

EGE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

(DOKTORA TEZİ)

WEB-TABANLI UZAKTAN EĞİTİM VE

BİR ALTYAPI TASARIMI

Birim BALCI

Bilgisayar Mühendisliği Anabilim Dalı

Bilim Dalı Kodu: 619.01.00

Sunuş Tarihi: 07.09.2007

Tez Danışmanı: Doç. Dr. Mustafa Murat İNCEOĞLU

Bornova –İZMİR

III

Birim BALCI tarafından doktora tezi olarak sunulan “Web Tabanlı Uzaktan Eğitim ve Bir Altyapı Tasarımı” başlıklı bu çalışma E.Ü. Lisansüstü Eğitim ve Öğretim Yönetmeliği ile E.Ü. Fen Bilimleri Enstitüsü Eğitim ve Öğretim Yönergesi'nin ilgili hükümleri uyarınca tarafımızdan değerlendirilerek savunmaya değer bulunmuş ve 07.09.2007 tarihinde yapılan tez savunma sınavında aday oybirliği / oyçokluğu ile başarılı bulunmuştur.

Jüri Üyeleri:

İmza

Jüri Başkanı : Doç.Dr. Mustafa Murat İNCEOĞLU

Raportör Üye : Yrd.Doç.Dr.Aybars UĞUR

Üye : Doç.Dr.Yalçın ÇEBİ

Üye : Prof.Dr.Mesut RAZBONYALI

Üye : Yrd.Doç.Dr. Vecdi AYTAÇ

ÖZET**WEB TABANLI UZAKTAN EĞİTİM VE
BİR ALTYAPI TASARIMI**

BALCI, Birim

Doktora Tezi, Bilgisayar Mühendisliği Bölümü

Tez Yöneticisi: Doç.Dr. Mustafa Murat İNCEOĞLU

Eylül 2007, 160 sayfa

Bu tezde, web tabanlı eğitimde eğitsel içerik hazırlama alanında, öğrenme nesnelерinin kullanılarak öğrenme nesnesi içerik paketlerinin oluşturulması amacıyla yönelik bir platform oluşturulmuştur. Platform, içerik paketi oluşturma, üst-veri tanımlamalarını yapabilmek, oluşturulan içerik paketleri arasında arama yapabilmeyi sağlayan bir eğitsel içerik paketi havuzuna ve birden çok dil desteğine sahip olma özelliklerinden dolayı, Türkiye’de tasarlanmış olan ilk uygulamadır.

Üç farklı kullanıcı grubunun tanımlandığı sistemde farklı işlevleri yerine getiren toplam 12 modül bulunmaktadır. Uygulama en genel haliyle eğitmenlere sisteme yüklenen ve paylaşımına açık olan kaynak dosyalarını kullanarak verdikleri dersler ile ilişkili ders içeriklerini hazırlama ve bunları içerik paketleri halinde depolama olanağı sunarken; öğrencilere de aldıkları derslerle ilgili paylaşımına açık olan içerik paketleri arasında arama, önizleme ve paketleri kendi yerel disklerine kopyalama imkanı vermektedir. Platform Microsoft IIS tabanlı çalışmaktadır ve sistemin kodlama kısmında ASP.NET ortamında, C# programlama dili kullanılmıştır.

Anahtar sözcükler: Web tabanlı eğitim, öğrenme nesneleri, SCORM, öğrenme nesnesi içerik paketi, içerik paketleme, öğrenme nesnesi havuzu.

ABSTRACT

**WEB BASED EDUCATION AND
AN INFRASTRUCTURE DESIGN**

BALCI, Birim

PhD. Thesis in Computer Engineering

Supervisor: Doç.Dr. Mustafa Murat İNCEOĞLU

September 2007, 160 pages

In this thesis, in the area of web based education and learning content design, a platform has been created to prepare the learning object content packages. This platform also has a content repository for the registered users. Due to the fact that this platform combines many properties together like preparing learning content packages according to the last version of the SCORM standard, editing metadata definitions, searching files and content packages from its own repositories and supporting multilanguage. This platform is the first application in Turkey that combines all these functions together.

There are 12 modules to make different functions and 3 user groups in the platform. By using the uploaded and sharable files in the platform, instructors can prepare content packages related to the lessons they give. On the other hand, students can search, preview and download the content packets related to the lessons they take.

Platform works on Microsoft IIS and ASP.Net environment and C# programming language is used.

Keywords: Web based education, learning objects, SCORM, learning object content package, content packaging, learning object repository.

TEŐEKKÜR

Bu alıőma sũresince fikirleriyle yol gũsteren ve desteęini hep hissettięim danıőmanın Sayın Do.Dr. Mustafa Murat İnceoęlu'na, manevi desteklerini esirgemeyen rektũrũm Sayın Prof. Dr. Mesut Razbonyalı ve hocalarım Sayın Prof. Dr. İsa Eőme, Sayın Prof. Dr. őaban Eren'e, yardımlarını esirgemeyen arkadaőlarıma, niőanlıma, uezellikle tũm stres ve sıkıntımı benimle beraber yaőayan canım anneme ok teőekkũr ederim.

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET V.....	V
ABSTRACT.....	VII
TEŞEKKÜR.....	IX
ŞEKİLLER DİZİNİ.....	XIV
ÇİZELGELER DİZİNİ	XVI
KISALTMALAR.....	XVII
1. GİRİŞ	1
2. ÖNCEKİ ÇALIŞMALAR.....	4
3. ÖĞRENME NESNELERİ VE SCORM STANDARDI.....	19
3.1 Öğrenme Nesnesi ve Nesne Havuzu Kavramları.....	19
3.2 SCORM Standardı	26
3.2.1 İçerik paketi bileşenleri.....	30
3.2.2 Üstveri38	
3.2.3 Sıralama ve Gösterim.....	57
4. TASARLANAN PLATFORMUN (LCDM) ÖZELLİKLERİ VE BENZERLERİNDEN FARKLARI	75

XII

4.1	İçerik Paketleme ve Üstveri İçin Türkiye'deki İlk Uygulama.....	75
4.2	SCORM 2004 Standardı Desteği.....	76
4.3	Web Tabanlı Uygulama ve Kolay Kullanıcı Arayüzü.....	76
4.4	Kaynak Havuzu ve ÖN Havuzu.....	76
4.5	Dosya ve ÖN İçerik Paketi Arama	77
4.6	Önizleme İmkânı.....	77
4.7	Dosya ve ÖN İçerik Paketi İndirebilme.....	78
4.8	Hazır İçerik Paketlerinin Sisteme Yüklenmesi	78
4.9	Eğitim Yönetim Sisteminden Bağımsız Kullanılabilme.....	78
4.10	Birden Fazla Dil Desteği.....	79
4.11	Üzerinde Çalışılan Platform.....	79
4.12	Sistemin Çalışması.....	79
5.	TASARIM	81
5.1	Kullanıcı Grupları ve Yetkileri	81
5.2	Platformda Tanımlı Modüller ve İşlevleri	84
5.2.1	Ders yöneticisi modülü	84
5.2.2	Grup yöneticisi modülü	87
5.2.3	Kullanıcı yöneticisi modülü.....	88
5.2.4	Dosya yöneticisi modülü	89
5.2.5	Proje yöneticisi modülü	94
5.2.6	Metadata modülü	98
5.2.7	Sequencing modülü.....	102
5.2.8	Paket yöneticisi modülü.....	103
5.2.9	Arama modülü	106
5.2.10	Önizleme modülü.....	107
5.2.11	Hazır paket yükleme modülü.....	109
5.2.12	Dil yöneticisi modülü.....	111
6.	LCDM İLE BENZER İŞLEVLERİ OLAN DİĞER UYGULAMALARIN KARŞILAŞTIRILMASI.....	113

XIII

7. LCDM PLATFORMUNUN FARKLI AÇILARDAN DEĞERLENDİRİLMESİ	129
7.1 Öğrenme Modelleri Açısından Değerlendirilmesi	129
7.2 Bilgi Yönetimi Açısından Değerlendirilmesi.....	131
7.3 İletişim Araçlarının Kapasitesi Açısından Değerlendirilmesi...	132
7.4 Öğretim Yazılımlarının Özelliklerine Göre Değerlendirme	132
8. SONUÇ	134
EK AÇIKLAMALAR.....	140
KAYNAKLAR DİZİNİ	148
ÖZGEÇMİŞ	160

ŞEKİLLER DİZİNİ

<u>Şekil</u>	<u>Sayfa</u>
3.1 Bir dersin yapısında bilgi ve öğrenme nesnelerinin yeri	24
3.2 SCORM İçerik Kümesi ve İçerik Yapısı.	29
3.3 Öğrenme Nesnesi İçerik Paketi.	30
3.4 Manifest Dosyası elemanları	31
3.5 İçerik Organizasyonu	32
3.6 VCard örneği.	48
3.7 SCORM 2004 Sequencing: Etkinlik ağacı örneği.	59
3.8 <sequencingRules> kurallarına ait durum ve hareketler	63
4.1 LCDM Çalışma platformu.	80
4.2 LCDM Sisteminin Çalışması.	80
5.1 Gruplar ve Hak sahibi oldukları modüller arası ilişkiler.	83
5.2 LCDM platformundaki modüller arasındaki ilişkiler.	85
5.3 LCDM platformundaki “ders yöneticisi” modülü arayüzü.	86
5.4 LCDM platformu “ders yöneticisi” modülü temel akış diagramı.	87
5.5 LCDM platformu “grup yöneticisi” modülü akış diagramı.	88
5.6 LCDM platformu “kullanıcı yöneticisi” modülü akış diagramı.	89
5.7 LCDM platformundaki “dosya yöneticisi” modülü arayüzü.	90
5.8 LCDM platformu “dosya yöneticisi” modülü akış diagramı.	93
5.9 LCDM platformundaki “proje yöneticisi” modülüne ait arayüz.	95
5.10 LCDM platformu “proje yöneticisi” modülü genel akış diagramı.	99
5.11 LCDM platformunda üstveri girişi ekranı	101
5.12 Sıralama bilgilerinin tanımlanması için bir ekran görüntüsü.	102
5.13 LCDM Paket Yöneticisi ekranı: Yayımcı girişi.	104

5.14 “Paket yöneticisi” modülü genel akış diagramı.	105
5.15 Paket arama işlemi akış diagramı.....	107
5.16 LCDM platformu “paket önizleme” işlemi temel akış diagramı. .	108
5.17 LCDM Paket önizleme.....	109
5.18 LCDM Hazır paket yükleme.....	110
5.19 “Hızır paket yükleme” modülü temel akış diagramı.....	111
5.20 “Dil yöneticisi” modülünün işleyişini gösteren akış diagramı.....	112
A.1 Dil tanımlamaları ile ilgili tablolar.....	140
A.2 Kullanıcıların tanımlandığı tablo	141
A.3 Dersler ve Kullanıcılar ile ilişkilendirildikleri tablolar	141
A.4 Dizinler tablosu	142
A.5 Dosyalar tablosu	143
A.6 Web adresleri tablosu	143
A.7 Projeler tablosu.....	144
A.8 Proje ile ilişkili versiyonlar tablosu.....	144
A.9 Organizasyonlar tablosu	145
A.10 Organizasyon detayları tablosu	146
A.11 Paketler tablosu	146
A.12 Tablolar arası ilişkiler.....	147

ÇİZELGELER DİZİNİ

<u>Çizelge</u>	<u>Sayfa</u>
2.1 İncelenen Eğitim Yönetim Sistemlerinin Karşılaştırılması	16
2.2 Üstveri editorleri	17
2.3 İçerik Paketleme ve Üstveri Editörleri.....	18
3.1 Öğrenme nesnelerinin tiplerine göre özellikleri	23
3.2 Bilgi nesneleri ve öğrenme nesneleri arasındaki farklar.....	24
3.3 Üstveri kategorileri, içerdikleri elemanlar ve veri tipleri	42
3.4 Structure ve AggregationLevel alanları arasındaki ilişki	46
6.1 LCDM ile incelenen LMS'lerin karşılaştırılması.	115
6.2 LCDM ile içerik paketleme ve üstveri araçlarını karşılaştırmak.	121
6.3 LCDM platformunun üstveri editörleri ile karşılaştırılması.	127

KISALTMALAR

ACL	Agent Communication Language
ADL	Advanced Distributed Learning
ADLNet	Advanced Distributed Learning Network
ARIADNE	Alliance of Remote Instructional Authoring and Distribution Networks for Europe
CAM	Content Aggregation Model
EAD	Encoded Archival Description
EM ²	An Environment for Editing and Management of Educational Metadata
GUID	Globally Unique Identifier
HTML	Hyper Text Markup Language
IEEE	Institute of Electrical and Electronics Engineers
IIS	Internet Information Server
IMS	Instructional Management Systems
IMS LOM	IMS Learning Object Metadata
IMS-LRM	IMS Learning Resource Metadata
ITS	Intelligent Tutoring Systems
IO	Information Object
IDDL	Institute for Distance and Distributed Learning
LCDM	Learning Content Design Manager
LCMS	Learning Content Management System
LO	Learning Object

XVIII

LOAT	Learning Object Authoring Tool
LOM	Learning Object Metadata
LOMGen	The Learning Object Metadata Generator
LOR	Learning Object Repository
LMS	Learning Management System
LTSC	Learning Technology Standards Committee
LTSN	Learning and Teaching Support Network
MARC	Machine –Readable Cataloguing
MOODLE	Moduler Object Oriented Dynamic Learning Environment
ÖN	Öğrenme Nesnesi
PIF	Package Interchange File
RELOAD	Reusable Learning Object Authoring and Delivery
RIO	Reusable Information Object
RLO	Reusable Learning Object
RTE	Run Time Environment
SCO	Sharable Content Object
SCORM	Sharable Content Object Reference Model
SN	Sequencing and Navigation
TEI	Text Encoding Initiative
TreeLOM	Tree View Learning Object Metadata
UKCMF	UK Learning Object METadata Common Framework
UPA	Uygulama Programlama Arabirimi

XIX

URL	Uniform Resource Locator
VLE	Virtual Learning Environments
WeLOAD	Web based Learning Object Authoring and Delivery
XML	Extensible Markup Language

1. GİRİŞ

Her geçen gün teknolojinin de gelişmesiyle zaman kavramının hayatımızdaki önemi artmakta ve bu, eğitim alanını da etkilemektedir. Bilginin miktar ve ayrıntı yönünden hızla artması, içeriklerin karmaşıklaşması, bilgiye hızlı erişim, bilginin çabuk güncellenmesi gerekliliği ve bilgi paylaşımı gibi kavramlar önem kazanmaktadır. Ertürk'ün (1975) de belirttiği gibi “eğitim, bireyin davranışında kendi yaşantısı yoluyla ve kasıtlı olarak istendik değişme meydana getirme süreci”dir. Eğitim ile ilgili yapılan tüm tanımlarda eğitimin davranış değişikliği oluşturduğu üzerinde durulmakta fakat mekanla ilgili bir ifade kullanılmamaktadır. Yani, öğrenmenin olduğu her durumda, zaman ve mekan bağımsız (Başkömürcü ve Öztürk, 1996; Blum and Sachs, 1999; Dzakira and Idrus, 2003; Balcı ve İnceoğlu, 2004) olarak davranışların değişmesine yol açan bir eğitim sürecinden söz edilebilir. Öğrenci sayısının artması, teknolojik gelişmeler ve İnternet kullanımındaki hızlı artış, öğretmenlerin sayıca yetersizliği gibi nedenlerle eğitimde bilgisayar kullanılması gereksinimi artmaktadır. Eğitimin öğrenciye en faydalı şekilde verilebilmesi için farklı eğitim teknolojileri (Alkan,1997; New Jersey Department of Education, 1998) uygulanmaktadır ve web-tabanlı uzaktan eğitim (e-eğitim) de bunlardan biridir.

E-eğitim, bilgisayar aracılığıyla eğitim (computer mediated education) (Klassen and Vogel, 2003), ağ üzerinden öğrenme (network learning), diğer bir ifade ile sanal öğrenme (virtual learning), klasik eğitimden tamamen farklı bir anlayıştır. Süreç açısından bakıldığında e-eğitimde kalabalık sınıflar, zaman sınırlaması, bireysel farklılıklar gibi sorunlar yaşanmamaktadır. Ancak içerik oluşturma, hazırlık aşamaları, sunuş biçimi ve değerlendirme aşamalarında e-eğitim, geleneksel eğitimden oldukça farklı süreçler içermektedir. Her öğrenciye kendi öğrenme hızında bir öğrenim sağlaması (Demirel vd., 2001; Balcı ve

İnceođlu, 2004), aynı eđitim platformunda olan kiřilerin birbirleri ile istedikleri zaman iletiřim kurabilmeleri (Yaman ve Hamedođlu, 2002), bilgisayarlarla daha kısa sũrede hızlı ve daha sistemli řekilde ođrenimin gerekleřebilmesi, sınırsız sayıda konu tekrarı imkanı, ođrenilenlerin daha kalıcı olması, laboratuarda yapılması tehlikeli ve pahalı olan deneylerin simũlasyonlarla yapılabilmesi, ođrencinin geleneksel eđitime gũre daha aktif rol ũstlenmesi, ođrencinin zamanını programlama (Demirel vd., 2001) ve daha ok arařtırma yapma alışkanlıđı kazanması gibi sebeplerle e-eđitim, eđitimde kaliteyi artırmaktadır.

E-eđitimin daha sađlıklı olarak uygulanabilmesi iin, hazırlıktan deđerlendirme ařamasına kadar her adımda daha iyi sonular verecek yũntemlerin arařtırılmasının ȳnemi tartıřılmazdır. “Her zaman ve her yerde eđitim” felsefesi ile “e-eđitim” yaygınlařmaktadır. Bunların sonucunda da eđitimde ierik ũretme, depolama, iletme, ođrenme ve kullanmada yeni sistemlerin geliřtirilmesi gerekmektedir.

E-eđitimin bařlarında ierik HTML sayfaları, animasyon/simũlasyonlar řeklinde hazırlanıp web’e konuluyor ve ođrencinin ȳnceden belirlenen sırada ierikleri gũrmesi sađlanıyordu. Buna ȳrnek olarak, lisans seviyesinde Bilgisayar Ađları dersi “Tıkanıklık Kontrolũ Konusu İin İnternet Tabanlı Bir Uzaktan Eđitim Uygulaması” (Balcı ve İnceođlu, 2004) alıřması gũsterilebilir. Bu alıřmada ođrenciler konu anlatımından sonra bir havuzundan rastgele seilen sorularla sınav olmakta ve geribildirim alabilmektedirler. Bu sistemin ođrencilerin ođrenmelerinde ve ođrendiklerinin kalıcılıđının sađlanmasında etkili olacađı řũphesizdir fakat genelde yeniden kullanımı zor, paylařılamayan ve farklı ođrenci ihtiyalarına cevap veremeyen bir niteliktedir.

Bu alıřmada, e-ođrenime olan ihtiya nedenleri temel alınarak, e-eđitimin ȳđretmen ve ođrenciye katkıları; e-eđitimde ierik geliřtirme sũre ve metotları; ieriđin paylařımı ve yeniden kullanımı gibi konular

incelenmiştir. İçeriğin elektronik ortamda hazırlanması sırasında yapılan en büyük hata, Aggarwal'ın (2003) da incelediği gibi geleneksel öğretimdeki saydamları Internet'e koymaktır. Oysa günümüzde Internet, veri bankaları ve veritabanı sistemleri web-tabanlı uzaktan eğitimin temellerini oluşturmakta; öğrenme nesneleri, öğrenme nesnesi (ÖN) içerik paketleri ve ÖN havuzlarının kullanımı ise yaygınlaşmaktadır. Bu konularda elde edilen veriler ışığında "ülkemizde web tabanlı uzaktan eğitimi nasıl bir adım daha ileriye götürebiliriz?" sorusuna cevap aranmıştır. Web-tabanlı bir uzaktan eğitim altyapı tasarımı yapılması yönünde ilerlenmiş; tasarım sürecinde, ilgili konularda geniş bir kaynak taraması yapılmıştır. Demirel ve arkadaşlarının (2001) belirttiği gibi öğretilecek materyal ve yazılımların hazırlanmasının web-tabanlı eğitimde süreci etkileyen en önemli noktalardan biri olduğu görülmüş ve "*web-tabanlı eğitimde içerik oluşturma ve oluşturulan içeriğin platform bağımsız olarak kullanımı*"nın bu çalışmanın konusunun temelini oluşturmasına karar verilmiştir.

Bu kapsamda, bölüm ikide, önceki çalışmalar incelenmiştir. Üçüncü bölümde, e-eğitimde içerik oluşturmak ve içeriğin platform bağımsız olarak kullanımını sağlamak için gerekli standart ve şartnameler incelenerek bilgi verilmiştir. Dördüncü bölümde, tasarlanan platformun özellikleri ve benzerlerinden farklılıkları açıklanmıştır. Platformun tasarımı, modüller arasındaki ilişkiler ise beşinci bölümde belirtilmiştir. Altıncı ve yedinci bölümlerde, tasarlanan platform benzer işlevleri yerine getiren diğer sistemlerle ve yazarlık araçları ile karşılaştırılmış ve öğrenme modelleri, iletişim araçları, eğitim yazılımlarının özellikleri gibi farklı kriterlere göre değerlendirilmiştir. Son bölümde ise, hazırlanan platforma ait sonuçlar özetlenmiş ve ilerideki çalışmalarla ilişkili olarak önerilerde bulunulmuştur.

2. ÖNCEKİ ÇALIŞMALAR

Web tabanlı uzaktan eğitimde bilgi işleme sırasında içeriğin Öğrenme Nesnesi (Learning Object-LO) halinde küçük birimler olarak ve farklı formatlarda tasarlanabilmesi, eğitsel içeriğin tek bir formata dönüştürülmesi için yaşanan zaman kaybını önlemektedir. Bir havuzda tutulan ve yeniden kullanılabilir olarak tasarlanan bu içerikler, öğrencilerin öğrenme ile ilgili gereksinimlerine göre organize edilerek ders/konu içerik paketleri elde edilmekte ve uyumlu bir eğitim yönetim sistemi üzerinde kullanılabilirler. Öğrenme nesnelerinin üstveri tanımlamalarının yapılması ve oluşturulan içeriğin sıkıştırılarak içerik paketi olarak saklanması ile bu içerikler farklı eğitim yönetim sistemleri ve yazarlık araçları arasında da taşınabilir olmaktadır.

Verilen bu bilgiler ışığında, konu ile ilgili olarak dünyada ve Türkiye’de yapılmış önceki çalışmalar incelenmiştir. Aşağıda, iki grup altında ÖN içerik paketi hazırlamak ve üstveri tanımlamaları yapmak amacıyla yapılan çalışmalardan bazıları ele alınarak özetlenmeye çalışılmıştır. Bunlardan bazıları eğitim yönetim sistemidir. Bunlar için Learning Management System (LMS), Sanal Öğrenme Ortamı (Virtual Learning Environments-VLE) terimleri de kullanılmaktadır.

İncelenen literatürde sadece öğrenme nesnelere ile ilgili olarak yapılan çalışmalara ilk olarak yabancı öğrencilerin Türkçeyi öğrenmesi için yeniden kullanılabilir öğrenme nesnelere kullanılarak hazırlanmış olan “Türkçe Merhaba” projesi örnek gösterilebilir. Bu sistem içerisinde resimler, sesler, örnekler, alıştırmalar birer varlık olarak tasarlanmıştır. Türkçe karakterlerin kolaylıkla kullanımını sağlamak amacıyla kullanılan bir Türkçe klavye benzetimi ve sistemdeki arayüzler gibi birçok değişik nesne, sistem içinde tekrar kullanılabilen bir yapıda tanımlanmıştır (Çağıltay, 2001; Çağıltay and Çağıltay, 2002, 2003).

Öğrenme nesneleri ile ilgili çalışmalara ikinci örnek olarak Indiana üniversitesinin müzik okulu kütüphanesinde bulunan müzik eserlerinin sayısallaştırılarak her müzik parçasının bir nesne olarak üstverileriyle tanımlandığı “Variations” projesi (Indiana University, 2001) verilebilir. Çevrimiçi ortamda öğretim görevlileri ve öğrenciler tarafından derslerde kullanılmak üzere paylaştırılmıştır.

Bir diğer uygulama Indiana Üniversitesi tarafından, 2.300’den fazla fotoğrafın tarih, numara, genel açıklamalar, başlık, özgün fotoğrafa ait kutu ve dosya numaraları gibi imler kullanılarak tanımlandığı “USSteel-2002” fotoğraf koleksiyonudur (Indiana University, 2002). Sayısal ortamda, bu fotoğrafların birer minyatür ve gerçek büyüklükteki görüntüsü ile fotoğraflara ait üstveri tanımlamaları saklanmakta ve çeşitli arayüzler ve tarama sistemleri aracılığıyla bu verilere erişim sağlanmaktadır. Sistem ilk, orta ve lise seviyesindeki bazı derslerde eğitim malzemesi olarak kullanılmaktadır.

