

**Papp Gyula**

# **eLearning szabványok**

elemző tanulmány



## Tartalomjegyzék

|   |           |
|---|-----------|
| <b>A SZABVÁNYOSÍTÁSRÓL .....</b>                      | <b>5</b>  |
| AZ E-LEARNING FOGALOMRENDSZERE .....                  | 5         |
| <i>LO – Learning Object</i> .....                     | 5         |
| <i>Metaadat</i> .....                                 | 6         |
| <i>eLearning rendszer</i> .....                       | 6         |
| LMS - Learning Management System .....                | 6         |
| CMS – Content/Course Management System .....          | 6         |
| LCMS – Learning Content Management System.....        | 6         |
| VLE – Virtual Learning Environment.....               | 7         |
| A SZABVÁNYOK SZÜKSÉGESSÉGE .....                      | 7         |
| A SZABVÁNYOSÍTÁS ELŐNYEI.....                         | 9         |
| <i>Együtműködési képesség</i> .....                   | 9         |
| <i>Újrahasznosíthatóság</i> .....                     | 9         |
| <i>Tartósság</i> .....                                | 9         |
| <i>Testre szabhatóság</i> .....                       | 9         |
| <i>Elérhetőség, keresetőség</i> .....                 | 9         |
| <i>Költségmegtakarítás</i> .....                      | 10        |
| A SZABVÁNYOSÍTÁS FŐBB TERÜLETEI .....                 | 11        |
| <i>Metaadat modell</i> .....                          | 11        |
| <i>Tartalom modell</i> .....                          | 11        |
| <i>Sorrend és navigáció</i> .....                     | 12        |
| <i>Futtatási környezet</i> .....                      | 12        |
| <i>A szabványosítás folyamata</i> .....               | 13        |
| SZERVEZETEK .....                                     | 15        |
| ADL.....  | 15        |
| AICC.....   | 16        |
| ALIC.....   | 16        |
| ARIADNE.....  | 16        |
| DCMI .....  | 16        |
| EdNA .....  | 17        |
| IEEE LTSC.....  | 17        |
| IMS.....  | 18        |
| ISO/IEC JTC1/SC36.....                                | 18        |
| O.K.I.....  | 18        |
| PROMETEUS.....  | 19        |
| WS-LT.....  | 19        |
| <b>SZABVÁNYOK ÉS SZABVÁNYJAVASLATOK .....</b>         | <b>20</b> |
| AZ AICC KEZDEMÉNYEZÉSEI .....                         | 20        |
| AZ IMS KEZDEMÉNYEZÉSEI.....                           | 21        |
| <i>Learning Resource Meta-data (LRM)</i> .....        | 21        |
| <i>Content Packaging (CP) – tartalomcsomag</i> .....  | 22        |
| <i>Question and Test Interoperability (QTI)</i> ..... | 22        |
| <i>Learner Information Package (LIP)</i> .....        | 23        |
| <i>Simple Sequencing (SS)</i> .....                   | 23        |
| <i>Enterprise &amp; Enterprise Service</i> .....      | 23        |
| <i>Learning Design (LD)</i> .....                     | 23        |
| <i>Digital Repositories (DR)</i> .....                | 24        |
| LOM.....  | 25        |
| <i>A LOM szerkezete</i> .....                         | 25        |
| SCORM .....   | 27        |
| <i>Hitelesítés</i> .....                              | 27        |
| <i>A SCORM fejlődése</i> .....                        | 28        |
| A SCORM 1.2 .....                                     | 29        |
| <i>A SCORM felépítése</i> .....                       | 29        |
| <i>Első könyv: Áttekintés</i> .....                   | 29        |
| <i>Második könyv: Tartalomhalmazási modell</i> .....  | 30        |
| A SCORM tartalomhalmazási modell áttekintése.....     | 30        |

|   |    |
|---|----|
| A SCORM tartalommodell .....                      | 30 |
| Tananyagelem (Asset).....                         | 30 |
| SCO (Sharable Content Object) .....               | 30 |
| Tartalomszervezés .....                           | 31 |
| Metaadatok használata .....                       | 33 |
| Metaadat kezelés .....                            | 33 |
| Tartalomcsomag .....                              | 35 |
| A manifest .....                                  | 35 |
| Submanifest alkalmazása.....                      | 37 |
| SCORM tartalomcsomag alkalmazási profil .....     | 38 |
| Sorrend és navigáció.....                         | 38 |
| <i>Harmadik könyv: Futásidejű környezet</i> ..... | 39 |
| API metódusok .....                               | 40 |
| Hibakódok .....                                   | 41 |
| Adatmodell .....                                  | 41 |
| A SCORM 2004 .....                                | 42 |
| <i>Negyedik könyv: Sorrend és navigáció</i> ..... | 44 |
| A Klaszter (Cluster).....                         | 45 |

# A szabványosításról

A számítógépek oktatási eszközként történő felhasználása egyidős az elektronikus számítógépek elterjedésével. A kezdetektől léteztek grandiózus tervek, amelyek a számítógépekre alapozott komplex oktatási rendszerekről szóltak, azonban erőforrások híján ezekre még sokáig kellett várni. A számítógépes oktatóprogramok a technikai fejlődés nyújtotta lehetőségek függvényében egyre újabb és újabb szolgáltatásokat nyújtottak, és egyre nagyobb mennyiségű tananyagot fogtak össze. Bár vitathatatlan, hogy az oktatóprogramok adott esetben sikeres eszköznek bizonyulnak, de több probléma is felmerült alkalmazásukkal kapcsolatban.

Ilyen például a differenciálás kérdése. Hogyan lehet megoldani a tananyag személyre szabhatóságát? Ez igen erőforrás-igényes feladat, továbbá újra felveti a nagyobb tárolókapacitás szükségességét. A tananyagoknak egy ilyen rendszerben dinamikusan változtathatóknak kell lenniük. Egy másik probléma a tananyag hasznosulásának, illetve hatékonyságának a kérdése. Az oktatás mindig is kétszereplős tevékenység: tanár – tanuló. Hogyan értesül a tanár a tanuló eredményességéről, illetve sikertelenségéről, továbbá hogyan tudja tetten érni ezek okait? A hálózatok (internet, intranet) terjedése megteremtette a lehetőséget a válaszok megszerzésére, valamint lehetőséget adott a tananyag dinamikusan változtatására, de ehhez újabb alkalmazásokat kellett kifejleszteni.

Az alkalmazások fejlesztése alapvetően két szemlélet mentén folyt. Az egyik irányzat a tanulási folyamatot kívánta modellezni, s ismeretelméleti alapon közelített a problémához. Törekvésük célja úgynevezett intelligens oktatási rendszerek (ITS) fejlesztése volt. A másik irányzat formai szempontok szerint közelített a problémához, és az üzleti racionalitás figyelembe vételével az oktatás eljárás-alapú modelljéhez igazodott.

A kilencvenes évek második felében e két szemlélet találkozott egymással, s generálta az eLearning szabványosítási folyamatot. A folyamat lényege az, hogyan lehetne a gazdaságosság figyelembe vételével olyan oktatási alkalmazásokat fejleszteni, amelyek hatékonyabbak a korábbiaknál. Ehhez a tartalmat el kell választani a vezérléstől. Viszont akkor egyértelműsíteni kell a tartalom és a futtató környezet közötti kommunikáció módját. Ez, és a fent említett szempontok együttesen vezettek az e-tanulás objektumorientált megközelítéséhez.

## Az eLearning fogalomrendszere

### LO - Learning Object

A megközelítés lényege az, hogy meghatározzuk a tartalom legkisebb önállóan is értelmes egységeit. Ehhez a tananyag dekompozíciója útján jutunk el. Amit kapunk, az a tananyagelem, más néven learning object (röviden LO).

Tananyagelem lehet egy kép, egy filmjelenet, egy animáció, egy képlet, egy szöveg, és így tovább. A legnehezebb a szövegnél megállapítani, hogy hol van az a határ, amikor már nem lehet tovább bontani. Illetve fordítva, amikor már struktúrát alkotnak a szövegrészek.

A tananyagelemeknek nem tartalmazhatnak utalást más tananyagelemre, nem hivatkozhatnak szöveggörnyezetükre, mert akkor sérül az újrahaznosíthatóságuk.

### **Metaadat**

Bár a tananyagelemek nem hivatkozhatnak egymásra, a közöttük fennálló logikai kapcsolat megteremtésére több eszközünk is van. Az egyik ilyen eszköz a metaadat, amely leírja, és egyben azonosítja is a tananyagelemeket. E logikai kapcsolódás megteremtésére három szinten kínálkozik lehetőség. A leglazább egy előre kiválasztott besorolási, osztályozási rendszer szerinti azonosítás, amely alapján azonos „jelentéskörbe” sorolunk tananyagelemeket. Ezt a meghatározást tovább pontosíthatjuk kulcsszavazással, s végül lehetőség van arra, hogy konkrétan hivatkozzunk egy vagy több a kapcsolódó tananyagelemre (lásd LOM). A tananyagelemek közötti tartalmi kapcsolat jelölésére, az elemek sorrendjének meghatározására pedig a manifest állomány szolgál.

### **eLearning rendszer**

Az eLearning rendszerekre hazánkban a keretrendszer kifejezés terjedt el. Azokat az alkalmazásokat értjük alatta, amelyeken keresztül a különböző szerepkörbe tartozó felhasználók (adminisztrátorok, szerzők, oktatók, tutorok és tanulók) hozzáférnek a tananyaghoz. Ezek az alkalmazások rendre moduláris felépítésűek, s attól függően, hogy mire helyezik a hangsúlyt, illetve annak függvényében, hogy hogyan alakul funkcionalitásuk más és másképp hívjuk őket.

### **LMS - Learning Management System**

Azaz tanulásirányítási rendszer. Feladata, hogy azonosítsa a felhasználókat, és jogosultságaiknak megfelelően hozzáférést biztosít számukra kurzusaikhoz, továbbá naplózza a felhasználók tevékenységeit. Kiemelkedően fontos a tanulói tevékenységek, és teljesítményadatok naplózása. A szabványokhoz igazodó LMS szerverekre jellemző, hogy a szabványos tananyagot jellemzően struktúrába szervezve, kurzusként tárolja.

### **CMS - Content/Course Management System**

Tartalomkezelő, illetve kurzuskezelő rendszer. A tartalomkezelő rendszerek nem alkalmasak az eLearning esetében megszokott tevékenységek naplózására. Bár jogosultságkezelés ezekben is van, de a naplóadatokból nem lehetséges elegendő pedagógiai releváns információt kinyerni, valamint inkább dokumentumkezelésről beszélhetünk esetükben, mint interaktív eLearning tananyagokról. A kurzuskezelő rendszerek viszont lényegüket tekintve LMS-ek.

### **LCMS - Learning Content Management System**

Azaz tanulási tartalom-kezelő rendszer, vagy tananyagkezelő rendszer. Jellegzetessége, hogy bár azonosításra itt is van lehetőség, de elsősorban a tananyagelemek tárolása a feladata. Valamint ebből kifolyólag mindig tartalmaz szerzői modult, amely segítségével a tárolt tananyagelemekből tananyagstruktúrákat, kurzusokat lehet építeni.

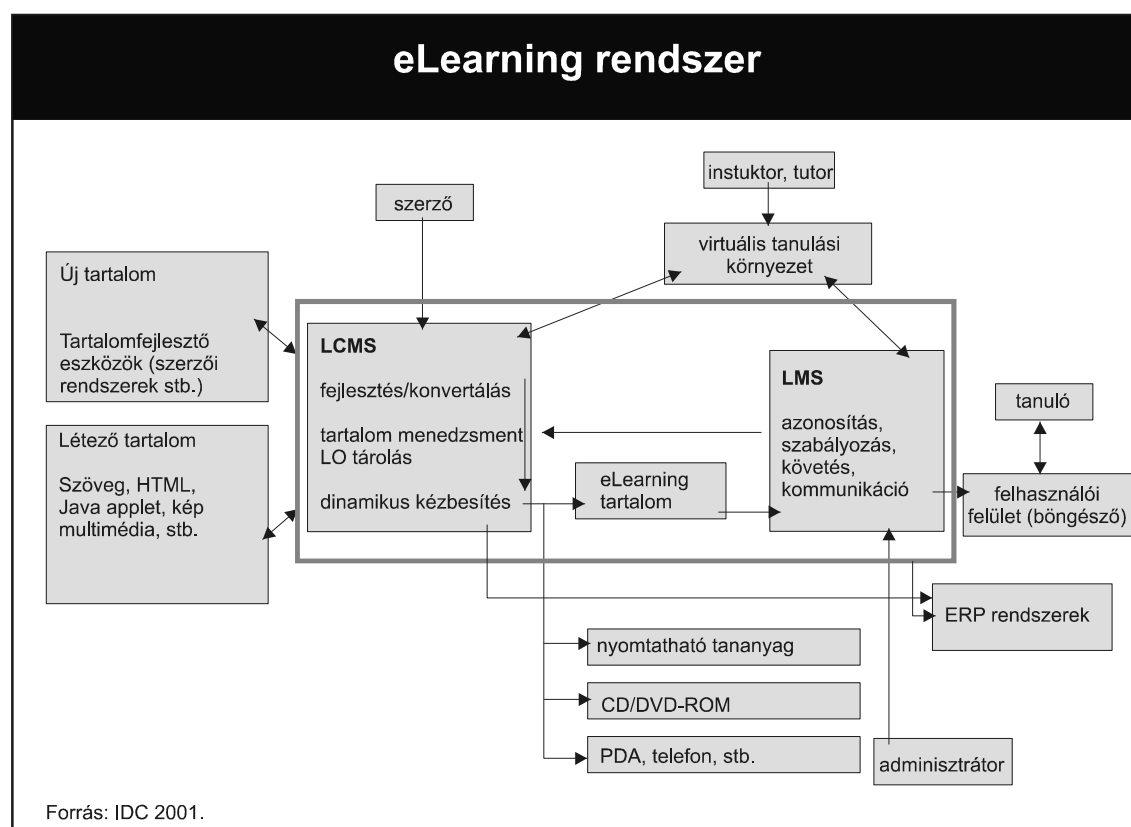
Az LCMS rendszerek is naplózhatnak, viszont nem elsődleges cél a tanulói tevékenységek követése, illetve a tanulói teljesítményadatok gyűjtése. Inkább a tananyagelemek végzett manipulációk nyomán követése a feladata. Ugyanakkor nem csak eLearning rendszerek számára képes publikálni, hanem CD/DVD formátumba, illetve nyomtatható formába is. LMS szerverrel együtt alkalmazva ideális eLearning rendszer alakítható ki vele.

## VLE - Virtual Learning Environment

Virtuális oktatási környezet. Az eLearning keretrendszerekben az a modul, amely a felhasználók – elsősorban a tanulók, oktatók, tutorok – számára kommunikációs felületet és együttműködési lehetőséget nyújt. Biztosítja azokat az eszközöket (csevegés, fórum, e-mail, üzenetváltás, faliújság, stb.), amelyek segítségével a hagyományos osztálytermi oktatásban megszokott tanítási-tanulási és szociális tevékenységek (pl.: kérdésfeltevés, megerősítés, fogadóóra, stb.) kvázi elvégezhetőek, illetve helyettesíthetőek.

Alapvetően két megvalósítási formája van a virtuális osztályterem használatának: szinkron és aszinkron. Az aszinkron képzési forma esetében a tanuló hálózaton keresztül saját ütemezése szerint halad a tananyag feldolgozásával. Ugyanakkor lehetősége van online eszközök használatára.

A szinkron képzés a résztvevők egyidejű jelenlétét feltételezi. Ebben az esetben többnyire élő video-, illetve hangkapcsolatot építenek ki a virtuális osztályterem résztvevői között.



1. ábra eLearning rendszer felépítése

## A szabványok szükségessége

A hagyományos számítógépes oktatóprogramok a mai napig egyedi megoldásokat tartalmaznak a tananyag tárolására, közvetítésére, a tanulói eredmények visszajelzésére. Egy konkrét témakörnek egyfajta konkrét realizációját nyújtják. A kész tananyag sem egészében, sem részeiben nem alkalmazható más kontextusban, más tananyag részeként egyrészt, mert fizikailag zárt egységet alkotnak, másrészt tartalmilag tele vannak az eredeti környezetére történő utalásokkal. (Bár létezik ajánlás erre a formátumra is: AICC fájl-alapú CMI rendszerek – AGR006) Tovább bonyolítja a dolgot, ha esetleg más operációs rendszer alatt szeretnénk használni a CD-t.

