

Az e-könyvek titkai, avagy amit a LIT-ről tudni érdemes*

„Akkor látjuk igazán, hogy mekkora technológiai csúcsteljesítmény egy könyv, amikor felül akarjuk múlni.”¹

A korunk mindennapjait „behálózó” online kommunikáció és az egyre fontosabbá váló digitalizáció világában az elektronikus könyv (e-könyv, e-book) egyértelműen korreprezentáns jelenségnek tekinthető. Az e-könyv mint fogalom és mint elnevezés is köztes – szinte mindenki mást ért rajta –, az viszont biztos, hogy a dokumentumok feldolgozási módszereinek kavalkádjában egy újabb megjelenítési formátumra bukkanhatunk benne. Számos – elsősorban angol nyelvű – cikk és tanulmány foglalkozott már az elektronikus könyvek témakörével, ám ezek inkább a formai jellemzőire, lehetőségeire és esetleges buktatóira helyezték a hangsúlyt. Technikai oldalról – ráadásul magyar nyelven – mindeddig egyetlen írás sem közelítette meg a területet, s talán éppen emiatt gyakorta felmerülnek a kérdések: Hogy használhatom? Mit tud? Hogyan készül?

Az e-könyv nehezen meghatározható „jelenség”

Az e-bookot nagyon nehéz definiálni, ezért itt és most nem is próbálkozom meg vele. A meghatározás azért problematikus, mert noha mostanra már viszonylag jól körülhatárolhatóak az ide sorolható formátumok, még mindig előfordul, hogy a szót következtetetlenül, vegyesen használják a források. Az interneten sok helyen bármilyen digitalizált szöveget e-booknak neveznek, csak legyen eleje meg vége, ráadásul egyesek a hiperszövegben (X)HTML írt regényeket, rövidebb szövegeket is e-booknak nevezik.

Mindezt tetézi, hogy manapság az e-könyvnek még a köznapi fogalma is kettős. Egyfelől speciális célszámítógépet, illetve hardvert jelent, a rajta megjeleníthető szöveggel együtt, amely vizuálisan és kezelhetősége szempontjából könyv benyomását kelti (ez a formája nem túl elterjedt, különösen nem Magyarországon, hiszen amellett, hogy drága, gyakorlatilag folyamatosan fejlesztés, de legalábbis elterjedés előtt áll – persze nem hazánkban). Másfelől pedig, általánosabb értelemben az interneten, illetve számítógépen olvasható hosszabb, egységes és szerkesztett, összefüggő szöveg, amelynél az oldalnagyság körülbelül akkora, mint egy igazi könyvnél, a szöveg elrendezése is hasonló, s tulajdonképpen még lapozgathatunk is benne.

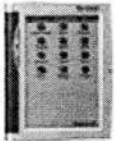
Eszközök, formátumok

Az előbb elmondottak alapján több célgépet/olvasóberendezést, illetve ezekhez kapcsolódóan több formátumot is megkülönböztethetünk. Nézzünk néhányat a teljesség igénye nélkül (1. táblázat).

Jól látszik, hogy bizonyos olvasószoftverek, formátumok nagyon eszközspezifikusak, ezért mi a *Neumann János Digitális Könyvtár és Multimédia Központban*² (továbbiakban *Neumann-ház*) – de természetesen a megközelítés igaz a MEK-re, vagy a HP e-book tárára is³ – igyekeztünk olyan e-book formátumokat választani/előállítani, amelyek a legelterjedtebb operációs rendszereken és olvasóberendezéseken is használhatóak. Ez a két megjelenítési formátum a *PDF* és a *LIT*. Míg az előbbi – noha csak félig sorolható az e-könyv kategóriába, ugyanis inkább egy nyomtatásra szánt, nyomdakész, általános dokumentumforgalmat megkönnyítő dokumentumformátum – nagyon jól használható Windows, Linux, OS/2 Warp és Mac Os környezetben, addig az utóbbi kizárólag Windows operációs rendszeren futtatható. (Az említett formátumban készített e-könyvek egyaránt olvashatók PC-n, notebookon és PDA-n.)

* A cikk anyaga részben elhangzott előadás formájában az MKE 2004. július 29–31. között Miskolcon megrendezett 36. vándorgyűlésén.

1. táblázat
E-book eszközök és formátumaik⁴

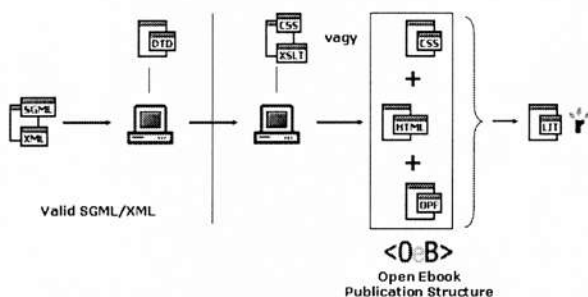
Az eszköz neve	Mintakép	eBook formátum
Franklin eBookMan		Mobipocket
Gemstar eBook		Gemstar eBook
Handheld PC		Palm Reader Microsoft Reader (LIT) Mobipocket
Handspring Visor		Palm Reader Mobipocket
hiebook		hiebook
Palm		Palm Reader Mobipocket
Pocket PC		Microsoft Reader (LIT) Mobipocket

A LIT-készítés technológiai praktikumai

A TMT rendszeres olvasói, vagy akik kicsit tájékozottabbak a könyvtári világban, biztosan tudják, hogy a Neumann-ház alapítása óta az SGML/XML⁵ technológiát használja tárolásra, és minden megjelenítési formátumot a későbbiekben lehetőség szerint ezekből állít elő. 2002 végéig ez kizárólag a HTML volt, hiszen addig csak így kínáltuk olvasásra a műveket, azóta viszont számos Adobe Readerrel⁶ olvasható PDF és Microsoft

Readerrel⁷ olvasható LIT látott napvilágot honlapunkon.

Mivel jelen cikk a szó szoros értelmében vett e-könyv formátummal – LIT – és annak technológiai kérdéseivel kíván foglalkozni, a következőkben a PDF-készítés ismertetésétől eltekintünk, és a Microsoft formátumára helyezzük a hangsúlyt. Hogy átlátható és érthető legyen, amiről beszélünk, az 1. ábrán a Neumann könyvtárban is alkalmazott e-book „gyártási” folyamatot mutatjuk be, amit a továbbiakban példákkal illusztrálva fogunk ismertetni.



1. ábra Egy lehetséges e-könyv készítési folyamat OPF segítségével

Noha a képen nincs jelölve, a kiindulópont mindkét esetben az RTF állomány – abból történik a tárolási formátumba való kódolás a TEI P4 DTD-je alapján. Ha elkészültek a teljes művet tartalmazó SGML/XML fájlok, és megfelelnek a DTD-ben meghatározott szerkezetnek, akkor érvényesnek (valid) tekinthetők, és indulhat a további munka.

