinek feldolgozására (ANSI/ISO 8859-1:1987 „Latin 1"). Ám a bővítésnek ebből az első köréből kimaradtak a kelet-európai nyelvek. Központi megoldás híján az egyéni útkeresés ideje jött el, így fejlesztették ki a speciálisan a magyar felhasználók igényeit szem előtt tartó ún. CWI kódtáblát. Ez azonban nem találkozott a nagy számítástechnikai cégek érdekeivel, amelyek érthető módon egy átfogó kelet-európai kódtáblában gondolkodtak. Az első változat, amelyet az IBM dolgozott ki, operációs rendszerében pedig a Microsoft is alkalmazott, 852-es kódlap néven vált ismertté, s részévé lett a magyar szabványnak is (Codepage 852 [Eastern Europe][36]; MSZ 7795-3:1992 / ASCII/PC).

Ezzel azonban még nem ért véget a magyar felhasználók kálváriája: grafikus operációs rendszerében a Microsoft áttért az ISO által is elfogadott, Latin 2-ként emlegetett kódtáblára (ISO 8859-2:1987[37]), amely lehetővé teszi az albán, cseh, angol, finn, horvát, ír (gael), lengyel, magyar, német, román, szlovák, szlovén és szoráb nyelvek karaktereinek egyidejű használatát. Természetesen ezt sem lehetett figyelmen kívül hagyni a magyar szabvány meghatározásakor (MSZ 7795-3:1992/ASCII), amelyben egy harmadik kódtábla is helyet kapott: a 'nagygépeknél' használatos EBCDIC[38] (Extended Binary-Coded Decimal Interchange Code – MSZ 7795-3:1992/EBCDIC).

A dolog egyetlen szépséghibája az maradt, hogy a Latin 1 és Latin 2 (illetve a további nyelveket bekapcsoló Latin 3–10) kódtáblák egymás alternatíváiként tudnak csak működni, ami azt jelenti, hogy ugyanahhoz a 8 bites kódhoz az egyikben ilyen, a másikban amolyan karakter rendelődik. Ily módon az eltérő kódtáblák által támogatott nyelvek (pl. a magyar és a francia) elvileg nem használhatók egy szövegen belül (a HTML-dokumentumok esetében pl. ez a mai napig leküzdhetetlen akadályt jelent).

Ennek a problémának a megoldását tűzték ki célul a *Unicode*[39] megálmodói: a karakterek kódolásához immár 16 bitet igénybe vevő kódtáblában a világ összes nyelvének (s nem csak, illetve elsősorban nem a latin betűs írásrendszerűeknek) összes karakterét szeretnék elhelyezni. A 2.0-s változatában nemzetközi szabványként is elfogadott kódtáblában (voltaképpen az ISO 10646-1:1993 'első fele': UCS-2 [Universal Character Set]; a teljes ISO 10646 4 bájtot tart fenn: UCS-4) 65 536 kódhely áll rendelkezésre, amelyek közül jelen pillanatban kb. 39 000-et definiáltak, 18 000-et későbbi használatra lefoglaltak, s 6000-et bocsátottak az egyes felhasználók privát használatára. De még ez utóbbi, hivatalosan szabad területnek szánt rész felosztásának, betöltésének koordinálására is született 'civil' kezdeményezés (ConScript Unicode Registry[40]).

A Unicode-ra épül a *Windows 1250-es jelű karakterkészlete*[41], s feltételezhető, hogy hamarosan minden népszerű grafikus operációs rendszerben problémamentesen megoldódik a Unicode karakterek kezelése.

3.3.3 Alternatív megoldások

Eltérő megközelítést képvisel az SGML korábbiakban már bemutatott módszere: a speciális, az ASCII-ban nem található karaktereket „entity"-ként, kizárólag ASCII-jeleket felhasználó „körülírásukkal" határozza meg (pl. á: **á**). Az SGML ilyen értelemben nemcsak platform-, de kódtáblafüggetlen kódolási rendszerként is működik. Az SGML Latin 1 készlete 62 karaktert tartalmaz, az ehhez kiegészítésként kapcsolódó Latin 2 készlet **<!ENTITY % ISOlat2 PUBLIC „ISO 8879-1986//ENTITIES Added Latin 2//EN">** 122 karaktere között megtalálhatók a magyar nyelvéi is. Az SGML-ben tehát szintén nem jelent problémát az ún. Latin 1 és Latin 2 karakterek párhuzamos használata.