A digitális tananyaggyártás igen hosszadalmas és költséges tevékenység. Az egyre bővülő oktatási piacon nincs értelme annak, hogy az egyes diszkrét tananyagelemeket újra és újra elkészítsék. Inkább az újrahasznosíthatóságukat, hordozhatóságukat kell megoldani.

A szabványosítás iránti igény általában annak a biztos jele, hogy az adott szegmensben – a mi esetünkben az eLearning területén – tömeges kereslet jelentkezik, s a fokozott kereslet költséghatékony kielégítése érdekében racionalizálni és egységesíteni kell az alkalmazott technológiát a sikeres megoldások mentén. Az eLearning technológiák globalizációja szintén ezt a tendenciát erősíti.

Természetesen fordított helyzetben is – ha rendelkezésünkre állnak bizonyos szabványok, de a kereslet még nem ölt tömeges mértéket – ajánlatos a szabványok alkalmazása, hiszen számos előnyt biztosítanak számunkra.

## A szabványosítás előnyei

A szabványosítás célja az, hogy átjárhatóságot biztosítson a tananyagok számára a különböző eLearning alkalmazások (keretrendszerek, szerzői rendszerek, tároló alkalmazások) között.

### Együttműködési képesség

Az együttműködési képesség a tananyagok szintjén azt jelenti, hogy amennyiben a tananyag teljes egészében megfelel egy adott szabványnak, akkor bármely – az adott szabványt támogató keretrendszerben lejátszható.

Az alkalmazások szintjén egyrészt beszélünk arról, hogy a különböző tananyaggyártók által készített tartalomcsomagok futtathatók ugyanabban a tanulásirányítási keretrendszerben (LMS) kihasználva a tartalomban definiált lehetőségeket. A másik vonatkozása az együttműködési képességnek, hogy a keretrendszer képes együttműködni más alkalmazásokkal. Ez jelentheti a vállalat pénzügyi-, HR alkalmazását, vagy ERP rendszerét; a felsőoktatásban pedig a különböző adminisztratív alkalmazásokat (Neptun, ETR, stb.).

### Újrahasznosíthatóság

Az újrahasznosíthatóságot az teremti meg, ha a tananyagot tananyagelemekből (learning object) építjük fel. Amennyiben a tananyagelemek teljesítik a rájuk vonatkozó megszorításokat (lásd: LO) akkor akár ugyanabban a tananyagban, akár más tananyagokban tetszőlegesen felhasználhatók különböző kontextusban.

### Tartósság

Elsősorban technikai szempontból érdekes a tananyag tartóssága. A cél az, hogy a tananyagelemek, s ez által a tananyag egésze minél tovább megőrizze kompatibilitását a különböző alkalmazások, keretrendszerek egymást követő verzióival.

Az elemi tananyagegységeket olyan formátumban kell tárolni, amely biztosítja a platformfüggetlenséget, vagy minden mérvadó platform támogatja az adott formátumot. Az elemek azonosítását szolgáló metaadatokat pedig el kell különíteni a tananyagelemektől.

### Testre szabhatóság

Ma már a technológia lehetőséget nyújt arra, hogy a tananyagelemeket az egyéni szükségletekhez, képességhez, illetve ízléshez igazítva szabadon csoportosíthassuk, s tömegesen személyre szabhassuk a tananyagot. Ez csupán a megfelelő alkalmazás kiválasztásának kérdése.

### Elérhetőség, kereshetőség

A tananyag flexibilis alakításának lehetősége akkor adott, ha mindig megtalálom a megfelelő tananyagelemet. Ez egyrészt szükségessé teszi a tananyagelemek egyértelmű azonosítását, indexelését, másrészt kategorizálását, osztályozását.

Ezért mind a tananyagelemeket, mind az azokból építkező struktúrákat kísérő adatokkal ún. metaadatokkal látjuk el. A struktúra különböző szintjein ezeknek a metaadatoknak más és más a funkcionalitásuk. Magasabb szinten ezek az információk teszik lehetővé, hogy a tanuló a neki leginkább megfelelő tananyagot válassza. Alacsonyabb szinten a tananyagfejlesztők munkáját könnyíti meg, hiszen rendkívüli módon felgyorsíthatja a

tananyagfejlesztés folyamatát, ha a különböző szolgáltatók által gyártott tananyagelemekhez hozzáférhetünk.

Természetesen ez azt feltételezi, hogy a tananyagelemeket nem kurzus formájában (vagy nem csak kurzus formájában) tároljuk, hanem elemi szinten.

Kiemelkedően fontos az egységes „metázás” akkor, ha közkincset alkotó nemzeti tananyagelem-adatbázist építünk.

### **Költségmegtakarítás**

A szabványok alkalmazásának következtében akár egy adott szervezeten belül, vagy akár különböző szervezetek között lecsökkenhet a tananyag előállításának ideje, valamint egy kritikus tömegű tananyagelem-mennyiség elérése után pedig radikálisan csökkenhetnek az előállítás költségei.

Míg a hagyományos osztálytermi képzés költségei a létszám növekedésével arányosan növekednek, a színvonalas eLearning képzések esetében a ráfordítás és a létszám viszonya fordított arányosságot mutat.

## A szabványosítás főbb területei

Nyilvánvaló, hogy a szabványosítás különböző részterületei között szoros összefüggés áll fenn. Az, hogy a tananyagelemekből hogyan építke fel egy konkrét tananyagot, kurzust, visszahat a metaadat modellre és befolyásolja a tanulói tevékenységek követését és az ezzel kapcsolatos adatmodellt.

Másrészt, mint ahogy az oktatás a valós élet részeként ezernyi szállal kötődik az élet különböző területéhez, ugyanúgy az oktatási szoftverek is összefüggésben állhatnak/állnak szoftveres környezetükkel. Így újabb és újabb szempontok merülnek fel, hogy az eLearning rendszerek ne csak pusztán önmagukban véve legyenek egységesek és hatékonyak, hanem feleljenek meg környezetük elvárásainak is.

### Metaadat modell

A metaadatoktól azt várjuk, hogy legyenek alkalmasak a tananyag különböző szintjeinek az egyértelmű leírására. Legyenek kellően általánosak, hogy minél szélesebb körben be tudják tölteni szerepüket, ugyanakkor a kellő általánosság mellett legyenek specifikusak is, hogy ki tudják elégíteni az oktatás sajátosságaiból eredő igényeket.

Mint korábban már szó volt róla, a metaadatok fő célja, hogy azonosítsák és leírják a tananyagelemeket. Ez egyrészt az egyed azonosítását jelenti, másrészt tartalmának, jelentésének az azonosítását. Ezen túl számos szempont felmerül:

- Milyen **összefüggésben** áll más tananyagelemekkel;
- Mivel minden tananyagelem egy állomány, ez az állomány milyen **technikai jellemzőkkel** bír, s milyen követelményeket támaszt a lejátszóval szemben;
- A tananyagot szánjuk valakinek. Van egy célcsoportja. Ugyanakkor felhasználásának számos pedagógiai jellemzője van, s ezeket az **oktatási információkat** is hordoznia kell;
- A tananyagot valakinek el kellett készíteni, adott esetben egy megrendelő számára. Szükséges a **szerzői jogi** állását jelezni;
- Egyáltalán, egy tananyagelemnek mi a **státusza** a tananyagban (verzió, érvényesség, stb.).

A tananyagoknak, illetve a tananyagelemeknek a leírása, osztályozása régi igény, és megvannak a hagyományai. A számítógépes oktatási anyagok elterjedésével egy időben a világ számos pontján alkottak leíró rendszereket, de ezek egymástól elszigetelten jelentek meg, esetlegesek voltak, s túlzottan magukon viselték a helyi sajátosságokat.

Manapság az eLearning tananyagok metaadatokkal való ellátására két rendszer terjedt el. A korábbi a „dublin core”, amelyet azzal a céllal hoztak létre, hogy segítségével megkönnyítsék az adatok keresését. Tehát nem kifejezetten tananyagelemekhez fejlesztették. A másik a LOM (Learning Object Metadata), melyet viszont épp ilyen céllal, az eLearning tananyagelemek és struktúrák leírására hozták létre.

### Tartalom modell

A tananyagelemek önmagukban nem tanulhatóak. Ahhoz, hogy elnyerjék értelmüket, fel kell építeni a tananyag, kurzus **struktúráját**. Mivel az építőkövek atomizálva vannak jelen, ez a struktúra-leírás egy újabb metaállományban történik. Ez az állomány határozza meg a tananyagelemek szerveződését leckékké, fejezetekké, témákká, kurzussá, tanfolyammá.

A szegmentálás mértéke, illetve a tananyag mérete, strukturáltsága az adott szervezet hagyományaitól, szabályaitól és a tananyag jellegétől függ. Az, hogy a tananyag hierarchiába szervezett egységeit hogy hívják, milyen fogalmakat használnak azonosításukra, szintén a szervezetre jellemző.

Ez a metaállomány a struktúra leírása mellett a lejátszó számára meghatározza, hogy fizikailag hol találhatóak a tananyagelemek.

A tananyagelemeket, a tananyag struktúráját leíró metaállományt, valamint a tananyag szabványosságát jelképező validációs állományokat szintén szabványos módon tömörítik, becsomagolják. Ezért beszélünk tartalomsomagról, tartalomsomagolásról (Content Packaging). Ezek a tartalomsomagok a szabványosságon túl már eleget tesznek a hordozhatóság kritériumának.

## **Sorrend és navigáció**

Bár a tartalomsomag metaállománya maga is meghatároz egyfajta sorrendiséget, amely elvileg a szerző által ideálisnak tartott tananyag-bejárás, tananyag-feldolgozási sorrendet reprezentálja, a sorrend definiálásának szabályai külön szabványajánlásban láttak napvilágot.

Az elmúlt néhány évben egyre nagyobb hangsúlyt kapnak a pedagógiai szempontok, így ez az indítvány arra kíván példát mutatni, hogy hogyan lehet alternatív tananyag-feldolgozási utakat definiálni azonos fizikai sorrend mellett.

Ennek leírása nem újabb állományba kerül, hanem a tananyag struktúráját leíró metaállományt kell módosítani (a jelenlegi gyakorlat szerint). Ez a megoldás kifinomultabb eszközöket ad a kezünkbe. Lehetőséget ad a szerzőnek, hogy egyértelműsítse a tananyag bejárásának módját a célcsoport számára leginkább megfelelő módon, s a vele adott szabályrendszer segítségével ezt a szabványt támogató keretrendszer be is tudja tartatni.

## **Futtatási környezet**

A tananyagot egy LMS kézbesíti a tanuló számára. Ahhoz, hogy erre képes legyen, „értenie kell” a tartalomsomag tananyagstruktúrát leíró metaállomány formátumát. Így lehetőség van arra, hogy a különböző szerzők által készített tartalom lejátszható legyen a keretrendszerben.

A lejátszáson túl az LMS kommunikációt (adatcserét) folytat a kézbesített tananyaggal. Naplózza a felhasználó tevékenységeit, valamint gyűjti a tanulók teljesítményadatait. Ez igen fontos tevékenység, hiszen e nélkül elégséges lenne egy közönséges webszerver a tananyag kézbesítésére. A felhasználói tevékenységek nyomon követésének számos oka van. Azon túl, hogy meg akarunk győződni arról, hogy a tanuló valóban feldolgozta a tananyagot, azaz valóban tanult, a követés során képződő adathalmaz rávilágíthat arra, hogy a tananyag mely része sikeres, s melyek a problémás pontok, amelyeket át kell dolgoznunk.

Az eLearning rendszerek dinamikus jellegűek. Az adatkövetés ad lehetőséget arra, hogy adott esetben a tutor közbeléphessen, ha valaki nem boldogul egy tananyagrészzel, vagy hibásan értelmezi.

Vagy képezheti a jogosultságkezelés alapját (mennyi idő van előírva a tanulónak egy tananyagegység elsajátítására).

Mindehhez két területet kell szabványosítani a tartalomcsomagok kezelésén túl. Meg kell határozni az **adatmodellt**, hogy melyek azok az adatok, amelyeket minden szabványos LMS rendszernek kezelni kell. Az adatmodell azt írja le, hogy tanulók mely tevékenységeiről szükséges adatot gyűjteni, hogyan hívják ezeket az adatokat, s milyen formátumúaknak kell lenniük (típus, értékkészlet).

A másik az **eljárásmodell** szabványosítása. Ennek azt kell leírnia, hogy milyen tevékenységeket kell elvégezni ahhoz, hogy hozzáférjünk a szükséges adatokhoz, s ezeket hogyan hívjuk. A hogyan kérdése a fejlesztő dolga.

Ezek azok a területek, amelyek a leglényegesebbek a szabványosítási folyamatban. Alapvetően mindegyik a működőképességet szolgálja, azt segíti elő, hogy megalkothatók legyenek szabványos eLearning rendszerek. A szabványosítás viszont számos más területre is kiterjed. Néhány példa erre:

- Tanulói információs csomag specifikációja;
- Kérdés és teszt együttműködési képesség specifikációja;
- Digitális tananyagtárházak specifikációja;
- Vállalati rendszerek együttműködési képességének specifikációja;
- Az oktatási folyamat tervezésének specifikációja;
- stb.

Az, hogy e specifikációkról csak címszavakban tettem említést, nem azt jelenti, hogy jelentőségük, vagy elfogadottságuk kisebb lenne. E szabványok többségét az adott terület fejlesztői figyelembe veszik, s beépítik alkalmazásaikba, viszont az előzőekben említett területek sokkal előrébb járnak a szabványosítási folyamatban, kiforrottabbak, így alkalmazásuk is.

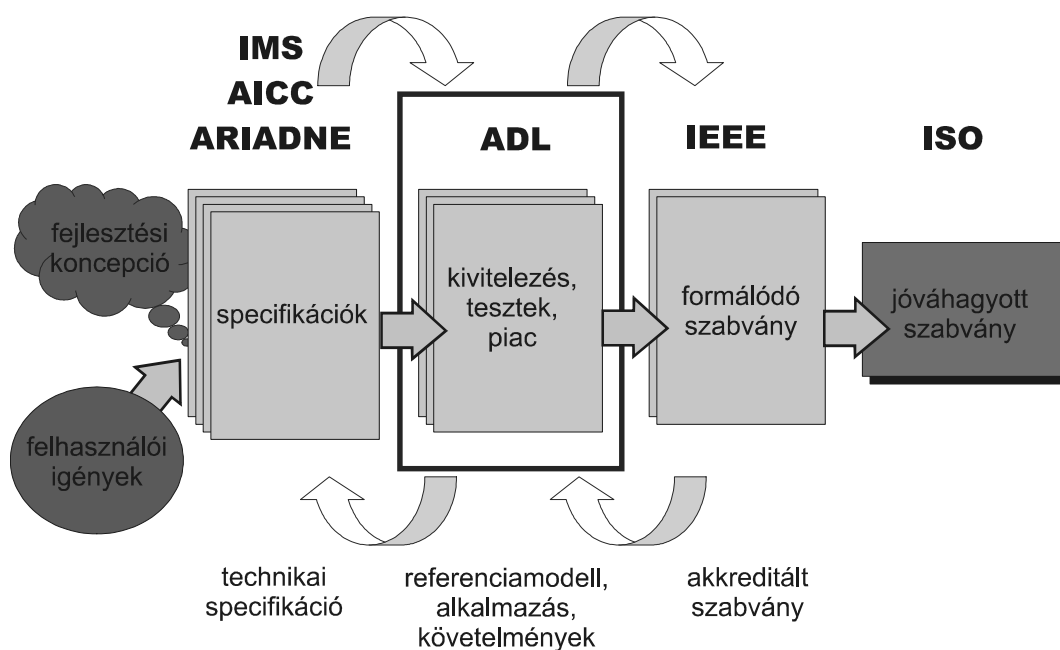
## A szabványosítás folyamata

Egy megoldás szabványként történő elfogadása igen hosszú folyamat, s a potenciális alkalmazók konszenzusán kell, hogy alakuljon. A folyamat a szükségletek felméréséből indul ki, s bizonyos fejlesztési koncepciók mentén vázlatos specifikációk megfogalmazásához vezet. Ezeket nyilvánosságra hozzák, hogy minél szélesebb körben megvitathassák, és újabb szempontokkal bővíthessék, vagy éppen fordítva egyszerűsítsék a megalkotott modellt.

A végleges specifikációk megalkotásával kezdetét veheti a gyakorlatba való átültetés folyamata. E folyamatnak nem csak laboratóriumok, független szervezetek, hanem sok esetben piaci résztvevők is részesei. A szakma vezető vállalatai aktívan részt vesznek a specifikációk formálásában, és élen járnak azok implementálásában.