OEB-érvényes XSL(T) és CSS⁸

Ábránkon továbbhaladva láthatjuk, hogy szükségünk lesz egy transzformációra képes ún. XSL-re (eXtensible Stylesheet Language – bővíthető stíluslapnyelv), melynek segítségével SGML/XML-ből előállítjuk azokat a(z) (X)HTML állományokat, amelyek már az olvasható művet tartalmazzák, adott esetben jegyzetekkel, ábrákkal és képekkel. Az említett *stylesheet* itt azonban csak a transzformációt végzi el, a kimeneti állományok külső megjelenéséért – betűméret, -típus, kiemelések stb. – nem felel; persze ez is a programozástól függ. Éppen ezért készítenünk kell olyan ún. CSS (Cascading StyleSheets – lépcsős stíluslap) utasításokat is, melyeknek a kívánt forma ábrázolása a feladatuk. Ezek elhelyezése – ahogy az 1. ábrán is látszik – opcionális, tehát lehetnek az XSL-ben, de külön fájlban csatolhatjuk később a(z) (X)HTML-hez is őket.⁹

Bármelyik utat is választjuk, a legfontosabb teendő az, hogy a programozás során implementáljuk a Nyitott eBook Specifikációt (Open eBook Specification).¹⁰ Erre egyrészt azért van szükség, mert a technológiáról szóló minden lényeges információ itt található meg, másrészt az OEB DTD-je olyan tiltásokat fogalmaz meg a HTML, XHTML és CSS nyelvek elemkészletével kapcsolatban, amelyeket nem célszerű figyelmen kívül hagyni. Ha mindent megfelelően készítettünk el, akkor olyan, böngészőben is megtekinthető (X)HTML állomány(oka)t kapunk, mely(ek) teljesen OEB érvényes(ek).

Példaként nézzük meg az „Open eBook Publication Structure 1.2”-es kiadásának (X)HTML kódrészletét (1. forráskód), amely külső CSS-re való meghívást tartalmaz. A továbbiakban célunk, hogy ebből az állományból állítsunk elő LIT formátumot.

```
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC
    "+//ISBN 0-9673008-1-9//DTD OEB 1.2
    Document//EN"
    "http://openebook.org/dtds/oeb-1.2/
    oebdoc12.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
xml:lang="en-US" lang="en-US">
  <head>
    <title>Open eBook Publication Structure 1.2</title>
    <meta content="text/x-oeb1-document" />
    <link rel="stylesheet" href="oeb12-oebps.css"
type="text/x-oeb1-css" />
  </head>
  <body>
    <h1 class="title">Open eBook&trade;
      <br />
      Publication Structure 1.2
    </h1>
    <h3 class="subtitle">RECOMMENDED
    SPECIFICATION August 27, 2002</h3>
    <p class="halftitledate">August 27, 2002</p>
    <p class="Copyright">Copyright &copy; 2002 by
    Open eBook Forum&trade;. </p>
    ...
  </body>
</html>
```

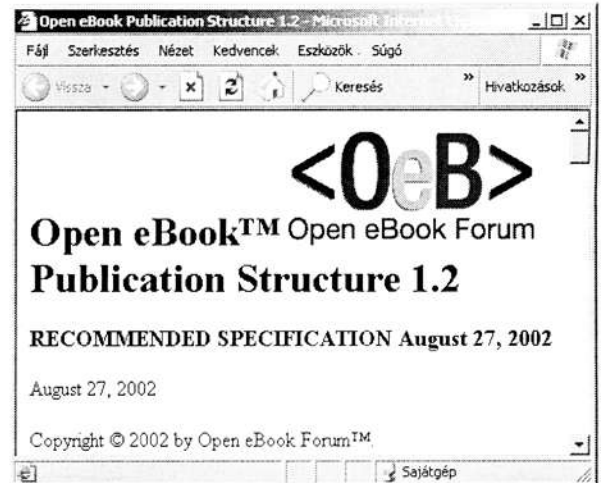
1. forráskód Az oeb12-oebps.html XHTML kódrészlete

Feltételezve, hogy a(z) (X)HTML-t már jól ismeri a kedves olvasó, nem térünk ki a kód minden részletére. Ennek ellenére láthatjuk, az XML vezérlési utasítást itt az OEB DOCTYPE-ja követi, meghatá-

rozandó az alkalmazott DTD-t. Az is egyértelmű, hogy XHTML-ben szintén a <html> </html> elemek veszik közre a dokumentumot, jelezve a böngésző számára a feldolgozandó állomány típusát. Szembetűnő azonban, hogy az említett címkék tartalma jelentősen kibővült, mégpedig az oldallal kapcsolatos vezérlőinformációkkal.

```
<html xmlns="http://www.w3.org/1999/xhtml"
xml:lang="en-US" lang="en-US">
```

A dokumentum gyökéreleme az xmlns attribútum használatával jelöli az XHTML névtérket – ezek az értéként megadott címen¹¹ találhatók meg. Jelen példa esetében a nyelv 1.0-s névtérét alkalmaztuk. Dokumentumunk nyelvhasználatát szintén a <html> címkében kell definiálni. Egy elem nyelvének megadásakor mind a lang, mind az xml:lang attribútumok használandók, ám nem hagyható figyelmen kívül, hogy az utóbbi értéke bármely más nyelvi deklarációval szemben elsőbbséget élvez – megadásának fontosságára a későbbiekben derül majd fény. Ha az 1-es forráskódot fájlba mentjük, és az állományt megnyitjuk valamelyik általunk használt böngészőben, a 2. ábrán látható eredményt kapjuk.



2. ábra OEB specifikáció szerint készített XHTML, CSS-szel formázva

Elkészült tehát a formáért felelős lépcsős stíluslap és megjelenítési formátum, amelyből a LIT-et konvertáljuk. De vajon mi a következő lépés? Hogyan tovább?