İncelenen literatürde görülen başka bir uygulama ise saydamların (slaytların) ÖN’lerine dönüştürülmesine yönelik bir çalışmadır. Bir başlık ve bir resim içeren saydamlardan oluşan bir sunumdaki resimlerin yeniden kullanımını sağlamak için saydamların öğrenme nesnesine dönüşümü otomatik olarak gerçekleştirilmektedir (Verbert et. al, 2004).

Günlük yaşamda öğrenmede, öğrenme nesnelерinin kullanımını artırmak amacıyla bu teknoloji ile yapay zekayı birleştiren bir proje ise incelenen diğer bir ilginç çalışmadır. Çoklu etmen (multi-agent) sistemi yaklaşımının zeki öğretim sistemleri (Intelligent Tutoring Systems–ITS) tasarımında kullanılmasının daha hızlı, verimli ve daha ucuz maliyetli sistemlerini; ÖN’leri ile çoklu etmen sistemlerin yakınsamasının “Akıllı Öğrenme Nesneleri”ni doğuracağına inanılmıştır (Silveria et. al, 2006). Bir etmen, bir ortam içinde sürekli ve otomatik bir şekilde çalışan bir yazılımdır. “Agent Communication Language- ACL” sayesinde diğer

etmenlerle mesaj yolu ile haberleşebilir. LMS etmeninin görevi öğrencilerin öğrenme ortamı ile etkileşimini yönetmek ve akıllı ÖN'leri ile haberleşmektir. Akıllı ÖN modeli, bir ajan, bir “manifest” dosyası ve öğrenme kaynaklarından oluşur. Ajan, ajanın nasıl çalışması gerektiğini açıklayan “manifest” dosyasını okuyabilir. Bu dosya, öğrenme içeriğini barındıran esas dosyalara yani öğrenme kaynaklarına referanslar içerir. Bir akıllı ÖN bu kaynakları öğrencilere dağıtır. Üstveri (IEEE LOM Standart 2004) bilgisi XML olarak tanımlanmış, içerik paketi IMS İçerik Paketleme Bilgisini (IMS 2004) kullanılmıştır (Silveria et al, 2006). Bu çalışmada, ÖN'lerinin arasındaki iletişim için ACL kullanılması ile öğrenme ortamlarında daha güçlü bir haberleşme sağlanacağı fikri ortaya atılmıştır. Bir diğer uygulama ise, web tabanlı öğrenme geliştirme araçlarının geliştirildiği “eLara” projesidir (Zhang et al., 2004).

Türkiye’de yapılan bu kapsamdaki uygulamalara bir örnek olarak, Boğaziçi Üniversitesi Bilgisayar Mühendisliği Bölümü “Bilgi İşleme Giriş” dersi içeriğinin elektronik ortama aktarılması verilebilir. Aktarılırken içeriğin farklı üreticilerin ürettikleri LMS’lerle birlikte çalışabilmesi için SCORM uyumlu tasarlanması zorunluluğu koyulmuştur (Sönmezer ve Bingöl, 2004).

Atatürk Üniversitesinde bir doktora tezi kapsamında ve nesneye dayalı içerik geliştirme uygulamasında kullanılmak üzere geliştirilmiş olan AtaNesA nesne ambarı ise işleyiş ve standartlara uygunluk açısından dünyadaki bir çok açık nesne ambarları ile aynı özelliklere sahiptir. Üst-veri tanımlamaları IMS Learning Object Metadata şartnamesi ile uyumludur. Orta ve yüksek öğretim seviyesinde kimya, fizik, biyoloji ve matematik derslerinin yanı sıra yüksek öğretim seviyesinde öğretim teknolojileri ve programlama dilleri derslerine yönelik 8.000’i aşkın ÖN içermektedir. Türkiye’nin ilk nesne ambarıdır ve Türkçedir. Web arayüzünden nesne kaydına imkan verir, kullanıcılara

nesne hakkında görüş bildirme, ilgilenilen nesneyi sepete ekleme ve nesne üstverilerini görme imkanı sunar (Karaman,2005).

Bu çalışmalar dışında, öğrenme içeriğinin paketlenmesi ve üstveri tanımlamalarının yapılması ile ilgili örnek çalışmalar, incelenen literatür arasında daha fazla yer tutmaktadır. Bu kapsamda ilk olarak, internet tabanlı bir kurs ve web sitesi tasarımı yapılmasına izin veren bir LMS olan “MOODLE” (Moodle, 2007), Moduler Object Oriented Dynamic Learning Environment, sayılabilir. Bir eğitim yönetim sistemi olduğundan, ders içeriği oluşturmak dışında forum, sınav, küçük sınav, sohbet gibi farklı etkinlikler de içermektedir. Moodle, Linux üzerinde Apache, PHP ve MySQL kullanılarak geliştirilmiştir ve Türkçe dahil olmak üzere 60 dil sürümü vardır. Sistemde, site yönetiminden sorumlu, kurs oluşturmak ve bunlara öğretmen atamak gibi işlemleri gerçekleştiren *administrator* ve kurs oluşturmak ve öğretmekle yükümlü *course creator* olmak üzere 2 kullanıcı vardır. Kullanıcıların sisteme girmeleri kimlik doğrulama gerektirmektedir. Moodle sisteminde oluşturulan kurslar arşiv amaçlı olarak sıkıştırılarak tek dosyaya paketlenilmektedir. Kurs içeriğinde kullanılacak kaynakların sisteme yüklenmesi sırasında bir seferde en fazla 2 Mbyte büyüklüğünde ve tek bir dosya yüklenebilmesine izin verilmiştir. Kurs ve web sitesi oluşturmak için kullanılan Moodle, Internet tabanlıdır.

Bir diğer açık kaynaklı eğitim yönetim sistemi “dotLRN” dir (dotLRN, 2007). dotLRN, Unix ve Linux işletim sistemlerinde çalışmaktadır. Moodle gibi kurs yönetimi, kurs ekleme, güncellemeleri görme, kurs üstverisinin girişi, “manifest” dosyasına bilgi girişi, klasörlere öge ekleme, ögelere üstveri ekleme, ögelere kaynaklar ekleme, ÖN’lerini araştırma, kursu sıkıştırılmış bir dosyaya çevirme, dosyaları diğer kurslarda da kullanabilmek gibi içerik hazırlamak ile ilgili işlevleri kapsamaktadır. dotLRN, bir ÖN havuzuna sahiptir ve oluşturulan

nesnelerin önzilmesi mümkündür. Yönetici, eğitmen ve öğrenci olmak üzere üç tip kullanıcı tanımlanmıştır. Eğitmen sisteme kişisel dosya saklama alanında kendi oluşturduğu ÖN'lerini görebilmektedir. Ayrıca, kurs arama, kurs ekleme ve ÖN arama işlemleri de yapabilir. Platformda, yüklenen tüm dosyalar birer sürüm numarası alır ve eski sürümler saklanabilmektedir. dotLRN sistemine sıkıştırılmış dosya formatında SCORM içerik paketleri eklemek mümkündür. Ancak, yüklenecek içerik paketi ile ilgili üstveri başlığı, tipi, telif bilgisi gibi bilgiler görüntülenerek, kursa isim verilip onaylanarak nesne havuzundan ilgili paketinin seçilmesiyle o kursa ait bilgilerin görülebilmektedir.

University of Louvain tarafından geliştirilmiş ve 28 dile çevrilmiş olan "CLAROLINE" (CLAROLINE, 2007)) ise web üzerinde kurs oluşturma ve yönetimi amaçlı açık kaynak kodlu bir öğrenme ortamıdır. Tercihen Apache web sunucu üzerinde, PHP ver4.1 ya da üstü ve MySQL ver3.23 ya da üzeri kullanılması tavsiye edilir. Sistem, SCORM paketlerini destekler. Sistem grup yönetimi, forum, tartışma, belge havuzları, öğrenci yazıları, takvim, sohbet, değerlendirme alanları, kullanıcı profili yönetimini, duyuru, gündem, ders açıklaması sağlamaktadır. Yönetici, öğretmen ve öğrenci olarak 3 tip kullanıcı vardır. Sisteme eklenebilecek dosya boyutu 10MB ile dosyaların toplam boyutu ise 30MB ile sınırlandırılmıştır. Sistemde ÖN hazırlamak ile ilgili bir kısım bulunmamaktadır. Ancak SCORM içeriği yüklenebilmektedir.

Granada Learning tarafından oluşturulan "LEARNWISE" (LEARNWISE, 2005) ise Web tabanlı içeriğin dağıtımını, çevrimiçi değerlendirme, işbirlikli araçlar, raporlama gibi özellikler sunan bir eğitim yönetim sistemidir. IMS içerik paketleme, IMS Metadata/IEEE LOM desteği vardır. Learnwise Publisher sayesinde içerik yazımı çevrimdışı olarak sağlanır. Kurs daha sonra standart IMS formatında kaydedilerek sunucuya yüklenebilir ya da LearnWise'a ait yayımlama

ortamında yayımlanabilir. Sistemde tamamen yeni bir içerik oluşturmak ve oluşturulan web sayfasını sisteme yüklemek mümkündür. Learnwise, açık kaynak kodlu bir yazılım değildir.

Bir başka LMS olan “Teknical's Virtual Campus” (Teknical's Virtual Campus, 2005) ise yeniden kullanılabilir ÖN'leri ve üstveri tanımlamaları yapmak için kullanılabilir. İçerik oluşturma işlemi Microsoft Word kelime işlemci programı ile yapılmaktadır. Sistemdeki tüm ÖN'lerinin üstveri tanımlamaları yapılmıştır ve bir havuzda tutulmaktadır. Bu havuz, IMS ve SCORM üstveri ve içerik paketleme şartnamelerine göre oluşturulan içeriği kullanmak üzere tasarlanmıştır.

Açık kaynak kodlu olan “ATUTOR” (ATUTOR, 2007), PHP ile ve MySQL veritabanı kullanılarak yazılmış başka bir eğitim yönetim sistemidir. Öğrenciler içerikle ilgili başlıkları “TILE” (TILE, 2007) denilen ÖN havuzundan çevrimdışı olarak çalışmak üzere araştırabilirler. Bu havuzdaki içerik paketleri yerel saklama alanlarına indirilip görülebilir ya da direk ATutor'a aktarılabilir. IMS/SCORM içerik paketleme şartnamelerine uyumlu olduğundan, içerik hazırlayanlar yeniden kullanılabilir içerikler hazırlayabilirler. Eğitimciler eğitsel içerikleri HTML ya da text formatı olmak üzere iki farklı şekilde oluşturabilirler. Bunlar yerel bir editörden yüklenebileceği gibi direk çevrimiçi olarak da girilebilir. Sisteme yüklenebilecek dosya boyutu 1MB ile sınırlandırılmıştır ve sisteme birden fazla sayıda dosya yüklemek istenirse, dosyalar sıkıştırılarak tek dosya halinde yüklenebilmektedir. Bu sistemde kullanılan TILE nesne havuzunun işlevselliği 3 kısımda incelenebilir. Bunlardan ilki paylaşılabilir ÖN'leridir ve ÖN havuzu bir kütüphane görevi görür. İkinci seviyede ÖN'nin yeniden amaçlandırılmasını sağlar. Öğrenme içeriği yazarların belirli müfredat gereksinimleri için eğitimcilerin bileşenleri ÖN'ne adapte edebilmek için değiş-tokuş edebilmesine imkan verir. Üçüncü

seviyede ise öğrencinin ihtiyaçlarına ve tercihlerine göre öğrenme içeriği yeniden şekillendirilebilir.

WebCT (WebCT, 2005) ise İngiltere'deki en popüler sanal öğrenme ortamıdır. İçerik yönetim ünitesi, IMS içeriğinin alınıp verilmesine imkan verir. İçerik yönetim ünitesi ReLOAD (ReLOAD, 2007), Reusable Learning Object Authoring and Delivery, içerik paketleriyle çalışır. Bunlar manifest içinde “metadata” elemanı içerirler. WebCT ile, birden fazla manifest kullanımında sorun yaşanabilmektedir.

Bu çalışma sırasında örnek ÖN içerik paketleri hazırlamak amacıyla sıkça kullanılan bir editör olan “ReLOAD Editor” ise, içerik paketi ve üstveri editörüdür ve web uygulaması değildir. Reusable Learning Object Authoring and Delivery anlamına gelen ReLOAD, bir Java uygulamasıdır ve açık kaynak kodludur. Diğer araçlar ile oluşturulmuş olan içeriğin paketlemesi, yeniden organize ederek mevcut içeriğin tekrar amaçlandırılması, içerik havuzlarında saklanmak üzere içeriğin hazırlanması ve son kullanıcıya içeriğin önizleme ile dağıtılması fonksiyonlarını yapabilmektedir. ReLOAD Editor’e eğitsel içeriğin bir LMS’e bağlı kalınmadan sanki tarayıcı üzerinden izleniyormuş gibi görüntülenebilmesi için “Reload Content Package Preview” fonksiyonu eklenmiştir. İçerik paketin haricinde bağımsız dosyalar için de üstveri tanımlamaları yapılabilmekte ve bunların bağımsız “xml” dosyaları olarak kaydedilmesine izin verilmektedir. Üstveri girişi için form ya da ağaç yapıda ya da her ikisinin karışık olduğu bir arayüz formatı seçimi kullanıcıya bırakılmıştır. Üst-veri sadece IMS-LRM (IMS Learning Resource Metadata) ile sınırlı kalmayıp, UKCMF v.1.0 (UK Learning Object Metadata Common Framework) ve LTSN v1.0 (Learning and Teaching Support Network) profilleri de seçilebilir. RELOAD Editor, IMS içerik paketi (v1.1.3) ve üstveri spesifikasyonlarına (v.1.2.2) uyum sağlar. Ayrıca editörün 1.3 sürümü, SCORM 1.2 paketlerini de

desteklemektedir. Editörün son sürümü (ver2.5.4) ise SCORM'un son sürümü olan SCORM2004 standardı ile uyumludur.

Reload Editor'ün web tabanlı ve halen geliştirilmekte olan sürümü (beta 0.9) “WeLOAD”, Web-based Learning Object Authoring and Delivery, (WeLOAD, 2007) kapsamında sisteme dosya yükleme, içerik paketi oluşturulması ve bunların arşivlenmesi, sıkıştırılmış dosyalarının indirilebilmesi gibi işlemler tamamlanmıştır. Üstveri girişi için, RELOAD Editor kullanılmaktadır. WeLOAD, PHP ve MySQL ile geliştirilmektedir. Yazılımın taşınabilir web sunucu kısmında Apache(1.3) server, PHP(5.04) ve MySQL (3.23.47) kullanılmış, WeLOAD kısmında ise PHP ve scriptler kullanılmıştır. İçerik paketi oluştururken dosyalar sunucuya tek tek yüklenmektedir, ya da web üzerindeki bir adres içeriğe eklenebilmektedir. Eklenen dosya tıkladığında dosya içeriğinin yeni bir pencerede görüntülenmesini sağlayan kolay kullanımlı bir arayüz oluşturulmuştur.

İçerik paketleme amaçlı kullanılacak başka bir editor olan “IMS Packet Editor” (IMS Packet Editor, 2007) ise Kasım 2001'de yayınlanmış olup IMS Global Learning Consortium'un “İçerik Paketleme Şartnamesi”nin bir uygulamasıdır (Cebeci ve Cimilli, 2004). IMS Paket Düzenleyici, ÖN'lerinin bir ders paketi (sıkıştırılmış zip dosyaları şeklinde) içine eklenmesi, organize edilmesi ve depolanmasına olanak sağlar. IMS Paket Düzenleyici ile yaratılan ders paketleri, IMS İçerik Paketleme şartnamesini destekleyen (Blackboard, WebCT, Macromedia Dreamweaver veya Microsoft LRN Toolkit gibi) öğrenim ortamları ve yazarlık araçları arasında işletilebilir ve yeniden kullanılabilir durumdadırlar. Düzenleyici, ÖN'lerini açıklamak üzere “IMS LOM Tanımlayıcı Bilgileri”nin (IMS Learning Objects Metadata)” yaratılması /düzenlemesini de yapabilmektedir. Yeni bir ders yaratırken, öncelikle kullanıcının pakete verdiği isim ile aynı isimde bir klasör ve de boş bir

zip dosyası yaratılmaktadır. Pakete dersler eklendikten sonra sırayla bu derslere öğeler (yani dosyalar, URL'ler) eklenir, tanımlamalar ve başlık bilgileri girilir. Sırayla dosyalar klasöre kaydedilerek güncellenir. Paket içinde bulunması gereken “Manifest dosyası” da güncellenerek klasör ve sıkıştırılmış dosya içine yerleştirilir. IMS Paket Editor, web tabanlı değildir fakat açık-kaynak kodlu bir uygulamadır.

LOAT- Learning Object Authoring Tool (Liu, Huang, Chao, 2005) ise Çince tasarlanmış bir ÖN yazarlık aracıdır. C++ ve Microsoft Access veritabanı kullanılmıştır. SCORM uyumlu ÖN yaratmak, uzaktaki bir ÖN havuzundan öğrenme kaynaklarını indirebilmek, “manifest” eklemek ve aynı zamanda öğrenme içeriğini tanımlamak, öğrenme kaynaklarını sıkıştırmak ve ADL'in sağladığı platforma yükleyebilmek gibi fonksiyonları vardır. Öğrenme ortamında “sistem geliştirici” ve “içerik sağlayıcı” olmak üzere 2 kullanıcı tanımlanmıştır. ÖN'leri ve üstveriler ayrı havuzlarda tutulmaktadır. LOAT sistemi 7 modülden oluşmuştur. Bunlar, eğitmen ile sistem arasındaki etkileşimi ve diğer modüllerle iletişimi sağlayan “User interface”, yerel makinada öğrenme kaynaklarını tutan “Learning Resource”, uzaktaki bir öğrenme nesnesi havuzuna erişimi sağlayan “Remote Learning Resource”, “manifest” dosyasını oluşturan ve kaynak eklenip silinmesine izin veren “Material navigation”, içerik paketi oluşturmak için “Content package” ile “Metadata” ve “Preview” modülleridir. Tanımlanan üstveriler, kataloglama ya da yeniden kullanıma uygun olması için, ilgili öğrenme nesneleri ile aynı dizinde tutulmaktadır.

Elektronik ÖN'leri oluşturmaya, doğrulamaya yönelik bir diğer araç “ELO Tool- Electronic Learning Object Tool”, aynı zamanda eğitsel üstveri tanımlamaları yapılmasına izin veren bir araçtır. IMS, Learning Object Metadata (LOM), Alliance of Remote Instructional Authoring and Distribution Networks for Europe (ARIADNE, 2005), DCMI (DCMI,

2007) gibi farklı üstveri şemaları arasında çeviri yapabilir (Liliana at all, 2003) ve farklı “xml” teknolojilerini destekler. Araç 3 modülden oluşmaktadır. “xml” dosyalarının tutulduğu bir havuz, ÖN’lerinin orijinal üstverilerini ELO üstveri formatına çeviren ve farklı üstveri şemaları arasında çeviri yapan bir dönüştürücü ve belge oluşturma, doğruluğunu onaylama, güncelleme, silme fonksiyonlarının yerine getirilmesini sağlayan bir belge oluşturucu içermektedir.

Aşağıda özetlenen uygulamalar ise sadece üstveri girişi amaçlı kullanılan programlardır ve eğitsel içerik paketleme ve SCORM standardına uyum özellikleri yoktur. Bu uygulamalar ImseVimse, LOM, EM², CLOMAT ve TreeLOM olarak sıralanabilir.

İsveç’te The Royal Institute of Technology in Stockholm’da Java dilinde yazılmış açık kaynak kodlu IMS tabanlı bir ÖN üstveri tanımlama editörü olan “Imse Vimse”, IEEE LOM spesifikasyonunu desteklemez (Nilsson & Palmér, 2002). Ayrıca bu editörün eğitsel içerik oluşturma ve paketleme gibi fonksiyonları da yoktur.

Sadece üstveri tanımlamalarını yapabilmek için kullanılan Learning Object Metadata (LOM) Editör (Technische Universität Darmstadt, 2001), IEEE LOM desteği olan ve Java dilinde yazılmış bir uygulamadır. Almanya Darmstadt University of Technology tarafından geliştirilmiştir.

EM² (Sampson et. al, 2002) yani “An Environment for Editing and Management of Educational Metadata” ise eğitsel üstveri oluşturmak, değiştirmek, veri ve altyapının doğrulanması, gelişen XML teknolojilerinin desteklenmesi, öğrenme standart ve spesifikasyonlarının desteklenmesi, standartlar arasında dönüşüm, üstveri yönetimi fonksiyonlarını yerine getirir. IMS LOM standardını destekleyen bir programdır. Java dili ve XML Document Object Model (XMLDOM) tabanlıdır ve Microsoft Windows NT platformunda yazılmıştır. Sistemde,

öğrenci, eğitmen ve yönetici olmak üzere 3 tip kullanıcı tanımlanmıştır. “xml” dosyaları havuzunda tutulan her üstveri dosyası aynı zamanda bir veritabanında da tutulmaktadır. Kullanıcı, “xml” dosyalarını bir eğitsel standarttan diğerine dönüştürebilir.

CLOMAT, Customizable Learning Object Metadata Authoring Tool (Malaxa, 2003) ise Florida State üniversitesinde SCORM tabanlı LOM girişini sağlamak için geliştirilen çok fonksiyonlu bir yönetim ortamıdır. Web tabanlı olarak ASP.NET ve C# , SQL veritabanı kullanılarak geliştirilmiştir. SCORM üstveri modeli ve yeni tanımlanabilecek bir üstveri modeli olmak üzere birden fazla üstveri modeli tanımlanmasına imkan verir. Birden fazla dil desteği vardır. Üstveri girişi ağaç yapı görüntüsünde bir arayüz üzerinden gerçekleştirilir.

Web üzerinden (html dosyaları ile) ÖN üstverisi oluşturmada kullanılan bir araç olan “LOMGen- The Learning Object Metadata Generator”, Kanada New Brunswick üniversitesinde 2004’de Java programlama dili kullanılarak yazılmıştır ve oluşturulan üstveri dosyaları CANLOM üstveri havuzunda tutulmaktadır (Singh, et.al, 2004).

Son olarak incelenen üstveri oluşturma aracı, Çukurova üniversitesi tarafından 2004 yılında geliştirilen ve bir ÖN yazarlık aracı olan “Tree View LOM Editor (TreeLOM)”dur (TreeLOM, 2006). Bu, web tabanlı bir editördür ve erişilebilen mevcut sürümü beta1.0’dır. dotNET platformunda geliştirilmiş bir projedir. MS Windows sunucu ve IIS (Internet Information Server), MS ASP.Net platformuna ek olarak C#, JS, HTML, DHTML dilleri kullanılmıştır. TreeLOM, bağımsız bir uygulama olarak ya da bir Learning Content Management System-LCMS’e ya da bir ÖN havuzuna alt sistem olarak dahil edilerek kullanılabilir. Uygulama, LOM Draft Standard (IEEE 1484.12.1-2002)’ını desteklemektedir. Ağaç görünümdeki arayüz, geleneksel ya da form görünümüne göre kullanıcının üstveri girişini daha kolay

yapabilmesini sağlamaktadır. Böylece üstveri elemanlarının tüm giriş alanları tek bir ekranda görüntülenebilmektedir.

Öğrenme nesnesi içerik paketi oluşturma ve üstveri girişi amacıyla kullanılan ve yukarıda incelenen araçlar üç farklı çizelge halinde karşılaştırılmıştır. Çizelge 1.1’de incelenen toplam yedi tane LMS içinde sadece dördü açık kaynak kodlu uygulamalardır. Eğitim yönetim sistemlerinin genelde programlama dili olarak PHP, veri tabanı olarak MySQL/SQL, web server olarak Apache kullanıldığı görülmektedir. Çizelge 1.2’de üstveri tanımlama amaçlı kullanılan araçlar görülmektedir. Bunlar Java programlama dili kullanılarak veya ASP.Net platformunda yazılmış, IMS ya da IEEE LOM standartlarını destekleyen yazılımlardır. Çizelge 1.3’te ise hem içerik paketleme hem de üst-veri tanımlama fonksiyonlarına sahip araçlar gösterilmiştir.

İncelenen araçlardan sadece ikisi, RELOAD ve TreeLOM, üstveri tanımlamaları için ağaç yapısında bir arayüz sağlar. Ancak ağaç yapısında veri girişi kullanıcının tüm kategorleri aynı ekranda görmesini sağladığından, form yapıya göre daha kolay kullanım imkanı verir. Çizelge 2.1, Çizelge 2.2 ve Çizelge 2.3’te veri içermeyen hücreleri tanımlayan özellikler ya ilişkili platform tarafından desteklenmemektedir, ya da incelenen kaynaklarda ilgili konuda bilgiye rastlanmamıştır.