Sokszor már ebben a szakaszban kiderül, hogy melyek azok az ajánlások, vagy az egyes ajánlásoknak melyek azok a részei, amit piac befogad. Ilyen például az IMS számos specifikációja, illetve az ADL referencia modellje a SCORM.

Ha egy specifikációról bebizonyosodik, hogy a gyakorlatban is jól működik, továbbküldik egy szabványosítási testületnek, hogy kívülállóként tesztelje, véleményezze azt. Az egyeztetések, finomítások folyamata még ilyenkor is – akár évekig is – elhúzódhat. Végül az akkreditált szabványt egy globális szabványosítási testület elé utalják (pl.: ISO), hogy nemzetközileg jóváhagyott szabvány váljék belőle.



**2. ábra A szabványosítási folyamat**

Jelenleg az eLearning területén egyetlen ilyen –nemzetközileg elismert szabvány létezik, a LOM (IEEE 1484.12.1 2002. július), de több ajánlás már az akkreditálás szakaszában van.

Egy már jóváhagyott szabvány alapvetően két típusba sorolható. *De jure* szabványról beszélünk akkor, amikor egy szabvány törvényileg elfogadott, és *de facto* szabványról beszélünk, ha az adott szabvány (vagy ajánlás) törvényileg még nem elfogadott, de a gyakorlatban jól bevált, s szakmai elfogadottsága széles körű.

Ilyen *de facto* (kvázi) szabványnak tekinthető a SCORM, amely az 1.2-es verzió megjelenése óta robbanásszerű gyorsasággal terjedt el világszerte.

## Szervezetek

Az eLearning szabványok megalkotása körül számos szervezet serénykedik. Ezeket minisztériumok, kormányzati szervek, szövetségek és egyesületek, alapítványok, piaci szereplők, egyéb nemzeti és nemzetközi szervezetek alapították, vagy szponzorálják. Noha lehetséges lenne számba venni őket, s ez igen terjedelmes adatbázist alkotna, mégsem szükséges.

Mára egyértelműen kikristályosodott, hogy melyek azok a szervezetek, amelyek meghatározóak az eLearning szabványok terén. Ugyanakkor több szervezet elévülhetetlen érdemeket szerzett a vezető szervezetek ajánlásainak kimunkálásában. Egyáltalán, az eLearning területén komoly együttműködés mutatkozik a különböző szervezetek között. Bevett gyakorlat, hogy tagokat delegálnak egymás munkacsoportjaiba, akik nemcsak az elvek alakításában vesznek részt, hanem az ajánlások gyakorlati alkalmazhatóságának megalapozásában is részt vesznek.

Lássuk a szabványosítási folyamatot befolyásoló fontosabb szervezeteket abc sorrendben:

### ADL

A szervezet neve: **Advanced Distributed Learning**

Alapításának éve: **1997.**

Webcíme: <http://www.adlnet.org>

A szervezetet az Amerikai Egyesült Államok Védelmi Minisztériuma és a Fehér Ház Tudományos és Műszaki Irodája, valamint a Munkaügyi Minisztérium hozta létre.

Az ADL a szabványalkotás folyamatában köztes szerepet tölt be. Nem törekszik önálló, új kezdeményezések megalkotására, hanem a különböző szabványajánló szervezetek (IMS, AICC, ARIADNE, IEEE) által kidolgozott specifikációk, és ajánlások alapján alkotja meg referencia modelljét, a SCORM-ot (Sharable Content Object Reference Model). Az ajánlásokat egységes rendszerbe foglalja. A SCORM gyakorlatban való beválása alapjául szolgál a végleges szabványajánlások megfogalmazásához.

Az ADL igen szerteágazó és gazdag kapcsolatrendszerrel büszkélkedhet. Együttműködési stratégiája az ADL Co-Lab hálózatra épül, amelyet a Védelmi Minisztérium 1999-ben hozott létre. Az ADL szorosabb vagy lazább partneri kapcsolatot több területtel ápol.

A kormányzati szférából elsősorban a katonai szervezetekkel tart fenn szorosabb kapcsolatot (18 szervezet). Valamint az Egyesült Államok szövetségi rendszerén keresztül a NATO tagországok védelmi minisztériumai, illetve az általuk delegált intézmények és szervezetek is részesei a kapcsolatnak. Ez a kapcsolatrendszer értelemszerűen zárt.

Ennél nyitottabb kapcsolatrendszert tesz lehetővé az akadémiai szektor számára meghirdetett együttműködő laboratóriumi hálózat. A fontosabb központokkal (Alexandria, Orlando, Madison) együtt 54 felsőoktatási intézmény, illetve konzorcium tagja a hálózatnak. A hálózat különböző tagintézményei közötti munkamegosztást az Alexandriai Co-Lab koordinálja. A hálózathoz az együttműködési megállapodás elfogadásával lehet csatlakozni egy akkreditációs folyamat után.

A vállalati szektorral, mint a SCORM alkalmazói körével úgynevezett „alkalmazói” kapcsolatot tart fenn az ADL. Jelenleg 115 céggel szerepel a közreműködők listáján. Az adaptálói körhöz rövid regisztrációs procedúra után bárki csatlakozhat. Az ADL mind

az akadémiai, mind a vállalati szektorba tartozó szervezetek számára biztosít szabadon hozzáférhető támogatást, szolgáltatásokat. Az ADL nemzetközi kapcsolatai is bővülnek. A Co-Lab hálózatnak már vannak tagjai Kanadából és az Egyesült Királyságból.

## **AICC**

A szervezet neve: **Aviation Industry CBT (Computer-Based Training) Committee**

Alapításának éve: **1988.**

Webcíme: <http://www.aicc.org>

Mint a legkorábban alapított szervezet az AICC úttörő szerepet töltött be a szabványosítási folyamatban. 1988-as alapítása óta meghatározó szerepet játszott az oktatási technológiák és az eLearning-alkalmazások fejlesztésének egységesítésében.

Az AICC ajánlásokat készít a CBT tananyagok fejlesztéséhez. A szervezet fő céljai:

- CBT rendszerek gazdaságos és effektív kidolgozása;
- A CMI (Computer Managed Instruction) rendszerek kompatibilitását biztosító irányelvek kidolgozása;
- Fórum biztosítása a CBT és a hasonló technológiák számára.

Eleinte a repülési ágazat számára készítettek javaslatokat. Tevékenysége, ajánlásai megalapozták az eLearning alkalmazások működésének alapelveit. Specifikációi illetve ajánlásai régóta *de facto* szabványként működnek a piacon.

## **ALIC**

A szervezet neve: **Advanced Learning Infrastructure Consortium**

Alapításának éve: **2000.**

Webcíme: <http://www.alic.gr.jp/eng/>

A konzorcium a prominens japán információs technológiai vállalatokat tömöríti. A szervezet tevélegesen részt vett több szervezet ajánlásának kidolgozásában és implementálásában.

## **ARIADNE**

A szervezet neve: **Alliance of Remote Instructional Authoring & Distribution Networks for Europe**

Alapításának éve: **1997.**

Webcíme: <http://www.ariadne-eu.org>

Az európai szervezet jelentősége egyrészt a különböző szervezetek (IMS, LTSC, ADL) ajánlásainak kidolgozásához nyújtott támogatásában van. Másrészt felsőoktatásban és a nagyvállalati szférában kifejtett – az eLearning megoldások, illetve általában véve a CBT megoldások alkalmazási elveket – népszerűsítő tevékenysége jelentős. A projekt olyan eszközök, és módszerek fejlesztésére összpontosít, amelyek számítógépes pedagógiai elemek és telematikával támogatott oktatási tananyagok előállítását, kezelését és újrahasznosítását szolgálják.

## **DCMI**

A szervezet neve: **Dublin Core Meta-data Initiative**

Alapításának éve: **1995.**

Webcíme: <http://dublincore.org>

A nemzetközi szervezetet az Ohio-beli Dublin-ban alapították. A szervezet fő célja, hogy a metaadat ajánlása segítségével megkönnyítsék az adatok keresését a hálózaton. Célkitűzésük...

- Az egyszerű előállítás és karbantartás;
- Széles körben érthető szemantika;
- Nemzetközi hatáskör (23 nyelven);
- Bővíthetőség.

## EdNA

A szervezet neve: **Education Network Australia**

Alapításának éve: **1995.**

Webcíme: <http://www.edna.edu.au/edna/go/pid/1>

Az ausztrál szervezetet az Oktatási Minisztérium alapította. Ausztrália – az ország adottságaiból kifolyóan – gazdag hagyománnyal rendelkezik a távoktatás területén. Az EdNA feladata az, hogy szerteágazó támogatást nyújtson az internet alapú oktatáshoz. Mint nemzeti szervezet, segítséget nyújt az oktatási piac különböző résztvevői számára többek között az alkalmazott technológiák é szabványok terén. Az EdNA-nak saját dublin core alapú metaadat szabványt fejlesztett, amely a hagyományos 15 elem mellett további nyolc saját elemet tartalmaz.

Az EdNA az IMS egyik fő együttműködő partnere.

## IEEE LTSC

A szervezet neve: **Institute of Electrical and Electronics Engineers Learning Technology Standards Committee**

Alapításának éve: **1995.**

Webcíme: <http://ltsc.ieee.org>

A non-profit szervezet az IEEE Computer Society Standards Activity Board közreműködésével jött létre azzal a céllal, hogy technikai szabványokat, ajánlásokat és irányvonalakat fejlesszenek ki programokhoz, eszközökhöz, technológiákhoz, amelyek az internetes oktatási anyagok, illetve rendszerek kidolgozását szolgálják. Biztosítják az alkalmazások közötti kompatibilitást, valamint a szabványok egy részét emeljék nemzetközi szintre.

Az LTSC-n belül közel 20 munkacsoportban folyik szabványalkotó munka. A munkacsoportok tevékenysége öt fő részterületen folyik. Ezek érintik egy átfogó eLearning referenciamodell megalkotását, a tanulói információkkal, profilokkal kapcsolatos, a tartalommal kapcsolatos szabványok és ajánlások megalkotását, valamint a CMI (Computer Managed Instruction) rendszerekre vonatkozó követelmények szabályozását.

Legismertebb fejlesztésük azonban a LOM (Learning Object Metadata) szabvány megalkotása volt.

## IMS

A szervezet neve: **IMS Global Learning Consortium Inc.**

Alapításának éve: **1997.**

Webcíme: <http://www.imsglobal.org>

Az IMS projekt a National Learning Infrastructure Initiative of EDUCASE keretein belül indult útjára. Az IMS a nemzetközi együttműködésre törekedve három centrumot hozott létre:

- **CETIS** az *Open University in Milton Keynes* és a *University of Wales-Bangor* közös irányítása alatt üzemel;
- **IMS Asia Centre** – Szingapúri központtal hozták létre és az *Infocommunication Development Authority* szponzorálja;
- **IMS Australia Centre** – A központot a *Department of Education, and Youth Affairs* szponzorálja.

A központok az IMS ajánlásainak terjesztésében működnek közre. Szervezik a társadalmi kapcsolatokat, felmérik a szükségleteket, begyűjtik az alkalmazott specifikációkkal kapcsolatos tapasztalatokat, konferenciákat, képzéseket, műhelyeket szerveznek.

Az IMS arra törekszik, hogy az elosztott tanulás területén előforduló minden alkalmazástípushoz az alkalmazások együttműködésére vonatkozó specifikációkat dolgozzon ki. Nagy hangsúlyt fektetnek a specifikációk termékekben történő implementálására, s ehhez sokrétű támogatást biztosítanak.

## ISO/IEC JTC1/SC36

A szervezet neve: **International Standardization Organization - Joint Technology Committee 1 Sub Committee 36**

Alapításának éve: **2000.**

Webcíme: <http://jtc1sc36.org/>

A szervezet a számítógép alapú oktatórendszerek működésének, kompatibilitásának és újrafelhasználhatóságának szabványosítását kívánják elősegíteni. Szabványalkotói tevékenysége az eLearning rendszerek minden lényeges területére kiterjednek. Az SC36 szorosan együttműködik más hasonló szervezetekkel annak érdekében, hogy a fejlesztés alatt álló szabványok minél inkább megfeleljenek a felhasználók elvárásainak. Másrészt szintén szoros kapcsolatot tart fenn az ISO egyéb bizottságaival, amelyek az érintőleges technológiák szabványosításában játszanak szerepet (pl.: SC24 – számítógépes grafika és képfeldolgozás).

## O.K.I.

A szervezet neve: **Open Knowledge Initiative**

Alapításának éve: **2001.**

Webcíme: <http://www.okiproject.org>

A szervezet célja az, hogy a különböző alkalmazások (elsősorban eLearning alkalmazások - LMS és ERP) számára olyan kapcsolódási felületet biztosítson, amely radikálisan növeli az alkalmazások közötti együttműködési lehetőséget, a különböző alkalmazások

közötti adatcserét és adatszinkronizálást. Az O.K.I. célja az, hogy egy nyílt specifikációt alkossanak erre a célra. A szervezet által fejlesztett specifikáció az Open Service Interface Definition (OSID). A szervezetet és a projektet a Mellon Alapítvány szponzorálja.

## **PROMETEUS**

A szervezet neve: **European Partnership for a Common Approach to the Production of eLearning Technologies and Content**

Alapításának éve: **1999.**

Webcíme: <http://prometeus.org>

A PROMETEUS elsősorban támogatói jellegű szervezet. Neve a PROMoting Multimedia access to Education and Training in the European Society kifejezésből származik. A szervezetnek számos különböző jellegű tagja lehet: oktatási hivatalok, vállalatok, szoftver és hardverforgalmazók, kiadók, stb.

A szervezet célja hogy serkentse a digitális multimédia tartalom és szolgáltatások – így az eLearning rendszerek számára fejlesztendő európai, illetve nemzetközi szabványok megalkotását. A PROMETEUS szervezi az érintett szervezetek és döntéshozók közötti párbeszédet és együttműködést.

## **WS-LT**

A szervezet neve: **CEN/ISSS (Information Society Standardization System) WorkShop on Learning Technology**

Alapításának éve: **1999.**

Webcíme:

<http://www.cenorm.be/cenorm/businessdomains/businessdomains/iss/index.asp>

A szervezet az európai szabványügyi hivatalokat fogja össze, melynek tagjai a nemzeti szabványügyi hivatalok. A munkacsoport az oktatási technológiák – így az eLearning – elfogadtatásában tölt be szerepet.

# Szabványok és szabványjavaslatok

## Az AICC kezdeményezései

Az AICC irányelveket, ajánlásokat, szabványajánlásokat és egyéb munkaanyagokat ad ki. Az ajánlások AGR-ek (AICC Guidelines and Recommendation) képviselik az AICC adott területre vonatkozó hivatalos közléseit. Minden AGR azonosítója „AGR”-rel kezdődik, s egy háromjegyű számmal folytatódik (egyéb dokumentumai is hasonló jelölésűek). Ezek az ajánlások néhány oldalas dokumentumok, s az CMI rendszerekkel szemben támasztott követelményektől az ajánlott videó-formátumokon keresztül a tanulásra alkalmas számítógép-konfiguráció definiálásáig mindent átfognak.

A mi szempontunkból a Web-alapú CMI (Computer Managed Instruction – számítógéppel támogatott oktatás) rendszerekre vonatkozó ajánlások az érdekesek. Az AICC CMI001 azonosítóval bocsátja ki a CMI rendszerek együttműködési képességére vonatkozó ajánlását (CMI Guidelines for Interoperability). Ennek legfrissebb verziója a 2004. augusztus 16-án elfogadott 4.0 verzió. Az ajánlást 1993 októbere óta fejlesztik. A CMI001 ajánlás 3.4-es változata képezi az alapját az IEEE LTSC 1484.11.1 számon futó, az adatmodellre vonatkozó és a 1484.11.2 számú, az API hívásokra vonatkozó szabványának.

A CMI001 a következő témakörökre terjed ki:

- **Kommunikációs adatmodell** – ez a rész írja le a tananyag és az LMS rendszer kommunikációja során kötelezendően, illetve opcionálisan követhető adatok listáját és ezek funkcionalitását. A listán 93 lehetséges adatelemet neveznek meg, melyből 15 lényeges (*core*) adat követése a kötelező. Ezek az adatok a tanuló azonosítását, illetve a tanulói tevékenységek követését szolgálják.
- **Tananyag struktúra adatmodellje** – itt a tananyag adminisztratív/információs jellegű adatait, a tulajdonképpeni kurzusadatokat definiálják.
- **Tananyagegységek sorrendje** – a fejezet a tananyag strukturális építkezéséről szól. A kurzusok létrehozására vonatkozó ajánlás, mely a tananyagegységek (Assignable unit) kapcsolódásának módját írja le.
- **Kommunikáció** – három különböző kommunikációs adatmodell kerül definiálásra:
  - **Fájl-alapú** – a hagyományos CD/DVD alapú, vagy lokális hálózatokon futtatott CMI alkalmazások számára a szöveges állományba történő naplózás kritériumai, illetve adatmodellje.
  - **HACP-alapú** – kommunikációs adatmodell web-alapú CMI rendszerek számára, amely az AICC saját (**HTTP AICC Communication Protocol – HACP**) protokolljára épül.
  - **API-alapú** – kommunikációs adatmodell web-alapú CMI rendszerek számára, amely JavaScript alapú API rendszert definiál.