Érdekes színfoltot jelent az Internet, ezen belül a World Wide Web rohamos terjedésének köszönhetően előtérbe került HTML nyelv. Mint afféle tisztességes SGML-alkalmazás, első pillantásra ez is megkerülni látszik a kódtáblaproblémát, hiszen a speciális karaktereket szintén a fenti struktúrájú „entity"-ként kódolja – legalábbis látszólag. Ugyanis a HTML-nek a gyakorlati megjeleníthetőség érdekében kompromisszumot kellett kötnie a meglévő technikai lehetőségekkel. Azaz, alkalmazkodnia kellett a 8 bites operációs rendszerekhez, s azok kódtáblaszisztémájához. Ennek megfelelően a HTML-dokumentumok fejlécében – csakúgy, mint az SGML-alapúakban – definiálhatjuk a szöveg kódolásakor használt kódtáblát, ám ennek alapértéke nem más, mint az ISO 8859-1. Az **á** formula sajnos csupán csalóka felszínnek bizonyul, mely alatt az **á** kód bújik meg, amelyre a HTML könyörtelenül le is fordítja ravasz körülírásunkat.

Az ebből eredő probléma a magyar nyelvű szövegekben – immár ismerős módon – a kis és nagy *ő* és *ű* betűknél jelentkezik. Míg az összes többi magyar karakter megtalálható, ráadásul ugyanazokon a kódhelyeken a Latin 1 készletben, addig ezek olyan helyekre tévedtek (245 és 251, illetve 213 és 219), amelyek a riválisi kódtáblában már mások (a portugál hullámos *ő*, illetve a francia kalapos *ű*) számára foglaltak. Ezt fogalmazza meg a magyar szövegeket HTML-ben kódolók azon empirikus tapasztalata, mely szerint „otilde-t kell írni, hogy magyar ő jelenjék meg". Ez azonban már attól függ, hogy olyan betűkészletet állítunk-e be a

felhasználói oldalon, amely az adott kódot a Latin 2-es táblának megfelelően interpretálja. A kódolói, szolgáltatói oldalon viszont mindaddig tisztázatlan állapot uralkodik, amíg meg nem történik az ISO Latin 2 kódtábla fejlécben való definiálása.

A fentiek figyelmen kívül hagyásával, „gondatlanul" használt HTML pedig azzal a veszéllyel jár, hogy a magyar nyelv történetének kései kutatói néhány évszázad távolából visszatekintve arra a megállapításra juthatnak majd, hogy a második évezred fordulóján a magyar nyelvben két párhuzamos írásrendszer élt: az egyik az elavult, kézírásos médiumra jellemző hagyományőrző, amely a tradicionális dupla éles ékezetes *ő*-höz és *ű*-höz ragaszkodott, a másik a haladó, a „bedrótozott" beszélők csoportja által támogatott, amely a hullámos *ő*-t és a kalapos *ű*-t részesítette előnyben. Pusztán a kódokból mindenesetre ez lesz kiolvasható.

A magyar nyelvű szövegek rögzítéséhez szükséges karaktereknek a fenti kódtáblák, illetve kódrendszerek szerinti kódjait összefoglaló *táblázatunkból* [42] látható, hogy a szabványok nem feltétlenül határozzák meg az összes rendelkezésre álló kódhely tartalmát; az „üres helyeket" a gyakorlatban alkalmazott karakterkészletek szabadon használhatják (pl. a 8 bites Windows-karakterkészletek bővebbek, mint az alapjukul szolgáló ISO 8859-es szabványok).

3.3.4 Anomáliák

A tökéletes megoldástól azonban még mindig elég távol vagyunk.

A mai magyar nyelvtani rendszer fonetikus alapokon nyugszik. Ennek megfelelően az érvényben lévő helyesírási szabályzat [8] több írásjegyből álló, de *önálló* betűknek tekinti a következőket: *cs, dz, dzs, gy, ly, ny, sz, ty, zs*. Ez a meghatározás nem pusztán formális: szerepet játszik a betűrendbe sorolásnál, vagy ami még fontosabb: az elválasztási szabályoknál. Előzetes teoretikus döntést igényel tehát, hogy a magyar nyelvű szövegek esetében egyszerűen írásjegyeket vagy a magyar ábécé betűit kívánjuk rögzíteni. (Az utóbbi esetben külön kódokat kellene alkalmaznunk a *t*, az *y* és a *ty* jelölésére; az előbbi esetben viszont semmi okunk sincs arra, hogy az *ü* kódjaként ne fogadjuk el az *„u + [umlaut]"* szintetikus kódolási formát.) Ám ezzel még korántsincs vége a bonyodalmaknak: a 12. pont „régi, ma már egyébként nem használatos betűk"-ként határozza meg a következőket: *aá, eé, eö, ew, oó, y, ch, cz, s, th, ts, w*. A sor végén pedig fatális módon ez áll: *stb. (A betűk, 3–13. pont)*.