Bu bölümde, öğrenme nesnelерinin web-tabanlı uzaktan eğitim sistemindeki yeri, ÖN’leri kullanılarak yapılmış çalışmalar, eğitsel içerik paketi hazırlama ve üstveri tanımlama araçları incelenmiş ve karşılaştırmalı olarak sunulmuştur.

Bölüm 3’te ise, tez çalışmasında tasarlanan platformun yeniden kullanılabilir eğitsel içerikler hazırlayabilmek için desteklemesi gereken standart ve şartnameler ile ilgili bilgiler verilmektedir

Çizelge 2.1 İncelenen Eğitim Yönetim Sistemlerinin Karşılaştırılması (LMS)

LMS	Açık k. Kodu	Dil	Veri Tabanı	İşletim Sistemi	LO hazırlama	LO havuzu	LO yükleme	üstveri	Üstveri havuzu	İçerik paketi desteği	On izleme
Moodle	✓	PHP	MySQL	Windows, Mac, Linux	✓	✓					
dotLRN	✓			Unix, Linux	✓	✓	✓	✓		✓	✓
Claroline	✓	PHP	MySQL		X		✓			✓	
Learnwise	X		SQL	Win NT /2000	Çevrimdışı olarak			Ims, IEEE LOM	✓	✓	✓
Teknical's Virtual C.					✓	✓		IMS, SCORM	✓	✓	✓
Atutor	✓	PHP	MySQL		✓	✓		IMS		✓	
WebCT							✓			✓	

Çizelge 2.2 Üstveri editorleri

Üstveri Editörleri	Açık k. Kodu	Dili	Veri Tabanı	İşletim Sistemi	Üstveri	Üstveri havuzu
Imsevimse	✓	Java			IMS LOM	
LOM Editor	✓	Java		Win, Unix, Linux	IEEE LOM	
EM²	✓	Java		Windows NT	IMS, IEEE LOM	
Clomat		ASP.NET C#	SQL	WIN	IEEE LOM	
LOMGen	✓	Java				CANLOM
TreeLOM	✓	ASP.NET C#		windows	IEEE LOM	✓

Çizelge 2.3 İçerik Paketleme ve Üstveri Editörleri

Üstveri Editörleri	Açık kaynak Kodu	Dili	Veri Tabanı	İşletim Sistemi	LO havuzu	Üstveri	Üstveri havuzu	İçerik paketi desteği	Ön izleme
RELOAD	✓	Java		Windows, Macintosh OSX, Linux		IMS, IEEE LOM		✓	✓
LOAT		C++	Access	Win	✓	✓	✓	✓	✓
ELO						IMS LOM			
WeLOAD	✓	PHP	MySQL						✓
IMS package editor	✓	Java				IMS LOM			

3. ÖĞRENME NESNELERİ VE SCORM STANDARDI

Bölüm 2’de ÖN’leri ve içerik paketlerinin oluşturulması, üstveri tanımlamalarını sağlayan araçlar incelenmiş ve karşılaştırmalı olarak gösterilmiştir. Burada ise, öğrenme nesneleri hakkında detaylı bilgi verilerek, yeniden kullanılabilir ÖN’leri ve içerik paketlerinin oluşturulmasında bir standart olan SCORM üzerinde durulmaktadır.

3.1 Öğrenme Nesnesi ve Nesne Havuzu Kavramları

Eğitim yönetiminde özel bir rolü olan öğrenme nesneleri, “bir eğitim yönetim sistemi tarafından sunulan, saklanan, kataloglanan ve raporlanan içerik parçaları (Aslantürk, 2002)” olarak tanımlanmaktadır. Bilgisayar programcılığındaki nesne yönelimli programlamayı baz alan ÖN’lerinde temel, içeriği parçalarına bölmek ve özel öğrenme hedeflerine göre yeniden birleştirmektir. Wiley (2000), öğrenme nesnelərini “öğrenmeyi desteklemek amacıyla yeniden kullanılabilen kaynaklar” olarak ifade etmiştir. IEEE Learning Technology Standards Committee-Learning Object Metadata (LTSC LOM) grubu (IEEE LOM, 2007), ÖN’lerini “teknoloji destekli öğrenim sırasında kullanılabilen, yeniden kullanılabilen, referans verilebilen sayısal ya da sayısal olmayan veriler” (IEEE, 2002) olarak tanımlamaktadır. Özetle, bir ÖN kendi başına bir bütün ve aynı zamanda bir bütünün parçası olabilen, üst verisi ile tanımlanan ve paylaşılabilen yeniden kullanılabilen (Aşkar, 2003) ve kişiselleştirilmiş öğrenme sağlayan varlıktır.

Öğrenme nesneleri, bilgi yakalamaya yönelik nesnelər, karar vermeyi destekleyen nesnelər, eğitim tasarımına yönelik nesnelər gibi farklı biçimlerde (Yalvaç ve Bayraktutan, 2004; Çağıltay ve Çağıltay, 2003) olabilirler. Eğitim tasarımına yönelik bir ÖN, basit bir resim dosyası, bir müzik veya metin parçasından bütün bir derse, hatta dersler havuzuna kadar büyük ve daha karmaşık yapılar olabilir. Örneğin bir

video filmi, prosedürler, hikayeler, simülasyonlar, değerlendirmeler, durum çalışmaları, sayısal resimler, canlandırma, eğitsel yazılım ve yazılım araçları, teknoloji destekli öğrenme sırasında atıf yapılan kişiler organizasyonlar, olaylar ya da metin, resim, diğer medya ve uygulamaları içeren web siteleri birer ÖN olabilir.

Öğrenme nesnelерinin eğitimci ve öğrencilere faydaları çeşitli kaynaklarda (Cebeci, 2003; Kaya ve Bingöl, 2004; Shepherd, 2000) anlatılmıştır. İçerik geliştiricilere faydaları bir dersi sıfırdan oluşturmak yerine hızlı, ucuz ve az emekle modüler ve paylaşılabilir parçaları kullanmak şeklinde özetlenebilir. Hazırlanan eğitim malzemeleri daha etkin ve yeterli olur. Ayrıca farklı platformlardan taşınabilirlik, dayanıklılık, yazarlık sistemleri arasında paylaşılabilirlik, web üzerinden erişim sağlarlar. Öğrenciler açısından bakıldığında ise “Yeteri kadarını alma (just enough)”, “zamanında ve hızlı şekilde ulaşma (just in time)” ve “kişiyeye özel öğrenim (just for person)” olanakları sunarlar. Yani öğrenci bir kursun sadece bilmediği kısımlarını alabilir, ÖN’leri araştırılabilir olduklarından istenilen içerik aratılarak istenildiği zaman çalışılabilir ve bireyler için özel olarak da içerik takibi yapılabilir.

Küçük ve yeniden kullanılabilir parçalar olan öğrenme nesnelерinin özellikleri aşağıdaki gibi ifade edilebilir.

- Bağımsız şekilde saklanabilirler.
- Zorunlu olmamakla birlikte üst-veri tanımlamaları sayesinde web üzerinden kolay erişilebilir ve güncellenebilirler (Shepherd, 2000; Cebeci, 2003; Kaya ve Bingöl, 2004; Balcı ve İnceoğlu, 2005).
- Etkileşimlidirler yani öğrencinin içeriği okuması, dinlemesi, cevap vermesi ve etkileşime girmesini sağlayabilirler.
- Üretilirler yani öğrencinin öğrenim ihtiyaçlarını karşılayacak biçimde otomatik olarak birleştirilebilirler.

- Uzunlukları birkaç saniye ile 30dk arasında değişebilir (Shepherd, 2000) fakat, genelde 2-15 dk arasındadır (Aşkar, 2003).
- Birden fazla ÖN biraraya gelerek bir ders ya da üniteyi oluşturabilir. Fakat öğrenme nesnelерinin geliştirilmeleri zaman alıcı ve zordur (Wiley, 2000; Aslantürk, 2002).
- Her ÖN, standart bir yöntem kullanarak hem yazarlık sistemleri arasında paylaşılabilir (Robson, 2000; Aslantürk, 2002; Balcı ve İnceođlu, 2005), hem de bir LMS ile iletişim kurabilmelidir. Fakat kullanılabilir olmaları için mutlaka bir LMS'e gerek yoktur. ÖN içerisinde ne olup bittiđi, LMS'ten bağımsız olmalıdır.

Öğrenme nesneleri temel (fundamental), birleştirilmiş-kapalı (combined-closed), birleştirilmiş-açık (combined-open), üretken-sunuş (generative-presentation) ve üretken-eđitsel (generative-instructional) olmak üzere 5 tipte olabilir (Wiley, 2000). *Temel* tipteki nesnelер, başka bir nesne ile ilişkilendirilmemiş dijital kaynaklardır. Eđer nesne, tasarım aşamasında yaratan kişi tarafından az sayıda dijital kaynađın bir araya getirilmesiyle oluşturulursa *birleştirilmiş-kapalı* tipte olurlar ve bunlar yeniden kullanım için erişilemezler. Örneđin bir video klipteki resimler ve ses kaydı birleştirilmiştir ve ne resimlere ne de sese ayrı ayrı erişim sağlanamaz. Diđer taraftan, istek geldiđinde çok sayıda dijital kaynađın bilgisayar tarafından birleştirilmesi ile *birleştirilmiş-açık* tipte ÖN'leri oluşturulur. Öđelerine yeniden kullanım için erişilebilen bu gruba en iyi örnek web sayfalarıdır çünkü web sayfasının resim, video, metin gibi bileşenleri yeniden kullanılabilir formatta bulunurlar. Dördüncü tip olan *üretken-sunuş* grubu temel ya da birleştirilmiş-kapalı tipteki nesnelерin birleştirilmesiyle oluşturulur. Son olarak birleştirilmiş-açık tipi dışındaki ÖN'lerinin birleşmesinden oluşan ve öğrencilerin etkileşimini bu kombinasyonlarla değerlendirmeye çalışan *üretken-eđitsel* tipteki ÖN'lerinden bahsedilebilir. Bu gruba örnek olarak bir dizi adımı

hatırlayıp uygulamayı içeren nesnelere verilebilir. Çizelge 3.1’de, ÖN tasarımında yeniden kullanılabilen bileşenler (*Reusable component objects*) ile ÖN’sini oluşturan bileşenlerin bağımsız olarak yeniden kullanılabilir olup olmadığı açıklanmaktadır.

Öğrenme nesnelere için “bilgi nesnelere (IO, information objects)”, “öğretimsel nesnelere”, “içerik nesnelere”, “ortam nesnelere (media objects)”, “bilgi bitleri” gibi çok sayıda farklı terim kullanılmıştır (Karaman, 2004). Ancak öğrenme nesnelere ile “bilgi nesnelere” birbirine karıştırılmaktadır. Çünkü ÖN “*bir hedef, bilgi, değerlendirme ve bunların bir öğretim stratejisi kullanılarak sunulmasından ibaret olan kendi başına çalışır birim*” (Karaman, 2004) iken, bilgi nesnelere (BN) “*sadece aktarılacak bilgilerden oluşan içerik birimi*” şeklinde tanımlanmaktadır. Yani fotoğraflar, videolar, animasyonlar, tablolar ve veritabanları yalnız başlarına sadece bilgi sağlayan parçalardır ve eğitsel anlamları yoktur. Bu nedenle, bunlara, pedagojik bazı değerlerin de eklenmesi gerekir. Birbirinden bağımsız, Yeniden Kullanılabilir Bilgi Nesnelere (RIO, Reusable Information Object)” biraraya gelerek “Yeniden Kullanılabilir Öğrenme Nesnelere (RLO, Reusable Learning Object) oluştururlar (Çağiltay ve Çağiltay, 2003). Çizelge 3.2’de BN’leri ve ÖN’leri arasındaki farklar belirtilmiştir.

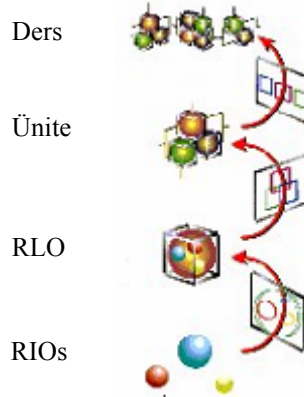
Çizelge 3.1 Öğrenme nesnelarının tiplerine göre özellikleri (Wiley, 2000).

	(1) Fundamental	(2) combined-closed	(3) combined-open	(4) generative-presentation	(5) generative-instructional
Eleman sayısı	Bir	Bir kaç	Çok	Birkaç- çok	Birkaç-çok
Kullanılan nesne tipleri	1	1, 2	Tümü	1, 2	1, 2, 4
Yeniden kullanılabilen bileşen nesneleri	Uygun değil	Yok	Var	Var / Yok	Var / Yok
Genel fonksiyonu	Bilgisayarda görüntüleme, sunma	Önceden tasarlanmış ders /uygulama	Önceden tasarlanan ders ve/veya uygulama	Bilgisayarda görüntüleme, sunma	Bilgisayarca oluşturulan ders ve/veya uygulama
Nesne bağımlılığı	Yok	Yok	Var	Var / Yok	Var
Nesnedeki mantık	Uygun değil	Yok /cevap anahtarına dayanan puanlama	Yok /alana bağı eğitim ve değerlendirme stratejileri	Alana bağı sunum stratejileri	Alan bağımsız sunum, eğitim ve değerlendirme stratejileri
Farklı içeriklerde kullanılabilirlik	Yüksek	Orta	Düşük	Yüksek	Yüksek
Aynı içerikte yeniden kullanılabilirlik	Düşük	Düşük	Orta	Yüksek	Yüksek

Çizelge 3.2 Bilgi nesneleri ve öğrenme nesneleri arasındaki farklar

Bilgi Nesneleri	Öğrenme Nesneleri
<ul style="list-style-type: none"> - Sadece Jpg, mov ya da html uzantılıdır. - Öğretimsel değeri olabilir. - boyut ve süre sınırı var. - Yönetilmeleri kolaydır. 	<ul style="list-style-type: none"> - Genellikle birden fazla dosyadan oluşur (html, jpg, mov) - Öğretimsel değeri olmak zorundadır. - Kesin bir dosya boyutu ve süre belirtilemez - Dizin yapısındadırlar. - Bir nesne, birden fazla dosya ve üst-veri kaydı içerebilir. - Yönetimi oldukça zordur - İçerik paketleme sözkonusudur.

Cisco RIO ve RLO'ların bir öğrenme sistemi hiyerarşisindeki yerini "Ders programı-Ünite-Modül-Ders-Konu/Sayfa" olarak göstermektedir. Ders, yeniden kullanılabilir öğrenme nesnelere, konu ya da sayfa ise yeniden kullanılabilir bilgi nesnelere karşılık gelmektedir. Bağımsız RIO'lar bir araya gelerek RLO'ları oluştururlar. Şekil 3.1'de, bir ders içinde RIO ve RLO'ların yerini göstermektedir.



Şekil 3.1 Bir dersin yapısında bilgi ve öğrenme nesnelерinin yeri (Cebeci, 2003).

Küçük parçalara ayrılarak hazırlanan ders içeriklerinin bir havuzda toplanması, farklı eğitim hedeflerine göre bu havuzdan alınan içeriğin belirli yöntemlerle biraraya getirilip kullanılması, havuzdaki içeriklerin güncellenebilmesi ÖN ambarı, diğer bir ifade ile ÖN içerik havuzları ile mümkün olmaktadır. Böylece, öğretim materyallerinin paylaşımı ve kullanımında zaman ve emek tasarrufu sağlanır, herkes en iyi kaynağa erişebileceği için öğretimin kalitesi artar ve kaynakların güvenli bir şekilde merkezi olarak yönetildiğinden emin olunur (Karaman, 2004)

ÖN, genelde nesnelere ve tanımlayıcı bilgileri olmak üzere 2 bölüme oluşur. İçerik parçaları ile birlikte tanımlanan *üstveriler* bu nesnelere taranarak ulaşılabilmesini kolaylaştırırken, ortak kullanımını da artırmaktadır. Nesnelere öğrenim amaçları doğrultusunda etkin ve doğru biçimde kullanılabilmesi için ÖN ambarlarının bulma, önizleme ve yayınlama işlevleri sağlaması gereklidir (Karaman, 2004).

Dünyada en çok bilinen nesne ambarı, özellikle yüksek öğretim öğretmen ve öğrencileri için tasarlanmış, ücretsiz ve açık bir kaynak olan Merlot (Merlot, 2007) nesne ambarıdır. Diğer nesne ambarlarına örnek olarak TheGateway (TheGateway,2007), SCHOOLNET (SCHOOLNET, 2007), Careo (Careo, 2007), POOL (POOL, 2005), EdoSource (EdoSource, 2007), ESCOT (ESCOT, 2007), Digital Library for Earth System Education-DLESE (DLESE, 2007), CEN/ISS (CEN/ISS, 2007), SMETE (SMETE, 2007), MARCOPOLO (MARCOPOLO, 2005), IMS (IMS, 2007), ADLNet (ADLNet, 2007), IEEE LTSC (IEEE LTSC, 2007), International Organization for Standardization –ISO (ISO, 2007), Alliance of Remote Instructional Authoring and Distribution Networks for Europe (ARIADNE, 2005), DCMI (DCMI, 2007) nesne ambarları verilebilir.

3.2 SCORM Standardı

SCORM, Sharable Content Object Reference Model (ADLNet, 2007), bilgisayar ve web-tabanlı öğrenme için yeniden kullanılabilen öğrenme içeriğini “eğitsel nesnelere” olarak oluşturmayı hedefleyen ve bilgisayar destekli eğitimde standartlaşma çalışmaları yapan bir kurum olan ADL (Advanced Distributed Learning) tarafından oluşturulmuştur.

ADL, öğrenim içeriği için farklı düzlemlerde birlikte kullanılabilir, üst-veri ve paketleme standartları aracılığıyla erişilebilir, yeniden kullanılabilir, maliyeti düşürürken üretkenliği arttıran araçlar, belirtiler, kılavuzlar, politika ve ön ürünler hazırlamaktadır.

SCORM ise, web tabanlı öğrenme içeriğinin erişilebilirliği, farklı platformlarda çalışılabilirliği, taşınabilirliği, teknolojik değişime dayanıklılığı, yeniden kullanılabilirliğini, hazırlanan eğitsel içerik, bireysel ve örgütsel ihtiyaçlara göre biçimlendirilebilirliğini sağlar. Bu sayede “Paylaşılabilir İçerik Nesnesi (SCO, Sharable Content Object)” denilen küçük, yeniden kullanılabilir ve herhangi bir LMS’ten bağımsız ÖN’leri oluşturulabilir ve bu SCO’lar farklı toplanlarda (aggregation- mesela kurslar) yeniden kombine edilebilir.

SCORM’un ilk sürümü olan ver1.0, “içerik toplam modeli” ve “çalışma anı ortamı”ni içermektedir ve 2000 yılında piyasaya çıkartılmıştır. Sonraki sürüme “içerik paketleme” ve “üstveri tanımlamaları” eklenerek ver1.2 adıyla piyasaya sürülmüştür. Son sürümü ise SCORM 2004’tür ve içerik ile ilgili “sıralama kuralları”ni da içermektedir. ADL, bir üst sürümünün çıkarılmayacağını duyurmuştur.

SCORM 2004, üç temel kitap toplamı olarak düşünülebilir. Bunlar, İçerik Modeli (CAM, Content Aggregation Model ver 1.3), Çalışma Anı Ortamı (RTE, Run-Time Environment ver 1.3), ve Sıralama ve İlerleme Kuralları (SN, Sequencing and Navigation ver 1.3) şeklindedir .

“Run Time Environment” kitabı, herhangi bir LMS’te hazırlanan içeriğin nasıl sunulacağını, öğrencinin izlemesi yani içeriğin nasıl çalıştırılacağını ve öğrenci bilgilerinin tutuluşunu belirleyen “Çalışma-Anı Ortamı”nı ifade etmektedir. RTE, SCO’ların farklı LMS’lere taşınabilmesini sağlar. Bileşenleri, içeriği başlatma (content-launch), içerik ile LMS’i konuşturmak üzere ortak bir yol yani Uygulama Programlama Arayüzü (UPA) ve bilgiyi öğrenene aktarmada kullanılan standartlaştırılmış bir veri modeli (data model) olarak sıralanır (ADL, 2006c). Bir anda yalnız bir SCO yüklenebilir ve etkin olabilir. SCO yalnızca LMS tarafından yüklenebilir. LMS, içerik paketinde bulunan içerik yapısına uygun biçimde kaynakların sıralanması ve dolaşılmasından sorumludur. Ekran görüntüsünün nasıl olacağı ya da öğrencilerin kaynaklar arasında nasıl dolaşacağı, LMS tarafından belirlenir (Aslantürk, 2002).

Uygulama Programlama Arabirimi ise, LMS ile yayınlanmakta olan kaynak arasında iletişimin kurulabilmesi için gerekli olan kuralları ve sorumlulukların standartlaştırılmasını sağlar. İçeriği barındıran web sayfaları bu UPA ile javascript sayesinde etkileşim sağlarlar. UPA, LMS’in kaynağın durumu (başlatıldı, bitti, hata oluştu) hakkında bilgilendirilmesini ve LMS ile SCO arasında veri alış verişinin gerçekleştirilmesini sağlayan bir düzenektir. UPA, bu iletişimin nasıl yapıldığını içerik geliştiriciden gizler (Aslantürk 2002). Arabirim tasarlanırken bir veri modelinin var olduğu varsayılmıştır. Kullanılabilecek bir veri modeli, *AICC*’nin “*cmi*” veri modelidir.

Geliştirilecek başka veri modelleri de SCORM Çalışma Ortamı tarafından kullanılabilir. Veri Modeli, LMS ile kaynak arasında iletilen, kaynakla ilgili bilgileri içeren standart veri elemanları kümesidir. LMS, veri elemanlarının oturumlar arasında yönetilmesinden sorumlu iken, içerik geliştiriciler de içeriklerin farklı LMS’lerde yeniden

kullanılabilirliğinin sağlanması açısından sadece bu elemanları kullanmak durumundadırlar. Ortak bir veri modeli oluşturmaktaki amaç, farklı LMS'lerin, içerik nesnelere ile ilgili olarak önceden tanımlanmış bilgileri izleyebilmesini sağlamaktır. (Aslantürk, 2002).

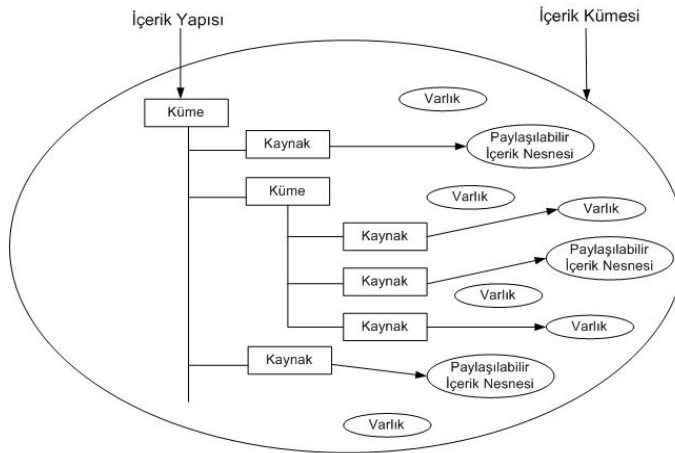
CAM, eğitsel içeriğin hazırlanışı, nasıl biraraya toplanacağı ve paketlenişini belirleyen "İçerik Kümesi Modeli", paylaşılabılır içerik nesnelere kapsamaktadır. Sequencing and Navigation kitabı ise eğitsel stratejilerin nasıl tasarlanacağını belirtir. Bu tez çalışmanın ana hatları "SCORM içerik paketleme" çerçevesinde çizildiğinden ve içeriğin farklı platformlarda çalışabilir şekilde tasarlanması için içerik paketleme, aranabilir olması için üstveri tanımlamaları, öğrenme içeriği organizasyonunun sıralanabilirliğinin sağlanması üzerinde durulduğundan, ilerleyen bölümler bu noktalar üzerinden anlatılmaktadır.

SCORM CAM ya da diğere bir ifade ile İçerik Kümesi Modeli, küçük ve basit eğitsel varlıkların bir araya getirilerek daha büyük ve karmaşık içeriklerin hazırlanması ve bunların uygun şekilde düzenlenmesini sağlar. CAM, içerik modeli (content model), üstveri, içerik paketleme (content packaging) olmak üzere üç bölümden oluşmaktadır. İçerik modeli bileşenleri, Varlıklar (*Assets*), SCO'lar ve İçerik Kümeleri (*Content Aggregations*) olarak sıralanır. Varlık, kullanıcıya sunulmak üzere hazırlanan ve herhangi bir web istemcisi tarafından kullanılabilen verilerin elektronik gösterimidir. Web sayfası, javascript fonksiyonları, XML belgesi, Flash nesnesi, JPEG, HTML fragment, GIF, WAV birer varlıktır (ADL, 2006a). Varlıklar birleşerek SCO'ları oluştururlar.