Az OEB csomag – OPF

Első lépésben egy újabb állományt kell készítenünk OPF (Open eBook Package File) kiterjesz-

téssel. Lényegét tekintve ez a fájl tartalmaz minden információt a készülő e-könyvről – dokumentum-összetevők, képfájlok stb. Hétköznapi példával élve olyasmiről van szó, mint a szakácsoknál a recept, csak ez meghatározott XML szerkezetet követ. A szóban forgó struktúra a következő:

Package – az OPF gyökereleme, egyedi azonosítóval rendelkezik. /Kötelező elem./

- **Metadata** – a dokumentum metaadatainak helye (title, author, publisher stb.). /ReaderWorks használata esetén elhagyható elem, egyébként kötelező./
- **Manifest** – az e-könyvet felépítő állományok (dokumentumok, képek stb.) listája, egyedi azonosítóval és médiatípus-meghatározással.
- **Spine** – az ebookot alkotó állományok lineáris sorrendjének meghatározására szolgál. /Kötelező elem./
- **Tours** – az e-könyv tartalmi „bejárása”, több <tour>-t tartalmazhat, de általában megegyezik a tartalomjegyzékkel. /Elhagyható elem./
- **Guide** – strukturális elemek megadása (tartalomjegyzék, előszó, bibliográfia stb.). /Elhagyható elem./

Minden OPF állománynak az OEB package DTD-nek megfelelően érvényes XML dokumentumnak kell lennie. Az OEB csomag vázát ennek alapján a 2. forráskód mutatja.

```
<?xml version="1.0"?>
<!DOCTYPE package PUBLIC
    "+//ISBN 0-9673008-1-9//DTD OEB 1.2
Package//EN"
    "http://openebook.org/dtds/oeb-
1.2/oebpkg12.dtd">
<package>
  <metadata />
  <manifest />
  <spine />
  <tours />
  <guide />
</package>
```

2. forráskód OEB package (OPF) „csontváz” a kötelező elemek kiemelésével

Noha az Open eBook Publication Structure 1.2-es specifikációja részletesen taglalja az egyes elemek jelentését és lehetséges tartalmi felépítését, a kötelezőket azonban – úgy gondoljuk – röviden érdemes áttekinteni.

<package />: A *package* az OPF gyökere, minden gyermekelemet ez tartalmaz. Egyetlen egyedi azo-

nosítója lehet – *unique-identifier* –, amely valójában attribútum. Értéke rendszerint megegyezik a *dc:Identifier* elem *id* attribútumának értékével.

```
<package unique-identifier="gmbartokpackage">
```

<metadata />: Az elektronikus könyv legfontosabb metainformációit tartalmazza. Egyik „gyermeke” a *dc-metadata*, amely a jól ismert 15 Dublin Core-elemet foglalhatja magában. Az alap DC-be nem sorolható metaadatok abba az *x-metadata* elembe kerülhetnek, melyek felépítése teljesen analóg az (X)HTML *head* részébe kódolható *meta* elemével. Példánkban *Bartók Béla: A népzeneről* című esszéjét írtuk le, melynek ára 1590 forint.

```
<metadata>
  <dc-metadata
    xmlns:dc="http://purl.org/dc/elements/1.0"
    xmlns:oebpackage="http://openebook.org/
namespaces/oeb-package/1.0">
    <dc:Title>A népzeneről</dc:Title>
    <dc:Creator role="aut">Bartók Béla</dc:Creator>
    <dc:Publisher>Neumann-ház</dc:Publisher>
    <dc:Type>Esszé</dc:Type>
    <dc:Identifier id="gmbartokpackage"
      scheme="ISBN">123456789X</
dc:Identifier>
  </dc-metadata>
  <x-metadata>
    <meta name="price" content="HUF 1590"/>
  </x-metadata>
</metadata>
```

Valószínűleg mindenkinek feltűnt, hogy a leírásban ékezetes karakterek szerepelnek. Eerre elsősorban magyar nyelvű elektronikus könyvek készítésekor lehet szükség. Ha a LIT konverzió során nem akarunk szembesülni karakterkódolási problémákkal, ne feledjük az XML vezérlési utasításban a Latin 2-es kódtáblát definiálni! Ezt könnyedén megtehetjük, ha az OPF első programsorában az egyébként opcionális *encoding* attribútum értékét az alábbiak szerint állítjuk:

```
<?xml version="1.0" encoding="iso-8859-2"?>
```

A beállított érték más ISO kódtáblákat is takarhat, de természetesen alkalmazható az Unicode UTF-8-as és 16-os kódolása is.

<manifest />: A *manifest* elem a készülő elektronikus kötetet alkotó állományok listája. Kötelezően tartalmaznia kell egy vagy több *item* elemet, melyek mindegyike rendelkezik *id*, *href* és *media-type*

attribútumokkal. Az *id* valójában egyedi azonosító, melyre majd a sorrendet meghatározó *spine*, *itemref* eleméből lehet hivatkozni. A *href* az egyes alkotórészek elérési útvonalát, a tulajdonképpeni URI-t definiálja, a *media-type* pedig az adott összetevő MIME¹² típusát határozza meg. A leendő e-könyv alkotóelemei (X)HTML, XML, TXT, JPG, PNG formátumúak lehetnek – ezekből néhányat felvillant példánk is.

```
<manifest>
  <item id="toc" href="gmbartoktoc.xml" media-
type="text/x-oeb1-document" />
  <item id="0000" href="gmbartok0000.html" media-
type="text/x-oeb1-document" />
  <item id="0001" href="gmbartok0001.html" media-
type="text/x-oeb1-document" />
  ...
  <item id="bartokface" href="face.png" media-
type="image/png" />
</manifest>
```

<spine /> A *spine* feladata az e-bookot alkotó állományok elsődleges lineáris olvasási sorrendjének meghatározása. Mindig tartalmaznia kell *itemref* elemet, azon belül pedig *idref* attribútumot. Ez utóbbi értéke megegyezik a *manifest*-ben lévő kapcsolódó *item id*-jével. Hogy érthetőbb legyen, vegyük szemügyre kódrészletünket! A *manifest* elem első *item*-jében elhelyezett XML állomány *id*-jének értéke "toc", ami teljesen megegyezik a *spine* első *itemref* elemében lévő *idref* értékével. Ha nem így lenne, akkor nem létező *id*-re való hivatkozás szerepelne az OPF állományban => konverziót követően a LIT nem tartalmazná a kötet tartalomjegyzékét.

Jól látszik az is, hogy itt már nem szerepel a *face.png*-re való utalás. Ha kicsit belegondolunk, és megértjük a *spine* működését, akkor funkciójából adódóan azonnal egyértelművé válik: a képre való hivatkozásra nincs szükség. Hogy miért? A magyarázat egyszerű: az `<itemref idref="bartokface" />` utasítás elhelyezése azt eredményezné a LIT-ben, hogy Bartók arcképe nemcsak a szövegtörzset kívánt helyén, hanem a dokumentum végén, mint a kötet alkotóeleme – bármely HTML összetevőhöz hasonlóan – külön is megjelenjen.

```
<spine>
  <itemref idref="toc" />
  <itemref idref="0000" />
  <itemref idref="0001" />
  ...
</spine>
```