Meglehetősen bonyolult a magyar nyelv írásjelhasználatának szabályozása is. Számítógépes dokumentumok létrehozásakor gyakorta figyelmen kívül hagyják a magyar *(nyitó) idézőjel*nek az angolszásztól való eltérését, vagy a *kötőjel*, a *nagykötőjel* és a *gondolatjel* közti különbséget. A kétjegyű betűkéhez hasonló problémát jelent a *három pont* esete, amely sajátos funkciójában semmiképpen sem tekinthető három mondatlezáró írásjel együttesének. Az ún. *belső idézőjelet* sem szabad összetévesztenünk a francia idézőjellel, hiszen a nyitó és záró jelek iránya éppen fordított. Meglepő módon a helyesírási szabályzat nem foglalkozik az *aposztróf* kérdésével, pedig ennek is több változata ismeretes *(Az írásjelek, 239–275. pont)*.

A szöveges dokumentumokkal kiemelten foglalkozó nyelv- és irodalomtudomány további érdekes szempontokat vet fel. A magyar nyelv története során nagy változásokon ment keresztül, nemcsak nyelvtanát, szókészletét, hanem helyesírását (illetve írásmódjait) tekintve is. Az igényes szövegrögzítés nem törekedhet e különbségek elfedésére. Éppen ellenkezőleg, a forrás minél tökéletesebb visszaadását kell megcéloznia, ami azonban lehetetlen az eredeti karakterek reprodukálása nélkül [9, 10][43]. Sajátos követelményeket támaszt továbbá a beszélt nyelv tudományos leírása; a magyar nyelvtudomány erre a feladatra saját rendszert fejlesztett ki, az ún. *egyezményes lejegyzést* [44] [11].

A fenti szempontok érvényesítésére természetesen egyetlen nemzetközileg elfogadott szabvány sincs felkészülve. A probléma lehetséges megoldása a Magyar Nyelv Történeti Kódtáblájának (MNyTK) kidolgozása, amelyben saját jogon (azaz külön kóddal) szerepelne minden, a magyar nyelvű szövegek rögzítéséhez szükséges *karakter*.

Az ideális MNyTK-nak tehát – első közelítésben – a következőket kellene tartalmaznia:

- a mai magyar ábécé 44 betűjét,
- a különleges írásjeleket,
- az egyes nyelvtörténeti korok sajátos betűit,
- a magyar nyelvű szövegek tudományos rögzítésében és feldolgozásában használatos speciális jeleket.

Az MNyTK csak abban az esetben lesz jól használható, ha kapcsolódni tud a nemzetközi szabványok valamelyikéhez. A fentiekben csupán a speciális igényeket próbáltuk meg összegyűjteni – magától értetődik, hogy egy magyar nyelven íródó szövegben is szükség lehet más nyelvek betűire; matematikai szimbólumokra; a nemzetközi fonetikai ábécé jeleire stb.

3.3.5 Megoldás: Unicode vagy SGML entity?

A probléma egyik megoldása a Unicode-hoz való csatlakozással képzelhető el: a magyar nyelvű dokumentumok speciális igényeit szem előtt tartó MNyTK-t (pontosabban azon részeit, amelyek

még nem szerepelnek a szabványban) a jelenleg még üres kódhelyeken kellene elhelyezni. Merész próbálkozás volna az MNyTK-nak a hivatalos Unicode táblába való felvételét indítványozni (ahogy a rovásírás esetében *már meg is történt*[45]), de a rendelkezésre álló terület nagyságának ismeretében talán nem reménytelen. Még ennél is kevesebb akadálya van annak, hogy a felhasználók önkényének átengedett szabad részek valamelyikét vegyük igénybe; ebben az esetben is érdemes volna azonban az ezen helyek betöltését koordináló civil kezdeményezéssel egyeztetni.

A másik lehetőség SGML-ben megalkotni mindazokat az „entity"-ket, amelyekre a magyar nyelvű szövegek rögzítéséhez szükségünk van, s ezek összességét magyar szabványként előírni, majd lehetőség szerint a központi SGML-forrásokkal is elismertetni.

Jelen pillanatban mindkét megoldás hosszú távon is kielégítőnek ígérkezik. Az SGML mellett szól a könnyen és platformfüggetlenül kezelhető ASCII-kódolás, másfelől azonban nagyobb mennyiségű szöveg esetében nem mellékes szempont, hogy míg a Unicode az *á* karakter tárolásához 2 bájtra tart igényt, addig az SGML 8 bájton (**á**) képes megtenni ugyanezt.