- **Kurzus struktúra definíció** – a különböző adatmodellek alapján a kurzus különböző jellemzőit más-más szöveges állományba kell rögzíteni. A fejezet ezen állományok szintaktikáját írja le.

Az AICC szabványalkotó tevékenysége mellett minősítéssel is foglalkozik. A minősítési rendszer kétszintű. A „*Designed to AICC Guidelines*” minősítést azok a cégek használhatják, amelyek maguk szavatolják, hogy termékük megfelel az ajánlásoknak.

Azonban mivel ez a megfelelés az esetek jelentős részében csak az ajánlások egy részének figyelembe vételét jelenti, nem biztosít együttműködést, kompatibilitást a különböző alkalmazások közt. Az AICC honlapjáról ehhez tesztprogramok tölthetők le.

Az „*AICC Certified*” magasabb szint. Ezt a minősítést azok a termékek kapják meg, amelyeket maga az AICC független tesztlaboratóriumokkal vizsgáltat be.

Hitelesítő tevékenysége a SCORM 1.2-es verziójának megjelenése óta háttérbe szorult. Jelenleg az alacsonyabb szintű minősítésre 2 termék van regisztrálva honlapjukon, míg a szigorúbb („az AICC igazolja”) kategóriában 14 termék található. Minden hitelesített szoftver az AGR-010 ajánlásnak felel meg, amely a web-alapú CMI rendszerekről szól. Korábban mindkét listában több tíz termék szerepelt.

Ez nem azt jelenti, hogy az AICC tevékenysége háttérbe szorulna, hanem azt, hogy a szabványosítási folyamatok előrehaladtával ajánlásai magasabb szintre kerültek

## Az IMS kezdeményezései

Az IMS az eLearning rendszerek minden lényeges vonatkozására igyekszik választ adni specifikációival. Szerteágazó kapcsolatrendszerének köszönhetően ajánlásait számos vállalkozás implementálja. Több területen is specifikációival kizárólagosságot élvez. Az IMS által fejlesztett specifikációk gyakorlati realizációja XML alapokon nyugszik.

Nagy hangsúlyt fektet specifikációi gyakorlatba való átültetésére, s bár az IMS nem folytat hitelesítési tevékenységet, az ajánlások implementálására vállalkozókat igyekszik maximálisan támogatni.

Ajánlásonként három részből álló dokumentációt bocsát ki. Az első kötet (Information model) ismerteti az adott ajánlást részletes leírás formájában. A második kötet (XML Binding) az XML-alapú megvalósításhoz szükséges ismereteket tartalmazza példák tükrében. Ennek alapján láthatnak hozzá az ajánlás bevezetésére vállalkozó fejlesztők a rendszer implementálásához. A harmadik kötet (Best Practice and Implementation Guide) tanácsokkal és a „legjobb gyakorlatra” való hivatkozásokkal segítik a fejlesztőket.

Az IMS a dokumentáción túl sémákat, validációs állományokat és konkrét példákat biztosít a fejlesztők rendelkezésére, amelyek segítségével lehetőség nyílik az azonnali kipróbálásra.

Legjelentősebb specifikációi a következők:

### Learning Resource Meta-data (LRM)

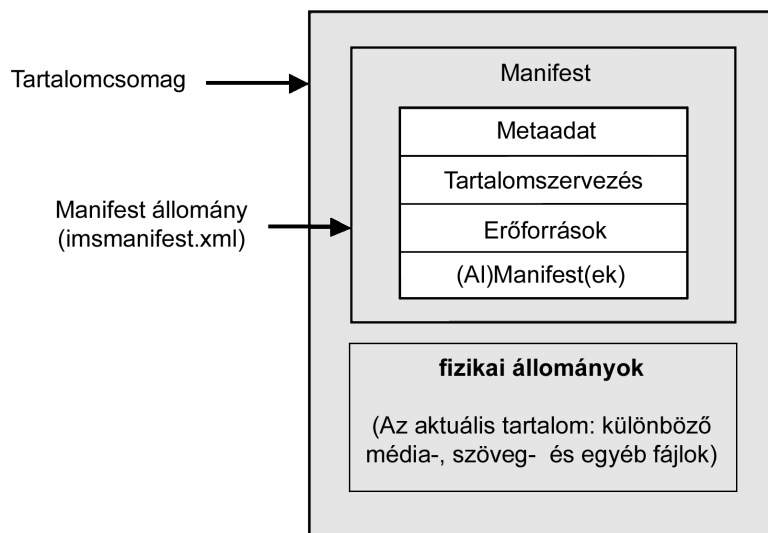
Az IMS a tananyagelemek és tananyagegységek metaadatokkal való felruházására az IEEE LTSC LOM szabványát használja, de nem hagyja magukra azokat sem, akik a dublin core szerint „metáznak”.

Az IMS LRM specifikációjának aktuális verziója: 1.3.

## Content Packaging (CP) - tartalomcsomag

Az IMS egyik legjelentősebb újítása a tartalom (tananyag) struktúrájának és hordozhatóságának leírására. A tartalomcsomag a kulcs a LMS számára a tananyag lejátszásához. Továbbá a tartalomcsomagban kerül leírásra a tananyag szerkezete, felépítése. A tartalomcsomag tulajdonképpen egy tömörített állomány, amely egyrészt tartalmazza a tananyagelemeket reprezentáló fizikai állományokat, valamint egy metaállományt, amely leírja a tananyag szerveződését.

Ez az úgynevezett Manifest állomány. Az LMS a csomag importálásakor ezt az állományt keresi. Az állomány neve kötött: imsmanifest.xml.



3. ábra A tartalomcsomag felépítése

A tartalomcsomag bevezetése forradalmasította az eLearning tananyagok forgalmazhatóságát. Az IMS CP specifikációját alkalmazza az ADL a SCORM referenciamodellben, s a Microsoft is erre alapoz az LRN név alatt futó „házi” szabványában. A mérvadó eLearning alkalmazások általában mind az IMS, mind a SCORM 1.2-es formátumú tartalomcsomagok befogadására fel vannak készítve.

Az IMS CP specifikációjának aktuális verziója: 1.1.4.

## Question and Test Interoperability (QTI)

Az IMS kérdés és teszt együttműködési modellje egyedülállónak tekinthető az eLearning szabványok területén. A tesztadatbázisok piaca egy jelentős szegmensét alkotja az oktatási piacnak. Azonban az alkalmazások jelentős része a mai napig a teszteket és a tesztadatbázisokat saját egyedi megoldásai alapján valósítja meg. A piacon ugyan bevett gyakorlat néhány piacvezető alkalmazás (WebCT, BlackBoard) tesztformátumának exportálása/importálása, de az igazi hordozhatóságra a QTI a megoldás. Az IMS a kérdések (kérdéstípusok) leírását egy XML állományban teszi meg.

Ma már egyre több alkalmazás épít erre a formátumra. Sokan úgy gondolják, hogy a specifikációt a SCORM a közeljövőben befogadja. A séma 22-féle kérdéstípus definiálására képes. A mai alkalmazások többsége kérdéstípusok tekintetében tíz alatt marad, vagy csak a legalapvetőbb kérdéstípusok használatára ad lehetőséget.

Az IMS QTI specifikációjának aktuális verziója: 2.0.

## Learner Information Package (LIP)

A tanulói információs csomag specifikációt a tanulói mobilitás támogatása érdekében hozták létre. Lényege az, hogy az együttműködő alkalmazások között lehetőség nyíljon a tanulói adatok cseréjére. Ezek az adatok nem pusztán az azonosítást szolgáló információk, hanem mintegy a tanuló „előéletét”, „pályafutását” is tartalmazzák. Tehát adott esetben a tanuló intézményváltáskor nem kezd „tisztalappal”, új intézménye számára nem „fekete doboz”, hanem rendelkezésre állnak a szükséges adatok. A személyes adatok kezelése azonban több problémát vetett fel (nem egységes törvényi szabályozás, biztonság, stb.), amely nehezíti az ajánlás terjedését.

Az IMS LIP specifikációjának aktuális verziója: 1.0.1.

## Simple Sequencing (SS)

Az ajánlás a tananyag különböző komponenseinek (tananyagelem, blokk, lecke, fejezet, stb.) sorba rendezésére, s a sorrend egzakt megjelenítéséhez ad eszközt. Segítségével a szerző a tananyagban meghatározhatja azt, hogy az LMS szerver hogyan jelenítse meg a tananyagot, s függőségek definiálásával a tanulót rá lehet kényszeríteni, hogy valóban csak az előírt sorrendben, illetve szabályok mentén dolgozhassa fel a tananyagot.

A specifikációt az ADL a SCORM 2004-ben implementálta, s az IEEE is számos workshopot rendezett az elmúlt, s az idei év során az SS alkalmazásának elterjesztése érdekében.

Az IMS SS specifikációjának aktuális verziója: 1.0.

## Enterprise & Enterprise Service

A specifikáció az LMS és az ERP rendszerek közötti együttműködést alapozza meg. Ehhez az együttműködéshez hoz létre egy információs modellt, melynek keretében több adatobjektumot definiál, s az adatobjektumok közötti adategyeztetés, adatcsere képezi a rendszerek közötti átmenetet. Az ajánlás támogatottsága igen jelentős, sok alkalmazás implementálja azt.

Az IMS Enterprise és Enterprise Service specifikációk aktuális verziója: 1.1, illetve 1.0.

## Learning Design (LD)

Korábban sok kritika illette az eLearning rendszereket, mert egyáltalán nem biztosított, hogy pedagógiai, módszertani szempontból megfelelő színvonalú tananyagokat játszanak le. Noha ez tulajdonképpen továbbra sem garantált, hiszen a digitális tananyagok színvonala, értékessége továbbra is a szerzőtől függ, illetve attól, hogy a tananyag teljesíti-e a képernyőről való tanulással szemben állított ergonómiai és módszertani követelményeket – e specifikáció ezen a helyzeten kíván javítani.

Az ajánlás logikája az, hogy igenis lehetséges meghatározni olyan módszereket, lépéseket, amelyek közelebb visznek minket a pedagógiai szempontból helyes tananyag megalkotásához. A szabványjavaslat ennek terminológiáját és eszközszerkezetét határozza meg. Hátterében a konstruktivista pedagógia elvrendszere húzódik meg.

A specifikáció 2002.-es megjelenése óta komoly népszerűségnek örvend elsősorban felsőoktatási körökben, s egyre több – elsősorban nyílt forráskódú – alkalmazás, tananyagszerkesztő születik hozzá.

Az IMS LD specifikációjának aktuális verziója: 1.0.

## **Digital Repositories (DR)**

A digitális tananyagtárházak specifikációja a nagy mennyiségű tananyag tárolásának, keresésének, szolgáltatásának témakörével foglalkozik. Mint ilyen komoly szerepet tölthet be LCMS alkalmazások fejlesztésében. Elsősorban a tárolás elveivel, technológiájával a tárolt „erőforrások” elérhetőségével, felkutatásával, kézbesítésével kapcsolatos problémákra keres megoldást.

Az IMS DR specifikációjának aktuális verziója: 1.0.

Az IMS a fentiekén túl – talán a teljesség igényével – még további specifikációkat dolgoz ki és fejleszt. Ezek a következők:

- ePortfolio;
- Sharable State Persistence;
- General Web Service;
- Resource List Interoperability;
- Access For All Meta-data;
- Vocabulary Definition Exchange;
- Reusable Definition of Competency or Education Objective.

## LOM

A LOM (Learning Object Metadata – tanulási objektum metaadat) IEEE 1484.12.1 számon jelenleg az egyetlen hivatalosan bejegyzett *de jure* nemzetközi eLearning szabvány. Feladata, hogy biztosítsa a tananyagelemeknek a működtető rendszerektől való függetlenségét. Fogadtatása, elterjedtsége széleskörű. A LOM-ra építi metaadatkezelését az IMS és a SCORM is.

A metaadatok jelentőségéről korábban már volt szó. Igazi jelentőségüket akkor nyerik el, ha a tananyagok tárolása elemi, atomi szinten valósul meg. Ekkor azonban – mint láttuk – számos előnyt biztosítanak a digitális tananyag-előállítók számára. Valójában a metaadatok segítségével számos szempontból jól kereshető katalógusokat tudunk létrehozni. Ez az elengedhetetlen alapja például a digitális tananyag-tárházak létrehozásának.

### A LOM szerkezete

A metaadatok az adatok – jelen esetben a tananyagelemek – leírására szolgálnak. Egy adott tananyagelem jellemzőit tároljuk metaadatként. A LOM ezeket a jellemzőket csoportokba sorolja. Egy-egy csoport egy-egy szerepkörnek megfelelő megközelítést reprezentál, hiszen egy tananyagelemnek más és más vonatkozásai érdeklik a tanárt, az adminisztrátort, a szerzőt, a forgalmazót, stb. A szabvány kilenc csoportot határoz meg. Ezek a következők:

1. **Általános** – a tananyagelem elsődleges azonosítására és általános leírására szolgáló szakasz. Az azonosítókon (ID, cím) túl rövid összefoglalót tartalmaz, azonosítja a tananyag nyelvét.
2. **Életciklus** – a tananyagelem státuszát, előéletét, verzióját tartalmazza, továbbá a közreműködő személyek, szervezetek adatait (név, cím, stb.) és szerepkörüket (szerző, engedélyező, stb.). Közvetve itt szabályozhatom a tananyagelem láthatóságát.
3. **Meta-meatadatok** – ez a rész magáról a metaadatokról ad információt. Azonosítja a használt metaadat-sémát.
4. **Technikai adatok** – például a formátum és méret. Itt kerül megnevezésre a tananyagelem elérési útvonala, és a kompatibilitási adatok.
5. **Oktatási adatok** – itt határozhatjuk meg például a célcsoportot (korosztályt, iskolatípust), a tananyag nehézségi fokát, a feldolgozás várható időtartamát, az interaktivitás mértékét.
6. **Tulajdonjogok** – itt kerülnek meghatározásra a felhasználás feltételei.
7. **Kapcsolatok** – ebben a szakaszban jelezhetjük az adott tananyagelem más tananyagelemekkel való kapcsolatát, azok adatainak megadásával.
8. **Kommentárok** – ennek a szakasznak a tananyagelem minősítésében van szerepe. Lehetőséget ad a tananyagelemek független mérvadó szaktekintélyek vagy szervezetek általi véleményezésre.
9. **Besorolás** – itt adhatjuk meg, hogy a tananyagelem – egy a szervezet által kiválasztott – besorolási rendszerben hol foglal helyet (pl.: DDC, ETB Tesaurus, UDC, stb.). Ez felveti annak a kérdését, hogy melyik rendszer a lesz a legalkalmasabb az általam fejlesztett/használt tananyagelemek besorolására. A választás

azonban magában hordozza az inkompatibilitás lehetőségét, hiszen külső forrásra mutat.

A szabvány 69 bejegyzést határoz meg, melyek némelyike kötelezően kitöltendő, mások kitöltése szabadon választható. A szabvány szabályozza, hogy egy-egy bejegyzésből hány darab engedhető meg egy tananyagelemhez, s szabályozza a bejegyzések típusát, s egyes esetekben a felvehető értékeket is előre meghatározza.

## SCORM

A szabvány-együttes teljes neve: **Sharable Content Object Reference Model**

A SCORM napjainkra az egyik legáltalánosabban elfogadott *de facto* eLearning szabványává vált. Nemcsak a piacvezető keretrendszereket-, szerzői- és egyéb eLearning alkalmazásokat gyártók adaptálják termékeikben, hanem a non-profit szféra, valamint a felsőoktatási intézmények fejlesztői is szívesen alkalmazzák. Hogy minek köszönheti népszerűségét, elfogadottságát? Nos, alapvetően a szabvány-együttes integráló jellege dominál. Továbbá sokat nyom a latba a szabvány gazdájának – az ADL-nek – az alapítók, és a szélesebb közösség irányában folytatott tevékenysége és szolgáltatásai.