Az OEB csomag felépítésének kellő mértékű áttekintése után térjünk vissza „Open eBook Publication Structure 1.2” címet viselő XHTML állományunkhoz, és kíséreljük meg elkészíteni hozzá az OPF állományt. Egyszerű dolgunk van, ugyanis mindössze egyetlen weblapot és egy képet kell „lekezelni”. A megoldást a 3. forráskód tartalmazza.

```
<?xml version="1.0"?>
<!DOCTYPE package PUBLIC
  "+//ISBN 0-9673008-1-9//DTD OEB 1.2
Package//EN"
  "http://openebook.org/dtds/oeb-
1.2/oebpkg12.dtd">
<package unique-identifier="oebps12package">
  <metadata>
    <dc-metadata
      xmlns:dc=http://purl.org/dc/elements/1.1/
      xmlns:oebpackage="http://openebook.org/
namespaces/oeb-package/1.0/">
      <dc:Identifier
        scheme="uri"
        id="oebps12package">http://openebook.org/
specifications/oeb-ps/1.2/
      </dc:Identifier>
      <dc:Identifier scheme="isbn">ISBN 0-9673008-1-
9</dc:Identifier>
      <dc:Title>Open eBook Publication Structure
1.2</dc:Title>
      <dc:Publisher>Open eBook Forum</dc:Publisher>
      <dc:Language>en-US</dc:Language>
    </dc-metadata>
  </metadata>
  <manifest>
    <item id="oeb12" href="oeb12-oebps.html" media-
type="text/x-oeb1-document" />
    <item id="logo" href="logo.png" media-type="image/
png" />
  </manifest>
  <spine>
    <itemref idref="oeb12" />
  </spine>
</package>
```

3. forráskód Az „Open eBook Publication Structure 1.2” oeb12-oebps.opf állományának kódja

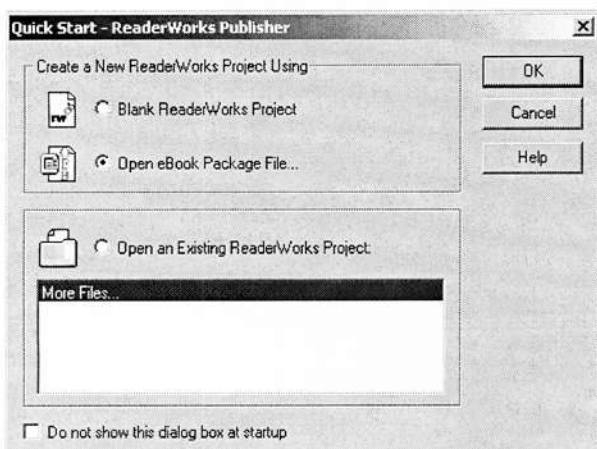
LIT konverzió ReaderWorksszel

Mielőtt az eljárás további részleteivel megismerkednénk, érdemes áttekinteni, hogy milyen fájlok készültek eddig:

- oeb12-oebps.html (A specifikáció részletét tartalmazó XHTML.)

- oeb12-oebps.css (Az XHTML megjelenéséért felelős lépcsős stíluslap.)
- logo.png (Az OEB logója, mely a címlapon szerepel.)
- oeb12-oebps.opf (A LIT összetevőit tartalmazó csomag.)

A továbblépéshez és egyben az átalakításhoz szükség van egy olyan konverterre, amely képes a felsorolt állományok LIT-té „kovácsolására”. Az interneten keresgélve a feladatra több alkalmas programot is lehet találni, ám mind közül legprofesszionálisabb az az OverDrive¹³ termék, amely a ReaderWorks nevet viseli. A szoftvernek két kiadása jelent meg, az ingyenesen letölthető Standard és a mintegy 119 USD-ért (~22 000 Ft) megszerezhető Publisher (3. ábra). Az előbbi működésében – ahogy ez a free verzióknál gyakran lenni szokott – bizonyos korlátozásokat vezettek be, de a transzformációra értelemszerűen ugyanúgy alkalmas, ahogy a fizetős változat. A következőkben az utóbbit fogjuk nagy vonalakban ismertetni és alkalmazni, ám mindvégig feltüntetve, hogy a Standardben mi nem működik.

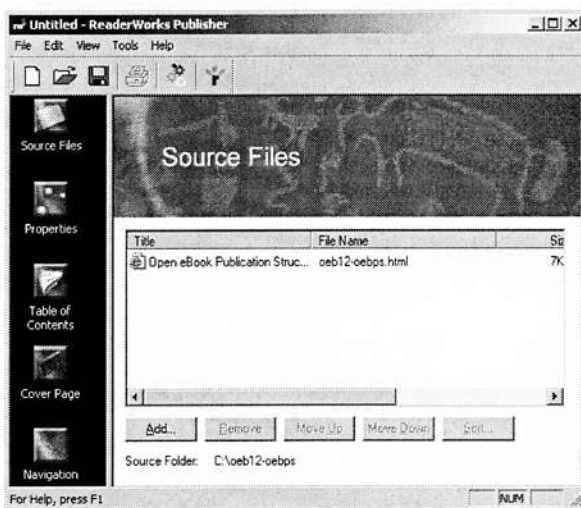


3. ábra A ReaderWorks Publisher 2.0 startablaka

A program letöltése és telepítése, illetve indítása után a következő lehetőségeket kínálja fel:

- új ReaderWorks Projekt kezdése,
- eBook csomag fájl (OPF) megnyitása,
- korábban mentett ReaderWorks Projekt (RWP) megnyitása.

Mivel korábban elkészítettük a csomagot, a 2. lehetőséget kell választani. Ezután kitallózható és betölthető az elkészített OPF állomány. Ha mindezzel kész vagyunk, akkor a Publisher kezelőfelülete tárul elénk (4. ábra).



4. ábra A ReaderWorks Publisher 2.0 kezelőfelülete

A betöltés után a forrásállományok listája (**Source Files**) menüpont aktivizálódik, és látható lesz a *spine*-ban meghatározott összetevő-sorrend. Ezen utólag még van lehetőség változtatni, illetve ha nem OPF-fel dolgozunk – lásd 3. ábra 1. pont –, akkor itt lehet beállítani a lineáris olvasási sorrendet.

Leendő e-könyvünk OPF-jének *dc-metadata* elemében beállított tulajdonságai a (**Properties**) menüpontban tűnnek fel. Ha a csomagban nem rögzítettük a metaadatokat, a rendelkezésre álló beviteli mezőkben lehet ezt megtenni. A metainformációk teljes elhagyása már csak azért sem ajánlott, mert például innen kerül a LIT kezdőlapjára a kötet címe és szerzője is.