SCO'lar çalıştığı LMS ile iletişime geçmek üzere SCORM Çalışma Ortamını kullanırlar. ÖN'leri SCORM uyumlu hazırlandıklarında elde edilecek ürün SCO olarak isimlendirilir. Bir SCO, farklı öğrenme hedefleri için farklı öğrenme deneyimlerinde kullanılabilmelidir. Daha

üst düzeyde öğrenme hedeflerini karşılayabilmek için bir ya da daha fazla SCO bir araya getirilebilmelidir. SCO'ların birleşmesiyle de elektronik kurs içeriği oluşur (Balcı ve İnceoğlu, 2005). SCO içindeki web sayfaları, LMS ile dolaylı yoldan haberleşen, kullanıcının SCO'yu görüp görmediği, ne kadar süre ile gördüğü, öğrenme etkileşimi içindeki performansı ve hakimiyetini gösteren JavaScript fonksiyonu içerirler (IDDL, 2006).

İçerik kümesi ise SCO'ların bir öğrenme birimi (konu, ünite, ders...) oluşturmak üzere nasıl birleştirileceğini belirtmek üzere kullanılan bir haritadır. İçeriğin öğrenciye hangi sıra ile sunulacağını gösterir. SCORM içerik paketi bir kursu, dersi ya da ilişkili içerik nesnelerini içerebilir. Şekil 3.2'deki yapıda "küme" bir kursa yani SCORM'da içerik organizasyonuna, bunun altındaki herbir altküme bir modüle ve her kaynak bir derse yani SCORM'da birer öğeye karşılık gelmektedir. Bu öğeler, birer kaynak ile ilişkilendirilir ve bu kaynaklar da ya bir varlığı ya da bir SCO'yu göstermek durumundadır.

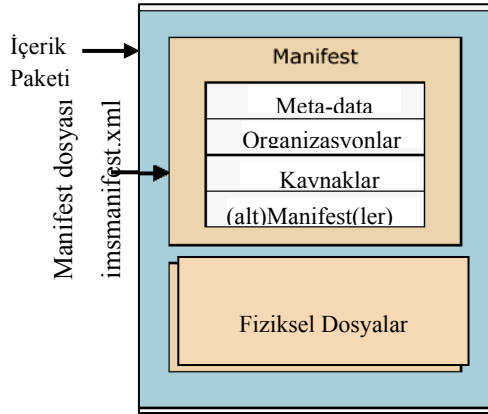


Şekil 3.2 SCORM İçerik Kümesi ve İçerik Yapısı (Aslantürk, 2002).

SCO'lar web tarayıcısı ile sunulabilirler fakat SCORM'da SCO'ların arasında bağlantı yoktur (ADL, 2006a). Eğer bir SCO içerisinden diğer SCO'lara bağlantı verilmemiş ise, bu SCO'lar arasında ilerlememizi LMS kontrol eder. İçerik paketinde bulunan `imsmanifest.xml` dosyasındaki organizasyon yapısı tanımlamasına bakarak LMS, SCO'lar arasındaki ilişkileri belirler, kullanıcı içeriği başlattığında bir ilerleme düzeni oluşturur ve içeriği bu düzene göre sunar. Bu LMS'e bağlı olarak genellikle ileri-geri okları sayesinde ilerlenen bir yapıdır (IDDL, 2006; ADL, 2006a).

3.2.1 İçerik paketi bileşenleri

Temel olarak içerik paketleme, paketi tanımlayan manifest kütüğünü, manifest dosyasının nasıl oluşturulacağını, manifest dosyası ve ilgili tüm fiziksel kaynak dosyalarının sıkıştırılmış bir dosya ya da CD-ROM gibi bir ortama nasıl aktarılacağını tanımlar. Şekil 3.3'te içerik paketi bileşenleri görülmektedir. Manifest dosyası (`imsmanifest.xml`), içerik paketi hakkında üstveri, içerik yapısı ve davranışını betimleyen seçimli bir `Organization` kesimi, paket içindeki kaynaklara başvuruların bir listesini bulundurur.



Şekil 3.3 Öğrenme Nesnesi İçerik Paketi.

Ayrıca manifest dosyası içinde altmanifestler de bulunabilir. Manifest dosyası XML (Extensible Markup Language) formatındadır ve <manifest>.....</manifest> etiketleri arasında tanımlı üç temel eleman içerir. Şekil 3.4’te de görülen bu elemanlar, "metadata", "organizations" ve "resources" elemanlarıdır (ADL, 2006a).



Şekil 3.4 Manifest Dosyası elemanları (ADL, 2004).

3.2.1.1 Manifest elemanının aldığı nitelikler

Manifest elemanı, XML formatındaki imsmanifest.xml dosyasının kök elemanıdır. Tüm eğitsel içerik organizasyonu yapısı, üstveri, kaynaklar ile ilgili bilgiler bu elemana ait etiketler arasında tanımlanır. Manifest elemanı 3 nitelik alabilir. Bunlardan "identifier" niteliği benzersiz bir değer almak zorundadır. "version", SCORM 2004 içerik paketleri için "1.3" değerini alır. Alabileceği son nitelik "xml:base" ise kullanımı seçimli olan bir niteliktir. Ayrıca burada, imsmanifest.xml dosyası içerisinde kullanılan etiketler ile ilgili olarak tanımlanması gereken "xml isim alanları" da belirtilir. İsim alanı belirtmek için "xmlns:" önekini takiben URL tanımı yapılır.

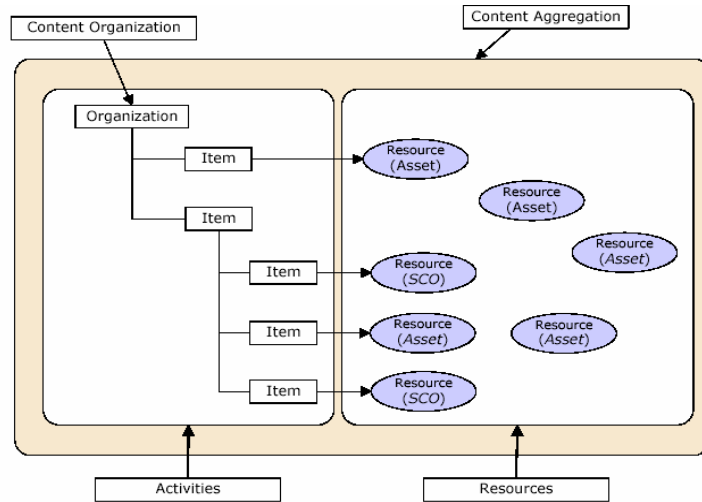
3.2.1.2 Manifest dosyası "metadata" elemanı

Bu eleman, içerik paketine ait üstveri bilgilerinin tanımlandığı bloğu ifade eder ve <metadtata>...</metadata> etiketleri arasında

tanımlanır. İçerik yapısının içine üstveri bileşeni olarak yerleştirilebileceği gibi, ayrı bir dosya olarak, içerik paketi içinde de bulunabilir. Üstveri tanımlamaları ileride ayrı bir başlık altında detaylı olarak incelenmiştir.

3.2.1.3 Manifest dosyası “organizations” elemanı

“Organizations” bloğu, eğitimcinin öğrenci tarafından eğitim içeriğinin kullanılması için belirlediği sırayı gösteren bir haritadır. Diğer bir ifade ile, kaynakların mantıksal olarak sıralanmasıdır (Şekil 3.5). Bir içerik paketinde, kök eleman olan “organizations” elemanı altında herbiri farklı bir yapıyı tanımlayan birden fazla “organization” bileşeni olabilir fakat, bunlardan bir tanesi varsayılan olarak tanımlanmalıdır. İçerik organizasyonu, içerik paketinin fiziksel yapısı ya da manifest’in yapısı ile karıştırılmamalıdır. İçerik paketindeki dosyalar genelde klasörler şeklinde organize edilirler. Fakat yapı kullanıcıya içerik paketini nasıl kullanacağını anlatmaz.



Şekil 3.5 İçerik Organizasyonu (ADL, 2004).

İçerik organizasyonundaki “activities” bölümü, başka “activities” bölümleri de içerirse, bu durumda SCORM SN kitabına göre küme (cluster) olarak isimlendirilir (ADL, 2006b). Bu basamaklandırma için bir sınır yoktur. “<organizations>” etiketi, Şekil 3.5’te (ADL, 2004) görülen Content Organization, içindekiler tablosu ve öğrenme etkinliklerinin hiyerarşik tanımını içermektedir.

Aynı Resource elemanına bir ya da birden fazla “item” elemanından referans verilebilir. Manifest’in resource bileşeni, paket içindeki fiziksel dosyalar gibi dış kaynakları tanımlar. Kaynakların toplamı “content” yani “içerik” olarak adlandırılır. Eğer kaynak bir LMS ile iletişim kuracak şekilde tasarlanırsa o zaman kaynak SCO; iletişim kurmayacaksa “asset” olarak adlandırılır. Birden fazla kaynak toplamına “resources” denir ve buna organizasyondan referans verilebilir.

Tanımlanma biçimi aşağıda görülen <organization> elemanı, üç nitelik ve yedi alt elemana sahip olabilir. Nitelikler yazar ya da yazarlık aracı tarafından tanımlanırlar. Bunlardan *identifier* zorunlu iken, *structure* niteliğinin kullanımı seçimlidir ve varsayılan değeri *hierarchical* olarak tanımlıdır. Bu eleman, varsayılan değeri *true* olan ve seçimli olan *adlseq:objectivesGlobalToSystem* niteliğini de alabilir.

```
<organizations>
  <organization identifier="TOC1">
    <title> Introduction to SCORM for LMS Vendors </title>
    <item identifier="ITEM1" identifierref="RESOURCE1"
      isvisible="true">
      <title>SCORM Run-Time Environment
        Requirements</title>
    </item>
    <item identifier="ITEM2" identifierref="RESOURCE2"
      isvisible="true">
      <title>LMS Conformance Requirements</title>
    </item>
  </organization>
</organizations>
```

Tanımlanan her organizasyon aşağıda sıralanan alt elemanlarını (ADL, 2006a) içerebilir.

- <title>, organizasyon başlığını belirtir.
- <item> elemanı, içerik organizasyonundaki bir etkinliği tanımlar. Organizasyon birden fazla etkinlik içerebilir. Her etkinliğin dolayısıyla her öge elemanın alabileceği nitelikler vardır. Bunlardan *identifier* o etkinliği tanımlar ve zorunlu olarak tanımlanması gereken bir niteliktir. Bu elemanın hangi kaynak ile ilişkilendirildiğini gösteren “*identifierref*” ve elemanın görünüp görünmeyeceğini gösteren ve varsayılan değeri “true” olarak tanımlı “*isvisible*” nitelikleri ise opsiyonel olarak kullanılmaktadır. Ayrıca yine opsiyonel olarak kullanılan “*Parameters*” niteliği, başlama zamanında kaynağa yollanan statik parametreleri içermektedir. Bu, eğer “item” elemanı bir kaynak elemanını gösteriyorsa yani *identifierref* niteliğine veri girişi yapılmışsa kullanılacak bir niteliktir. Her “item” elemanı ayrıca tanımlı birer başlık, “item” ve “metadata” alt elemanlarını içerebilirler (ADL, 2006a).
- <adlcp:timeLimitAction> elemanı, etkinlik için maksimum süre aşıldığında yapılacak işlemi tanımlar. Süre ve yapılacak işlem SCO tarafından kontrol edilir. Bu eleman “exit, message”, “exit, no message”, “continue, message”, “continue, no message” değerlerini alabilir. Bu ve aşağıdaki diğer iki elemanın XML isim alanı http://www.adlnet.org/xsd/adlcp_v1p3 olarak belirtilmiştir (ADL, 2006a). Kaynak paketleri için bu eleman görüntülenmez.
- Aşağıda <item> etiketi altında tanımlı olan <dataFromLMS> elemanının kullanımı görülmektedir. Bu eleman, işlem başlatıldıktan sonra <item> ile gösterilen kaynak yani SCO

tarafından beklenen “başlatma” verisini sağlar. LMS’i değil, sadece SCO’ları ilgilendirir.

```
<organization>
  <item identifier="ITEM3" identifierref="RESOURCE3"
    isvisible="true">
    <title>Content 1</title>
    <adlcp:dataFromLMS>Some SCO Information
  </adlcp:dataFromLMS>
    <adlcp:completionThreshold> 0.75
  </adlcp:completionThreshold>
  </item>
</organization>
```

- Yukarıdaki kodda görülen <adlcp:completionThreshold> alt elemanı ise, bu tanımlamanın yapıldığı öğenin referans ettiği SCO kaynağınca kullanılacak eşik değerini tanımlar. Alabileceği değer 0.0 ile 1.0 arasında olabilir.
- <adlnav:presentation> alt elemanı, bağlı olduğu elemanın gösterimi ile ilgili bilgi içerir. Her <item> elemanında en fazla 1 kere bulunabilir. http://www.adlnet.org/xsd/adlnav_v1p3 XML isim alanı ve “adlnav” öneki olarak tanımlıdır (ADL, 2006a). Presentation elemanı nitelik almaz, <navigationInterface> isminde tek bir alt elemana sahiptir. Bu alt eleman, verilen bir öğrenme etkinliği için navigation arayüzü gösterimi gereksinimlerini kapsar. <navigationInterface> alt elemanı nitelik almaz fakat LMS’in öğrencinin spesifik olayları tetiklemesine izin veren kullanıcı arayüzü aygıtlarını sağlamadığını belirten tek bir alt eleman, <hideLMSUI>, içerir. <hideLMSUI> elemanı, bağlı olduğu eleman içinde istenildiği kadar bulunabilir. Nitelik ve alt eleman içermez, “string” tipte değer alır. LMS’in bir sonraki ilerleme isteğine izin verebilmesi için sağladığı kullanıcı arayüzünün alabileceği değerleri tanımlar. Bu değerler, LMS’in bir öncekine ilerleme cihazını görüntülememesini sağlayan “previous”, bir sonrakine ilerle bağlantısını göstermeyen “continue”, terket

ilerleme bağlantısını göstermeyen “abondon” çıkış ilerleme bağlantısını göstermeyen “exit” değerleridir. Aşağıda <adlnav:presentation> elemanının kullanım biçimi görülmektedir.

```
<organization>
  <item identifier="ITEM3" identifierref="RESOURCE3"
    isvisible="true">
    <title>Content 1</title>
    <adlnav:presentation>
      <adlnav:navigationInterface>
        <adlnav:hideLMSUI>continue</adlnav:hideLMSUI>
        <adlnav:hideLMSUI>previous</adlnav:hideLMSUI>
      </adlnav:navigationInterface>
    </adlnav:presentation>
  </item>
  <metadata>
    <adlcp:location> activities/activity1MD.xml
  </adlcp:location>
  </metadata/>
</organization>
```

- <metadata> elemanı ise, organizasyon içindeki üst-veri yani Content Organization’a ait üstveri bilgisini tanımlar. “http://www.imsglobal.org/xsd/imscp_v1p1” isim alanına bağlıdır.
- <imsss:sequencing> elemanı, içerik kümesindeki organizasyonların sıralanma bilgilerini ifade eden alt elemandır. Her organizasyon <sequencing> elemanını seçimli olarak alabilir. Sıralama kuralları ileride detaylı olarak açıklanmaktadır.

3.2.1.4 **Manifest dosyası “resources” elemanı**

Resources elemanı, manifest içindeki “organizations” kısmında her organizasyon altında tanımlanan öğelerin gösterdiği kaynak dosyalara referansların toplandığı kısımdır. Sadece <resource> elemanını alabilir. Birden fazla “resource” elemanı içerirse, bir hiyerarşi sözkonusu değildir (ADL, 2006a). <resource> elemanının alabileceği nitelikler şunlardır. Identifier, tanımlanması zorunludur ve yazar ya da

yazarlık dili tarafından sağlanır. `Type`, kaynağın tipini gösterir ve mutlaka tanımlanır. `Href`, opsiyoneldir ve bir URL (Uniform Resource Locator)'dir. `xml:base`, opsiyoneldir. Manifest'teki dosyalar için ilişkili yol sağlar. `adlcp:scormType`, SCORM kaynağının tipini tanımlar. Mutlaka `SCO` ya da `asset` değerlerinden birini almalıdır. `adlcp:persistState`, opsiyoneldir. Bir kullanıcı teşebbüsünden diğer teşebbüse kadar veriyi kararlı kılacak bir araç üzerinde ısrar eder.

`<Resources>` elemanının `<resource>` elemanı alt elemanlar da içerebilir. Bunlar `<metadata>`, `<file>`, `<dependency>` elemanlarıdır. `<metadata>` alt elemanı, kaynağı tanımlayan SCO üstveri ya da asset(varlık) üstveri bilgisidir. Diğer alt eleman olan `<file>` ise, bu kaynağın dayandığı dosyaların listesini verir. Bir `resource` elemanı altında birden fazla `<file>` alt elemanı olabilir. `<file>`, alt elemanı `href` attribute'unu zorunlu olarak alır. Aşağıda "imsmanifest.xml" dosyası içerisinde `<resources>` elemanının nasıl tanımlandığı ve aldığı alt elemanları görülmektedir.

```
<resources>
  <resource identifier="R_A2" type="webcontent"
    adlcp:scormType="sco" href="scol.html">
    <file href="scol.html"/>
    <file href="assets/image1.gif">
    <metadata>
      <adlcp:location>assets/asset1.xml</adlcp:location>
    </metadata>
  </resource>
  <resource identifier="R_A5" type="webcontent"
    adlcp:scormType="asset"
    href="pics\distress_sigs_add.jpg">
    <file href="pics\distress_sigs_add.jpg"/>
  </resource>
</resources>
```

Aşağıda tanımlanma biçimi görülen `<dependency>` alt elemanı ise, bu kaynağın dayandığı başka bir kaynağı belirtmektedir. Zorunlu olarak `identifier` niteliğini alır.

```

<resources>
  <resource identifier="R_A2" type="webcontent"
    adlcp:scormType="sco" href="scol.html">
    <file href="scol.html"/>
    <dependency identifierref="R_A5"/>
  </resource>
  <resource identifier="R_A5" type="webcontent"
    adlcp:scormType="asset"
    href="pics\distress_sigs_add.jpg">
    <file href="pics\distress_sigs_add.jpg"/>
  </resource>
</resources>

```

3.2.2 Üstveri

Üstveri (metadata), veri hakkında veri anlamına geldiğinden, ÖN Üstverisi, LOM ya da kısaca “öğrenme nesnelere hakkında tanımlayıcı veri” olarak açıklanabilir (Anido et al., 2003). Her ÖN için, içerik hazırlayıcıların arama yapmak ve gerekli ÖN’lerine ulaşabilmek için kullanacağı tanımlamaların bulunması gereklidir (Wiley, 2000). Bir ÖN’nin üstverisi genel olarak nesnenin başlığı, tanımı, konusu, dili, hedef kitlesi, anahtar sözcükleri, türü, sınıfı, katalog numarası, içeriğinin organizasyonu, öğrenciye içeriğin hangi sırada sunulacağını ve nesnenin oluşturulduğu tarih, içerik, dosya biçimi, geliştiren kişi ve kurum, iletişim bilgisi, sürüm numarası, telif hakkı gibi bilgileri içerir. Böylece içerik havuzundan arama yapılarak ÖN’ne erişebilir (Balcı ve İnceoğlu, 2005). İçerik paketinin üstverisinin olmasının avantajları arasında LMS’e yüklenen bazı bilgiler sadece üstveri içinde bulunması, kursların ve derslerin tanımlarının sağlanması, SCORM 2004’de içerik hiyerarşisinin belli seviyeleri için üstveri tanımlamalarını yapılabilmesi sayılabilir.

SCORM 2004 şartnamesinde üstveri tanımlamaları IEEE LOM tanımlamalarına göre yapılmaktadır. Bu şartnameye göre ÖN üstveri tanımlamaları dokuz başlık altında sıralanan toplam 69 etiket içerir. Bu kategoriler IMS şartnamesinde de benzer şekilde tanımlanmıştır. ÖN

üstverisinde, ÖN'lerinin arama ve indeksleme amacıyla kullanılacak tanımlama bilgilerinin oluşturulmasında standart olarak XML kullanılmaktadır. Üstveri tanımlarını içeren 9 grubun herbiri farklı veri elemanları içermektedir. Tanımlamalarda baz alınan şema LOMv1.0 olarak belirtilmiştir. Buna göre, veri elemanının ismi, açıklaması, izin verilen değerler sayısı(size), değerlerin sıralanışının anlamı olup olmadığı (order), izin verilen değerler kümesi (value space) ve veritipi (LangString, DateTime, Duration, Vocabulary, CharacterString, Undefined) alanları bulunmaktadır. SCORM 2004 üçüncü ve son sürümünde, üstveri tanımlamalarında kullanılan elemanların hepsinin kullanımı seçimidir. Yani içerik hazırlayan kullanıcı, üstveri tanımlaması yapıp yapmayacağına ya da hangi aşamada üstveri tanımlaması yapacağına tamamen kendisi karar vermektedir. Ancak, arama ve yeniden kullanılabilirliği arttırması sebebiyle üstveri tanımlamaları tavsiye edilmektedir. IEEE LOM üstveri elemanlarını barındıran gruplar aşağıda listelenmiştir.

1. Genel (General): Başlık, açıklama, konu, tanımlayıcı gibi ÖN'ni bir bütün olarak tanımlayan genel bilgileri,
2. Yaşamdöngüsü (Lifecycle): Sürüm numarası, durum, editor gibi kaynağın geçmiş ve şimdiki durumları ve kaynağın gelişimine etki eden kişiler hakkındaki bilgileri,
3. Üst-üstveri (Meta-metadata): Üst-verinin kendisi hakkında bilgileri,
4. Teknik (Technical): Kaynağın formatı, boyutu, lokasyonu (URL), kullanılacak tarayıcı, işletim sistemi gibi teknik gereksinimleri ve özellikleri hakkında bilgileri,
5. Eğitsel (Educational): kaynağın tipi, hedef kitlesi, öğrenme süresi, zorluk derecesi gibi, kaynağın eğitsel ve pedagojik özellikleri,
6. Haklar (Rights): Gerekli ödeme ve telif hakkı bilgileri gibi kaynağın kullanım hak ve koşulları ile ilgili bilgileri,

7. İlişki (Relation): Bu kaynak ve hedeflenen diğer kaynaklar arasındaki ilişkiler hakkında bilgileri,
8. Açıklama (Annotation): Kaynağın eğitsel kullanımı ve yorumların ne zaman, kimler tarafından oluşturulduğu bilgilerini,
9. Sınıflandırma (Classification): Bu kaynağın farklı bir sınıflandırma sistemindeki yerini betimleyen tanımlamaları içermektedir.

SCORM standardına göre üstveri bilgilerinin tanımlanması, biri `imsmanifest.xml` dosyasında ilgili yerlere gömülmek, diğeri ise bağımsız “xml” dosyaları şeklinde kaydedilip manifest dosyasında ilgili yerlerden referans verilerek tanımlanabilmektedir. Aşağıda içerik paketi manifest dosyası için, XML uzantılarını kullanarak üst-veri eklenmesini gösteren bir örnek program kodu verilmiştir.

```
<manifest xmlns:lom = "http://ltsc.ieee.org/xsd/LOM">
  <metadata>
    <schema> ADL SCORM </schema>
    <schemaversion> 2004 3rd Edition </schemaversion>
    <lom:lom>
      <general>
        <lom:title>
          <lom:string language="en-US"> Title for the
            Package </lom:string>
          </lom:title>
          .....
        </lom:lom>
      </metadata>
    </manifest>
```

Aşağıdaki örnekte ise isim alanının kullanım sırasında tanımlanması gösterilmiştir.

```
<manifest>
  <metadata>
    <schema> ADL SCORM </schema>
    <schemaversion>2004 3rd Edition </schemaversion>
    <lom xmlns:lom="http://ltsc.ieee.org/xsd/LOM">
      <general>
        <lom:title>
```

```

        <lom:string language="en-US"> Title for
        the Package </lom:string>
    </lom:title>
</general>
    .....
</metadata>
</manifest>

```

Ayrıca, manifest dosyasına `<location>` elemanını kullanarak da üstveri eklemek mümkündür. `<location>` elemanı SCORM Content Model Component’ı tanımlar ve üst-verinin nerede olduğunu gösterir. Bu bir URI olabilir. İsim alanı “http://www.adlnet.org/xsd/adlcp_v1p3” olarak tanımlanmıştır (ADL 2006a). Bununla ilgili örnek aşağıda görülmektedir.

```

<manifest>
  <metadata>
    <schema> ADL SCORM </schema>
    <schemaversion>2004 3rd Edition </schemaversion>
    <adlcp:location>course/metadata/course.xml
    </adlcp:location>
  </metadata>
</manifest>

```

Bu üstveri tanımlamaları manifest etiketi için gösterilmiş olmasına rağmen, diğer kademelerde de aynı şekilde kullanılabilirler. Metadata etiketi nitelik içermez fakat `<schema>`, `<schemaversion>` ve `<lom>` olmak üzere 3 alt eleman içerir. `<Schema>` elemanı, “ADL SCORM” değerini alabilir ve içerik paketinin SCORM gereksinimlerine uygun olarak oluşturulduğunu gösterir. `<Schemaversion>` ise, SCORM 2004 sürümü için “2004 3rd Edition” değerini alır.