Bár a SCORM nem terjed ki az eLearning rendszer minden komponensére (hiányzik például a tanulókra vonatkozó információk, vagy a tesztek szabványos kezelése), gyakorlatias mivolta miatt, s hogy egységbe gyúrva kapjuk a különböző ajánlások legjobbjait kedveltté tette az adaptálók körében.

### Hitelesítés

Az ADL több lehetőséget biztosít arra, hogy ellenőrizhessük, illetve hitelesíthessük hogy egy adott termék, vagy tananyag részben vagy egészben megfelel-e az általa kidolgozott szabványoknak. Lehetőség van arra, hogy a fejlesztő maga bizonyosodjon meg terméke megfeleléséről. Erre a célra az ADL honlapjáról mindig letölthető az éppen aktuális támogatott SCORM verzióknak megfelelő tesztszoftver, melynek segítségével önteszteket végezhetünk. Valamint néhány tananyagpéldát is találhatunk a honlapon arra a célra, ha nem rendelkezünk még SCORM kompatibilis tananyaggal alkalmazásunk teszteléséhez.

- SCORM 1.2 – Conformance Test Suite 1.2.7 (Self Test)
- SCORM 2004 – Conformance Test Suite 1.3.3 (Self Test)

A teszter szoftver és a SCORM verziók között egyezés van a fő és az első mellékverzió számában. A tesztszoftvernek a finomításoknak, hibajavításoknak köszönhetően újabb és újabb változatait adják ki. A harmadik szám ezeknek a változásoknak a jelölésére szolgál.

Az ADL jelenleg a SCORM-nak két verziójához biztosít hitelesítési eljárásokat, illetve támogatást. Mivel a jelenleg piacon lévő alkalmazások és tananyagok túlnyomó többsége a SCORM 1.2-es változatát támogatják, továbbra is fenntartják ennek a verzióknak a hitelesítését, valamint az új verzióknak – a SCORM 2004 (SCORM 1.3.1) – való megfelelés hitelesítését. Az öntesztek a következő területeken végezhetőek el:

- LMS (Learning Management System) megfelelési teszt;
- Tartalomcsomag megfelelési teszt;
- SCO (Sharable Content Object) RTE (Run-Time Environment) megfelelési teszt;
- Metaadat megfelelési teszt;
- Manifest megfelelési teszt (csak SCORM 1.3.1).

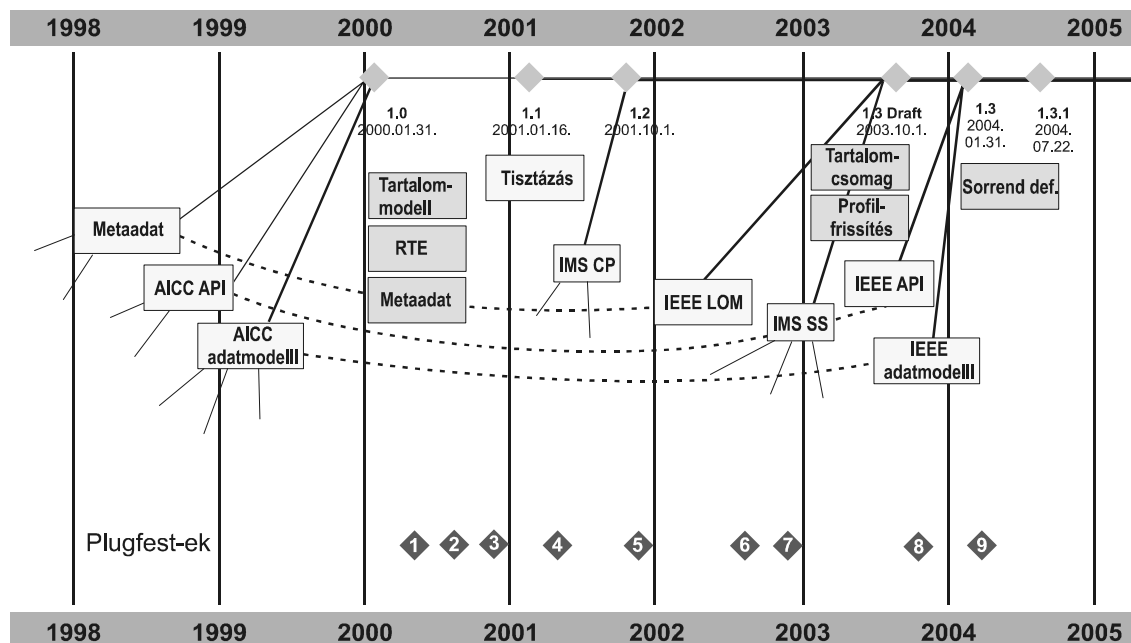
Természetesen az egyes tesztek funkcionalitásukat tekintve jelentős eltéréseket mutatnak a verziók közötti különbségek miatt.

LMS rendszerek esetében a megfeleltetésnek három szintje van: RTE1, RTE2, RTE3. Az első szint esetében a tartalomhalmozási modellnek való megfelelés, a tananyagelemek (SCO, asset) korrekt indítását és kezelését, a korrekt API implementációt, és a kötelező adatmodell elemek támogatását várják el. A második szinten a további adatmodell elemek támogatását, míg a harmadik szinten az összes adatmodell elem támogatását várják el.

Ha viszont valaki termékének hitelesítését hivatalosan is szeretné elismertetni, akkor termékét alá kell vetnie az ADL hitelesítési eljárásának, s át kell adnia valamely Hitelesítési Teszt Centrumnak. A tesztelést általában a Wisconsin Testing Organization végzi. Mind a SCORM 2004-re, mind a SCORM 1.2-es verziójára lehet kezdeményezni a hitelesítést, bár ez utóbbit csak 2005 januárjáig tervezte támogatni az ADL, de a SCORM 2004 lassú terjedése miatt eltekintett ettől.

## A SCORM fejlődése

A SCORM fejlődésvonala jól szemlélteti az eLearning szabványok terén végbemenő letisztulási folyamatot. Az évek során – többek között magának a SCORM-nak köszönhetően – körvonalazódott, hogy melyek azok az ajánlások, amelyek a gyakorlatban is jól beválnak és melyek azok a szervezetek, amelyek a folyamat éllovasaivá váltak.



4. ábra A SCORM fejlődése

Jelenleg a szabványnak két változata, verziója élvez támogatást: a SCORM 1.2 és a SCORM 2004, melynek verziószáma jelenleg 1.3.1. Az 1.2-es változat – mint a 4. ábrán láthatjuk – immáron négy éve használatos. Igen elterjedt mind a tananyagfejlesztők, mind a keretrendszereket és szerzői rendszereket gyártók körében. Úgy is fogalmazhatnánk, hogy kritikus tömeget képvisel a piacon lévő alkalmazások és tananyagok tekintetében. Ez idő alatt azonban újabb elvárások fogalmazódtak meg a szabvánnyal szemben, melyek egy részét az új verzióban realizálták.

A korábbi verziók közül említést érdemel még az 1.1-es változat. Ez volt az első igazán sikeres verzió. Sok tananyagot fejlesztettek e szerint, s néhány LMS rendszer a mai napig támogatja. Jelentős különbség az újabb verziókhöz képest a tananyag struktúrájának leírásában mutatkozik. Ez a változat még nem az IMS tartalomcsomagolási elveit követ-

te, hanem az AICC CMI001 3.4-es verziójának tartalomstruktúra formátumát (CSF – Content Structure Format). A tananyag felépítését egy ún. CSF XML állományban írták le, az említett AICC ajánlás terminológiáját használva.

## A SCORM 1.2

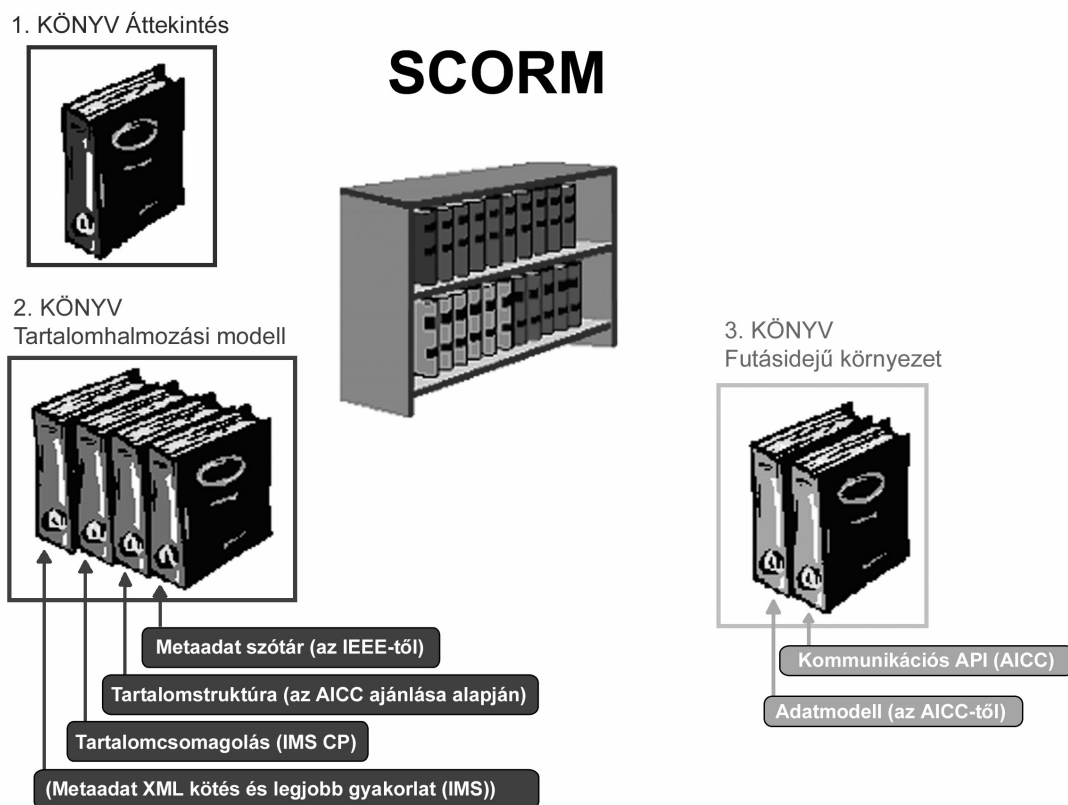
### A SCORM felépítése

Az ADL a SCORM szabvány-együttes felépítését, alkalmazásának módját nyilvános dokumentációban teszi hozzáférhetővé. A dokumentáció az ADL honlapjáról letölthető (<http://www.adlnet.org>). A dokumentáció gerincét alkotó „könyvek” a következők:

- Áttekintés;
- Tartalomhalmozási modell;
- Futásidejű környezet;

Időről-időre a felmerülő hibák korrigálására, illetve apróbb pótlások, vagy egyszerűsítések miatt az ADL kiegészítéseket ad ki. Továbbá külön dokumentumban is hozzáférhetővé teszi a megfelelőségi követelmények listáját az LMS rendszerekre, valamint a tananyagszervezés különböző szintjeire vonatkozóan.

Az ADL ezeken túl hálózatán hozzáférhetővé teszi a szabvánnyal kapcsolatos előadásokat és bemutatókat.



5. ábra A SCORM 1.2 felépítése

### Első könyv: Áttekintés

A könyv eredeti címe: SCORM Overview Book.

Az első könyv célja, hogy általános tájékoztatást adjon az ADL szerepéről, küldetéséről, kitűzött céljairól. Egy rövid ismertetőből megismerkedhetünk az e-Learning fejlődésével, az internet e-tanulásra kifejtett hatásával, a Web-alapú tanulás jelentőségével. Olvashatunk az intelligens oktatási rendszerekről (ITS), s a velük szemben támasztott követelményekről, a tananyag személyre szabhatóságának jelentőségéről, s a SCORM hatásáról az eLearning-ben. Megismerkedhetünk a már vázolt ADL Co-Lab hálózat rendeltetésével, szervezeti felépítésével, s feladataival.

Röviden bemutatásra kerül az ADL koncepciója a SCORM bevezetésével kapcsolatosan, a SCORM eLearning iparban betöltött szerepének vázolója. Ezek után kerül ismertetésre a szabvány felépítése. A tárgyalás módja követi a szabvány részterületeinek könyvekre osztását. Nos, lássuk a könyveket, azaz magát a szabványt!

## **Második könyv: Tartalomhalmozási modell**

A könyv eredeti címe: SCORM Content Aggregation Model (CAM) Book.

A második könyv elsősorban a szerzői rendszerek gyártóinak és a tartalomfejlesztőknek ad eligazítást. Mint a cím is árulkodik, elsősorban a tartalom előállításával, szervezésével, kezelésével kapcsolatos kérdések kerülnek itt tárgyalásra. A könyv öt fő részre tagolódik, s ezek a következők:

- A SCORM tartalomhalmozási modell áttekintése;
- A SCORM tartalommodell;
- SCORM metaadat kezelés;
- A SCORM tartalomcsomag;
- SCORM sorrend és navigáció

### **A SCORM tartalomhalmozási modell áttekintése**

A fejezet néhány oldalon rövid eligazítást ad a könyv szerkezetéről, valamint ismerteti a többi könyvvel való összefüggéseket, illetve útbaigazítást ad a könyv használatáról, tartalmáról.

### **A SCORM tartalommodell**

A tartalommodell írja le a SCORM-ban használatos komponenseket, amelyekből maga a tananyag felépül. Meghatározza a hierarchikusan építkező tananyagstruktúra szintjeit, és a szintek kezelését.

### ***Tananyagelem (Asset)***

A SCORM követi az eLearning környezetben megszokott objektum-orientált szemléletet. A kiindulópont itt is a tananyagelem, amely a legkisebb, logikailag tovább nem bontható egység. Egy kép, ábra, animáció, XML dokumentum, hangfájl, film, vagy szöveg. Ezt a SCORM **Asset**-nek nevezi. Az Asset (tananyagelem) az építőköve a tananyagstruktúráknak.

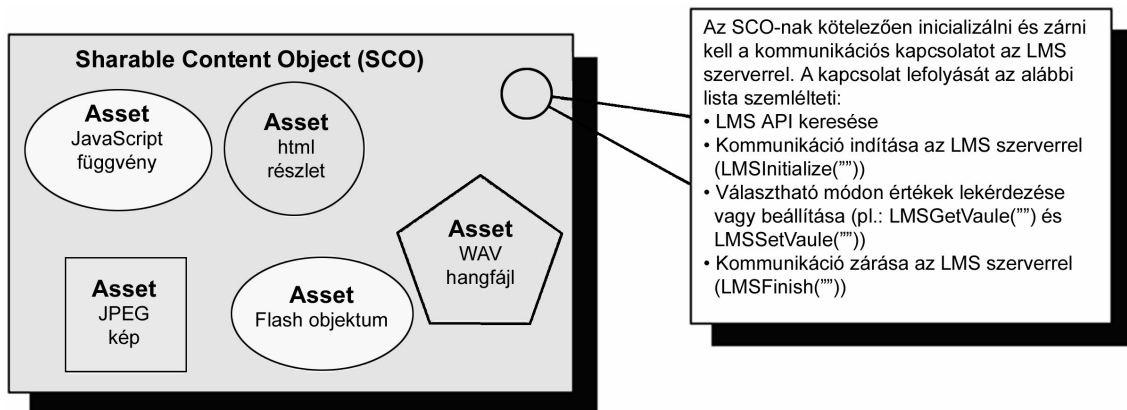
A tananyagelem bizonyos jellemzőit, adatait metaadat állományban írjuk le. Ez teszi lehetővé a tananyagelem jól kereshetőségét, illetve újrahasznosíthatóságát.

### ***SCO (Sharable Content Object)***

A tartalommodellben a következő szintet a SCO, más néven megosztható tartalomobjektum képviseli. Az SCO egy vagy több tananyagelemből (Asset) áll. Az újrahaszno-

síthatóság érdekében az SCO-kat kis egységekbe célszerű szervezni. Ugyanakkor gondoskodnunk kell arról, hogy ne tartalmazzon a környezetére való utalást, mert akkor nem tudjuk más tananyagba illeszteni, új kontextusban felhasználni.

Az SCO a legkisebb tananyagegység, amely **kommunikál** a tanulásirányítási szerverrel (LMS). Így az SCO-nak tartalmaznia kell a kommunikációhoz szükséges kódrészleteket. Természetesen feltételezzük, hogy az LMS szerver eleget tesz a SCORM RTE követelményeinek. Az SCO és az LMS kommunikációja az SCO tartalmát hordozó html oldalon elhelyezett szkriptek biztosítják.