A (**Table of Contents**) pontban előre elkészített tartalomjegyzék állomány betöltésére nyílik lehetőség. Példánkban nem szerepel ilyen, ugyanis a szöveget hordozó XHTML állományban helyeztük el a tartalmat alkotó egyes fejezetcímeket és a kapcsolódó könyvjelzőket is. Amennyiben semmiféle tartalomjegyzékkel nem rendelkezünk, ám mégis szeretnénk, hogy az e-könyvnek legyen ilyen funkciója, ismét a ReaderWorksöt kell segítségül hívni, hiszen a program egy könnyen kezelhető speciális varázsló segítségével lehetőséget nyújt tartalomjegyzék generálására.

Mint minden hagyományos értelemben vett könyvnek, az e-könyvnek is lehet egyedi tervezésű borítólapja (**Cover Page**). Ez a funkció sajnos az ingyenesen letölthető Standardben nem hozzáférhető – ott automatikusan a ReaderWorks grafikája

kerül a borítólapra konverzió során –, ám a Publisher több lehetőséget is kínál:

- könyvtárkép elhelyezése (library image) – a Microsoft Reader Libraryjében jelenik meg, formátuma GIF, JPG vagy PNG lehet;
- borítólap elhelyezése (cover image) – GIF, JPG vagy PNG;
- Pocket PC-re optimalizált borító és könyvtárkép elhelyezése – GIF, JPG vagy PNG.

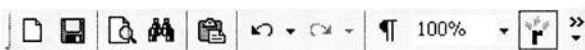
A borító funkció legnagyobb haszna – amellet, hogy esztétikusabbá és szebbé teheti a könyvet – elsősorban annak egyedivé tételében rejlik, hiszen egységes arculat kialakításával utalhat a forrásgyűjteményre, esetleg e-kiadóra, ugyanakkor követheti az eredeti mű küllemét is. Példánkhoz mi is készítettünk borítólapot, melyet természetesen el is helyeztünk a készülő e-bookon – lásd 10. ábra.

A navigáció (**Navigation**) menüpont szintén csak a Publisherben elérhető, ám mivel funkciója teljesen megegyezik az OPF *tours* eleméével – akár ott is elkészíthető –, így Standardben való nélkülözésének nincs nagy jelentősége.

Áttekintve a program legfontosabb funkcióit, nincs más hátra, mint a „Build eBook” menüpontra vagy ikonra kattintani, és a szoftver a megadott forrásokból elkészíti azt a LIT formátumot, amelynek megtekintésére azonnal lehetőség is nyílik a telepített Ms. Reader futtatásával – lásd bővebben A LIT használata c. fejezetet.

LIT-konverzió Worddel

Láthattuk, hogy milyen folyamatok során lesz SGML/XML-ből LIT, és valljuk be, korántsem meglepő, ha néhányan elborzadtak az egésztől. Éppen ebből a megfontolásból kiindulva – vagyis hogy nem érthet mindenki az XML alapú technológiákhoz – fejlesztette ki a Microsoft azt a LIT-konverter plugint,¹⁴ amelynek segítségével korrektúrázott RTF vagy DOC kiterjesztésű szövegeinkből egyetlen kattintással létrehozhatjuk a LIT formátumú könyvet. A program Word 2000-től telepíthető, és ugyanúgy beépül a szövegszerkesztőbe, mint ahogy azt az Adobe Acrobat is teszi (5. ábra).



5. ábra A Wordbe épült LIT-konverter plugin ikonja

Az ikonra kattintva megnyíló párbeszédablakban megadható a mű címe, szerzője és a kimeneti állomány neve – alapesetben ezeket az „Adatlap”-

ról veszi a program –, nem utolsósorban pedig elhelyezhető a GIF, JPG vagy PNG formátumú borítólap. A megoldás tehát praktikus, és nem igényel előképzettséget, ennek ellenére inkább egyszerűbb, mindenféle speciális formázástól, bonyolult táblázatoktól stb. mentes köteteknél célszerű használni, ha azt akarjuk, hogy a végeredmény szép is legyen.

A Microsoft Reader és a LIT képességei

A LIT formátumú e-könyv lapozható, megjelenése könyvszerű. Beépített teljes szövegű keresést tartalmaz, jegyzetelhetünk, rajzolhatunk és kiemelhetünk benne, majd mindezt úgy tárolhatjuk, hogy közben az e-könyv szövege nem változik semmit. A formátum az olvasóprogram segítségével képes a jegyzetanyagok megjelenítésére is. Oldalanként az igazi könyvnél kicsit kevesebbet, megközelítőleg 30-40 sort – bár ez a beállított betűmérettől függ – és kb. soronként 50 karaktert tartalmaz. Hardver- és szoftverfüggő, zárt formátumról van szó, amely nem nyomtatható, és az oldalankénti szerkezet lezártsága miatt egyben nem másolható.

Miért kell a LIT, ha van PDF?

A PDF remek formátum ahhoz, hogy dokumentumokat és nyomdai információkat rögzítsünk. Mégis ez egy statikus fájl, nem jók, jobban mondván inkább kötöttek az arányai, és ez használatát behatárolja. A Microsoft Reader dinamikus olvasómotor, ami azt jelenti, hogy az aktuálisan használt képernyő méretéhez képes igazítani a LIT dokumentumoldalt. A dinamikus dokumentumoldal legnagyobb előnye pedig az, hogy optimális lehetőséget teremt az olvasásra, és számunkra ez a legfontosabb. Az olyan szoftverrel, amely támogatja a dinamikus dokumentumoldalt, a könyvek és dokumentumok bármilyen képernyőméreten jól olvashatóak, sőt a szövegméret is növelhető vagy csökkenthető az olvasó igénye szerint.

Elválasztási problémák

A Microsoft Reader dinamikus olvasómotorjának elválasztási problémái a szoftver nyelvfüggőségéből adódnak. Akik használták már a LIT formátumot, biztosan tudják, hogy a programnak csak angol, német, francia, olasz és spanyol nyelvű verziója létezik.¹⁵ Ez részben azt jelenti, hogy nincs magyar kezelőfelülete, illetve helyes szöveg-elválasztásra is csak a felsorolt nyelveken képes. De mi történik akkor, ha magyar nyelvű dokumen-

tumot nyitunk meg olvasásra? Nos, sajnálatos módon az is elválasztódik, ám nem a magyar helyesírási szabályok, hanem pl. az angol nyelvtan alapján – angol Reader esetén. Ekkor találkozhatunk olyan „szörnyűségekkel”, mint amilyet a 6. ábra is mutat.

énekek, de nem azt az önálló életet élők, mint a mesterek más, melyek és szebb műveik...

6. ábra Helytelen elválasztás a Microsoft Readerben

Valljuk be őszintén, nem kell nyelvészseninek lenni ahhoz, hogy belássuk, ez nem maradhat így, mert nemcsak a „magyarosok”, hanem még az informatikusok szemét is bántja. Szerencsére léteznek megoldások a probléma orvoslására – némelyek egyszerűbbek, mások bonyolultak –, de kezdjük inkább az előbbiekkal.