SCORM 2004’ten önceki sürümlerde üstveri elemanlarının bazılarının tanımlanması zorunlu, bazıları seçimli iken (IEEE, 2002), son sürümle birlikte tüm üstveri etiketlerinin kullanımı seçimli olmuştur. Yukarıda da sözü geçen ve dokuz kategoride incelenen, toplam 69 üstveri elemanları aşağıda Çizelge 3.3’te görülmektedir.

Çizelge 3.3 Üstveri kategorileri, içerdikleri elemanlar ve veri tipleri
(ADL, 2006a)

Eleman no	Eleman adı	Veri Tipi
1	General	
1.1	Identifier	CharacterString (max 1000 karakter)
1.1.1	catalog	
1.1.2	entry	
1.2	Title	LangString (max 1000 karakter)
1.3	Language	
1.4	Description	LangString (max 2000 karakter)
1.5	Keyword	LangString
1.6	Coverage	LangString (max 1000 karakter)
1.7	Structure	Vocabulary
1.8	Aggregation Level	Vocabulary
2	LifeCycle	
2.1	Version	LangString (max 50 karakter)
2.2	Status	Vocabulary
2.3	Contribute	
2.3.1	Role	Vocabulary
2.3.2	Entity	CharacterString
2.3.3	Date	DateTime
3	Meta-Metadata	
3.1	Identifier	CharacterString
3.1.1	Catalog	
3.1.2	Entry	
3.2	Contribute	
3.2.1	Role	Vocabulary
3.2.2	Entity	CharacterString
3.2.3	Date	DateTime
3.3	Metadata Schema	CharacterString
3.4	Language	CharacterString
4	Technical	
4.1	1- Format	CharacterString
4.2	2- Size	CharacterString
4.3	3- Location	CharacterString
4.4	4- Requirement	
4.4.1	OrComposite	
4.4.1.1	Type	Vocabulary
4.4.1.2	Name	Vocabulary
4.4.1.3	Minimum version	CharacterString
4.4.1.4	Maximum version	CharacterString
4.5	Installation Remarks	LangString (max 1000 karakter)
4.6	Other platform requirements	LangString
4.7	Duration	Duration

Çizelge 3.3 Üstveri kategorileri, içerdikleri elemanlar ve veritipleri
(devam)

5	Educational	
5.1	Interactivity Type	Vocabulary
5.2	Learning Resource Type	Vocabulary
5.3	Interactivity Level	Vocabulary
5.4	Semantic Density	Vocabulary
5.5	Intended End User Role	Vocabulary
5.6	Context	Vocabulary
5.7	Typical Age Range	LangString
5.8	Difficulty	Vocabulary
5.9	Typical Learning Time	Duration
5.10	Description	LangString
5.11	Language	CharacterString
6	Rights	
6.1	Cost	Vocabulary
6.2	Copyright and other restrictions	Vocabulary
6.3	Description	LangString
7	Relation	
7.1	1- Kind	Vocabulary
7.2	2- Resource	
7.2.1	Identifier	CharacterString
7.2.1.1	Catalog	
7.2.1.2	Entry	
7.2.2	Description	LangString
8	Annotation	
8.1	1- Entity	CharacterString
8.2	2- Date	DateTime
8.3	3- Description	LangString
9	Classification	
9.1	Purpose	Vocabulary
9.2	Taxon Path	
9.2.1	Source	LangString
9.2.2	Taxon	
9.2.2.1	Id	CharacterString
9.2.2.2	Entry	LangString
9.3	Description	LangString
9.4	Keyword	LangString

LOMv1.0 tanımlamasında, kategoriler ve veri elemanlarının sıralanışı bilgiseldir. Örneğin “5.Educational” kategorisi, “1.General”dan önce gelebilir. Aynı şekilde “1.General” kategorisindeki “1.3.General.Language”, “1.2.General.Title”dan önce gelebilir.

3.2.2.1 General kategorisi

Bu kategorideki tanımlamalar, ÖN’ni bir bütün olarak tanımlayan genel bilgileri içermektedir. Aşağıda bu kategorideki veri elemanlarının tanımlanması görülmektedir (ADL, 2006a).

“Identifier” tanımlamasındaki “Catalog” elemanı, kaynak için giriş içeren katalog sistemlerinin ismini tanımlar. URI, URN, DOI, POI, ISSN, ISBN, XRI gibi değerler alabilir. Ancak tavsiye edilen değer URI’dır (ADL, 2006a). “Entry” alanı ise, öğrenme nesnesi için verilen katalog içindeki girişin değeridir. Başlık ya da başka bir tanımlayıcı olabilir. Listelenen her katalogun ilişkili bir entry’sinin olması gerekir. Bu değer, uygulama tarafından istenildiğinde oluşturulabilir.

```
<lom>
  <general>
    <identifier>
      <catalog>URI</catalog>
      <entry>http://www.adlnet.gov/content/CO_01</entry>
    </identifier>
    <title>
      <string language="en">Title for the LO object </string>
    </title>
    <language>en</language>
    <description>
      <string language="en">Textual description</string>
    </description>
    <keyword>
      <string language="en">learning object</string>
    </keyword>
    <coverage>
      <string language="en">Circa, 16th century France</string>
    </coverage>
    <structure>
      <source>LOMv1.0</source>
      <value>atomic</value>
    </structure>
    <aggregationLevel>
      <source>LOMv1.0</source>
      <value>2</value>
    </aggregationLevel>
  </general>
</lom>
```

“Title” veri elemanı, ÖN’nin başlığını ifade eder. (“tr,” “Öğrenme Nesnesi Nedir”) gibi. Nesne birden fazla başlık içeriyorsa, en genel olan önce yazılır, diğeri ise onu takiben parantez içinde verilebilir. “Language”, dil tanımlaması “dilkodu-ÜLKEKODU” şeklinde ilk kısım küçük harfle dil kodu, ikinci kısım büyük harfle ülke kodu şeklinde verilebilir. Örneğin “en”, “en-GB”, “de” gibi. ISO 639:1998 ve ISO 3166-1:1997 kod kümelerini kullanır. Data tipi en fazla 100 karakterlik CharacterString tiptedir. “Description” ise ÖN’nde yazılı olacak metni ifade eder. Yani kaynağın içeriğinin tanımıdır. Mesela (“tr,” “Bu video çalışmasında Leonardo da Vinci’nin hayatı ve çalışmaları açıklanmıştır”) gibi. Genel tanımlama ÖN’nin eğitsel tanımlaması ile karıştırılmamalıdır.

“Keyword”, ÖN’nin başlığını tanımlar ve anahtar kelime tanımlaması yapılır. mesela (“en,” “Mona Lisa”) gibi. “Coverage” ise ÖN’nin başvurduğu zaman, kültür, coğrafya ya da bölge gibi bilgilerin belirtildiği kısımdır. Mesela (“en,” “16th century France”). En fazla 10 değer girilebilir. Örneğin ÖN Fransa’da 16. yüzyıldaki tarım ile ilgili ise 1.5.General.Keyword kısmında (“tr,” “tarım”) ve 1.6.General.Coverage kısmında ise (“tr,” “16.yüzyıl Fransa”) tanımlaması yapılabilir.

“Structure” alanı, nesnenin yapısını ifade eder. (1)*Automic*, (2)*collection*, (3)*networked*, (4)*hierarchival* ya da (5)*linear* değerlerini alabilir. Automic, daha küçük parçalara bölünemeyen en küçük yapıdır. Collection, aralarında belirtilmiş bir ilişki olmayan nesnelere topluluğudur. Networked, belirtilmemiş ilişkiler içeren nesnelere topluluğudur. Hierachical, ağaç yapısında ilişkilendirilmiş nesnelere topluluğu; Linear ise, tam olarak sıralı olan nesnelere topluluğudur yani önceki sonraki ilişkisinde bağlantılı nesnelere ifade eder. “Aggregation level” ise nesnenin fonksiyonel yapısını ifade eder. 1-4 arasında değer verilebilir. (1)- işlenmemiş ortam verisi (raw media data), (2)-birinci

seviye toplamı, Ders, (3)-ikinci seviye toplamı. Kurs ve (4)en yüksek seviyeyi yani sertifika verilecek kurs kümelerini ifade etmektedir. “Structure” ve “AggregationLevel” alanlarını içeren bir örnek Çizelge 3.4’te görülmektedir.

Çizelge 3.4 Structure ve AggregationLevel alanları arasındaki ilişki

LO	1.7 Structure	1.8 AggregationLevel
Mona Lisa dijital resmi	Atomic	1
Mona Lisa dijital resmi ile ilgili bir ders	Collection ya da network	2
Mona Lisa ile ilgili bir kurs	Linear	3
Farklı kaynaklardan Mona Lisa ile ilgili derslerin toplamı	Collection	3
Mona Lisa ile ilgili tarih, tanım vs içeren bir küme kurs	Linear ya da Hierarchical	4

3.2.2.2 LifeCycle kategorisi

Bu kategorideki veri elemanları ve kullanılışları aşağıda görülmektedir (ADL, 2006a).

```

<lom>
  <lifeCycle>
    <version>
      <string language="en">1.0 alpha </string>
    </version>
    <status>
      <source> LOMv1.0</source>
      <value>final</value>
    </status>
    <contribute>
      <role>
        <source> LOMv1.0</source>
        <value>author</value>
      </role>
      <entity> BEGIN:VCARD&#13;#10;VERSION:2.1&#13;#10;FN:
        Joe Friday&#13;#10;END:VCARD</entity>
      <date>
        <dateTime>2002-12-12 </dateTime>
      </date>
    </contribute>
  </lifeCycle>
</lom>

```

```

    <description>
      <string language="en">A description for the
                                date</string>
    </description>
  </date>
</contribute>
</lifeCycle>
</lom>

```

“Version” elemanı üstveri tanımlamasının sürümünü belirtir. Örneğin (“en,” “1.2.alpha”). “Status” alanı kaynağın o anki durumunu tanımlar ve draft, final, revised, unavailable değerlerini alabilir. Bu alan unavailable olarak tanımlanmış ise ÖN’ne erişilemez.

“Contribute” veri alanı, ÖN’nin oluşturma giriş ya da yayın sırasında katılımı olan kişi ya da organizasyonları ifade etmektedir. Bu eleman üç alt eleman içermektedir. Bunlardan “Role” alanına girilebilecek değerler yazar (author), yayımcı (publisher), editör (editor), grafik tasarımcı (graphical designer), içerik sağlayıcı (content provider), script yazarı (script writer), eğitim tasarımcısı (instructional designer), bilinmeyen(unknown) ya da “terminator” gibi değerler olabilir. “Terminator”, değeri ÖN’ni erişilmez kılar.

“Entity” alanı ise ÖN’ne katkıda bulunan kişilerin önem sırasına göre girildiği elemandır. Burada yazar/katılımcılar için elektronik kartvizit oluşturulur. Role alanına giriş yapıldıysa mutlaka burada bir “vCard” tanımlanmalıdır. Eğer role alanındaki kişi silinirse ilişkili “vCard” da silinecektir. Bu alan vCard, IMC vCard 3.0, IETF RFC 2425:1998 ve IETF RFC 2426:1998 spesifikasyonlarına (vCard and vCalender, 2007) göre değer alır. Şekil 3.6’da bir VCARD örneği görülmektedir. Date ise katılım tarihi bilgisini içerir. (“2004-08-25”) gibi kullanılabilir.

BEGIN:VCARD	← başlangıcı
FN:Joe Friday	
TEL:+1-919-555-7878	
TITLE: Area Administrator Assistant	
EMAIL	
TYPE=INTERN	
ET:jfriday@host.com	
END:VCARD	← sonu

Şekil 3.6 VCard örneği

3.2.2.3 metaMetadata kategorisi

Bu kategori, ÖN'ni değil, üstveri kaydının kendisi tanımlanır. Aşağıda bu kategorideki veri elemanları ve kullanımları gösterilmektedir (ADL, 2006a).

```

<lom>
  <metaMetadata>
    <identifier>
      <catalog>URI</catalog>
      <entry>http://www.adlnet.gov/metadata/MDO_01</entry>
    </identifier>
    <contribute>
      <role>
        <source>LOMv1.0</source>
        <value>creator</value>
      </role>
      <entity>BEGIN:VCARD&#13;&#10;VERSION:2.1&#13;&#10;FN:Joe
        Metadata Creator&#13;&#10;END:VCARD</entity>
      <date>
        <dateTime>2002-12-12</dateTime>
        <description>
          <string language="en">This date represents
            the date the creator finished authoring
            the metadata.</string>
          </description>
        </date>
      </contribute>
      <metadataSchema>LOMv1.0</metadataSchema>
      <metadataSchema>ADLv1.0</metadataSchema>
      <language>en</language>
    </metaMetadata>
  </lom>

```

“Identifier” alanı, üstveri kaydını tanımlayıcı bilgidir. “Catalog” alanı URI, URN, DOI, POI, ISSN, ISBN, XRI gibi değerler olabilir. Ancak tavsiye edilen değer URI olarak belirtilmiştir (ADL, 2006a). “Contribute” veri alanı, üstverinin oluşumundan itibaren durumunu etkileyen kişi ya da kuruluşların önem sırasına göre belirtildiği kısımdır. LyfeCycle kategorisindeki Contribute alanı gibi tanımlanır. “Metadata Scheme” veri alanı, üstveri oluşturulurken yazarın kullandığı standardı belirtir. Şu an için IEEE LOM tanımlaması için kullanılan değer “LOMv1.0” şeklindedir (ADL, 2006a). “Language” veri alanı ise, üstveride kullanılan dili ifade eder. Girilen değer, tüm LangString değerleri için varsayılan dildir olarak kabul edilir. Eğer burada yazılı bir dil yoksa, LangString değerleri için varsayılan bir dil yok demektir (ADL, 2006a).

3.2.2.4 Technical kategorisi

Üstveri tanımlamalarındaki bu kategorideki veri elemanları, ÖN için gereken teknik koşulları ve karakteristikleri açıklamaktadır. Aşağıda detaylar görülmektedir. “Format” veri alanına girilecek olan değer, kaynağa ya da kaynaktaki içerik kümelerine ulaşmak için gerekli olan yazılımları ifade eder. “video/mpeg”, “ application/x-toolbook”, “text/html” gibi MIME type değerler (RFC2048:1996) olabilir. Ya da “nondigital” değerini alabilir. Eğer nesnenin MIME tipi tanımlaması mümkün değilse, “application/x-unknown” yazılmalıdır. HTML sayfası için bu “text/html” olarak değiştirilmelidir. “Size” veri alanı, nesnenin byte cinsinden boyutudur. Digit’ler 0 – 9 arasındadır. Örneğin “4200” şeklinde veri girilebilir.

```
<lom>
<technical>
  <format>text/html</format>
  <size>1024</size>
  <location>Lesson01/Module01/Resources/SC001.htm</location>
  <requirement>
```

```

<orComposite>
  <type>
    <source>LOMv1.0</source>
    <value>browser</value>
  </type>
  <name>
    <source>LOMv1.0</source>
    <value>ms-internet explorer</value>
  </name>
  <minimumVersion>5.0</minimumVersion>
  <maximumVersion>6.0</maximumVersion>
</orComposite>
</requirement>
<installationRemarks>
  <string language="en">This activity requires the client
    browser to have a Macromedia Flash plugin
    installed.</string>
</installationRemarks>
<otherPlatformRequirements>
  <string language="en">Sound card, Min. RAM: 16Mb, Video
    card and display: at least 800 X 600 pixels x
    256 colors</string>
</otherPlatformRequirements>
<duration>
  <duration>P5Y</duration>
  <description>
    <string language="en">Length of time to play
    simulation</string>
  </description>
</duration>
</technical>
</lom>

```

“Location” veri alanı, ÖN’ne erişimde kullanılacak ifadeyi belirtir. Yani bu üstveri ile tanımlanan ÖN’nin fiziksel olarak nerede bulunduğunu gösterir, (<http://host/id>) gibi. Bu genelde bir URL bazen de text açıklaması olabilir. Eğer başka içerik dosyaları içeren bir paket içinde üstveri dağıtılacaksa, o paket içindeki içerik dosyasına üstveri dosyasından bir URL sağlanmalıdır (ADL, 2006a). Metadata Editor bu değişkeni “http://” ile başlayanlar için URI olarak, diğerleri için ise “TEXT” olarak tanımlar. Eğer kullanıcı farklı bir URL girerse, üstveri dosyasında bu değişkenin değerini değiştirmesi gerekecektir.

“Requirement” elemanı ÖN’ni kullanabilmek için gereken teknik yeterlilikleri gösterir. Birden fazla seçenek “OrComposite” etiketi

kullanılarak birden fazla seçenek OR ile birbirine bağlanır. Bunun alt elemanlarından “Type”, ÖN’ni kullanabilmek için gereken teknoloji tipi (hardware, software, network vs) gösteri ve *Operating system, browser* değerlerini alabilir. “Name”, gereken teknolojinin ismini belirtmek için kullanılır ve alabileceği değer teknoloji tipine göre değişmektedir. Eğer teknoloji tipi “operating system” olarak belirtilmiş ise name’in içeriği *pc-dos, ms-windows, macos, unix, multi-os, none* olabilir. Eğer teknoloji tipi “browser” olarak belirtilmiş ise name alanı *any, netscape communicator, ms-internet explorer, opera, amaya* olabilir. “Minimum version” ve “maximum version” alanları ise gerekli teknolojinin en düşük ve en büyük sürümlerinin belirtildiği alanlardır.

“Installation Remarks” veri elemanı alanında, bu öğrenme nesnesinin nasıl yükleneceği açıklanır. Mesela (“en,” “unzip the zip file and launch index.html in your web browser”) gibi. “Other Platform Requirements” ise gereken diğer yazılım/donanımların belirtildiği bir veri elemanıdır. (“en,” “sound card”) gibi bir veri girişi yapılabilir. “Duration” veri elemanı ise daha çok ses, film ve animasyon içeren LO’lerin çalıştırılma süresini ifade etmek için kullanılır. Örneğin, “PT1H30M”.

3.2.2.5 Educational kategorisi

ÖN’sine ait eğitsel ve pedagojik karakteristiklerin tanımlandığı kategoridir. Bu kategori altında tanımlı elemanlar ve kullanım biçimleri aşağıda gösterilmiştir. Burada tanımlı elemanlardan “interactivity type”, active, expositive ve mixed değerlerini alabilir.

```
<lom>
  <educational>
    <interactivityType>
      <source>LOMv1.0</source>
      <value>mixed</value>
    </interactivityType>
  </learningResourceType>
```

```

        <source>LOMv1.0</source>
        <value>figure</value>
    </learningResourceType>
    <interactivityLevel>
        <source>LOMv1.0</source>
        <value>very low</value>
    </interactivityLevel>
    <semanticDensity>
        <source>LOMv1.0</source>
        <value>very low</value>
    </semanticDensity>
    <intendedEndUserRole>
        <source>LOMv1.0</source>
        <value>learner</value>
    </intendedEndUserRole>
    <context>
        <source>LOMv1.0</source>
        <value>training</value>
    </context>
    <typicalAgeRange>
        <string language="en">18-</string>
    </typicalAgeRange>
    <difficulty>
        <source>LOMv1.0</source>
        <value>easy</value>
    </difficulty>
    <typicalLearningTime>
        <duration>PT1H30M</duration>
        <description>
            <string language="en">Average length of time to
            experience the activity.</string>
        </description>
    </typicalLearningTime>
    <language>en-US</language>
</educational>
</lom>

```

Aktif öğrenme yaparak öğrenmedir. Bu belgeler simülasyon, anket ve alıştırmaları içerirler. “Expositive” öğrenme ise pasif öğrenmedir. Bu belgeler makale, video klipleri, grafikler, ve hypertext belgelerini içerir. Her ikisinin birleşimi ise “mixed” interaktivite tipini oluşturur. “learningResourceType”, exercise, simulation, questionnaire, diagram, figure, graph, index, slide, table, narrative text, exam, experiment, problem statement, self assessment, lecture değerlerini alabilir. En baskın olan tip en başa yazılmalıdır. “interactivityLevel”, very low, low, medium, high, very high

değerlerini alabilir. Interaktivite tipi active olan bir LO için interactivite seviyesi yüksek olmalıdır.

Anlamsal yoğunluk (Semantic density), öğrenme nesnesinin kısalığına, büyüklüğüne, ses ya da video gibi kaynaklarının süresine bağlıdır. Zorluktan bağımsızdır. Very low, low, medium, high, very high değerlerini alabilir. Açıklayıcı metin, resim ve devam etmek için tıklanacak bir buton içeren bir ekran düşük anlamsal yoğunluğa sahiptir. Kısa metin, resim ve bazı değerlerin değişmesi amaçlı kullanılan butonlar içeren bir ekranın anlamsal yoğunluğu ise yüksek olarak tanımlanabilir. 30 dakika süren bir video belgesi düşük anlamsal yoğunluğa sahip olabilirken 5 dakikalık aynı videonun özeti yüksek anlamsal yoğunluğa sahip olabilir. Bir matematik probleminin text gösterimi orta, sembolik gösterimi ise çok yüksek anlamsal yoğunluğa sahip olabilir (IEEE, 2002).

“IntendedEndUserRole”, ÖN'nin kimin için tasarlandığı bilgisini tutar. “Teacher, author, learner, manager” değerlerinden birini alabilir. “Context”, ÖN'nin bulunacağı ve kullanılacağı ortamı belirtir. LOM v1.0 için alabileceği değerler school, higher education, training, other şeklinde olabilir. “Typical Age Range”, kullanıcının yaş aralığı bilgisini tutan veri elemanıdır. Mesela “7-9” ya da “18-” ya da “en,” “adults only” gibi değerler alabilir. “Difficulty”, hedef kitle için bu ÖN ile çalışma zorluk derecesini gösterir. Alabileceği değerler “very easy, easy, medium, difficult, very difficult” şeklinde tanımlanmıştır. “typicalLearningTime”, tahmini öğrenme süresini gösterir. “Description”, ÖN'nin nasıl kullanılacağına dair bilgileri içerir. Örneğin (“en,” “teacher guidelines that come with a textbook” ya da “başlığın bir açıklaması olarak kullanın”) şeklinde veri girilebilir. “Language” ise olası kullanıcının kullandığı dili belirtir. (“en” “tr”) gibi.