4. Nem szöveges objektumok

4.1 Nem karakteres minták: kotta, térkép

4.1.1 Kotta

A kotta a digitális átírás szempontjából leginkább olyan szöveghez hasonlítható, amelyben az egymást követő karakterek sora mellett még kiegészítő információt hordozó mellékjeleket is találunk. A hatvanas évek óta számos kísérlet történt a kétdimenziós kottaírás egydimenziós, azaz karakteres megjelenítésére, azonban egyetlen rendszer sem érte el a szabvánnyá válást. A kottadigitalizálással kapcsolatos kutatások áttekintéséhez a legjobb kiindulópont talán az Európai Unió bécsi központú *Harmonica*[46] projektje, amelynek célja, hogy feldolgozza és bemutassa az ezen a téren elért eddigi nemzetközi eredményeket. Az alábbiakban bemutatjuk a főbb metanyelvajánlásokat, illetve -kísérleteket – részben a Harmonica projekt alapján.

ZIPI Music Parameter Description Language (MPDL)[47]

A *Computer Music Journal*ban 1994-ben közzétett leíró nyelv. A hangjegyeket kiegészítő zenei paraméterek leírására szolgál. A legáltalánosabban elfogadott paramétereket definiálja csak, és nyitva hagyja a lehetőséget továbbiak leírására.

Music Notation Interchange File Format (NIFF)[48]

1995-ben készült, grafikus információt minimális mértékben használó, sokoldalú, flexibilis leíró formátum.

Unicode javaslat (ISO/IEC 10646)[49]

Javaslat zenei anyagok, kották karakteres kódolására a Unicode még üres kódhelyein. A 220 elemből álló kódkészlet a nyugat-európai zene kottajelölésének teljes körű leírását célozza, alapja a *Common Music Notation*, amelyet számos forrásból kiegészítettek.

Thesaurus Musicarum Italicarum[50]

Az Utrechti Egyetem kutatási projektjének egyik célja, hogy SGML-alapú kottaátíró rendszert készítsen, vagyis kidolgozza a kotta dokumentumtípus DTD-jét. Kiindulási alapul a TEI-ajánlást használják, amelyben új, kiegészítő elemkészletet definiálnak az 1961–75 között kidolgozott DARMS rendszer alapján.

4.1.2 Térkép

Napjainkban zajlik a térképészet digitális forradalma. Számos térképészeti és térinformatikai kutatás foglalkozik térkép-digitalizálással, illetve digitális térképek különféle célú felhasználásával. Ezek áttekintéséhez kiváló kiindulópont az Eötvös Loránd Tudományegyetem *térképtudományi tanszékének honlapja*[51]. Digitális könyvtárak is foglalkoznak térinformatikai szolgáltatással, így például a *Berkeley Digitális Könyvtár* [52] vagy az *Alexandria Projekt*[53]

A térképek azonban napi használati tárgyak, amelyek a felhasználó számára értéküket vesztik abban a pillanatban, amint már nem felelnek meg az ábrázolt valóságnak. Így e térinformatikai kutatások célja is elsősorban a naprakész, sokoldalúan használható térképek előállítása, és a muzeális, illetve történeti értékű régebbi, tartalmában elavult térképek digitalizálásának kérdése háttérbe szorul. Mivel ezek jelkészlete rendkívül nehezen vagy egyáltalán nem írható le szabványos metaadatokkal, egyelőre az egyedüli megoldásnak a nagyfelbontású szkennelés tűnik. Ez természetesen megnehezíti a weben keresztüli hozzáférést.

Muzeális értékű térképek digitális publikálására úttörő hazai példa *Szántai Lajos: Atlas Hungaricus, Magyarország nyomtatott térképei (1528–1850)* c. kétkötetes könyve (Akadémiai Kiadó, 1996), melynek CD-ROM melléklete 45 térképet tartalmaz kétféle minőségben: 640x480 pixeles, 8 bit/pixel színmélységű tömörített GIF változatban, illetve

2000x1400-tól 4000x3000 pixelig terjedő méretben, 24 bit/pixel színmélységben, tömörítetlen TIFF fájlokban. E kétféle tömörítés lehetővé teszi, hogy az olvasó a pillanatnyi igényei és technikai lehetőségei szerinti változatot használja.