Balra igazított szöveg

Ha lépcsős stíluslapunkban balra igazítottá tesszük a szövegre vonatkozó utasításokat, azzal elérjük, hogy az SGML/XML -> XHTML -> LIT konverzió eredményeként létrejött állomány szövege nem választódik el, mivel ebben az esetben a Reader elválasztó modulja sem lép működésbe.

Wordből történő konverzió esetén egyszerűen csak balra kell igazítani a szöveget (7. ábra), majd a már ismertetett módon átkonvertálni a DOC vagy RTF állomány(oka)t. (Zárójelben jegyezzük meg, hogy DOC, RTF -> LIT konverzióval létrehozott fájlokban – a szöveg igazításától függetlenül – nincsenek elválasztási problémák, mert a szövegszerkesztőből létrehozott LIT-et nem választja el a Reader.) Ezek a megoldások kevésbé elegánsak, hiszen a könyvek rendszerint sorkizártan, nem pedig balra igazítva olvashatók, másrészt igényesebb helyeken nem a Word állományok képezik a szövegtárolási formátumot, ennek ellenére egyszerűsége miatt ezt alkalmazza a HP is saját e-book tárában.¹⁶

szükséges a kutatók terület nyelvének bizonyos ismerete, különben nem tudni megalkotni bizonyos vonatkozásokat szó és

7. ábra Balra igazított szövegrészlet a Microsoft Readerben

Helyesen elválasztott szöveg

Amennyiben azt szeretnénk, hogy Readerünk magyar helyesírás szerinti elválasztást alkalmazzon, a következők szerint kell eljárni. Speciális elválasztó programok alkalmazásával, mint pl. a Morpho-Logic „Helyes-e?”¹⁷ nevet viselő szoftvere, ún. unit separatorok (US)-ek (&#x1F;) helyezhetők el a

szöveget tartalmazó SGML/XML állományokban. A fájlok bármely karakteres szerkesztőben – jEdit, TextPad, EditPlus stb. – való megnyitásával, majd a separatorok feltételes elválasztójelekre (soft hyphen – ­ ­ ­) cserélésével – természetesen mindez az alkalmazott kódtáblától függ – elérhető, hogy az SGML/XML -> XHTML -> LIT konverzió után megfelelően elválasztott szavakat találunk (8. ábra).

tükröz, de egyben teljesen át van írtva a nyugat-európai zene sablonjaival. Az egrotikum és banalítás összekapcsolása kompro-

8. ábra Sorkizárt szövegrészlet helyes elválasztással a Microsoft Readerben

Felhőtlenül azonban még ennél a technológiánál sem örülhetünk, mert azzal, hogy szavaink – noha az olvasó számára láthatatlan elválasztójelekkel „töltődtek fel” – a Reader számára kereshetetlené válnak, ugyanis keresőmotorja észreveszi a forrásban megbúvó entitásokat. Kétségkívül ez hiba az olvasószoftverben, de nem tudni, hogy javításra van-e lehetőség. Ezzel pedig elveszítjük az e-könyv egyik legfontosabb tulajdonságát, a kereshetőséget.

Ennél sokkal kisebb probléma – de probléma –, hogy a Helyes-e? – függetlenül attól, hogy kitűnő és hasznos program – „csak” az általa ismert szavakat választja el. Persze van lehetőség az ismeretlen szavak algoritmikus elválasztására is, ám ez nem igazán jó megoldás, mert jelentős részük vagy idegen, vagy összetett, és ezeket nem lehet hangzási alapon helyesen elválasztani. Ellenben a program képes elválasztási kivételszótárt is használni. Ez azt jelenti, hogy ha a helyesírás-ellenőrzéskor minden ismeretlen szót felvesszünk elválasztással, akkor a dokumentum minden szava el lesz választva. Az elválasztási kivételszótár teljes szóalakot vesz figyelembe, vagyis minden ismeretlen szót annyiszor kell felvenni, ahány különböző szóalakban szerepel a dokumentumban. A szófelvétel segíthet azzal, hogy a szó algoritmikus elválasztását alapértelmezettként megadja a program.

A szoftver ismer néhány többértelmű szóalakot, amely bármely értelmében szerepelhet általános szövegben, és elválasztása az értelemtől függ. Ilyen pl. a „legelőre”. Ezeket a szavakat csak azokon a helyeken választja el, amelyek a szó értelmétől függetlenek, pl. „legelő-re”.

Lehetnek olyan szavak, amelyeket valóban nem lehet elválasztani, mert rövidke, vagy azért, mert

pl. az ssz -> sz-sz elválasztást nem lehet soft hyphenel kezelni, így az „asszonnyal” szóra azt kell visszaadni, hogy nincs elválasztása. Az el nem választott szavakat viszont automatikusan elválasztja a Microsoft Reader – nagy valószínűséggel helytelenül –, tehát ezekben az esetekben ezt valahogy meg kell akadályozni. De hogyan?

Elválasztás nélküli szöveg

Ahogy azt az 1. forráskód is tartalmazza, a <html> elem attribútumaként definiálható a nyelvhasználat, tehát az ott látott amerikai angol mintájára megadható a magyar is – xml:lang="hu" lang="hu". A nyelvi kód nevezett címkében való attribútum értékent definiálása, az SGML/XML -> XHTML -> LIT átalakítás után azt fogja eredményezni, hogy az olvasószoftver nem választja el a szöveget, az pusztán sorkizártan jelenik meg (9. ábra). Mindez pedig egyértelműen azt bizonyítja, hogy a Reader elválasztáskezelése valóban nyelvfüggő, és a szoftver magyar elválasztót használna, ha lenne neki. Azonban a programhoz nem áll rendelkezésre ilyen, ráadásul a Microsoftnak ehhez SDK-ja (Software Development Kit = szoftverfejlesztői környezet) sincs, így a beépítés egyelőre nem lehetséges. Lehetett olvasni arról, hogy a redmondi óriás megállapodott egy céggel, amely minden nyelvre beépít nekik algoritmikus elválasztót, ám az említett megoldás azt vetíti előre, hogy nem terveznek olyan nyílt kivitelezést, amit pl. az Office-ban használnak, vagyis „bedrótozott” technikára lehet továbbra is számítani.¹⁸

Igaz, maig is élnek, de nem azt az önálló életet élik, mint a mesterek más, mélyebb és szebb művei...