3.2.2.6 Rights kategorisi

Öğrenme nesnesini kullanmak için bedel ödemek gerekip gerekmediğini, ÖN’ne Copyright ve diğer sınırlamaların uygulanıp uygulanmadığının açıklandığı kategoridir. Aşağıda ilgili elemanların kullanımı görülmektedir. Alabileceği değerler (yes, no) şeklindedir. “Description” alanında ise eğer ilk iki alandan biri *yes* değerini almışsa nesne kullanım koşullarına ait açıklamalar içermelidir.

```
<lom>
  <rights>
    <cost>
      <source>LOMv1.0</source>
      <value>yes</value>
    </cost>
    <copyrightAndOtherRestrictions>
      <source>LOMv1.0</source>
      <value>yes</value>
    </copyrightAndOtherRestrictions>
    <description>
      <string language="en">For additional information or
        questions regarding copyright, distribution and
        reproduction, contact Joe Developer at
        joe_developer@someorganization.org</string>
    </description>
  </rights>
</lom>
```

3.2.2.7 Relation kategorisi

Öğrenme nesneleri arasındaki ilişkilerin tanımlandığı bölümdür. Bu kategorideki veri elemanlarının tanımlanmaları aşağıda görülmektedir. Bunlardan “Kind”, ÖN ile hedef ÖN arasındaki ilişkinin doğasını belirtir. *ispartof*, *haspart*, *isvarsof*, *hasversion*, *isformatof*, *hasformat*, *references*, *isreferencedby*, *isbasedon*, *isbasisfor*, *requires*, *isrequiredby* değerlerini alabilir. “Resource”, ilişkinin kurulacağı hedef ÖN’ni gösterir. Identifier-catalog değeri olarak URI seçilmesi tavsiye edilir (ADL, 2006a). Entry alanı ise identifier’ın gerçek değeridir. “Description” ise hedef öğrenme nesnesinin açıklamasıdır.

```

<lom>
  <relation>
    <kind>
      <source>LOMv1.0</source>
      <value>isbasedon</value>
    </kind>
    <resource>
      <identifier>
        <catalog>URN</catalog>
        <entry>urn:ADL:1234-45FD</entry>
      </identifier>
      <description>
        <string language="en">Microsoft MSCE</string>
      </description>
    </resource>
  </relation>
</lom>

```

3.2.2.8 Annotation kategorisi

Öğrenme nesnesinin eğitsel kullanımına dair komutlar ve bu komutların kim tarafından ne zaman oluşturulduğu bilgisini verir. İlgili alanlara veri girişi aşağıda görüldüğü şekilde yapılabilir. Kullanımı opsiyoneldir ve üstveriyi oluşturan kişi haricinde herkes tarafından kullanılabilir. “Description” alanına veri girişi, (“tr,” “bu video klibi öğrencilerimle kullandım. Resmin belli yerlerine zoom yapmaktan çok zevk aldılar.”) şeklinde yapılabilir.

```

<lom>
  <annotation>
    <entity>BEGIN:VCARD&#13;&#10;VERSION:2.1&#13;&#10;FN:Joe
      Author&#13;&#10;END:VCARD</entity>
    <date>
      <dateTime>2001-07-30T10:14:35.5+01:00</dateTime>
      <description>
        <string language="en">Date and time annotation was
          created</string>
      </description>
    </date>
    <description>Learners will need to understand the
      fundamentals of Windows programming in order to

```

```

        grasp the concepts described in this learning.
    </description>
</annotation>
</lom>

```

3.2.2.9 Classification kategorisi

ÖN'nin nasıl bir sınıflandırma içinde olduğunu gösteren kategoridir. İçerdiği elemanların tanımlanmaları aşağıda görülmektedir (ADL,2006a).

```

<lom>
  <classification>
    <purpose>
      <source>LOMv1.0</source>
      <value>skill level</value>
    </purpose>
    <taxonPath>
      <source>
        <string language="en-US">ADL SCORM
          Concepts</string>
      </source>
      <taxon>
        <id>I</id>
        <entry>
          <string language="en-US">Content Aggregation
            Model</string>
        </entry>
      </taxon>
      <taxon>
        <id>I.A</id>
        <entry>
          <string language="en-US">Content Packaging
            Fundamentals</string>
        </entry>
      </taxon>
      <taxon>
        <id>I.A.3</id>
        <entry>
          <string language="en-US">Resource
            Fundamentals</string>
        </entry>
      </taxon>
      <taxon>
        <id>I.A.3.a</id>
        <entry>
          <string language="en-US">Packaging
            SCOs</string>
        </entry>
      </taxon>
    </taxonPath>
  </classification>
</lom>

```

```

        </entry>
    </taxon>
</taxonPath>
<description>
    <string language="en-US">Describing and packaging
        SCOs in a SCORM Content Package</string>
</description>
<keyword>
    <string language="en-US">Packaging SCOs</string>
</keyword>
</classification>
</lom>

```

Tanımlı alt elemanlardan “Purpose”, sınıflandırmanın amacını ifade eder. “discipline, idea, prerequisite, educational objective, accessability, restrictions, educational level, skill level, security level, competency” değerlerinden birini alabilir. Bu elemandan sadece bir tane olabilir. “TaxonPath” elemanı, belli bir sınıflandırma sistemindeki taxonomic yolu tanımlar. İçerdiği “Source” elemanı, sınıflandırma sisteminin adını (mesela (“en,” “ACM”), (“en,” “MESH”) gibi) gösterir. Bu data elemanı resmi ya da kullanıcı tanımlı bir taxonomy kullanabilir. Değeri ISO/IEC 10646-1:2000 repertuarından seçilir. “Taxon” ise önceden tanımlı bir etiket ya da terime sahip olan bir düğümdür. {[“12,” (“en,” “Physics”)], [“23,” (“en,” “Acoustics”)]} örneğinde, “12” değeri Taxon’un Id’si, (“en,” “Physics”) ise entry’sidir. “Description” elemanı, maksimum 2000 karakterlik LangString tipinde veri içerebilirken, “Keyword” elemanı ise, en fazla 1000 karakterlik LangString tipinde veri içerebilir.

3.2.3 Sıralama ve Gösterim

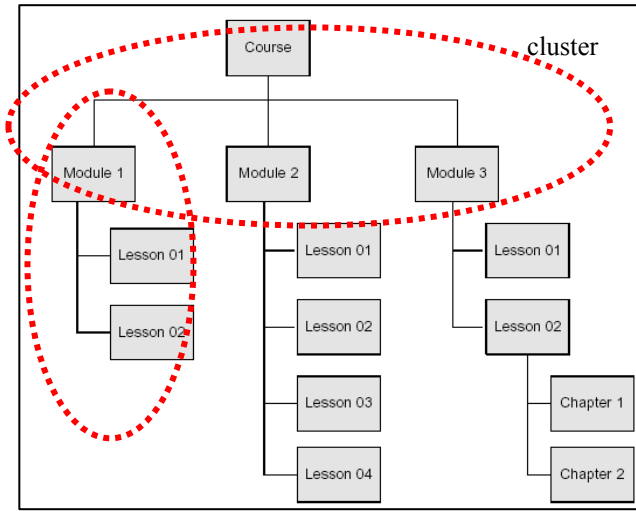
Öğrenciye dağıtılmak için tanımlı her öğrenme etkinliği kendisi ile ilişkili bir içerik nesnesine yani bir SCO ya da bir varlığa (asset) sahiptir. SCORM 2004 şartnamesindeki “Sequencing and Navigation” (SN) kitabı (ADL, 2006b), eğitsel içerikte kullanılan paylaşılabılır içerik nesnelere ve varlıkların sıralanmasındaki temel LMS sorumluluklarını içermektedir.

“Sequencing and Navigation” kitabı, IMS Simple Sequencing spesifikasyonunun SCORM ortamına nasıl uygulandığını açıklar. Navigation ise ilerleme olaylarını başlatacak olan kullanıcı arayüzü aygıtlarının varlığını kabul eder. Bu aygıtlar LMS tarafından sağlanır ya da içerik nesnesi içine gömülü durumda olur. Öğrenci böyle bir aygıtı tetiklediğinde LMS olayı ilgili ilerleme isteğine çevirir, isteği işleme sokar ve bir sonraki öğrenme etkinliğini belirtebilir (ADL, 2006a).

IMS SS, öğrenme etkinliklerinin yapısını tanımlayan bir “etkinlik (aktivite) ağacı” tanımlar. Şekil 3.7’de etkinlik ağacı örneği görülmektedir. Eğitsel içerik paketindeki içerik organizasyonu, farklı platformlarda çalışabilen bir etkinlik ağacının yapısını göstermektedir. Bu yapı içerisindeki <organization> elemanı altında tanımlı bir öğe, bir kursu, modülü, ünite ya da bir dersi ifade edebilir. Bu eleman etkinlik ağacının kökü ve altında tanımlanan her <item> elemanı ise bir öğrenme etkinliği konumundadır. SCORM uyumlu LMS’ler bu içerik organizasyonunu etkinlik ağacına çevirirler. Öğrenci etkinlik ağacındaki bir içerikle etkileşime geçince, LMS bir sonraki öğrenme etkinliğini bilebilmek için sıralama bilgisini değerlendirir. Sıralama bilgisine bağlı olarak her öğrencinin aynı içerikle olan deneyimi farklı olabilir. Sıralama bilgisi içerik sağlayıcı tarafından tanımlanır (ADL, 2006b).

SCORM sıralama davranışları, hangi ilerleme talebinin sıralama oturumunu başlatacağını belirtmektedir. Fakat, bu ilerleme talebinin nasıl ve ne zaman tetikleneceğini belirtmez. Etkinlik ağacının kök etkinliğindeki bir girişimin başlamasından bitimine kadar geçen süre *sıralama oturumu* (sequencing session) olarak adlandırılır. Genellikle LMS, sisteme dahil olmak, kursa başlamak gibi sistem olaylarında bir “ilerleme başlat talebi” (*Start navigation request*) yürütür. Eğer, bir önceki sıralama oturumu “*Suspend All*” ilerleme talebi sebebiyle sonlandıysa, LMS’in “start” yerine “*Resume All*” ilerleme talebini

yüürlüğe koyması gerekir. Bazı durumlarda ne *Start* ne de *Resume All* ilerleme talepleri başarılı olmaz, sadece geçerli bir “*Choice*” yani seçim ilerleme talebi, sıralama oturumunu başlatabilir. Bu durumda, geçerli olacak “*Choice*” ilerleme talebini tetikleyecek mekanizmayı sağlamak LMS’in sorumluluğundadır. Sıralama oturumunun bitmesi, etkinlik ağacındaki kök etkinlik tarafından “*Exit sequencing request*” talebinin yürütülmesiyle gerçekleşir. Bu durum ya *Exit All* (e.g. logout) ya da *Suspend All* (e.g. pause) ilerleme talebine neden olur ya da etkinlik ağacının köküne çıkış işlemi gerçekleştirilir.



Şekil 3.7 SCORM 2004 Sequencing: Etkinlik ağacı örneği.

Sıralama stratejilerinin XML’e uyarlanması ve XML’in imsmanifest dosyasına yerleştirilmesi ile ilgili bilgidir ve bu bilgi manifest içindeki <item> ya da <organization> elemanlarına, bunların çocuğu olarak eklenebilir. Sıralama bilgisini oluşturmanın 2 yolu vardır. Bunlardan ilki “<sequencing>” elemanını kullanmaktır. Bu eleman, verilen etkinlik için tüm sıralama bilgisini enkapsüle eder.

İkinci yol ise `<sequencingCollection>` elemanı kullanmaktır. Bu ise, farklı etkinlik tarafından önceden kullanılmış sıralama bilgisinin yeniden kullanılması için toplanmasını sağlar. `<sequencing>` elemanı altında tanımlanabilen sıralama elemanlarının listesi aşağıda verilmiştir.

```

<sequencing>
  <controlMode / >
  <sequencingRules>
    <preConditionRule>
      < ruleConditions>
        < ruleCondition>
      < / ruleConditions>
      <ruleAction />
    </preConditionRule>
    <postConditionRule>
      < ruleConditions>
        < ruleCondition>
      < / ruleConditions>
      <ruleAction />
    </postConditionRule >
    <exitConditionRule>
      < ruleConditions>
        < ruleCondition>
      < / ruleConditions>
      <ruleAction />
    </exitConditionRule >
  </sequencingRules>
  <limitConditions / >
  <auxiliaryResources / >
  <rollupRules>
    <rollupRule>
      <rollupConditions>
        <rollupCondition/>
      </rollupConditions>
    </rollupRule>
  </rollupRules>
  <objectives>
    <primaryObjective>
      <minNormalizedMeasure>
      <mapInfo/>

```

```

        </primaryObjective>
        <objective>
            <minNormalizedMeasure/>
            <mapInfo/>
        </objective>
    </objectives>
    <randomizationControls / >
    <deliveryControls />
    <constrainedChoiceConsiderations/>
    <rollupConsiderations / >
</sequencing>

```

<sequencing> elemanı ana elemandır ve on adet alt eleman içerir. LCDM platformunda da sıralama kurallarının eklenmesi sırasında bu eleman kullanılmaktadır. Barındırdığı tüm alt elemanlar için de geçerli olan XML isim alanı “<http://www.imsglobal.org/xsd/imsss>” ve isim alanı öneki “*imsss*” olarak tanımlanmıştır. <sequencing> elemanı, kullanımları opsiyonel olan iki nitelik alır. Bunlar *ID* ve *IDRef* nitelikleridir. ID, sıralama bilgi kümesine atanan benzersiz bir değerdir ve sadece <sequencingCollection> elemanının alt elemanları için tanımlanabilir; <organization> ya da <item> elemanlarının çocukları için ID değeri tanımlanmaz. IDRef ise ID değeri için referanstır. XML belgesi içerisinde biryerde kullanılan, yeniden kullanılabilir sıralama bilgisine bağlantı yapmak için kullanılır.

3.2.3.1 <controlMode> alt elemanı

Sıralama talepleri yürütülürken bir kümeye (cluster yani ana etkinlik ve birinci dereceden çocuklar, Bkz. Şekil 3.7) ilerleme isteklerinin nasıl etki edeceğini tanımlamasına izin veren elemandır. Küme içindeki ana etkinlik, küme hakkındaki sıralama bilgisini içerir. Kümeye dahil olmayan çocuklar ise ilişkili içerik nesnelerini içerirler. Bu eleman, bir etkinlik için birden fazla kontrol bilgisini tutan saklama alanı

gibidir ve <sequencing> elemanı içinde kullanımı seçimlidir ve en fazla 1 kere bulunabilir. Bu elemanın alt elemanları yoktur. Alabileceği nitelikler hakkında aşağıda örnek tanımlama gösterilmiştir. Buradaki tüm alt elemanlar “true” ya da “false” değerlerini alabilir.

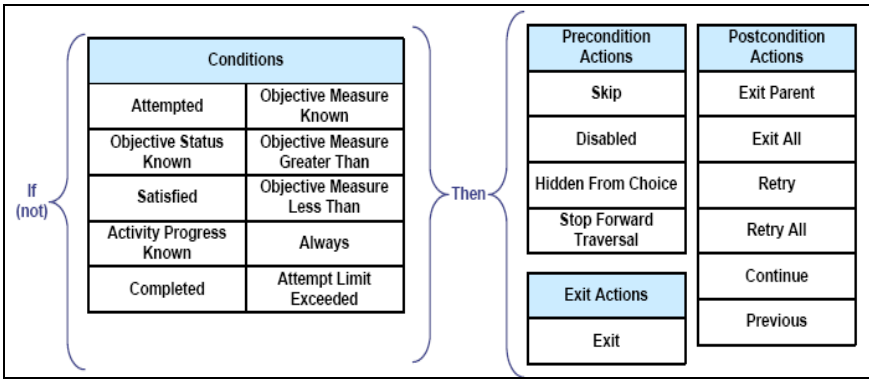
```
<imsss:sequencing>
  <imsss:controlMode choice="false" choiceExit="false"
    flow="true" forwardOnly = "true"/>
</imsss:sequencing>
```

<controlMode> elemanına ait “Choice” niteliğinin kullanımı seçimlidir ve varsayılan değeri “true” olarak belirtilmiştir (ADL, 2006b). Öğrencinin küme içinde bir sıralamaya bağlı kalmadan istediği etkinliği seçebileceğini gösterir. “false” değerini alırsa, öğrenci küme içerisinde herhangi bir etkinliği seçemez.

ChoiceExit niteliği, sıralama talebinin işlenmesi durumunda, etkinliğin aktif olan çocuğunun sonlandırma yetkisinin olup (true, varsayılan değer) olmaması (false) durumunu gösterir. Flow niteliği ise, opsiyonel olarak kullanılır ve etkinliğin çocuğunun akış sıralama talebine izni verilip verilmemesi (false, varsayılan değer) durumu için tanımlanmıştır. forwardOnly niteliği, etkinlik ağaç yapısında geriye dönük hedeflere izin verilmemesi (true) ya da verilmesi (false) durumunu gösterir. Kullanımı opsiyoneldir ve varsayılan değeri “false”dur. useCurrentAttemptObjectiveInfo niteliğinin de kullanımı opsiyoneldir. Etkinliğin çocuğunun ilerleme bilgisinin sadece kural değerlendirmelerinde kullanılması (true, varsayılan değer) ya da kullanılmamasını (false) ifade eder. Varsayılan değeri “true” olan useCurrentAttemptProgressInfo niteliğinin ise kullanımı opsiyoneldir. Etkinliğin alt elemanı için ilerleme bilgisi, “eğer bu bilgi etkinliğin güncel, o anki girişimi sırasında kayıt edildiyse kural değerlendirmeleri ve rollup’ta kullanılabilir (true) ya da kullanılamaz (false)” anlamındadır.

3.2.3.2 <sequencingRules> alt elemanı

Bu eleman, <sequencing> elemanı içerisinde kullanımı seçimlidir ve birden daha fazla kere kullanılabilir. Üç tane alt eleman içerir. Bunlar <preConditionRule>, <exitConditionRule>, <postConditionRule> elemanlarıdır. Bu elemanlara uygulanabilecek kurallar listesi Şekil 3.8’de görülmektedir.



Şekil 3.8 <sequencingRules> kurallarına ait durum ve hareketler (ADL, 2006b)

<preConditionRule> alt elemanı, belli bir etkinliğin dağıtımı ve sıralama kararlarını kontrol eden eylemlerin tanımlandığı kısımdır. Bu tür eylemleri içeren kurallar, etkinliğin dağıtılıp dağıtılmayacağına karar vermede kullanılır. <sequencingRules> elemanı içerisinde seçimli olarak birden fazla kere kullanılabilir. Bu elemanın <ruleConditions> ve <ruleAction> olmak üzere iki alt elemanı vardır. <ruleConditions> elemanı, içerisinde tanımlanan elemanların nasıl biraraya getirildiğini gösteren, varsayılan olarak “all” (tüm kural durumlarının doğru olması durumu) değerini alan “*conditionCombination*” niteliğini alabilir. Bu niteliğin alabileceği diğer bir değer ise “any” değeridir ve kural durumlarının içlerinden sadece birinin doğru olması durumunu işaret eder.

<ruleConditions> elemanı altında tanımlı birden fazla <ruleCondition> alt elemanı olabilir. Bu eleman 4 farklı nitelik alabilir. Bunlardan *referenceObjective* niteliği, durum değerlendirilmesi sırasınca kullanılan etkinlik ile ilişkili olan nesneye ait tanımlayıcıyı (*identifier*) belirtir. Kullanımı opsiyoneldir ancak varsayılan değeri “No” olarak belirlenmiştir. Boşluk değeri alamaz. Alacağı değer bir <objective> elemanının ya da <primaryObjective> elemanının *objectiveID* değeri olabilir. Bir diğer nitelik olan *measureThreshold* niteliği ise, ölçme-tabanlı durum değerlendirmelerinde kullanılan eşik değerini gösterir. Varsayılan olarak tanımlanan değeri 0.0 ve değer aralığı ise -1.0000 ile 1.0000 arasındadır. *Operator* niteliği, duruma uygulanacak olan operatörü gösterir. Varsayılan değeri “noOp” şeklinde olup, kuralı olduğu gibi kabul eder. Alabileceği diğer değer “not” ise kural değerlendirmesi sırasında aynı olan durumu ihmal eder. *condition* niteliği ise, zorunlu olarak tanımlanması gereken bir niteliktir. Alabileceği değerler “activityProgressKnown, attemptLimitExceeded, attempted, completed, objectiveStatusKnown, timeLimitExceeded, objectiveMeasureKnown, satisfied, outsideAvailableTimeRange, objectiveMeasureLessThan, objectiveMeasureGreaterThan, always” şeklindedir.

<ruleAction> elemanı ise, tanımlanan kuralın doğru yani “true” olarak değerlendirilmesi durumundaki, arzu edilen sıralama davranışını gösteren elemandır. <ruleAction> elemanı, <preConditionRule>, <postConditionRule> ya da <exitConditionRule> elemanları içerisinde sadece 1 kere tanımlanabilir ve bağlı olacağı elemana göre değişiklik gösterir. Alabileceği tek nitelik “*action*” niteliğidir ve arzu edilen sıralama davranışını gösterir. Eğer “action”, bir <preConditionRule> içerisinde tanımlı ise, bu durumda action attribute’u *skip*, *disabled*, *hiddenFromChoice*, *stopForwardTraversal* değerlerinden birini alabilir.

Eğer `<postConditionRule>` içinde tanımlı ise, *exitParent*, *exitAll*, *retry*, *retryAll*, *continue*, *previous* değerlerinden birini alabilir. Ve son olarak eğer `<exitConditionRule>` içinde tanımlı ise, sadece *exit* değerini alabilir. Aşağıda, `<sequencingRules>` elemanı içinde `<preConditions>` elemanının kullanımı görülmektedir.

```
<imsss:sequencing>
  <imsss:sequencingRules>
    <imsss:preConditionRule>
      <imsss:ruleConditions>
        <imsss:ruleCondition condition = "satisfied"/>
      </imsss:ruleConditions>
      <imsss:ruleAction action = "disabled"/>
    </imsss:preConditionRule>
  </imsss:sequencingRules>
</imsss:sequencing>
```

<postConditionRule> elemanı, belirli bir etkinlik için sıralama kararlarını kontrol eden işlemlerin tanımlarını içerir. Seçimli olarak ve en fazla 1 kere kullanılabilir. Nitelik almaz. Alacağı elemanlar, `<preConditionRule>` elemanındaki gibi, `<ruleConditions>` ve `<ruleAction>` alt elemanlarıdır.

<exitConditionRule> alt elemanı ise, bir etkinliğe uygulanan girişim sonlandırıldığında uygulanacak işlemleri içeren kuralları tanımlar. Bağlı olduğu `<sequencingRules>` elemanı içinde seçimli olarak ve en fazla 1 kere kullanılabilir. Bu eleman nitelik almaz.. İçerdiği alt elemanlar ise aynen `<preConditionRule>` elemanına bağlı olarak tanımlanan alt elemanlardır ve onlar gibi kullanılırlar.

3.2.3.3 <limitConditions> elemanı

Etkinlik üzerindeki girişimler ve o girişim için izin verilen maksimum süre ile ilgili tanımlamaları içeren elemandır. `<sequencing>` elemanı içinde seçimli olarak ve en fazla 1 kere tekrarlanabilir. Alt eleman içermez. Fakat, *attemptLimit* ve *attemptAbsoluteDurationLimit*

olmak üzere 2 tane nitelik alır ve bunların kullanımı da seçimlidir. “attemptLimit”, etkinlik için maksimum girişim sayısını belirler ve varsayılan değeri 0 (sıfır)’dır. “attemptAbsoluteDurationLimit” ise kullanıcının etkinlik ile etkileşim içerisinde olduğu zaman, etkinlik üzerindeki tek bir girişim ile harcayabileceği maksimum zamanı gösterir fakat bu süre, etkinliğin geçici olarak durdurulduğu zamanı kapsamaz.

3.2.3.4 <auxiliaryResources> elemanı

ADL, bu elemanı tedbir olarak kullanmaktadır. Yedek, yardımcı kaynakların kullanımı hakkında tanımlama gereksinimleri ile ilgili olarak “auxiliary kaynaklar SCO olabilir mi”, “bunlara sıralama bilgileri uygulanabilir mi” gibi kaygılar olmaktadır (ADL, 2006b; IMS, 2003).

3.2.3.5 <rollupRules> elemanı

<sequencing> elemanı içinde seçimli olarak ve en fazla 1 kere bulunabilen bir elemandır. Kullanımları isteğe bağlı olan, rollupObjectiveSatisfied, rollupProgressCompletion ve objectiveMeasureWeight olmak üzere 3 nitelik olabilir. “rollupObjectiveSatisfied” niteliği rollup’ta bulunan etkinlik ile nesnenin memnun edici durumunun ilişkilendirildiğini gösterir. Varsayılan değeri “true”dur. “rollupProgressCompletion” niteliği, etkinliğin tamamlanma durumunun, bağlı olduğu ana etkinlik için rollup’ta bulunduğunu ifade eder. Varsayılan değeri 1.0000’dur ve kullanımı opsiyonel olan bir niteliktir. <rollupRules> elemanı alt eleman olarak sadece <rollupRule> elemanını içerebilir. <rollupRule> alt elemanı, opsiyonel olan childActivitySet, minimumCount ve minimumPercent niteliklerini alabilir. “childActivitySet” niteliğinin varsayılan değeri “all” olarak belirtilmiştir ve rollup durumunu değerlendirmek için kimin veri değerlerinin kullanıldığını gösterir.

Alabileceği değerler “all, any, none, atLeastCount, atLeastPercent” şeklindedir. “*minimumCount*” niteliği ise sadece *childActivitySet* niteliğinin değeri “atLeastCount” olarak tanımlandığında kullanılır ve varsayılan olarak “0” sıfır değerini alır. “minimumPercent” niteliği ise sadece *childActivitySet* niteliğinin değeri “atLeastPercent” olarak belirtilmiş ise tanımlanabilir ve varsayılan değeri “0.0000” şeklinde tanımlanmıştır. Bu nitelik ile belirtilen çocukların en azı eğer true değerine sahip rollup durumu içeriyorsa, “rollup rule condition” “true” olarak değerlendirilir. Alabileceği değer aralığı 0.0000 to 1.0000 şeklinde tanımlanmıştır. Aşağıda bu elemanın kullanımına bir örnek verilmiştir.

```
<imsss:sequencing>
  <imsss:rollupRules>
    <imsss:rollupRule childActivitySet = "all">
      <imsss:rollupConditions>
        <imsss:rollupCondition condition=
          "attempted"/>
      </imsss:rollupConditions>
      <imsss:rollupAction action = "completed"/>
    </imsss:rollupRule>
  </imsss:rollupRules>
</imsss:sequencing>
```

<rollupRule> elemanın alabileceği alt elemanlar <rollupConditions> ve <rollupAction> elemanlarıdır. Bunlardan <rollupConditions> alt elemanı, tek bir rollup kuralı içinde uygulanan bir küme durum için konteyner görevi görür. <rollupRule> elemanı içinde sadece 1 kere bulunur. Kullanımı opsiyonel olan ve varsayılan olarak “any” değerini alan (diğer değer “all”) ve rollup durumlarının nasıl kombine edildiğini gösteren “conditionCombination” niteliğini alabilir. <rollupConditions> elemanın, rollupRule içinde uygulanacak olan bir durumu tanımlayan <rollupCondition> adında alt elemanları vardır. <rollupCondition> elemanın alabileceği nitelikler sırayla “operator”, (varsayılan değeri “noOp” alabileceği diğer değer “not”) ve kural için durum elemanını tanımlayan ve tanımlanması zorunlu olan

“condition” niteliğidir. Condition niteliğinin alabileceği değerler, “satisfied, objectiveMeasureKnown, objectiveStatusKnown, attempted, completed, activityProgressKnown attemptLimitExceeded,, timeLimitExceeded, outsideAvailableTimeRange” şeklinde belirlenmiştir (ADL, 2006a). Tanımlı diğer alt eleman, <rollupAction>, kural içinde uygulanacak olan durumu belirtir. Bağlı olduğu <rollupRule> elemanı içinde sadece 1 kere bulunur. Alt eleman içermez. Aldığı nitelik “action” niteliğidir. Kural doğru olarak değerlendirilirse, bu nitelik beklenen davranışı tanımlar. Alabileceği değerler ise “satisfied, notSatisfied, completed, incomplete” şeklindedir.