4.2 Belső minta nélküli objektumok: képek

A képek és fakszimilék digitalizálását a washingtoni Kongresszusi Könyvtárban működő nemzeti digitális könyvtár gyakorlata alapján mutatjuk be, ez ugyanis jól tükrözi a lehetséges és elterjedt megoldásokat, ugyanakkor *megfelelően van dokumentálva*[54] is.

A projekt elsősorban szolgáltató könyvtár, nem múzeumi, illetve archiváló jellegű. Ezzel együtt minden dokumentumot a technika engedte legnagyobb hűséggel rögzítenek, az olvasók számára szolgáltatott változat minőségét pedig az adatátviteli lehetőségekhez, illetve a felhasználói igényekhez igazítják. A szolgáltatás elsősorban WWW-alapú, de a későbbi, nagyobb igényű technikai lehetőségek előtt is nyitva hagyják az utat a TEI formátumú szövegek, és az archiválási célokra készített nagyfelbontású digitális képek.

4.2.1 Képgyűjtemények

A Kongresszusi Könyvtár háromféle minőségben digitalizál minden képet:

- *Bélyegkép* – a bibliográfiai leíráshoz kapcsolódik, és lehetővé teszi, hogy a felhasználó eldöntse, szüksége van-e a nagyobb változatra. Adatok: 8 bit/pixel színmélység, GIF formátum a saját tömörítés eljárásával, 200x200 pixeles méret.
- *Referensz* – a ténylegesen használt változat. Adatok: 24 bit/pixel színmélység, JPEG formátum 10:1-es tömörítési aránnyal, különféle méretben 500x400 pixeltől 1000x700 pixelig. Tervezik nagyobb felbontás szolgáltatását is: 2000x1400-tól 4000x3000 pixelig.
- *Archivált példány* – tömörítés nélkül, illetve a jövőben minőségvesztés nélküli tömörítéssel rögzítik, elsősorban reprodukciók és későbbi esetleges újratömörítés céljára. Ehhez a változathoz jelenleg a felhasználók nem férhetnek hozzá. Adatok: 24 bit/pixel színmélység, TIFF formátum tömörítés nélkül. A felbontás változó, jelenleg 500x400-tól 1000x700 pixelig, később esetleg nagyobb felbontásban. Csak a legnagyobb felbontású változatot tárolják.

Fontos megjegyezni, hogy a TIFF formátum lehetővé teszi a digitalizált képpel együtt szöveges leíró információk rögzítését is. Ennek felhasználását, illetve az így készíthető képi adatbázis lehetőségét elemzi *Manfred Thaller* [12], a Kongresszusi Könyvtár pedig *részletes kódolási utasítást*[55] dolgozott ki az 5.0 verziójú TIFF képekhez.

4.2.2 Fakszimilék

A projekt kísérleteket folytat fakszimile és kereshető szövegfájl együttes rögzítésére. A szövegrögzítés alapvetően a TEI-ajánlás szerinti SGML-metakódokat használja, de a fejléc (*header*) adatai minimális mértékben vannak csak kitöltve, mivel a könyvtár önálló bibliográfiai adatbázist vezet. A WWW-n szolgáltatott szövegeket butítják, vagyis az interpretáló jellegű TEI kódokat megjelenítésorientált HTML kódokká alakítják. A felhasználó hozzáférhet az SGML-változathoz is. A szövegek rögzítésénél 0,05 százalék a megengedett hibaarány.

A digitális fakszimiléket többnyire tónusos képként rögzítik, bár tapasztalataik azt mutatják, hogy nyomtatványok és vonalas rajzok esetében a kétszínű bitmap jobban használható. Mivel az elv az, hogy a lehető legtöbb információt kell rögzíteni, ilyenkor is elkészítik maximális felbontással a tónusos változatot, noha az olvasó csak a kétszínű bitmapet használja.

Mivel a digitális fakszimile egy szövegfájlhoz tartozik, itt nem készítenek bélyegképet, csak referensz és archív változatot:

- *Referensz* változat – színmélysége fekete-fehér másolatnál 8 bit/pixel, színesnél 24 bit/pixel, tömörített JPEG formátum, 150 dpi felbontással.
- *Archivált példány, tömörítés nélkül* – színmélysége azonos a referenszpéldányéval, tömörítetlen TIFF formátumban, 300 dpi felbontással.
- *Archivált példány, tömörítéssel* – kísérleti változat, melynek színmélysége és felbontása azonos a tömörítés nélküli változattal, azonban a fájl tömörített JPEG formátumú. A projekt irányítói közül egyesek úgy vélik, a fakszimile esetében elegendő az olvashatóság megőrzése, vagyis a JPEG-tömörítésből eredő minőségromlás még a megengedhető határon belül van.