6. ábra: Megosztható tartalomobjektum (SCO)

A kapcsolat során az SCO-nak az API hívások közül kettőt kötelezően meg kell hívnia, ezek az `LMSInitialize("")` és a `LMSFinish("")` eljárások. A tananyag természetétől függően opcionálisan más eljárások hívása is lehetséges.

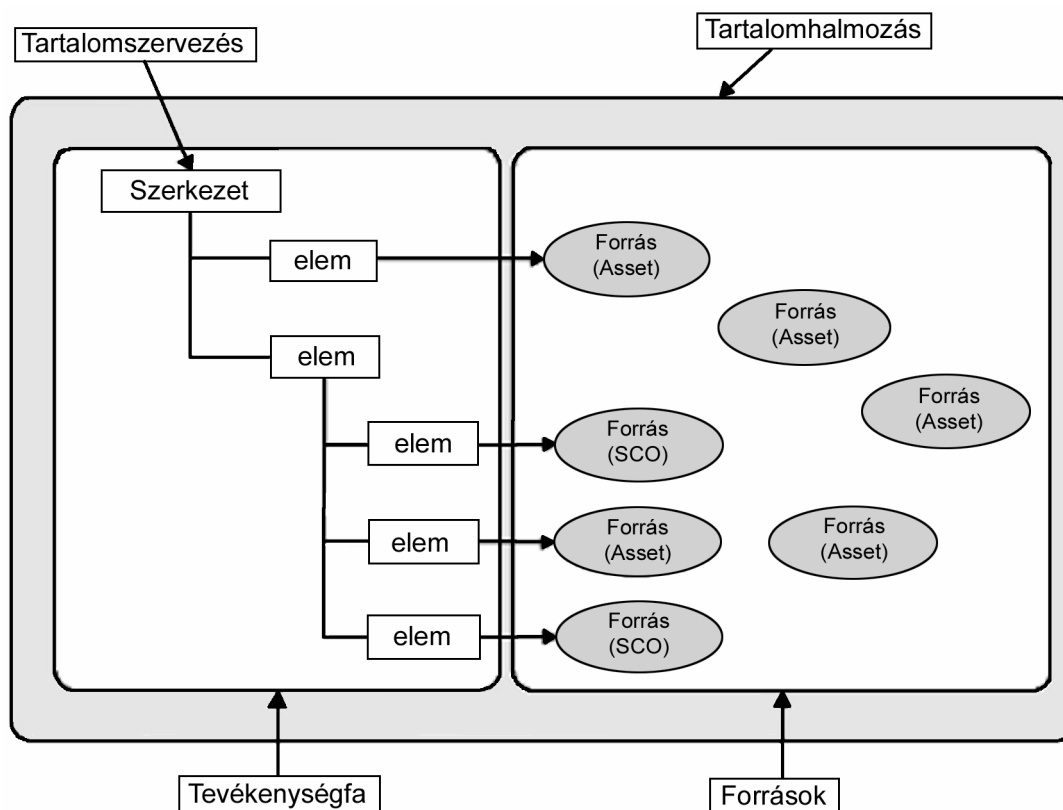
Az SCO-k a tananyagelemekhez hasonlóan metaadatokkal kerülnek leírásra. Jelentéstartalmuk szélesebb, mint a tananyagelemeké, ezért több információt szükséges nyilvántartanunk róluk.

### Tartalomszervezés

Minden tananyag rendelkezik egy struktúrával, amely magán viseli az adott tananyag, ismeretanyag logikáját, az adott tudományterület ismeretanyagának tárgyalási módját, az adott szervezet oktatási módszereit. A digitális tananyagok esetén is strukturáljuk a tananyagot. Ez az esetek többségében egy hierarchikus szerkezetet jelent, de elképzelhetőek ettől eltérő struktúrák is (lineáris, hálós, stb.).

Minden szervezetre jellemző a tananyagok tagolási stratégiája. Szükséges kialakítani egy közös terminológiát a tananyag egyértelmű és hatékony szegmentálása érdekében. Például: kurzus, lecke, modul, lap, stb. A SCORM esetében az elemi építőkövek adóttak: SCO, Asset. Ezekből kell felépíteni magát a tananyagot, amelyet most nevezünk kurzusnak.

A SCORM-ban a tananyag struktúrájának leírása tulajdonképpen szintén metaadatokkal történik. El kell választanunk egymástól a tartalom szervezését – amely a tananyag-struktúra logikai leírása – a tartalomhalmozástól (aggregation) – amely a fizikai tananyagelemek és tananyagegységek, tanulási egységek felhalmozása. Az előző egy szemléleti megközelítése a problémának, a másik inkább technikai jellegű, amely a tananyagelemekhez, illetve tananyagegységekhez biztosít hozzáférést. A tananyag elvi felépítése a tartalomcsomagban testesül meg.



7. ábra: Tartalomszervezési modell

A SCORM 1.2 a tartalomszervezést a tananyagstruktúra meghatározása irányából közelíti meg. A fejlesztő a struktúra felvázolásával deklarálja az általa ideálisnak tartott tananyagbejárás sorrendet. Tehát az egymást követő egységek szekvenciája képviseli a tananyag feldolgozásának javasolt útját. Ezt a tananyagegységekben definiált előfeltételek használatával lehet elérni. Lesznek olyan elemek, amelyek elágazásokat tartalmaznak. Ebben az esetben magukban hordozzák az elágazás bejárásához szükséges szabályokat. Ezek a tananyagegységek ebből kifolyólag nem tesznek eleget az újrahasonosíthatóság feltételeinek. A megoldás az lehet, ha az ilyen egységek nem hivatkoznak külső tananyagelemre, vagy tananyagegységre, s így nem kell az LMS-nek megjeleníteni őket.

A SCORM a tartalomhalmozással és tartalomszervezéssel kapcsolatos metaadatokat az IEEE LTSC LOM (Learning Object Metadata) ajánlása alapján írja le. A SCORM a következő típusú tartalomra vonatkozó metainformációk nyilvántartására ad lehetőséget:

- **Tartalomhalmozási metaadat** (Content Aggregation Meta-data) – leíró információt biztosít a tartalomhalmozás egészéről (tartalomcsomagról), valamint lehetővé teszi ezen információk hozzáférhetőségét;
- **Tartalomszervezési metaadat** (Content Organization Meta-data a SCORM 2004-ben a tartalomszervezési metaadatok veszik át a tartalomhalmozás szerepét, így a tartalomhalmozás részben csak a metaadat-séma azonosítása marad meg kötelezően) – alkalmazásának célja például kereshetőségének a biztosítása a tananyagtárolóban, illetve leírja a tananyag szerkezetét;
- **Tevékenység metaadat** (Activity Meta-data csak SCORM 2004-ben) – leírja a tanulói tevékenységeket, szabályozza a tananyagegységek hozzáférhetőségét;

- **SCO metaadat** – használata biztosítja az SCO felhasználástól független tárolását, újrahasznosításának lehetőségét és kereshetőségét;
- **Tananyagelem metaadat** (Asset Meta-data) – lehetővé teszi a tananyagelem azonosítását, kereshetőségét például a tananyagtárolóban a tananyagfejlesztés során. Leírja a tananyagelem jellemzőit, ez által rendezhetővé, csoportosíthatóvá válik. Biztosítja a tananyagelem újrahasznosíthatóságát;

### **Metaadatok használata**

A különböző egységek metaadatainak a megjelenítésére a tartalomcsomagban öt szintet deklaráltak. Ezek a következők:

- **Manifest:** A manifest szint írja le a teljes tartalomcsomagot. Tulajdonképpen ez a rész felel meg a tartalomhalmozási metaadatoknak;
- **Organization:** Ez a szint írja le az oktatási egységek szerveződését. Hogy kurzusról, leckéről vagy modulról van szó, azt a fejlesztő dönti el. Ez a szint felel meg a tartalomszervezési metaadatoknak.
- **Item:** Ez a szint írja le a tevékenységeket, amelyek akár hierarchikusan egymásba ágyazva jelennek meg. Jellemzően környezetfüggő. Amikor egységeket kapcsolunk össze a SCORM tevékenység metaadatok használata kötelező.
- **Resource:** Ez a szint ír le egy tananyagegységet, vagy egy tananyagelemet környezetétől függetlenül. Használata kötött: vagy az SCO metaadatokat, vagy a tananyagelem (Asset) metaadatokat kell használnunk. Ezt az `adlcp:scormType` attribútum értéke határozza meg, ami vagy "sco", vagy "asset".
- **File:** Egy tananyagelemet (Asset) ír le környezetétől függetlenül. Szervezési szempontból a Resource alá tartozik.

### **Metaadat kezelés**

A metaadatok alkalmazásának lényege, hogy a tartalomcsomag struktúra önálló szintjeit olyan leíró adatokkal lássa el, amelyek lehetővé teszik az adott egység...

- azonosíthatóságát;
- rendeltetésének meghatározását;
- a tananyag rendszerben elfoglalt státuszának meghatározását;
- leírják technikai jellemzőit;
- bizonyos pedagógiai jellemzőit;
- tulajdonjogi állását,
- kapcsolatrendszerét más tananyagegységekkel;
- felhasználására vonatkozó plusz információkat (pl.: ajánlás);
- egy általánosan elfogadott kategóriarendszerben elfoglalt helyét – besorolását.

A metaadatok alkalmazása megkönnyíti a komponensek jól kereshetőségét, tartalmuk feltárását és lehetőséget biztosít újrahasznosíthatóságukra (Asset, SCO). A metaadatok használatának előnye akkor mutatkozik meg, amikor egy adott keretrendszeren belül a tananyagot felépítő elemek, egységek tárolása elemi szinten valósul meg, nem pedig a kurzushoz kötődve. Ilyenkor az egyetlen példányban, egységes elvek szerint tárolt tananyagelemekre több kurzus is hivatkozhat (LCMS).

Az LMS szintén használja a metaadatokat. Egyrészt informálja a tanulót a tananyag felépítéséről, másrészt segítséget nyújt az egység futásidő alatti megjeleníthetőségéről.

A SCORM metaadat alkalmazási profilja az IMS metaadat ajánlására és az IEEE LTSC Learning, Object Metadata szabványának 6.1-es (Draft) változatára épül.

A SCORM CAM részletesen leírja a használatos elemek használatának módját, meghatározza alkalmazásának szabályait. A leírás a következő információkat tartalmazza az elemekről:

- Az elem neve: <general>;
- XML névtér (Namespace): <http://ltsc.ieee.org/xsd/LOM>;
- XML névtér előtagja (Namespace prefix): lom;
- XML ábrázolásának módja: <general>;
- A SCORM-ban az elemre vonatkozó követelmények, hogy mely előfordulása esetén mennyi lehet belőle;

| SCORM Metaadat alkalmazási profil | Metaadat előfordulási követelmény     |
|-----------------------------------|---------------------------------------|
| Content Aggregation               | 0 vagy 1                              |
| Content Organization              | 1 és csakis 1 (csak a SCORM 2004-ben) |
| Activity                          | 1 és csakis 1 (csak a SCORM 2004-ben) |
| SCO                               | 1 és csakis 1                         |
| Asset                             | 1 és csakis 1                         |

### 1. táblázat Metaadat-szintek

- Az adattípusa, s a megengedett hossza, valamint a típusra vonatkozó szabvány megadása (ha van pl.: közreműködő személy – vCard): a <general> anyaelem, nincs típusa;
- A lehetséges értékét, illetve ha van, az alapértelmezett szókészletet: a <general> anyaelem, nincs értéke;
- Alárendelt elemeinek listája, ha van;
- Példa – rövid kódrészlet az alkalmazás módjáról.

A SCORM javaslatot tesz arra, hogy a tartalomszervezés különböző szintjein mely elemek kitöltése kötelező és melyek kitöltése választható opcionálisan.

- K jelöli, ha az elem használata kötelező;
- O jelöli, ha az elem opcionálisan választható.
- F jelöli, ha az elem foglalt, azaz nem a szerző tölti ki (például a keretrendszer generálja).

| No.   | Elem               | Tartalom-halmazás | SCO | Asset |
|-------|--------------------|-------------------|-----|-------|
| 1     | <general>          | K                 | K   | K     |
| 1.1   | <identifier>       | F                 | F   | F     |
| 1.2   | <title>            | K                 | K   | K     |
| 1.3   | <catalogentry>     | K                 | K   | O     |
| 1.3.1 | <catalog>          | K                 | K   | O     |
| 1.3.2 | <entry>            | K                 | K   | O     |
| 1.4   | <language>         | O                 | O   | O     |
| 1.5   | <description>      | K                 | K   | K     |
| 1.6   | <keyword>          | K                 | K   | O     |
| 1.7   | <coverage>         | O                 | O   | O     |
| 1.8   | <structure>        | O                 | O   | O     |
| 1.9   | <aggregationLevel> | O                 | O   | O     |

**2. táblázat Metaadat elemek használata a SCORM 1.2-ben (részlet)**

## Tartalomcsomag

A tartalomcsomag biztosítja a szabványos utat a tananyagcsere lehetőségére a különböző rendszerek között. A tartalomcsomagban kerül leírásra a tananyag szerkezete, viselkedése, s itt kerülnek felhalmazásra a tananyagot alkotó egységek. A tartalomcsomag használatától azt várjuk, hogy a különböző tanulásirányítási szerverek (LMS) között hordozhatóvá válik a tananyag, annak átalakítása, konvertálása, manipulálása nélkül.

A SCORM tartalomcsomag ajánlása az IMS tartalomcsomag specifikációjára épül, annak egyfajta implementációja. Ugyanakkor a SCORM tartalomcsomag több olyan plusz információt is tartalmaz, amely az IMS ajánlásában nem szerepel, de a gyakorlatban szükség lehet rá (pl.: az LMS-sel való kommunikációban). A különböző tartalomcsomagok az ajánlás eltérő verzióira épülhetnek. A legfrissebb változat az IMS Content Packaging Specification 1.1.3.

A tartalomcsomag egy tananyagegységet reprezentál. Mérete nem specifikus, nincs meghatározva, hogy mekkorának kell lennie. A tananyag fejlesztője dönti el, hogy a tananyag mely szintjét képviseli – az egész kurzust, vagy annak egy-egy részegységét készíti el csomagként, esetleg több összefüggő kurzust egy csomagban. Miért csomagoljuk be a tananyagot? Az indok a kis méret, így az interneten is könnyen lehet tananyagot cserélni.

A tartalomcsomag tulajdonképpen a klasszikus számítástechnikai szóhasználat szerint is csomag, hiszen valójában egy .zip archív, azaz egy tömörített állomány. Alapvetően két fő részre tagolódik. Az egyik része egy speciális XML állomány, amely a tartalom struktúrájának, szerveződésének leírása. A másik része azoknak a fizikai állományoknak a gyűjteménye, amelyekből maga a tananyag felépül.

A SCORM a tartalomcsomagot PIF-nek (Package Interchange File) nevezi. Bár eredetileg a SCORM 1.2-es változata többféle tömörítési eljárással készült csomagot is megenged, az elvárás a PIF állományokkal szemben az, hogy feleljenek meg az RFC 1951-es ajánlásnak.

## A manifest

A manifest állomány egy XML dokumentum, amely leltárszerűen tartalmazza az adott csomag tartalmát és szerkezetét. Elég rugalmas forma arra, hogy különböző bonyolult-

sági fokú tananyagot leírjon, s a mérete is változatos lehet. Az LMS alkalmazások a manifest-ből nyerik a tananyag megjelenítéséhez szükséges információkat. A manifest tartalmazhat újabb manifest-et beágyazva, s ez akár több szint mélységig ismételhető. Az egymásba ágyazások mélysége nincs specifikálva.

A lényeg az, hogy a tartalomcsomag mindig tartalmazzon egy legfelső szintű manifest dokumentumot. A manifest állomány neve kötelezően `imsmanifest.xml` kell, hogy legyen. Mind az `imsmanifest.xml`-nek, mind a többi sémaleíró állománynak (különböző .DTD és .XSD állományok) a csomag gyökerében kell elhelyezkedni.

A manifest felépítése a következő:

- Meta-data
- Organization
- Resources
- Submanifest(s)

A négy rész saját belső struktúrával rendelkezik. Arra, hogy egy-egy részen belül hogyan helyezkednek el a különböző lehetséges elemek, valamint, hogy melyikből mennyi szerepelhet egy szinten, a szabvány részletesen leírja. A SCORM CAM minden szint lehetséges elemét definiálja, alkalmazásának szabályait rögzíti, és rövid kódrészlettel szemlélteti az alkalmazás módját. Ezek az információk elemenként a következők:

- Az elem megnevezése (például: `<schemaversion>`);
- Mely elemhez kötődik (például: alárendeltje valaminek);
- XML névtér – Namespace (például: `http://www.imsglobal.org/xsd/imscp_v1p1`);
- Az XML névtér előtagja (például: `imscp`);
- XML ábrázolásának módja, a „tag” neve (például: `<schemaversion>`);
- A SCORM-ban az elemre vonatkozó követelmények, hogy mely előfordulása esetén mennyi lehet belőle (például: Content aggregation – egy, és csakis egy);
- Az adat típusa (például: `CharacterString`), s a megengedett hossza;
- A lehetséges értékét, illetve ha van, az alapértelmezett szókészletet;
- Tulajdonságait, ha szerepeltetni kell azokat, illetve a megadás módja;
- Alárendelt elemeinek listáját, ha vannak;
- Példa – rövid kódrészlet az alkalmazás módjáról.