3.2.3.6 <objectives> elemanı

Bu eleman, bir etkinlik ile ilişkilendirilmiş olan “objective” kümeleri için saklama birimi görevi görür. Bağlı olduğu <sequencing> elemanı içinde en fazla 1 kere tanımlanabilir ve nitelik tanımı yapılmaz. <primaryObjective> ve <objective> olmak üzere 2 alt eleman içerir. Her etkinlik en azından 1 tane “primary objective” elemanı içermelidir.

<primaryObjective> alt elemanı, etkinlik ile ilişkili olan “rollup” kurallarına katılan nesneyi tanımlar. Eğer <objectives> elemanı tanımlanmış ise, <primaryObjective> elemanının tanımlanması zorunludur. Bu eleman bağlı olduğu <objectives> elemanı içerisinde sadece 1 kere bulunabilir. Her ikisi de seçimli olarak kullanılan 2 tane nitelik alabilir. İlk nitelik, varsayılan değeri “false” olan satisfiedByMeasure niteliği, etkinlik ile ilişkili olan objective’in yeterli olması durumunda herhangi başka bir metot yerine <minNormalizedMeasure> elemanının kullanılabilceğini gösterir. İkinci nitelik ise etkinlik ile ilişkili objective’in ID’sini belirten objectiveID niteliğidir. Burada boşluk karakteri kullanılmaz. Eğer bir <primaryObjective> elemanı bir <mapInfo> içerirse, objectiveID niteliği

gereklidir. Eğer içermez ise opsiyoneldir. XML veri tipi “anyURI” olarak tanımlanmıştır. Aşağıda da görüldüğü üzere, <primaryObjective> alt elemanı da alt elemanlar içerebilir. Bunlar <minNormalizedMeasure> ve <mapInfo> elemanlarıdır.

```
<imsss:objectives>
  <imsss:primaryObjective objectiveID = "PRIMARYOBJ"
    satisfiedByMeasure = "true">
    <imsss:minNormalizedMeasure>0.6
  </imsss:minNormalizedMeasure>
  <imsss:mapInfo targetObjectiveID = "obj_module_1"
    readNormalizedMeasure = "false" writeSatisfiedStatus =
    "true" />
  </imsss:primaryObjective>
</imsss:objectives>
```

<minNormalizedMeasure> elemanı “objective” için minimum yeterlilik (satisfaction) ölçüsünü tanımlar. Değeri -1 ile 1 arasındadır. Varsayılan değeri “1.0”dır. Eğer bu eleman, bir “primaryObjective” için minimum değer tanımlamak üzere kullanılmaktaysa, LMS bu değeri *cmi.scaled_passing_score* (ADL, 2006c) ifadesini başlatmak için kullanacaktır. Bu eleman bağlı olduğu <primaryObjective> ve <objective> elemanları içinde en fazla 1 kere kullanılabilir. Nitelik ve alt eleman içermez.

Burada tanımlanabilecek diğer alt eleman <mapInfo> elemanıdır. Paylaşılan global bir nesneye (ya da nesneden) etkinliğin yerel nesne bilgisini eşleştirmek etmek için kullanılır. Her etkinlik için sınırsız sayıda “objective maps” olabilir. Alt eleman içermez fakat nitelik alabilir. Bu nitelikler, zorunlu olan “targetObjectiveID”, global paylaşılan objective için tanımlayıcıdır. Boş karakter olamaz. XML Data Type: xs:anyURI olarak tanımlanmıştır. “readSatisfiedStatus” niteliği ise seçimlidir ve varsayılan olarak “true” değerini alır. Yerel “objective” için gelişme, ilerleme adımı (progress) tanımlanmamış ise, tanımlı yerel “objective” için memnuniyet durumu tanımlı paylaşılan “global

objective”den elde edilebilir (true ya da false değerleri). Üçüncü nitelik “*readNormalizedMeasure*” niteliğidir, seçimlidir ve varsayılan değeri true’dur. Tanımlanmış yerel “objective” için normalize edilmiş ölçünün, ölçü tanımlanmadığı zaman tanımlı paylaşılan “global objective”den alınabileceğini (true ya da false) belirtir. “*writeSatisfiedStatus*” niteliği ise varsayılan olarak “false” değerini alır. Tanımlı yerel objective için yeterlilik durumu, tanımlı paylaşılan global objective’e transfer edilebilir. Son olarak “*writeNormalizedMeasure*” niteliği seçimlidir ve varsayılan olarak “false” değerini alır.

“*readSatisfiedStatus*” ve “*readNormalizedMeasure*” içinde eğer bir objective için birden fazla <mapInfo> elemanı mevcut ise, sadece bir tanesinin *readSatisfiedStatus* niteliği ve *readNormalizedMeasure* niteliği true olarak belirlenir.

Write Objective Map’leri, yani *writeSatisfiedStatus* ve *writeNormalizedMeasure*, için bir etkinlikte eğer aynı *targetObjectiveID*’yi paylaşan birden fazla objective <mapInfo> elemanı var ise bu durumda objectives’lerden sadece bir tanesinin *writeSatisfiedStatus* niteliği true değerini ve *writeNormalizedMeasure* niteliği için true değerini alabilecektir.

<objectives> elemanı altında tanımlı ikinci eleman ise <objective> alt elemanıdır. Bu, etkinlik ile ilişkisi olmayan rolup’a katkısı olmayan nesnelere tanımlar. Sadece <primaryObjective> elemanı tanımlanmış ise kullanılabilir. <objectives> elemanı içinde 0 ya da daha fazla kere bulunabilir. Bu eleman, <primaryObjective> elemanının aldığı niteliklerin aynısını yani “*satisfiedByMeasure*” ve “*objectiveID*” niteliklerini alabilir. Alabileceği alt elemanlar da <primaryObjective> elemanının aldığı elemalar yani <minNormalizedMeasure> ve <mapInfo> elemanlarıdır.

3.2.3.7 <randomizationControls> elemanı

Sıralama süreci sırasında bir etkinliğin çocuklarının (kendisine bağlı alt etkinliklerin) nasıl sıralanacağı ile ilgili tanımlamaları barındırır. Alt eleman içermez. Bağlı olduğu <sequencing> elemanında en fazla 1 kere kullanılabilir. Tanımlanması opsiyonel olan 4 nitelik alabilir. Bunlardan varsayılan değeri “never” olan “randomizationTiming” niteliği, etkinliğin çocuklarının sıralanışının ne zaman meydana geleceğini gösterir. Alabileceği diğer değerler “once” veya “onEachNewAttempt” şeklindedir. “selectCount” niteliği ise etkinlik ile ilişkili çocuk etkinliklerden seçilmesi gereken çocuk etkinlik sayısıdır ve varsayılan değeri 0’dır. “reorderChildren” niteliği, çocuk etkinliklerin sıralanışının rastgele olup olmayacağını belirler. Varsayılan değeri “false”dur. “selectionTiming” niteliğinin ise varsayılan değeri “never”dir ve seçimin ne zaman olması gerektiğini gösterir. Alabileceği diğer değerler “never, once, onEachNewAttempt” şeklindedir.

3.2.3.8 <deliveryControls> elemanı

Sıralama süresince bir etkinliğin çocuklarının sıralanışı ile ilgili tanımlamalarını içerir. Alt eleman içermez. <sequencing> elemanı içinde en fazla 1 kere bulunabilir. Kullanımı opsiyonel olan üç nitelik alabilir. Bunlardan ilki “tracked”, girişim (attempt) için “objective progress” bilgisi ve “activity/attempt progress” bilgisinin kaydedilip kaydedilmeyeceğini gösterir. Varsayılan değeri “true” olarak tanımlıdır. “completionSetByContent” niteliği, etkinlik için tamamlanma kararının SCO tarafından verilip verilmeyeceğini gösterir. Varsayılan değeri “false” olarak tanımlanmıştır. “objectiveSetByContent” niteliği ise objective’in memnun edici değerinin SCO tarafından atanacağını belirtir ve varsayılan değeri false’dur.

3.2.3.9 <constrainedChoiceConsiderations> elemanı

Sıralama süreci boyunca “choice navigation” taleplerinin nasıl engelleneceği ile ilgili tanımlamaları içerir. Sadece tanımlandığı etkinlik için uygulanabilen bir seçenektir. Alt eleman içermez. <sequencing> elemanı içinde en fazla 1 kere bulunabilir. XML isim alanı http://www.adlnet.org/xsd/adlseq_v1p3 ve **isim** alanı öneki “adlseq” şeklinde tanımlanmıştır (ADL, 2006b). İki nitelik alabilir. Bunlar opsiyoneldir. “preventAction” niteliği varsayılan değeri “false”dur ve çocuk etkinlikler üzerindeki girişimler (attempt’ler), o anki etkinlik ebeveyn etkinlik olmadığı sürece başlatılmaz. “constrainChoice” niteliğinin varsayılan değeri de “false”dur ve sadece “next” mantığında ilerlenen bir sonraki etkinlik seçimini sağlar.

3.2.3.10 <rollupConsiderations> elemanı

Bir etkinliğin kullanım (rollup) süreci içerisine ne zaman dahil olacağını gösteren tanımlamaları içeren elemandır. Bağlı olduğu <sequencing> elemanında en fazla 1 kere kullanılabilir. Seçimli olarak kullanılacak 5 nitelik alabilir. Bunların varsayılan değerleri “always” olarak belirtilmiştir. “requiredForSatisfied”, etkinliğin hangi durum altına dahil edildiğini gösterir. “requiredForNotSatisfied” niteliği ise etkinliğin bağlı olduğu ebeveyn “not satisfied rollup” kuralının değerlendirilmesindeki durumu tanımlar. Diğer nitelik, “requiredForCompleted”, ebeveyn tamamlanmış bir kullanım değerlendirmesi için durumu belirtir. “requiredForIncomplete” niteliği, tamamlanmamış bir kullanım kuralı için durum tanımlaması yapmada kullanılır. Son nitelik, “measureSatisfactionIfActive” etkinlik aktif iken kullanım sıracınca memnuniyet hesaplamada değer (measure) kullanılıp kullanılmayacağını belirtir. Sadece bu niteliğin varsayılan değeri “false” olarak tanımlanmıştır. Niteliklerin alabilecekleri

diğer deęerler ise hepsinde aynı olmakla birlikte, şöyledir. Etkinlięin kullanım kuralı işlemlerine her zaman dahil olacağını gösteren “always”; sadece etkinlięin deneme girişiminde bulunduğu zamanlarda kullanım işlemine dahil olacağını gösteren “ifAttempted”; eęer etkinlik atlanmamışsa dahil olacak olan “ifNotSkipped” ve eęer etkinlik geçici olarak durdurulmamışsa dahil edilecek anlamında olan “ifNotSuspended”.

3.2.3.11 <sequencingCollection> elemanı

<sequencingCollection> elemanı, <sequencing> elemanı ile tanımlanan bir küme sıralama bilgisinin yeniden kullanılmasını sağlayan bir elemandır. Bu eleman nitelik almaz. Özellikleri şöyle özetlenebilir:

- eęer bir <sequencingCollection> istenmiş ise, sadece 1 sıralama koleksiyonu olabilir.
- Eęer bir <sequencing> elemanı, “sequencingCollection” içinde tanımlı bir <sequencing> elemanını göstermekteyse, IDRef nitelięinin kullanılması zorunludur ve “sequencingCollection” içinde tanımlı <sequencing> elemanı ile belirtilen ID’yi referans göstermelidir.
- Eęer bir <sequencing> elemanı bir <item> ya da <organization> elemanının çocuk elemanı olarak tanımlanmış ise, ID nitelięinin kullanılmasına izin verilmez.
- Bu başlık altında tanımlı tüm <sequencing> elemanları için, ID nitelięinin kullanılması zorunludur ve IDRef nitelięinin tanımlanmasına izin verilemez. Bu, diğer sıralama bilgilerine referans verilen sıralama bilgisinin deęiştirilmesini önlemek için konmuş bir kuraldır.

- <sequencingCollection> elemanı, <imscp:manifest> elemanı içinde en fazla 1 kere kullanılabilir (ADL, 2006b). Aşağıda bu elemanın kullanımını gösteren bir kod parçası örnek olarak verilmiştir.

```
</resources>
<imsss:sequencingCollection>
  <imsss:sequencing ID = "pretest">
    <imsss:controlMode choice = "false" choiceExit =
      "false" flow = "true" forwardOnly="true"/>
    <imsss:sequencingRules>
      <imsss:preConditionRule>
        <imsss:ruleConditions>
          <imsss:ruleCondition condition =
            "satisfied"/>
        </imsss:ruleConditions>
        <imsss:ruleAction action = "skip"/>
      </imsss:preConditionRule>
    </imsss:sequencingRules>
  </imsss:sequencing>
</imsss:sequencingCollection>
</manifest>
```


4. TASARLANAN PLATFORMUN (LCDM) ÖZELLİKLERİ VE BENZERLERİNDEN FARKLARI

Web tabanlı uzaktan eğitimde, yazarlık araçlarının öğrenme ortamlarındaki önemi artmaktadır. Ayrıca öğrenme nesnesi havuzları da bilgi yönetiminde kritik nokta haline gelmektedir. Bu çalışma kapsamında, ÖN'lerinin bir ders paketi şeklinde organize edilmesi, depolanması ve üstveri tanımlamalarına bağlı olarak aranabilmelerini sağlamak, kullanılabilirliklerini artırmak amacıyla, bir üst-veri tanımlama ve ÖN içerik paketi oluşturma platformu hazırlanması kararlaştırılmıştır. Tasarlanan platforma “LCDM- Learning Content Design Manager” adı verilmiştir. Bu bölümde, bölüm üçte anlatılan SCORM standardı temel alınarak tasarlanan platformun başlıca özellikleri açıklanmaktadır. Literatürde görülen önceki çalışmalar da değerlendirilerek, tasarımı yapılan sistemin özellikleri saptanmış ve tasarlanan platformun temel özellikleri ve diğer çalışmalardan farkları aşağıdaki gibi belirlenmiştir.

4.1 İçerik Paketleme ve Üstveri İçin Türkiye'deki İlk Uygulama

İncelenen kaynaklarda görüldüğü kadarıyla gerek yurtdışında gerekse yurtiçinde yapılan çalışmalarda tasarlanan araçlar genellikle ya içerik paketi oluşturmak ya da üstveri girişi sağlamak üzere tasarlanmış uygulamalardır. Türkiye'de ÖN içerik paketi hazırlamak amacıyla tasarlanan bir araca, incelenen kaynaklarda rastlanmamıştır. Çukurova Üniversitesince tasarlanan TreeLOM (TreeLOM, 2006) ise, sadece üstveri giriş işlevini yerine getiren bir programdır. Yurtdışında tasarlanan araçlardan sadece ReLOAD ve IMS Package Editor hem üstveri girişi hem de içerik paketleme işlevlerini gerçekleştirmek için kullanılan ve farklı standartları destekleyen iki araçtır, ancak web tabanlı değildir.

Bu çalışmada tasarlanan platform, LCDM, ise Türkiye’de yapılan, web üzerinden eğitsel içerik oluşturarak ÖN içerik paketleri hazırlamak ve üstveri tanımlamalarını yapabilmek için gerekli fonksiyonları içeren ve üniversite seviyesinde kullanılması düşünülen bir uygulamadır.

4.2 SCORM 2004 Standardı Desteği

İncelenen kaynaklarda, içerik paketleme ve üstveri girişi için kullanılan araçlardan SCORM’u destekleyenler SCORM 2004 öncesindeki sürümlerle uyumludur. LCDM platformu, son sürüm olan SCORM 2004’ü desteklemektedir. Böylece platform, IEEE LOM şartnamesine uygun üstveri tanımlamaları yapmak, IMS İçerik Paketleme kurallarına göre eğitsel içeriği paketlemek ve gene IMS Simple Sequencing’e göre sıralama kurallarını uygulamayı mümkün kılmaktadır.

4.3 Web Tabanlı Uygulama ve Kolay Kullanıcı Arayüzü

Tasarlanan platform, etkileşimliliği ve yeniden kullanılabilirliği sağlamak, zamanı iyi kullanmayı kolaylaştırmak için, kolay kullanıcı arayüzlerine sahip olacak şekilde web tabanlı olarak tasarlanmıştır.

4.4 Kaynak Havuzu ve ÖN Havuzu

Web tabanlı olması LCDM platformuna erişimi her zaman ve her yerden mümkün kılacağından, bilgi paylaşımının artırılması amacıyla platformda iki ayrı havuz oluşturulmuştur. Bunlardan biri farklı tipteki kaynak dosyalarının tutulduğu ve istenirse paylaşımına açıldığı *Kaynak Dosyaları Havuzu*, diğeri ise ÖN içerik paketlerinin tutulduğu ve kayıtlı kullanıcıların erişimine açık olan *Öğrenme Nesnesi İçerik Paketi Havuzudur*. Her iki havuzdan da farklı kriterlere göre arama yapmak mümkündür. Kaynak ve içerik paketi havuzları sayesinde, sistemde yüklü bir dosyadan ve yine sistemde tanımlı bir eğitsel içerik paketi projesinden uyarlama imkanı ile yeni bir dosya ve proje oluşturulması

mümkündür. Böylece LCDM sistemi, eğitsel içerik oluşturulması ile ilgili olarak kaliteyi ve verimi artırmış ve maliyeti düşürmüş olmaktadır. Ayrıca, sistemde 2 ayrı havuz olması, hangi dosyanın kim tarafından ve ne zaman sisteme yüklendiği bilgisi ve hangi içerik paketinin kim tarafından, ne zaman oluşturulduğu bilgisi öğrenilebilmektedir.

4.5 Dosya ve ÖN İçerik Paketi Arama

Sistemdeki eğitsel içerik hazırlama yetkisi olan kayıtlı kullanıcılar, sisteme kendi yükledikleri dosyalar arasından arama yapabilecekleri gibi, aynı zamanda başkalarının sisteme yüklediği ve “paylaşılabilir” olarak işaretlediği dosyalar arasından da arama yapılabilir ve bu dosyaları kendi oluşturacakları eğitsel içeriklerde kullanabilirler. Ayrıca, sistemdeki kayıtlı kullanıcıların tümünün kendilerinin oluşturduğu ve diğer kullanıcıların oluşturduğu tamamlanmış eğitsel içerik paketleri arasından Dosya/Paket Adı, Dosya/Paket Tipi (MIME Type), Oluşturan Kişi ve Oluşturulma Tarihi kriterlerine göre arama yapılması mümkündür. Birden fazla kritere göre arama yapılmak istenilirse, kriterler mantıksal “VE” işlemine alınarak arama gerçekleştirilir.

4.6 Önizleme İmkani

LCDM platformu, iki farklı önizlemeyi desteklemektedir. Bunlardan biri daha çok kullanıcının kaynak dosyalarını sisteme yüklerken kullanacağı dosya bazında önizleme, diğeri ise oluşturulan ÖN içerik paketlerinin görüntülenebileceği proje bazında önizleme fonksiyonlarıdır. Dosya bazında önizleme için sistemdeki kaynak dosyalar üzerine tıklanarak içeriğin yeni bir tarayıcı penceresinde görüntülenmesi sağlanmıştır. Paket bazında önizlemede ise, öğrencinin göreceği içerik yapısı aynı sırada fakat özel olarak tasarlanmış bir web arayüz üzerinden yeni bir tarayıcı penceresinde görüntülenebilmektedir.

4.7 Dosya ve ÖN İçerik Paketi İndirebilme

Eğitsel içerik hazırlama yetkisi olan kayıtlı kullanıcılar, sistemdeki kaynak dosyalarını kendi disklerine geri yükleyebilirler. Bu özellik, kullanıcının bir projede kullandığı dosyaları kendi yerel diskinde saklayarak kaynaklarını tüketmesini engeller ve istenildiğinde geri alınan dosya üzerinde değişiklik yaparak platforma farklı bir dosya halinde yeniden yüklemesini sağlar. Eğitsel içerik paketlerinin yerel diske kopyalanması ise iki farklı şekilde gerçekleşmektedir. Bunlardan birincisi SCORM formatında hazırlanan ve uyumlu bir eğitim yönetim sistemine yüklenerek çalıştırılabilecek olan paket formatının kopyalanması, ikincisi ise oluşturulan paketteki eğitsel içeriğin doğrudan herhangi bir makinedeki herhangi bir web tarayıcı sayesinde görüntülenmesini sağlayacak olan önizleme formatının kopyalanmasıdır.

4.8 Hazır İçerik Paketlerinin Sisteme Yüklenmesi

LCDM platformunda, başka uygulamalar kullanılarak hazırlanan SCORM 2004 uyumlu içerik paketlerinin yüklenmesi ve çalışabilir hale getirilmesi mümkündür. Dışarıdan yüklenen içerik paketleri, yükleyen kullanıcıya bağlı olarak kaydedilir ve sistem dışından alındığına dair işaretlenir. Sıkıştırılmış dosya olarak sisteme yüklenen paketler, açılarak veritabanındaki ilgili tablolara işlenir. ReLOAD ile oluşturulan içerik paketlerinin sisteme yüklenmesi ve çalıştırılması test edilmiştir.

4.9 Eğitim Yönetim Sisteminden Bağımsız Kullanılabilme

LCDM platformunun bir LMS üzerinde çalışmasına gerek yoktur. Web tabanlı çalışabilmesi, sisteme dosya yükleme, dosyaları kullanarak içerik paketleri oluşturabilme ve bu paketler arasından arama yapabilme, içerik önizleme ile kullanıcı ilgili içeriği bir LMS'e bağlı kalmadan takip edebilmektedir.

4.10 Birden Fazla Dil Desteđi

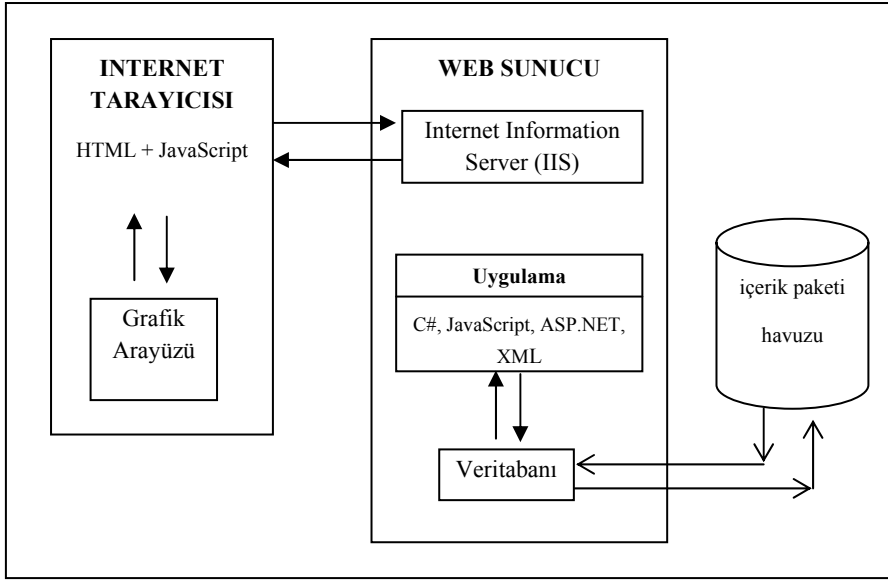
LCDM platformunda, Őu anda Trke ve İngilizce olmak zere iki dil tanımlanmıŐtır. Diller ve arayzler zerindeki ifadelerin bu dillerdeki karŐılıkları veritabanında tutulmakta, kullanıcı istediđi dili setiđinde dođrudan o dile ait arayz ile karŐılaŐmaktadır. Ayrıca, platforma yeni dil eklenmesi ve mevcut dil tanımlamaları zerinde gncellemeler yapılabilmesi iin bir arayz tasarlanmıŐtır.