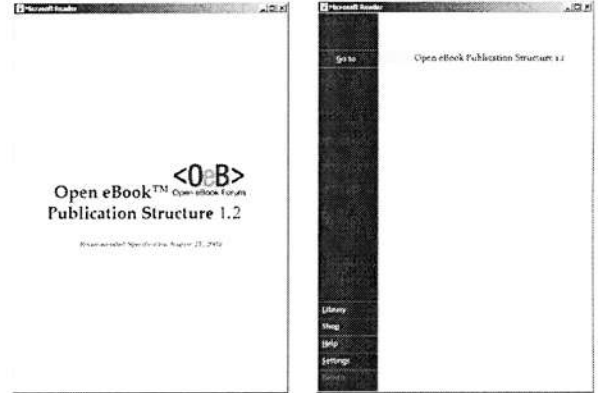
9. ábra Sorkizárt szövegrészlet elválasztás nélkül a Microsoft Readerben

Összehasonlítva a 6. ábrán látható kiinduló állapottal, bátran kijelenthetjük: elértük a kívánt célt. Találtunk egy olyan módszert, melynek kivitelezése nem érinti a kötetek forráskódú szerkesztését, hiszen teljes mértékben automatizálható a bemutatott attribútum(ok) és érték(ek) (X)HTML-ben való elhelyezése. Ráadásul szövegünk immár nem tartalmaz helytelen elválasztásokat, és továbbra is teljes mértékben kereshető maradt.

A LIT használata

A LIT formátumú e-könyvek használatához a Microsoft honlapjáról mindössze le kell töltenünk a Microsoft Reader¹⁹ programot, majd PC-nkre vagy

PDA-nkra kell telepíteni. (A folyamat nem bonyolultabb, mint a PDF formátum olvasásához szükséges Acrobat Reader installálása.) ReaderWorks-szel készített OEB specifikációrészletet tartalmazó e-bookunkat betöltve az olvasószoftverbe, először a könyv borítója, majd címlapja tárul elénk (10. ábra).



10. ábra Az elkészült e-könyv a Microsoft Readerben megnyitva (borítólap és címlap)

A címlap bal oldali sávjában a legfontosabb funkciók érhetők el, úgymint:

- Go to (tartalma pozíciófüggő, ám lényegében a szoftverben való navigálást segíti);
- Library (a korábban már megnyitott könyvek listája);
- Shop (a Microsoft website e-book üzlete, illetve a szoftver aktivációs lehetősége);
- Help (a Reader súgója);
- Settings (a beállítási lehetőségek – betűméret, clear type, képernyő, hang és annotációk);
- Return (visszaugrás a legutóbb megtekintett oldalra).

A következőkben nem célunk az olvasószoftver részletes ismertetése – egyszerűségéből adódóan különösebb nyelvismeret nélkül is bárki percek alatt áttekintheti –, inkább ábrákkal illusztrálva néhány hasznos funkciójára helyezzük a hangsúlyt, melyhez a Neumann-ház gyűjteményének kötetait hívjuk segítségül.

Bármilyen könyvet olvas az ember, gyakran előfordul, hogy a tartalommal kapcsolatosan olyan gondolatai támadnak, amelyeket le szeretne írni. A papír alapú könyvben azonban vétek megjegyzéseket elhelyezni, így aztán maradnak a cédulák. Ezekkel viszont az a probléma, hogy elkeveredhetnek, elveszhetnek, és még sorolhatnánk. E-könyvünk ezt a problémát egészen elegánsan

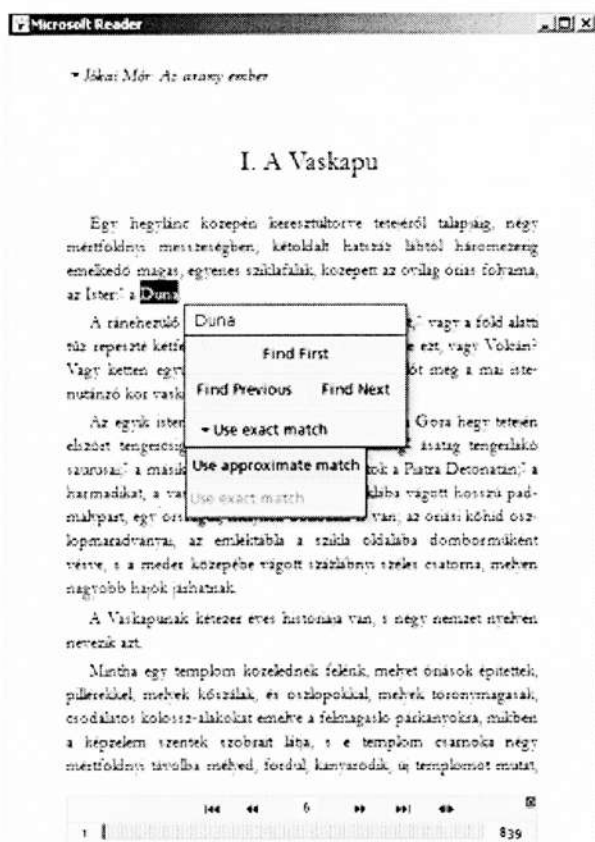
oldja meg, ugyanis lehetőséget biztosít pl. szavakhoz köthető olyan egyéni megjegyzések beszúrására (Add Text Note), melyek automatikusan eltárolódnak, ám később egy kattintással előhívhatók, szerkeszthetők és törölhetők. Azokat a szavakat, mondatrészeket stb., melyekhez korábban megjegyzéseket csatoltunk, vagy netán valamilyen szempontból különös jelentőséggel bírnak számunkra, tetszőleges színnel kiemelhetjük (Add Highlight), majd a későbbi könnyű megtalálás érdekében az oldalakat könyvjelzőkkel (Add Bookmark) megjelölhetjük. Ez utóbbiak a lap jobb oldalán helyezkedhetnek el, és kattintással a kívánt oldalra lehet ugrani segítségükkel. Noha kevésbé lényeges funkció, azért érdemes megemlíteni, hogy a Reader rajzolási lehetőséget (Add Drawing) is biztosít az egyes e-könyvlapokon. Az így készült rajzok, jelölések természetesen bármikor törölhetők, ugyanakkor színük is változtatható (11. ábra).



11. ábra Jegyzet, rajz, könyvjelző, kép és kiemelés a Readerben (Részlet Herman Ottó: Arany, Tompa, Petőfi és a népköltés madárvilága c. kötetéből)

A felsoroltakon kívül szintén a kiemelt szolgáltatások között említhető a keresés (Find...) funkció, amelynek segítségével adott szó után pontosan (Use exact match), vagy hozzávetőlegesen, közelítőleg (Use approximate match) kutathatunk az e-könyv teljes szövegében (12. ábra).

Végezetül már csak egyetlen fontos dologról nem tettünk említést, pedig ez is nagy lehetőségeket hordoz magában. Arról a text-to-speech²⁰ modulról van szó, amelyet angol, francia és német nyelvhez fejlesztettek ki, s melynek hozadéka, hogy az említett nyelveken írt/készített e-könyveket akár fel is olvastathatjuk a szoftverrel – sajnos magyarhoz ilyen kiegészítés sem létezik.