A manifest állomány első része (`<metadata>`) a metaadatokkal kapcsolatos információkat tartalmazza. Ez elsősorban a sémának és a sémaverzióknak az azonosítását jelenti, amely szerint a csomag készült. Viszont adott esetben tartalmazhat a LOM szerinti tagolásban valódi metaadatokat, amely a csomag szintű tananyagegységet azonosítja, illetve jellemzi.

A második rész (`<organization>`) a tulajdonképpeni struktúraleírás. Egy, és csakis egy legfelső szintű Organization elem megengedett, s ezen belül hierarchikusan tagolva kell definiálni azokat az egységeket (Organization és Item), amelyek reprezentálják a tananyag felépítésének logikai szerkezetét.

A SCORM tartalomcsomag bizonyos pontokon túllép az IMS által deklarált tartalomcsomag modellen. Újabb elemeket definiálnak, amelyek a tananyag felhasználásának

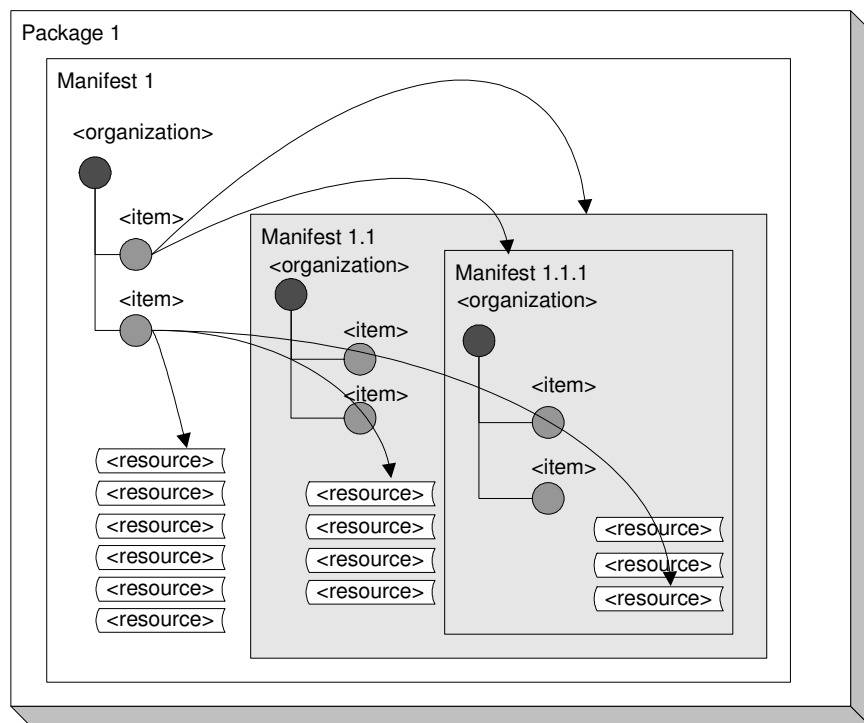
módjával, szabályaival kapcsolatosak. Ilyenek például a `<timeLimitAction>`, vagy a `<dataFromLMS>`, stb. Ezek az elemek előtagjukban is meg vannak különböztetve az IMS tartalomcsomag elemeitől. Előtagjuk `adlcp`. Ezeket az információkat az `Organization` rész tartalmazza.

## Submanifest alkalmazása

A tartalomcsomagban mindig kell lennie egy legfelső szintű manifest-nek, hiszen ez írja le a tartalomcsomagot. Az, hogy használunk-e submanifest-eket, a tartalomfejlesztő döntésének függvénye. A submanifest használatára akkor lehet szükség, ha a fejlesztő el akarja különíteni a tananyag részeit (leckék, modulok).

A submanifest használatának módja szabályozva van. Kikötés, hogy a felsőbb szintű manifest `<item>` elemének `identifierref` attribútumában a következőkre hivatkozhatunk:

- Az aktuális manifest-ben található `<resource>` elem azonosítójára;
- Az alárendelt manifest `<resource>` elemének azonosító attribútumára;
- A submanifest közvetlen leszármazottja a manifest-nek, vagy az abban található beágyazott manifest-nek.
- A hivatkozás két formája nem megengedett:
  - Az `<item>` elemnek tilos hivatkozni másik `<item>` elemre, vagy egy submanifest `<item>` elemére;
  - Az `<item>` elemnek tilos hivatkozni `<organization>` elemre, vagy egy submanifest `<organization>` elemére.



8. ábra Megengedett hivatkozások

## SCORM tartalomcsomag alkalmazási profil

A tartalomcsomag alkalmazási profil rendeltetési célja az, hogy felvilágosítást adjon az IMS tartalomcsomag specifikációjának alkalmazásáról a SCORM kompatibilis csomagok készítéséhez. Praktikus segítséget ad az alkalmazóknak és definiálja azokat a kiegészítéseket, amelyekkel a SCORM épített be az ajánlásba. A profil leírja, hogy a tartalomhalmozási modellnek megfelelően hogyan kell csomagolni a komponenseket. A SCORM két típusát ismeri a tartalomcsomagoknak:

- Forráscsomag és,
- Tartalomhalmozási csomag.

A forráscsomag használata egy eljárási mód arra, hogy a különböző tananyagegységeket (Asset, SCO) mindenféle tananyag-szervezési és tantervi osztályozás nélkül cserélhetővé tegye az oktatási rendszerek között. Tehát az ilyen csomagoknak nincs semmilyen logikai struktúrája, következésképp az LMS-nek nem kell kézbesíteni a tanulók számára.

A tartalomhalmozási csomag viszont értelemszerűen tartalmazza a tananyag szerveződésére vonatkozó információkat. A SCORM ajánlást tesz a két csomagtípus alkalmazásával kapcsolatosan a manifest elemek használatára.

| No.       | Elemek                          | Forráscsomag | Tartalomhalmozási csomag |
|-----------|---------------------------------|--------------|--------------------------|
| 1         | <manifest>                      | K            | K                        |
| 1.1       | identifier                      | K            | K                        |
| 1.2       | version                         | O            | O                        |
| 1.3       | xml:base                        | O            | O                        |
| 1.4       | <metadata>                      | O            | O                        |
| 1.4.1     | <schema>                        | O            | O                        |
| 1.4.2     | <schemaversion>                 | O            | O                        |
| 1.4.3     | {Meta-data}                     | O            | O                        |
| 1.5       | <organizations>                 | K            | K                        |
| 1.5.1     | default                         | T            | K                        |
| 1.5.2     | <organization>                  | T            | K                        |
| 1.5.2.1   | identifier                      | T            | K                        |
| 1.5.2.2   | structure                       | T            | O                        |
| 1.5.2.3   | adlseq:objectivesGlobalToSystem | T            | O                        |
| 1.5.2.4   | <title>                         | T            | K                        |
| 1.5.2.5   | <item>                          | T            | K                        |
| 1.5.2.5.1 | identifier                      | T            | K                        |
| 1.5.2.5.2 | identifierref                   | T            | O                        |

K – kötelező, O – opcionális, T – tiltott

### 3. táblázat Manifest elemek használata a csomagtípusokban (részlet)

#### Sorrend és navigáció

A sorrend és a tananyag bejárásának szabályozása a SCORM 1.2-es változatában a tananyagban elhelyezett előfeltételeken keresztül valósul meg, s ezt az adatmodellben definiált – a tananyagegység feldolgozásának státuszát rögzítő – változó figyelésével oldhatunk meg. Ezek értéke a következő:

- passed;
- completed;
- browsed;

- failed;
- not attempted;
- incomplete.

A SCORM új verziójában a tananyag bejárásának szabályozására jelentősen több, s kifinomultabb lehetőség kínálkozik, de több költséggel is jár.

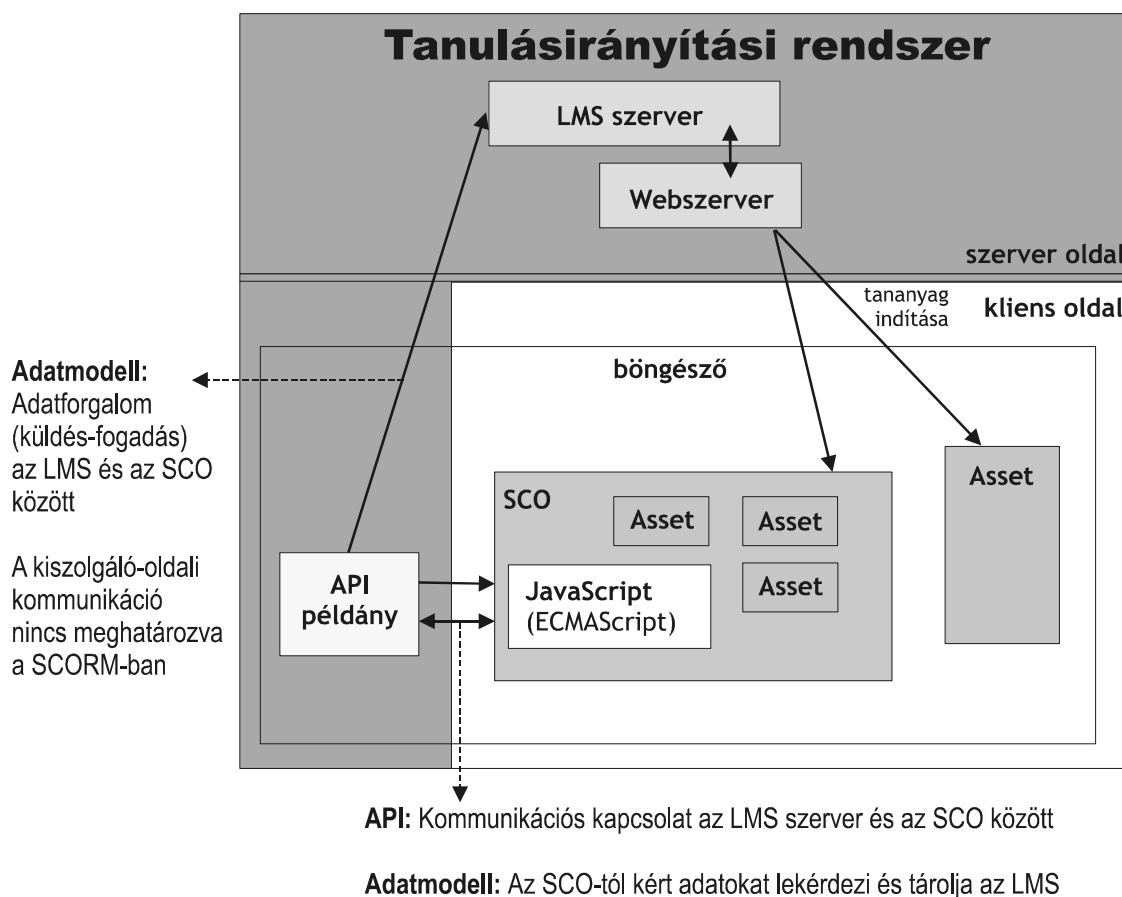
### Harmadik könyv: Futásidejű környezet

A könyv eredeti címe: SCORM Run-Time Environment (RTE) Book.

A SCORM RTE írja le a tartalom hasznosulásának módját, azaz hogyan jut el tananyag a tanulóhoz, a tartalom hogyan kommunikál az LMS szerverrel és miféle adatokkal írható le a rendszerben a tanulói tevékenységek sokasága. Mit és hogyan kell nyomon követnünk ahhoz, hogy rendelkezésünkre álljanak mind a tanulókról, mind a tananyagról gyűjtendő adatok.

Ehhez egyrészt specifikálni kell azt az eljárásmodellt, ami lehetővé teszi alkalmazások fejlesztését a tartalom megjelenítéséhez, másrészt szükségünk van egy adatmodellre, ami leírja, hogy milyen adatok gyűjtésére van szükség, azok típusát és viselkedését.

A SCORM RTE definiál erre egy modellt. A modell célja, hogy a tartalom újrahasznosítható és interoperábilis legyen a keretrendszerek között. Az ADL a 9. ábrán látható módon képzelel el a rendszer működésének módját.



9. ábra SCORM RTE koncepciója

Amíg a tanuló kapcsolatban van a tartalommal az LMS értékelve a tanuló teljesítményét és navigációs kéréseit, a kért tevékenység azonosítása után kézbesíti a tanuló számára a soron következő tananyagegységet. A szükséges információkat az LMS a tartalomcsomagból a manifest organization részéből nyeri. Ez határozza meg az úgynevezett tevékenységfát, ami a tartalom szerveződését reprezentálja. Magához az indítható tananyagegységhez az item és a resource szakaszban elhelyezett bejegyzéseken keresztül jut. Az így elért egységek már indíthatóak.

A tanuló tartalomhoz való csatlakozásának, s a session menedzselésének kezelésére a SCORM egy időkezelési modellt ír le, részletezve a tanuló csatlakozásának és a tananyag feldolgozásának lehetséges módozatait, s a képződő adatok kezelésének módját.

A SCORM tartalomhalmozási modell három komponenst definiál, amely a tartalom egy-egy szintjét képviseli. Ezek a tananyagelem (Asset), tartalomegység (SCO), és a tartalomszervezés (content organization). Az LMS ez utóbbiból nyeri a szükséges információkat (például: hol van egy adott SCO). Az SCO helymeghatározására az LMS URL-t használ.

Ehhez a manifest állományban a következő információkat kell megadni a resource bejegyzés alatt:

- xml:base deklaráció;
- indítási paraméter;
- href deklaráció.

Két típus indítható ilyen módon. A tananyagelemet (Asset) képes az LMS indítani, de nem használ API hívást, s nem kommunikál az LMS-sel. Az SCO viszont változatos kommunikációt folytathat az LMS-sel, sőt minimálisan egy inicializálási és egy lezárási hívást tartalmaznia kell.

## API metódusok

A SCORM 1.2-es verziója az AICC CMI001 Guidelines of Interoperability című dokumentumára alapozta futásidejű környezetének definícióját. E szabvány írja le az API-t, amelyek segítségével a futásidejű szolgáltató (RTS) a tartalommal kommunikál. Az RTS az a szoftver, amely ellenőrzi a tananyag-kézbesítés végrehajtását, biztosítja az erőforrások elosztását és adatátvitel vezérlését. Az API teszi lehetővé a kommunikációt a tartalom és az RTS között, melyet jellemzően egy LMS szerver valósít meg.

Minden API függvénynév és paraméternév nagybetű-érzékeny, hívásuknál/megadásuknál pontosan kell hivatkozni rájuk. Minden paraméter characterstring típusú, ezért idézőjelek között kell megadnunk őket. A SCORM a használt API metódusokat három csoportba sorolja.

- **Session metódusok** – a tartalom (SCO) és az LMS közötti kommunikáció indítását és lezárását végzik: `LMSInitialize("")`, `LMSFinish("")`.
- **Adatátviteli metódusok** – az SCO és az LMS közötti adatcserét, értékek átadását, lekérdezését valósítják meg: `LMSGetValue("")`, `LMSSetValue("")`, `LMSCommit("")`.
- **Kiegészítő metódusok** – használatukkal kiegészítő kommunikációt folytathatunk az SCO és az LMS között (például: hibakezelés): `LMSGetLastError("")`, `LMSGetErrorString("")`, `LMSGetDiagnostic("")`.

A SCORM RTE dokumentáció részletesen leírja a metódusokat, használatuk módját. A függvények használatára vonatkozóan ajánlásokat fogalmaznak meg, például az ellenjavalt alkalmazásukra felhívva a figyelmet, s kitérve néhány problematikus helyzetre is. A dokumentáció a következő információkat tartalmazza a függvényekről:

- A függvény neve: LMSInitialize
- A függvény szintaxisa: `return_value = LMSInitialize(parameter);`
- A függvény leírása: elindítja, inicializálja az LMS-sel való kommunikációt;
- Paraméterei: ("" ) üres characterstring (üresen kell hagyni);
- A függvény visszatérő értéke: két visszatérő értéke van "true" és a "false".