A projekt nem foglalkozik Adobe PDF formátumú fakszimilék készítésével, de nem zárja ki az elméleti lehetőséget, hogy más digitális könyvtárak ezt a technológiát használják fakszimilék készítésére. Ugyanakkor kísérleteket folytat vonalas ábrák és nyomtatványok CCITT, vagyis a faxgépek által használt formátumú tömörítésére, TIFF formátumban, 300 dpi felbontással. Ugyancsak folynak kísérletek a nyomtatott tónusos fotók raszterszerkezete és a szkenner közötti interferenciából adódó zavaró hullámkörök kiszűrésére, azonban a minden szempontból megnyugtató megoldás még nem ismert.

A digitális fakszimile és a szövegfájl egymáshoz rendelésére a projekt a Berkeley Egyetemen kifejlesztett *Ebind szoftvert* [56] használja, amely lehetővé teszi a TEI formátumból HTML-re konvertált fájlokba GIF formátumú bélyegképek beillesztését.

A mikrofilmen tárolt anyagok digitalizálása természetesen további kérdéseket is felvet. A washingtoni kísérletek során a negatív film szkennelése bizonyult a legcélravezetőbbnek, ugyanis ezen látszanak legkevésbé a fizikai sérülések és a por. A képeket JPEG fájlokban rögzítették, 8 bit/pixel színmélységben. A digitalizálás során az olyan mikrofilmkockákat, amelyeken egy könyv két oldala volt látható, oldalanként külön-külön rögzítették, azonban a kéziratlapokat minden esetben egyetlen képen.

4.3 Időalapú dokumentumok: hang és mozgókép

Jelenleg e két médium digitalizálása még egyáltalában nem tűnik véglegesen megoldottnak, a washingtoni projekt fel van készülve arra, hogy a tömörítési technikák fejlődésével esetleg újra kell digitalizálni az eddigi anyagokat. Emellett gondot okoz a webalapú szolgáltatás is, hiszen a jobb minőséget adó, de nagyobb helyi tárkapacitást és átviteli sebességet igénylő letölthető fájlok, illetve a gyengébb minőségű, de kényelmesebben használható *streaming* megoldások között kell megtalálni az arany középutat.

A könyvtár a következő formátumokat használja:

- *Letölthető hangfájl*: Microsoft[57] WAVE formátum, 22,05 kHz mintavétel, 16 bit, mono.
- *Streaming hangfájl*: RealAudio[58] formátum, 14,4-es modemre tömörítve (1997 elején).
- *Mozgóképfájl*: 320x240 pixel képméret, 30 kép/s, MPEG-1[59] tömörítés, illetve ezzel azonos minőségű Quicktime[60] fájl. Az átlagos fájlméret 9 Mbájt/perc. A washingtoni projekt jelenleg még nem kínál streaming videoformátumot.

A fenti formátumok egyike sem megfelelő az eredeti, analóg médiumot kiváltó archiválásra. Ennek kérdéseivel foglalkozik a már említett Harmonica[61] projekten kívül az Ausztrál Nemzeti Könyvtár[62] és az amerikai Kongresszusi Könyvtár[63] egy testülete is.

Végül megjegyezzük, hogy előkészítés alatt áll az *MPEG-7 szabvány* [64], amely – a TIFF formátumhoz hasonlóan – lehetővé teszi kereshető, szöveges leíró információ rögzítését az audio-video adattal fizikailag azonos fájlban. A nemzetközi szabvány első vázlata 1999 végére, a szabvány elfogadása 2001-re várható.

5. Javaslat helyett

A számítástechnika a leggyorsabban változó területek közé tartozik, ez aligha szorul bizonyításra. Nemcsak a folyamatosan cserélődő hardverek és szoftverek világa tűnik kaotikusnak, de a szabványok, szabványjavaslatok és kváziszabványok is egymással rivalizálnak – miközben folyamatosan változnak, fejlődnek.

A digitális rögzítés egyedül üdvözítő módszereinek kijelölése kétségkívül lehetetlen feladat. A követendő eljárás olyan technikák választása, amelyeknél biztosítottnak látszik az esetleg szükségessé váló konvertálások elvégezhetősége. Ennek legfontosabb feltétele a kódolási rendszer lehető legteljesebb mértékű átláthatósága, és az információvesztéstől mentes kódolás.