12. ábra Keresés a Microsoft Readerben (Részlet Jókai Mór: Az arany ember c. regényéből)

Következtetések

Az e-book jövője egyelőre nem határozható meg pontosan, de a téma aktualitását mutatja, hogy 2004-ben, a Bath Egyetemen megrendezett Digitális Könyvtárak Európai Konferenciáján is előadás hangzott el a szakterület kapcsán.²¹ Kezd az is körvonalazódni, hogy legnagyobb lehetősége az oktatásban – szakkönyvek, tankönyvek, útikönyvek, kézikönyvek, „rég” könyvek szolgáltatásában – rejlik. Szorosan az oktatáshoz kapcsolódva, a lehetséges felhasználási területek között szerepel még a tudományos jellegű alkalmazás, vagyis minden olyan terület, ahol a szótárszerű rákeresés fontosabb lehet, mint az olvasási élmény.

A képernyőről való információgyűjtés a PC-k, Notebook-ok, PDA-k, PDA – mobil kombók terjedésével egyre inkább magától értetődővé válik. Jelen pillanatban azonban az a helyzet, hogy a mai kor embere nem szeret képernyőről olvasni, ha nem muszáj – elsősorban éppen emiatt bukott

meg az e-book üzleti célú alkalmazása. Ám még ha nem is szeretünk annyira képernyőről olvasni, információszerezési szokásaink azért változnak, tehát olvasásunk módja is változik – több, de rövidebb szöveget olvasunk. Ráadásul mivel az internet a képi rövidségű szövegek médiuma, noha a könyveket nem oda tervezték, azok forgalmát le lehet ott bonyolítani, célunk pedig részben ez.

Összességében tehát kijelenthetjük, alternatív megjelenítési formátumként érdemes szolgáltatni e-könyveket, hiszen a számítógépen felnövekvő új nemzedék minden bizonnyal már ilyen dokumentumokat is használni fog. Ha pedig az oktatás az e-bookra is nevel, elképzelhető, hogy hatékonyabban átmentheti a könyvkultúrát a jövőbe.²²

Jegyzetek

- ¹ F. TÓTH Krisztina: Az elektronikus könyv helye a digitális kultúrában. <http://www.mediakutato.hu/cikk.php?i=119>, a letöltés időpontja: 2005-03-02.
- ² Neumann-ház – Bibliotheca Hungarica Internetiana – Gondolkodó magyarok. <http://www.neumann-haz.hu/e-book/index.html>, 2004-12-27.
- ³ Magyarországon a Neumann-ház, a HP és a Magyar Elektronikus Könyvtár (MEK) foglalkozik nagyobb számban Microsoft Readerrel olvasható LIT formátumú e-könyvek előállításával.
- ⁴ EbookMall, Device Comparisons. <http://ebookmall.com/knowledge-collection/device-comparisons.htm>, 2004-08-16.
- ⁵ BÍRÓ Szabolcs: A szövegfeldolgozás modern eszközei: az SGML/XML nyelvek. = TMT, 51. köt. 10. sz. 2004. p. 453–459.
- ⁶ Adobe Reader download page. <http://www.adobe.com/products/acrobat/alternate.html>, 2004-08-16.
- ⁷ Microsoft Reader for Desktop and Laptop PC. <http://www.microsoft.com/reader/downloads/pc.asp>, 2005-01-03.
- ⁸ Az XSL és CSS nyelvek ismertetésétől eltekintünk, mert amellett, hogy nem szorosan kapcsolódnak a LIT-konverzióhoz – inkább csak egyfajta lépcsőfoknak tekinthetők a folyamatban – messze túlmutatnak e cikk célján és keretein.
- ⁹ Abban az esetben, ha már korábban is az SGML/XML technológiát használtuk a digitalizált művek tárolásához, a(z) (X)HTML-t pedig publikálásához, akkor természetesen nem kell új XSL-t és CSS-t készíteni, a már meglévők módosításával elérhető a kívánt eredmény.

- ¹⁰ Az OEB specifikációt egy kiadókból, könyvkereskedőkből, szoftverfejlesztőkből és a könyvpiar körébe tartozó egyéb intézményekből álló szervezet, az Open eBook Forum gondozza. Eddig három kiadása jelent meg ajánlás formájában: 1.0, 1.0.1 és 2002. augusztus 27-én az 1.2-es. <http://www.openebook.org/specs.htm>, 2004-08-16.
- ¹¹ Az XHTML névterek definícióinak helye. <http://www.w3.org/1999/xhtml>, 2004. 08. 13.
- ¹² Multipurpose Internet Mail Extensions – Sokcélú internetes levélbővítések. Segítségével internetes levelekhez nem szöveges fájlokat lehet csatolni.
- ¹³ OverDrive, Inc. – ReaderWorks. <http://www.overdrive.com/readerworks/>, 2004-08-16.
- ¹⁴ Read in Microsoft Reader. <http://www.microsoft.com/reader/developers/downloads/rmr.asp>, 2004-08-16.
- ¹⁵ Lásd 7. pont.
- ¹⁶ HP e-book tár. http://www.hpebook.hu/index_1.my, 2005-01-04.
- ¹⁷ MorphoLogic – Helyes-e? (Helyesírás-ellenőrző és elválasztó program.) http://www.morphologic.hu/h_he1.htm, 2005-01-03.
- ¹⁸ PÁL Miklós: Microsoft Reader (LIT) – magyar elválasztás (elektronikus levél). 2004. december 6.
- ¹⁹ Lásd 7. pont.
- ²⁰ Microsoft Text-to-Speech Package. <http://www.microsoft.com/reader/developers/downloads/tts.asp>, 2005-01-04.
- ²¹ MALAMA, Chrysanthi–LANDONI, Monica–WILSON, Ruth: Fiction Electronic Books: A Usability Study. = Research and Advanced Technology for Digital Libraries: 8th European conference, ECDL 2004, Bath, UK, September 2004 : proceedings / Rachel Heery, Liz Lyon (eds.). Berlin, New York: Springer, 2004. p. 69–79. ISBN 3-540-23013-0
- ²² PÁLFI Norbert: IRODALOM, SZÖVEG, INFORMÁCIÓ. Könyv a G2 galaxisban, avagy az ebook helyzete és kilátásai. <http://magyar-irodalom.elte.hu/vita/tpn.html>, 2004-08-16.

Beérkezett: 2005. I. 6-án.



Bíró Szabolcs

informatikus-könyvtáros,
a Neumann János Digitális Könyvtár
és Multimédia Központ
Kht. könyvtár-informatikai osztályán
osztályvezető-helyettes.
E-mail:
biro.szabolcs@neumann-haz.hu