A kommunikáció státuszát tekintve három különböző állapot képzelhető el: nem inicializált, futó, lezárt. A három esetre vonatkozóan meg van határozva, hogy mely függvényeket lehet meghívni az adott esetben.

## Hibakódok

A használt IEEE szabvány rendelkezik a lehetséges hibakódok értéktartományáról, amelyet 0 és 65535 között határoz meg. E tartomány alapvetően két részből áll. Az egyik tartományt a szabvány lefoglalja magának, s meghatározza, hogy mely résztartományban mely hibakódok helyezkedhetnek el. Ez a tartomány 0-tól 999-ig tart. A másik tartományt a keretrendszer-gyártók számára biztosítja egyedi hibakódok definiálása céljából.

| Hibakód kategóriák              | Hibakód tartomány |
|---------------------------------|-------------------|
| nincs hiba                      | 0                 |
| általános hiba                  | 100 – 199         |
| szintaktikai hiba               | 200 – 299         |
| RTS hiba                        | 300 – 399         |
| adatmodell hiba                 | 400 – 499         |
| kivitelező által definiált hiba | 1000 – 65535      |

### 4. táblázat Hibakódok tartomány kiosztása

A dokumentáció az egyes hibakódokról részletes funkcionális leírást ad, néha-néha példákkal illusztrálva.

## Adatmodell

A könyv túlnyomó részét az adatmodell részletes leírása foglalja el. Az adatmodell elemek neve közös előtaggal kezdődik: ez a „cmi”. Ez jelzi az LMS-nek, hogy adatmodell elemről van szó. Az elemek névadása az ECMAScript ajánlás szabályai szerint kell, hogy történjen: pontjelölésű karakterstring. A nevek felépítése a következő: *cmi.elemlév*.

Mielőtt az adatmodell elemek részletes bemutatása elkezdődne, az adatmodellre vonatkozó szabályok kerülnek ismertetésre. Ezek az elemek csoportosításának rendje, az előfordulásuk (számosságuk) szabályai, a használatos kulcsszavak és foglalt határolójelek ismertetése. Ezt követi az adattípusok, s azok felépítésének, használatának bemutatása.

Az adatmodellben 49 elem használatára nyílik lehetőség, de ezek közül 14 elem használata kötelező.

Az egyes elemek leírása a következő séma szerint történik:

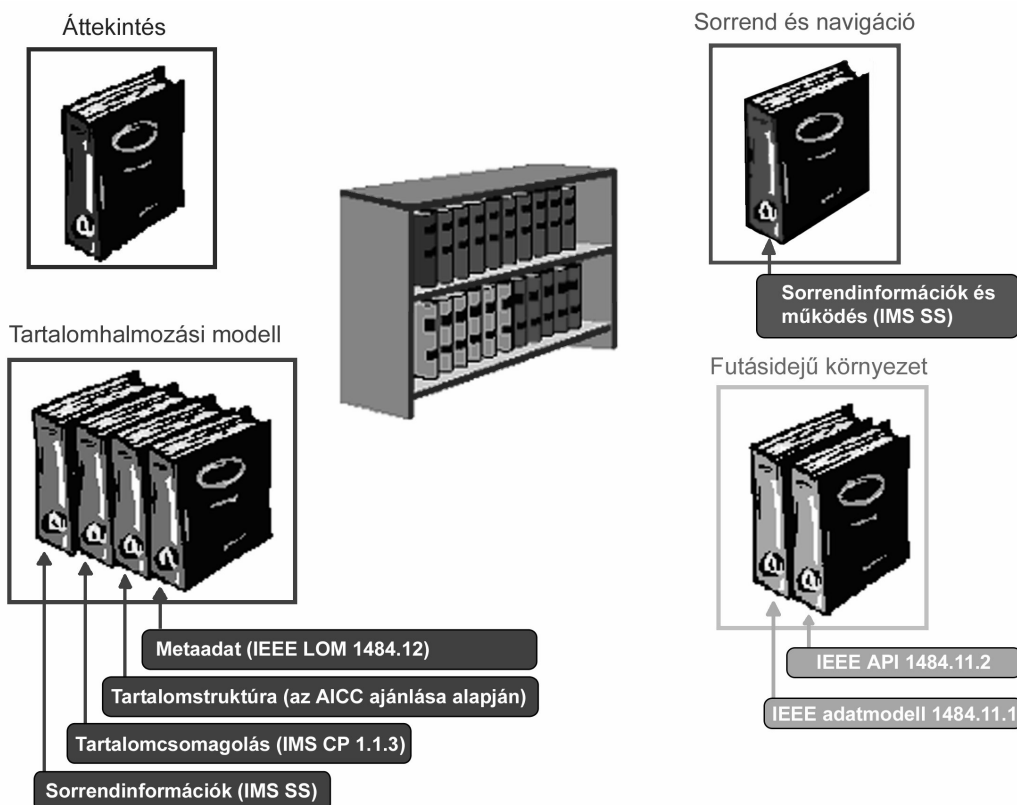
- Az elem szabványos neve pontjelöléssel;
- Adatmodell implementációs követelmények:
  - adattípus;
  - értéktartomány;
  - formátum
- LMS működési feltételek;
- SCO működési feltételek;
- API implementációs követelmények;
- Kiegészítő működési feltételek;
- Példa.

Az LMS rendszerek minimális megfeleléséhez (RTE1) a következő 14 adatelem használata kötelező:

- cmi.core.\_children
- cmi.core.student\_id
- cmi.core.student\_name
- cmi.core.lesson\_location
- cmi.core.credit
- cmi.core.lesson\_status
- cmi.core.entry
- cmi.core.score.\_children
- cmi.core.score.raw
- cmi.core.total\_time
- cmi.core.exit
- cmi.core.session\_time
- cmi.suspend\_data
- cmi.launch\_data

## A SCORM 2004

A SCORM 2004 igen rövid vázlatos specifikációs időszak után vezette be az ADL (2003. október – 2004. január). A változtatásnak két oka volt. Az egyik, hogy a szabványosítás során bizonyos specifikációk státusza megváltozott. A futtatási környezet szempontjából érdekes adatmodell, illetve az API hívásokra vonatkozó ajánlások akkreditálását az IEEE a kezébe vette. Továbbá elfogadásra került a LOM, így az új változatban lecserélésre került a LOM korábbi 6.1-es Draft specifikációja. A másik ok a verzióváltásra az, hogy az ADL Sequencing and Navigation néven – negyedik könyvként implementálta az IMS Simple Sequencing specifikációját.



**10. ábra A SCORM 2004 felépítése**

Ez bizonyos értelemben nem hozott radikális változást, viszont a tartalomhalmozási modell némileg átszerveződött, mivel a sorrend és navigáció definiálása tulajdonképpen a manifest állományban történik.

Ami viszont a verziók közötti inkompatibilitáshoz vezet: megváltoztak az API-hívások nevei. A változásokat a 4. táblázat tartalmazza.

| típus | SCORM 1.2             | SCORM 2004 (1.3.1) |
|-------|-----------------------|--------------------|
| 1     | LMSInitialize("")     | Initialize("")     |
| 1     | LMSFinish("")         | Terminate("")      |
| 2     | LMSGetValue("")       | GetValue("")       |
| 2     | LMSSetValue("")       | SetValue("")       |
| 2     | LMSCommit("")         | Commit("")         |
| 3     | LMSGetLastError("")   | GetLastError("")   |
| 3     | LMSGetErrorString("") | GetErrorString("") |
| 3     | LMSGetDiagnostic("")  | GetDiagnostic("")  |

**5. táblázat API függvények a SCORM verziókban**

Részben ez a változtatás, részben a manifest állományban bekövetkező változások szükségessé teszik az alkalmazások áttervezését. A SCORM 2004 (1.3.1) támogatásához új verziók bevezetésére van szükség.

Ez a folyamat a SCORM korábbi sikereihez képest kissé lassan folyik, s bár az ADL komoly erőfeszítéseket tesz az új verzió népszerűsítésére, ma még kevés termék támogatja a SCORM 2004-et. Az ADL által hitelesített 92 alkalmazás közül csak 11 felel meg a SCORM 2004-nek, ezek közül is csak 5 az LMS rendszer, szemben az 1.2-es változatot támogató 50 hitelesített LMS alkalmazással

(forrás: ADL <http://www.adlnet.org/scorm/certified/index.cfm>)

## Negyedik könyv: Sorrend és navigáció

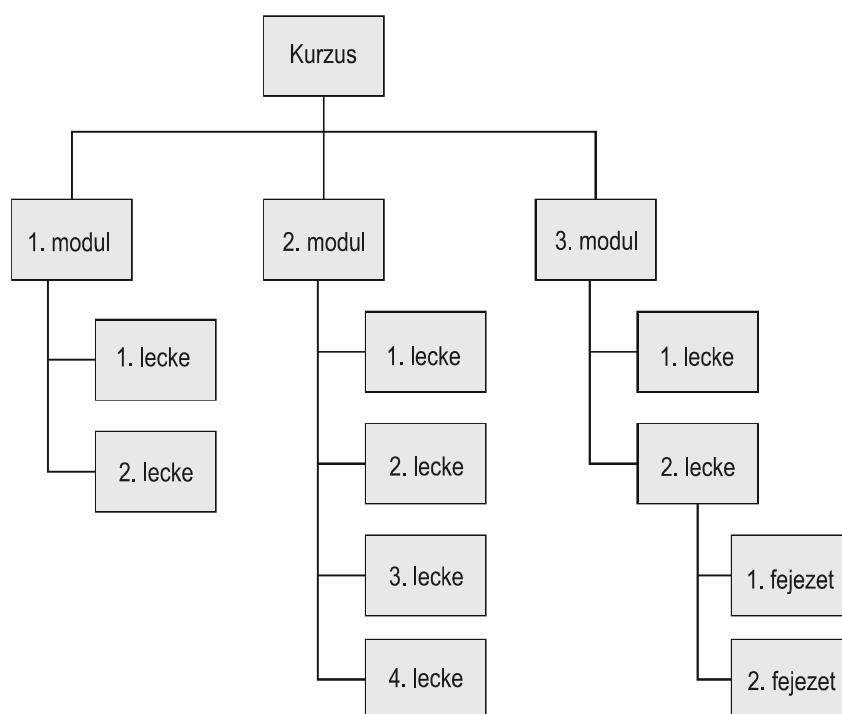
A könyv eredeti címe: SCORM Sequencing and Navigation (SN) Book.

A SCORM 1.2-es verziója a sorrend definiálására, és a tananyag bejárására kevés eszközt nyújt. A negyedik könyv új fejezetet nyitott a SCORM szabvány életében. Bár a verzióugrás (1.2 ⇒ 1.3 ⇒ 1.3.1) nem volt nagy, úgy gondolom, hogy nem véletlen az, hogy az új verzió megkülönböztető nevet kapott – SCORM 2004 – ami eddig nem volt jellemző a szabvány jelölésére. A formálódó szabványok több változtatást kényszerítettek ki, de az új nevet talán a sorrend-definiálás lehetőségének köszönhetjük.

A SCORM SN könyv alapja az IMS Simple Sequencing Specification ajánlása. Noha az IMS specifikációjának az egyszerű jelzőt adta, s alapjában véve néhány széles körben elterjedt bejárési stratégiát specifikál, ez nem zárja ki azt, hogy ne lehetne mesterséges intelligencia alapú sorrend-definíciós modellt alkotni (lásd: Brandt Dargue: Using HLA & SCORM 2004 For Adaptive Training Based on Student Performance. International Plugfest 2004. február 16-19.).

A specifikáció célja az, hogy a tananyag bejárásának egyfajta definícióját adja. Lehetőséget teremtsen arra, hogy adott esetben a tananyagot meghatározott módon lehessen csak bejárni. Ennek az alapja nem más, mint a tartalomhalmozási modell. A kiindulási pont a tartalom struktúrájának leírása, ami a manifest állományban testesül meg. Úgy is felfoghatnánk, hogy a tartalomnak a manifest-ben leírt struktúrája a szerző által ideálisnak tartott bejárési útvonalat reprezentálja. Nos, az igazság az, hogy egy hierarchikusan építkező, elágazásokat tartalmazó tananyagnál máris hiányoznak az egyes útmutató információk, amelyek a sorrendet hivatottak jelezni.

A SCORM SN ezeket az információkat biztosítja a tananyagfejlesztők számára, sőt, egyenesen szabályozhatják a tanulói tevékenységeket, függőségeket definiálhatnak, stb..



11. ábra Példa a tevékenységfára

A SCORM-ban a bejárás alapját a tevékenységfa, vagy másképp az aktivitásfa reprezentálja. Mint már szó volt róla, a tevékenységre vonatkozó információk a tartalomcsomagból származnak. Ha a 11. ábrán látható tevékenység gráf elemeit a tartalomcsomag megfelelő bejegyzéseknek feleltetem meg, akkor a kurzus tulajdonképpen az <organization>-nek, a modulok pedig egy-egy <item>-nek felelnek meg.

Az LMS a tartalomszervezést tevékenységfává képezi le. Egy tevékenységfa azt a belső hierarchikus adatstruktúrát ábrázolja, amely tanulói tevékenységet definiálja. Amikor a tanuló kapcsolatban van a tartalmat ábrázoló tevékenységfával, az LMS értékeli a sorrendre és a követésre vonatkozó információkat és meghatározza a tanulóra vonatkozó sorrendet és tevékenységet. Továbbá választási lehetőséget biztosít a tanuló számára az adott feltételek alapján. Így a tananyagfejlesztőnek lehetősége van, arra, hogy ugyanazon struktúra mellett más-más tananyag-bejárás szabályokat alkosson.

### A Klaszter (Cluster)

A SCORM bevezeti a klaszter fogalmát a résztvékenységek jelölésére. A klaszter megfelel egy szint tevékenységsorozatának, amely lehet a tananyagstruktúra fogalmi szerint egy modul, lecke, vagy fejezet, illetve a hierarchia csúcsán álló, az elágazások indulópontjait tartalmazó terület is klaszternek minősül. A tevékenységfa levéleleme nem lehet klaszter!

A SCORM által felvázolt elképzelés a valós tanulói tevékenységekből és lehetséges eseményekből indul ki. Számba veszi azokat a lehetséges történéseket, amelyek követe- se, kezelése megoldandó, mind a tananyag, mind a tanuló szempontjából, s szimulálja a kvázi pedagógiai környezetet.

A probléma kezelésére a következő modellstruktúrát definiálták:

- Sorrend definíciós modell – definiálja a tartalomfejlesztő számára azokat az elemeket (változókat), amelyek a sorrend vezérléséhez szükségesek;
- Sorrendkezelési modell – leírja a sorrendre vonatkozó változók kezelésének módját, ezen belül;
  - Követési modell – a tanulói tevékenységek adatainak a gyűjtése;
  - Aktivitás állapot modell – az aktivitásra vonatkozó információk kezelésének, lekérdezésének módja;
  - Sorrend definíciós modell – mint fent;
- Navigációs modell – a tanuló navigációs kéréseinek kezelését írja le.

Mint látszik, a SCORM a sorrend és tanulói tevékenységek kezelésére egy adatmodellt definiál. A különböző tevékenységek, események kezeléséhez változókat használ, s azokat osztályozva csoportokba szervezi. Maguknak a tevékenységeknek a megvalósítására, a feltételek feldolgozására pedig kifejezéseket használ. A kifejezés a következőképpen épül fel:

**If** [feltétel beállítása] **Then** [tevékenység]

A specifikáció a különböző jellegű tevékenységek szervezésére a következő eszköz-rendszert biztosítja:

- Sequencing control modes;
- Constrain choice modes;
- Sequencing rule description;
- Limit conditions;
- Auxiliary resources;
- Rollup rule description;
- Rollup controls;
- Rollup consideration controls;
- Objective description;
- Selection Controls;
- Randomizations controls;
- Delivery controls;

Maga a specifikáció csupán a definiálás módjával, a lehetséges értékekkel, alapértelmezett értékekkel foglalkozik. Szemléltetésként néhány konkrét példát tartalmaz a dokumentum. Konkrét tananyag-feldolgozási stratégiákra nem találunk példákat, de az ADL honlapján találunk ezekkel kapcsolatos bemutatókat, amelyek részletezik az egyes bejárási stratégiák esetén a különböző változókra értelmezett értékeket.

Hol jelennek meg a sorrendre és a navigációra vonatkozó információk? Jellemzően a tartalomcsomagban, a manifest állomány méretét radikálisan növelve találjuk ezeket az információkat. Másrészt nyilvánvalóan a felhasználókról képződő adatokat az LMS szervernek kell nyilvántartani.