A digitális könyvtárnak tehát nemcsak beszerzési, de rögzítési, illetve archiválási politikával is rendelkeznie kell. Az ezekre vonatkozó irányelveket pedig nem szabad egyszer s mindenkorra eldöntötteknek tekinteni, hanem – adott esetben egy speciálisan erre a feladatra létrehozandó munkacsoport keretei között – folyamatosan ellenőrizni, újragondolni, s az átalakuló technikai lehetőségeknek, illetve olvasói elvárásoknak megfelelően frissíteni kell.

Irodalom

[1] HORVÁTH I.: Pour une histoire nouvelle de la littérature hongroise. Előadás 1996. szeptember 12-én a IV. Nemzetközi Hungarológiai Kongresszus nápolyi ülésszakán.

[2] PAPP T.: Disztichon alfa (Első magyar versgenerátor).

[3] HORVÁTH I.: Szöveg. = 2000, 1994. november, p. 42–53. (További elérési utak: Internet Expo Magyar Pavilon, Oktogon megálló[65], MEK Társadalomtudományi olvasó[66])

[4] DÁVIDHÁZI P.: A hatalom szétosztása: (poszt)modernizáció a szövegkritikában. = Helikon, 3–4. sz. 1989. p. 328–343.

[5] HORVÁTH I.–H. HUBERT G.–FONT Zs.–HERNER J.–SZŐNYI E.–VADAI I.–RUTTNER T.–GÁL Gy.: Répertoire de la poésie hongroise ancienne. Paris: Nouvel Objet, 1992.

[6] GÁL Gy.: A 'Répertoire de la Poésie Hongroise Ancienne' adatmodellje. = Irodalomtörténeti Közlemények, 3. sz. 1989. p. 267–272.

[7] LOTMAN, J. M.: Szöveg – modell – típus. (Szerk.: Hoppál Mihály) Budapest, 1973.

[8] A magyar helyesírás szabályai. Budapest, Akadémiai Kiadó, 1984. A betűk (3–13. pont), Az írásjelek (239–275. pont).

[9] VARJAS B.: Paleográfiai útmutató 15–17. századi magyar nyelvű kéziratok olvasásához. Budapest, ELTE könyvtártudományi tanszék, OSZK – KMK, 1982.

[10] V. ECSEDY J.: A régi, magyar nyelvű nyomtatványok betűkarakterei (1533–1800). MKSz, 1986.
[11] R. MOLNÁR E.: Leíró magyar hangtan. Kézirat, Budapest, Tankönyvkiadó, 1990.
[12] THALLER, M.: The archive on the top of your desk? On self-documenting image files. = Fikfak, J.–Jaritz, G. (szerk.): Image processing in history: towards open systems. St. Katharinen, 1993. (Halbgraue Reiche zur Historischen Fachinformatik Band A16, p. 21–44.)
[13] GABLER, H. W.: A kiadói szöveg születése: a számítógép bába-szerepben. Helikon, 1989. p. 3–4, p. 425–426.
[14] HÁRTÓ G.: A grafikai mozzanat a szövegben. = Literatúra, 2. köt. 2. sz. 1995. p. 204–212.
[15] HORVÁTH I.: Számítógépes költészet magyarul[67].
[16] HORVÁTH I.: Bölcsészet a bábeli könyvtárban. = 2000, 1997. május, p. 61–63.
[17] PAJZS J.: Számítógép és lexikográfia. Budapest, MTA Nyelvtudományi Intézete, 1990.
[18] PAPP T.: Múzsával vagy múzsa nélkül? (Irodalom számítógépen.) Budapest, Balassi Kiadó, 1992.
[19] STOLL B.: Szövegkritikai problémák a magyar irodalomban. Budapest, 1987.
[20] SZÖRÉNYI L.: Ars mutilandi Hungarica, azaz a csonkítás mestersége magyar módra. = Gondolatjel, 1984. 2. sz. p. 14–22; 3. sz. p. 20–21; 4. sz. p. 18–20; 6. sz. p. 16–24.
[21] SZÖRÉNYI L.: Szöveggondozás – magyar módra (Delfinológiai vázlat). = SZÖRÉNYI L.: Múltaddal valamit kezdeni. Budapest, Magvető Kiadó, 1989. p. 250–279.
[22] TURI L.: Számítógép az irodalomtudományban (szakdolgozat). Eötvös Loránd Tudományegyetem Tanárképző Főiskolai Kar, 1992. MEK Társadalomtudományi olvasó[68].

Elektronikus hivatkozások

[1] *http://www.dlib.org/dlib/July95/07arms.html*
[2] *http://sunsite.berkeley.edu/*
[3] *http://sunsite.berkeley.edu/Info/standards.html*
[4] *http://lcweb.loc.gov/marc/*
[5] *http://www.loc.gov/marc/marcdtd/marcdtdback.html*
[6] *http://sunsite.berkeley.edu/Z39.50/*
[7] *http://sunsite.berkeley.edu/SICI/version2.html*
[8] *http://www.acl.lanl.gov/URC/*
[9] *http://sunsite.berkeley.edu/FindingAids/*
[10] *http://www.doi.org*
[11] *http://purl.org/metadata/dublin_core*
[12] *http://iconclass.let.ruu.nl/*
[13] *http://candl.let.ruu.nl/Research/marburg/descript.htm*
[14] *http://www.idg.hu/expo/oktogon/balassa/hiszov.htm*
[15] *http://www.mek.iif.hu/porta/bbs/golden.txt*
[16] *http://www.mek.iif.hu/porta/bbs/golden2.txt*
[17] *http://www.promo.net/pg/*
[18] *http://www.sil.org/sgml/sgml.html*
[19] *http://www.w3.org*
[20] *http://www.uic.edu/orgs/tei/*
[21] *ftp://ota.ox.ac.uk/pub/ota/TEI/dtd/teilite.dtd*
[22] *http://sunsite.berkeley.edu/MLA/guidelines.html*
[23] *http://www.sil.org/sgml/acadapps.html*
[24] *http://www.lpl.univ-aix.fr/projects/multext/*
[25] *http://nl.ijs.si/ME*
[26] *http://www.loria.fr/~romary/Telri/*
[27] *http://info.ox.ac.uk/ctitext/publish/comtxt/ct15/burnard.html*
[28] *http://www.hti.umich.edu/*
[29] *http://etext.virginia.edu/*
[30] *http://www.softquad.com*
[31] *http://www.w3.org/XML/*
[32] *http://www.microsoft.com/standards/xml/xmlparse.htm*
[33] *http://www.microsoft.com/truetype/unicode/cscp.htm*
[34] *http://web.ansi.org*
[35] *http://www.iso.ch*
[36] *http://AS400BKS.rochester.ibm.com:80/cgi-bin/bookmgr/BookMgr.cmd/BOOKS/QB3AQ500/F.28*
[37] *http://sizif.mf.uni-lj.si/linux/cee/charset.html*
[38] *http://AS400BKS.rochester.ibm.com:80/cgi-bin/bookmgr/BookMgr.cmd/BOOKS/QB3AQ500/F.40*
[39] *http://www.unicode.org/*
[40] *http://www.indigo.ie/egt/standards/csur/*
[41] *http://www.microsoft.com/typography/unicode/1250.htm*
[42] *http://www.neumann-haz.hu/tanulmany/*
[43] *makarkod.htm*
[44] *http://www.neumann-haz.hu/tanulmany/tortenet.htm*
[45] *http://www.neumann-haz.hu/tanulmany/fonetika.htm*
[46] *http://www.dkuug.dk/JTC1/SC2/WG2/docs/n1686/n1686.htm*
[47] *http://www.kfs.oeaw.ac.at/harm/home.html*
[48] *http://cnmat.CNMAT.Berkeley.EDU/ZIPI/mpdl.html*
[49] *http://www.motu.com/pages/NIFF.net.html*
[50] *http://www.lib.virginia.edu/dmmc/Music/UnicodeMusic/*
[51] *http://candl.let.ruu.nl/*
[52] *http://lazarus.elte.hu*
[53] *http://elib.cs.berkeley.edu/*
[54] *http://alexandria.sdc.ucsb.edu/*
[55] *http://lcweb2.loc.gov/ammem/formats.html*
[56] *http://lcweb2.loc.gov/ammem/formats.html*
[57] *http://sunsite.berkeley.edu/Ebind*
[58] *http://www.microsoft.com*
[59] *http://www.real.com*
[60] *http://www.mpeg.org*
[61] *http://www.apple.com/quicktime/*
[62] *http://www.svb.nl/project/harmonica/harm_deliv.htm*
[63] *http://www.nla.gov.au/niac/meetings/tech.html*
[64] *http://lcweb.loc.gov/film/*
[65] *http://drogo.cselt.stet.it/mpeg/*
[66] *http://www.idg.hu/expo/oktogon/balassa/hiszov.htm*
[67] *gopher://gopher.mek.iif.hu:70/hh/porta/szint/tarsad/irodtud/szoveg/*
[68] *http://www.idg.hu/internetto/babel/szkolt.htm gopher://gopher.mek.iif.hu:70/hh/porta/szint/tarsad/irodtud/turi1/*

Beérkezett: 1998. V. 22-én.