4.11 zerinde alıŐılan Platform

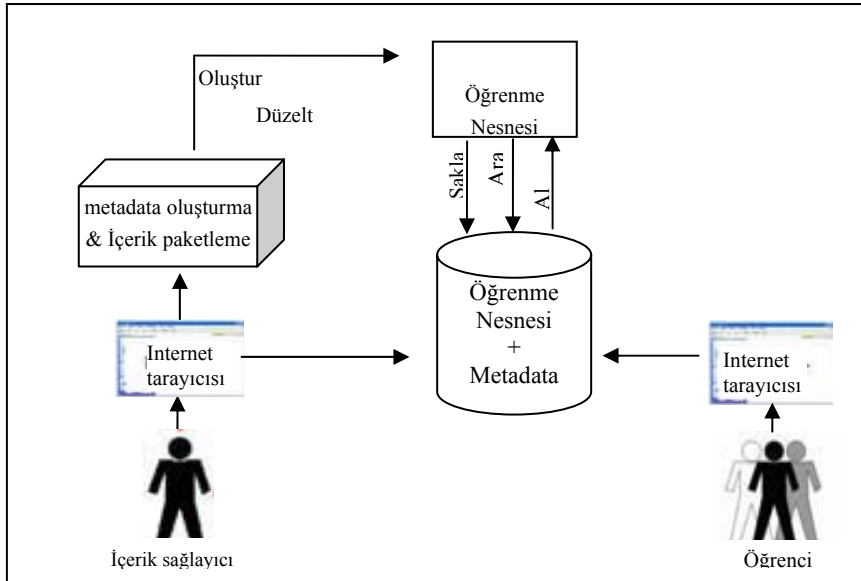
LCDM platformunda Őekil 4.1’de de grldđi zere, Internet Information Server (IIS), ASP.NET, XML, C#, javascript dilleri ve veritabanı olarak da SQL veritabanı kullanılmıŐtır. Veritabanı ile ilgili detaylı bilgiler Ek Aıklamalar A’da verilmiŐtir. LCDM platformunun niversite dzeyinde kullanımı amalanmıŐtır. Gerek kullanılması dŐnlen niversitede kullanılan programlar olmasından ve gerekse kolay bulunabilir olmalarından dolayı Windows, ASP.NET ve SQL veritabanı seilmiŐtir. Kolay bir kullanıcı arayz ile eriŐilebilecek olan sistem IIS server zerinden alıŐmakta ve arka planda da alıŐan uygulamalar ve veritabanı bađlantısı sayesinde dosya ve ierik paketlerine eriŐilerek ilgili fonksiyonlar yerine getirilmektedir.

4.12 Sistemin alıŐması

Őekil 4.2’de grldđi zere, ierik sađlayıcı, kullandığı tarayıcı zerinden sisteme giriŐ yaptıktan sonra, st-veri oluŐturma ve ierik paketleme aracı sayesinde kendi hazırladıđı đrenme nesneleri ierik paketlerine eriŐerek gncelleme yapabilir, yeni đrenme nesnesi ierik paketleri oluŐturabilir. Aynı Őekilde đrenci de Internet tarayıcısı zerinden sisteme dahil olarak yetkilerine gre ierik paketi havuzuna eriŐerek arama, nizleme ve dosya indirme iŐlemlerini yapabilmektedir.



Şekil 4.1 LCDM Çalışma platformu.



Şekil 4.2 LCDM Sisteminin Çalışması.

5. TASARIM

Bu bölümde, LCDM sisteminin önceki bölümlerde anlatılan özelliklere sahip olacak ve alt yapıyı destekleyecek şekilde SCORM standardına uygun olarak tasarlanması ile ilgili bilgiler verilmektedir.

5.1 Kullanıcı Grupları ve Yetkileri

LCDM platformunda, Şekil 5.1’de de görüldüğü üzere, üç grup kullanıcı tanımlanmıştır. Bunlar, ÖN içerik paketlerini hazırlayacak olan *yayımcı*, hazır paketler arasından ders/konu takibi yapabilecek olan *öğrenci* ve tüm sistemin yönetiminden sorumlu olan *yönetici*’dir. Aşağıda her üç kullanıcı grubunun yapabileceği işlemler özet olarak listelenmiştir. Detaylar, ilerleyen kısımlarda sistemde tanımlı modüller açıklanırken ilgili başlıklar altında verilmiştir.

Yayımcı grubundaki kullanıcıların, yani eğitmenler, Dosya Yöneticisi, Proje Yöneticisi, Paket Yöneticisi, Metadata Yöneticisi, Sıralama, Arama ve Önizleme modüllerine erişim hakkına sahiptirler. Bir yayımcı, aşağıdaki fonksiyonları yerine getirebilir.

- Sisteme dosya yükleyebilir ve URL tanımlaması yapabilir.
- Tamamlanmış olan içerik paketlerinin listesini görebilir.
- Sistemdeki dosyaları kullanarak bir içerik paketi için organizasyon, proje oluşturabilir
- Oluşturduğu eğitsel projeyi sıkıştırarak saklamaya hazırlayabilir.
- Dosya ve paket bazında önizleme yapabilir.
- Kendi dosyaları ve paylaşımına açık diğer dosyalar arasından arama yapabilir.
- Dosya ve paket bazında arama yapabilir.
- Dosya bazında indirme yapabilir.

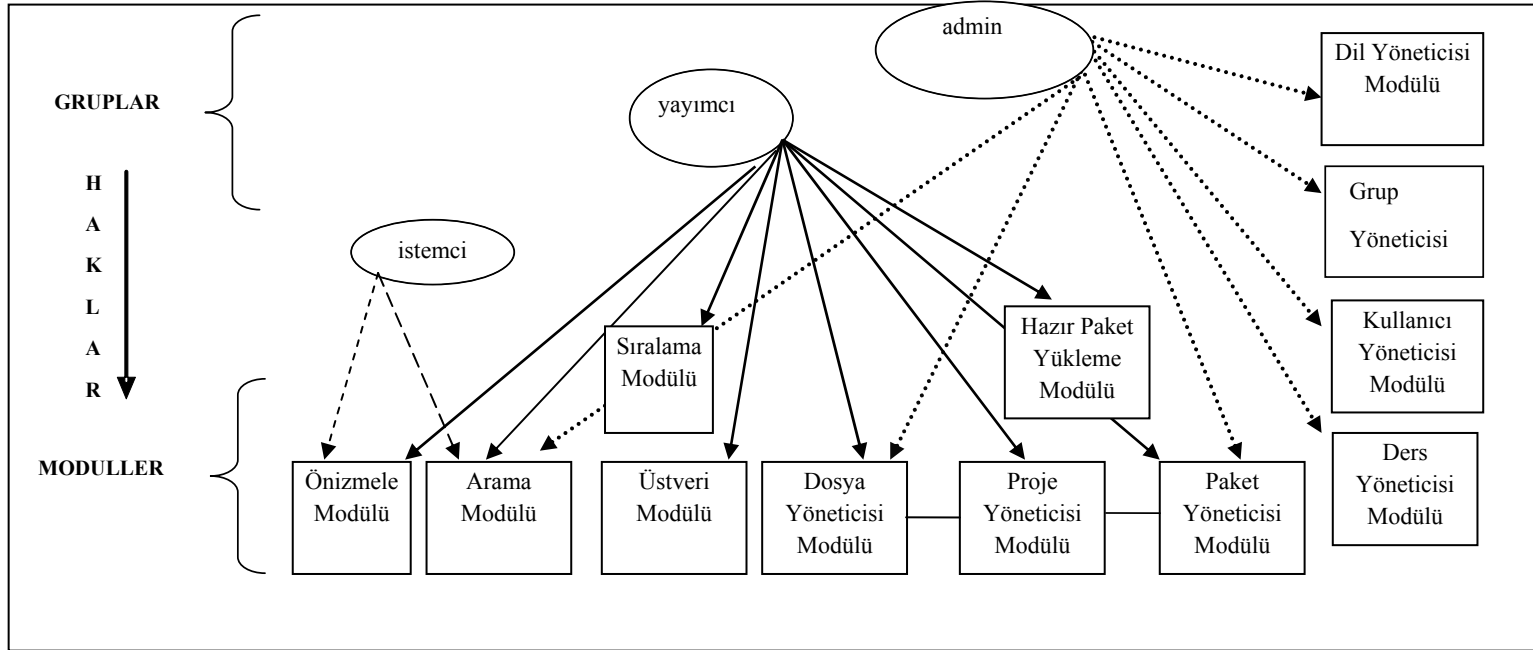
- Paket bazında, hem SCORM 2004 uyumlu hem de önizlenebilen formattaki paketleri yerel makinesine kopyalayabilir.
- Dosya / organizasyon / paket bazında üst-veri girişi yapabilir.
- Organizasyon / konu başlığı (item) bazında sıralama kuralları, yani içeriğin öğrenciye hangi sırada gösterileceğini belirleyen kurallar, ekleyebilir.

İstemci grubundaki kullanıcılar yani öğrenciler, sistemde aldıkları dersler kapsamındaki tamamlanmış içerik paketleri ile ilgili olarak aşağıda açıklanan işlemleri yapabilirler.

- Platformdaki içerik paketlerini ve bunlarla ilgili bilgileri görebilirler.
- İçerik paketleri arasında arama yapabilirler.
- İstedikleri içerik paketinin önizlemesini görebilirler.
- İçerik paketlerini kendi makinelerine indirebilirler.
- İçerik paketinin önizlenebilen formatını kendi disklerine indirebilirler. Böylece eğitsel içeriği bir LMS'e bağımlı olmadan kendi makinelerindeki internet tarayıcı aracılığı ile görüntüleme imkanı sağlanmış olmaktadır.

Yönetici grubundaki kullanıcılar ise aşağıdaki fonksiyonları gerçekleştirme yetkisine sahiptirler.

- “Kullanıcı Yöneticisi” modülü ile sistemde yeni kullanıcı tanımlamak, kullanıcının aldığı ya da verdiği dersleri belirleyebilir ve ilgili güncellemeleri yapabilirler.
- “Grup Yöneticisi” modülü ile mevcut gruplara dahil olan kullanıcıları düzenleyebilir ve sistemde yeni gruplar tanımlayabilirler.



Şekil 5.1 Gruplar ve Hak sahibi oldukları modüller arası ilişkiler.

- “Ders Yöneticisi” modülü sayesinde sistemde yeni ders tanımlayabilir, mevcut derslerle ilgili güncellemeler yapabilirler
- “Dosya Yöneticisi” modülüne erişerek, yayımcıların platforma yükledikleri dosyalar, yükleme zamanı, URL tanımlamaları, o kullanıcı ile ilişkili izin yapısı gibi bilgileri görebilirler ve müdahale edebilirler.
- “Paket Yöneticisi” modülünde yayımcı ve verdiği dersler bazında arama yapabilirler.

“Birden fazla dil desteği (multilanguage) fonksiyonu ile ilgili olarak, sisteme yeni bir dil ve o dile ait tanımlamaları girebilirler.

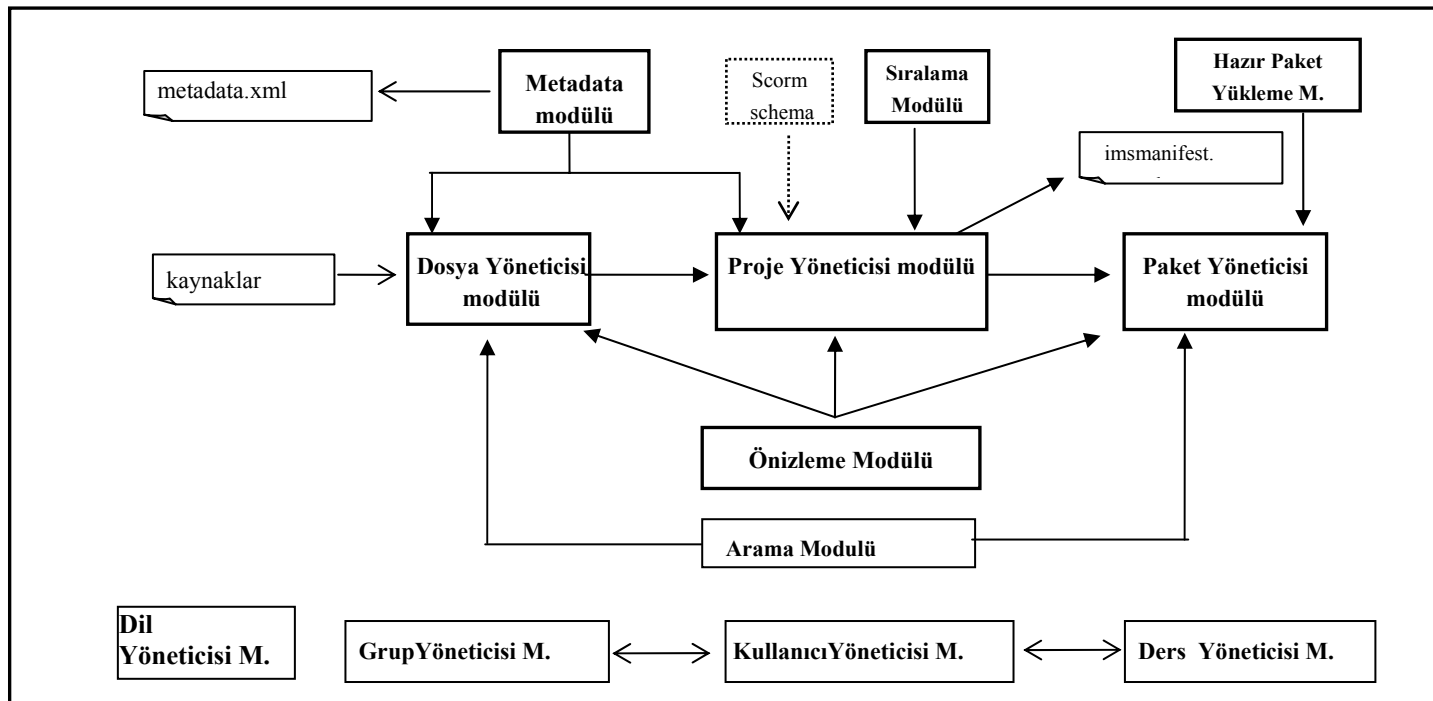
5.2 Platformda Tanımlı Modüller ve İşlevleri

LCDM platformunda toplam 12 tane modül tanımlanmıştır. Bunlar, Şekil 5.2’de de görüldüğü gibi, “Grup”, “Kullanıcı”, “Ders”, “Dosya”, “Proje”, “Paket” yönetimi modülleri, “Önizleme”, “Arama”, “Metadata”, “Sıralama”, “Hazır Paket Yükleme” ve “Dil Yönetimi” modülleridir. Aşağıda bu modüller ve işlevleri ayrı ayrı başlıklar altında açıklanmıştır.

5.2.1 Ders yöneticisi modülü

Ders yöneticisi modülü, sadece yönetici grubundaki kullanıcıların erişime açık bir modüldür ve şu fonksiyonları yerine getirmektedir:

- Sistemde tanımlı dersler, bu derslere ait kısa açıklamalar, dersin eklenme tarihi gibi bilgilerin listelenmesi
- Sisteme yeni ders eklenmesi ve dersin silinmesi
- Sisteme kayıtlı kullanıcılar arasından dersi veren yayımcıların ve dersi alan öğrencilerin seçilmesi



Şekil 5.2 LCDM platformundaki modüller arasındaki ilişkiler.

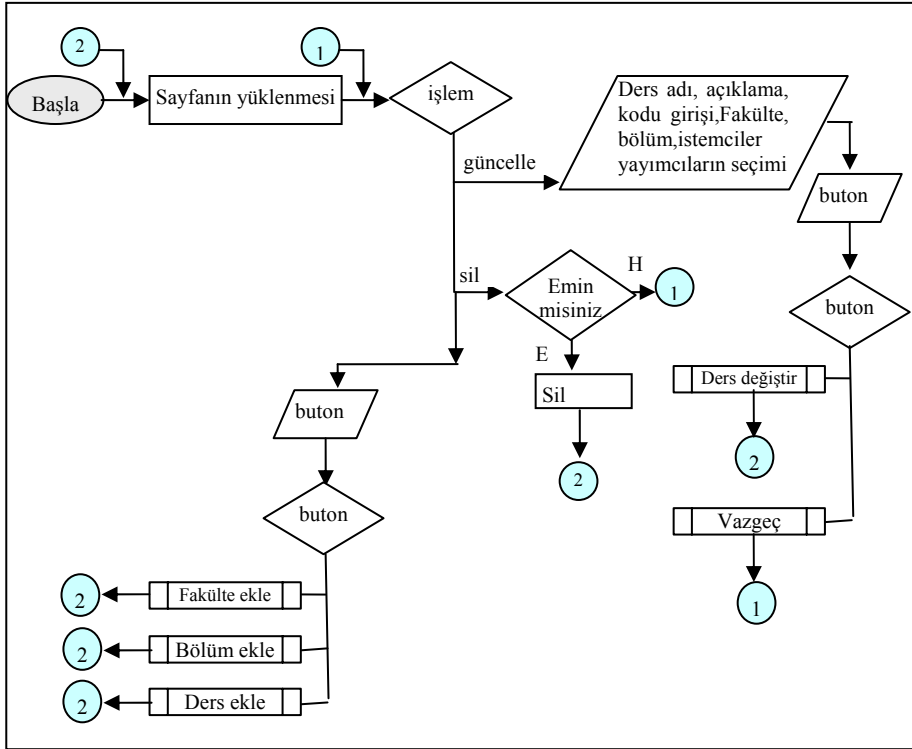
- Derse ait ders adı, açıklaması, kodu, dersi hangi yayımcıların verdiği ve hangi istemcilerin (öğrencilerin) aldığı gibi bilgilerin güncellenmesi.

LCDM platformunda bir ders tanımı yapmak için, sistemin üniversite seviyesinde kullanılmasının amaçlanması nedeniyle, sırayla önce bir fakülte tanımı yapmak ya da dersin verileceği fakülteyi seçmek, o fakülte altındaki bölümleri sisteme tanımlamak ya da bölümü seçmek ve en son seçilen bölüm altında ders tanımlamalarını yaparak dersi oluşturmak gerekmektedir. Yayımcıların hazırlayacağı eğitsel içerikler sistemde tanımlı olan derslerle ilişkili olarak oluşturulacağından, bu hiyerarşik yapı hem yayımcı, hem istemci hem de yönetici grubundaki kullanıcılar için kullanım kolaylığı sağlamaktadır. Şekil 5.3'te "Ders Yöneticisi" modülü sayfası görülmektedir. Bu modülün işleyişi, en genel biçimde, ekranda görülen temel işlemlere göre çıkartılan ve Şekil 5.4'te görülen akış şeması üzerinden de anlaşılabilir.

The screenshot shows the LCDM (Learning Content Design Manager) interface. The top navigation bar includes the LCDM logo and the text 'Learning Content Design Manager'. A user profile box on the right shows the user name 'Kullanıcı Adı: birimb' and the role 'Yönetici', with a 'Logout' button. The main navigation menu includes 'Dosya Yöneticisi', 'Ders Yöneticisi', 'Grup Yöneticisi', 'Paket Yöneticisi', 'Dil Yöneticisi', and 'Kullanıcı Yöneticisi'. The 'Ders Yöneticisi' section is active, showing buttons for 'Fakülte Ekle', 'Bölüm Ekle', and 'Ders Ekle'. Below these buttons, there is a text block explaining that users can view and manage courses in the system, including updating and deleting them. A table at the bottom lists the courses:

Dersin Kodu	Ders Adı	Açıklama	Eklenme Tarihi	
ISLT 301	Yöneylem		08-48-2007	Güncelle Sil
ISLT101	İşletmeye Giriş temel konular	09-36-2007		Güncelle Sil

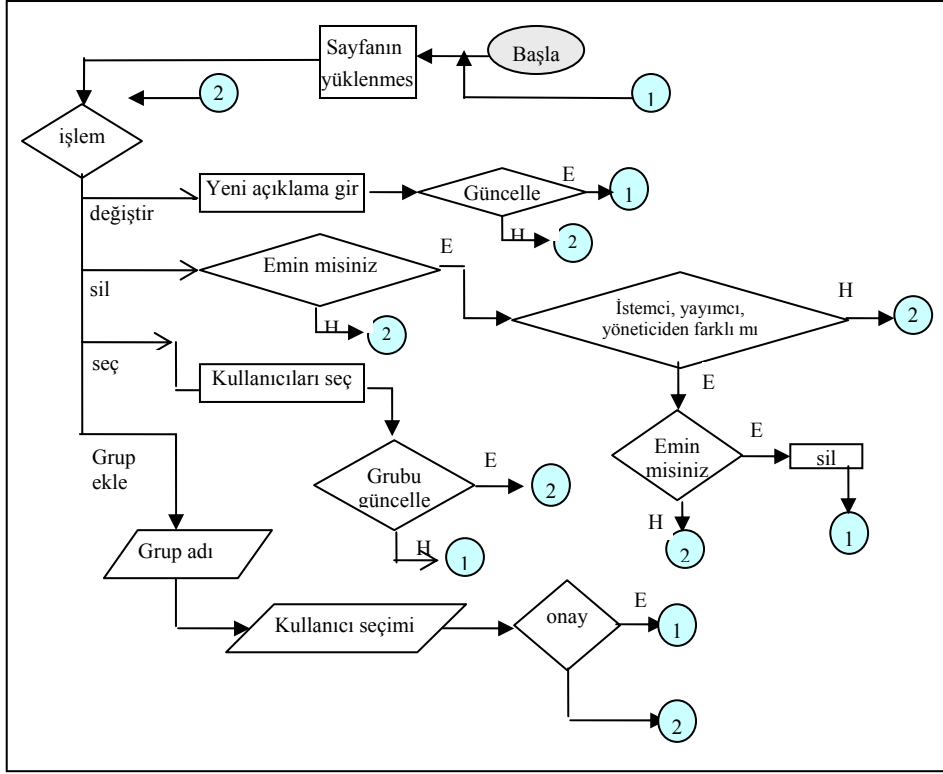
Şekil 5.3 LCDM platformundaki "ders yöneticisi" modülü arayüzü.



Şekil 5.4 LCDM platformu “ders yöneticisi” modülü temel akış diagramı.

5.2.2 Grup yöneticisi modülü

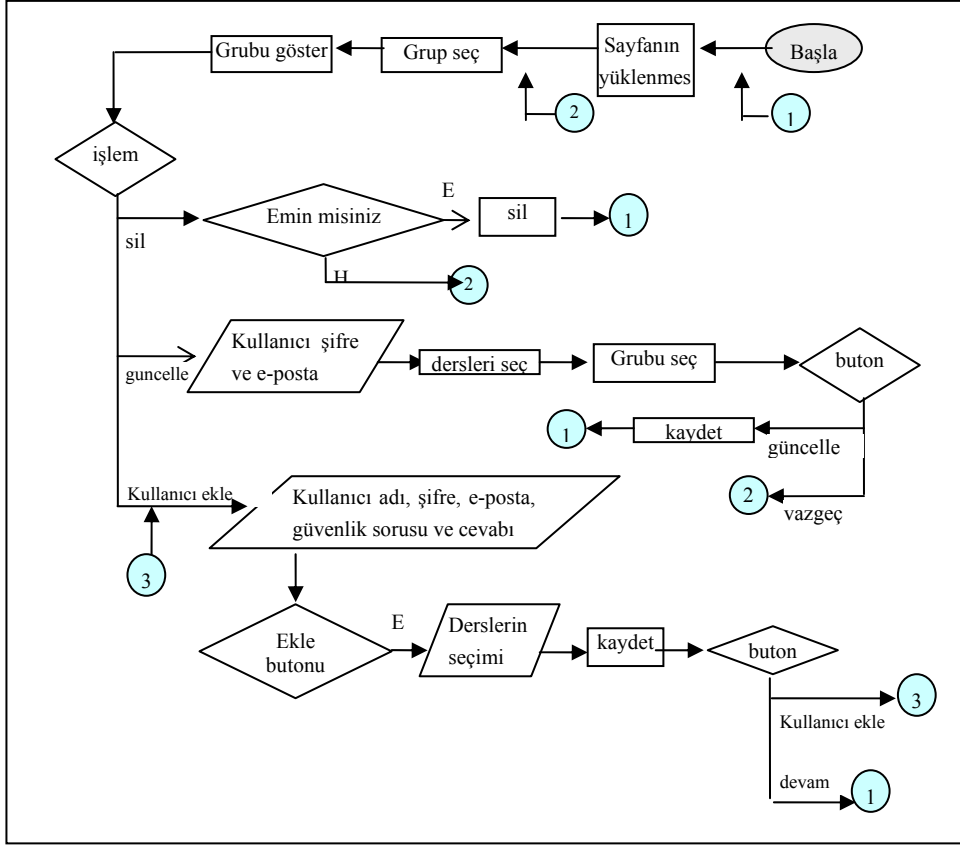
Grup yönetimi sadece yönetici grubundaki kullanıcıların erişimine açıktır. Bu modül sayesinde bir yönetici, LCDM platformunda yeni kullanıcı grubu oluşturabilir; oluşturulan bir gruba sistemdeki diğer kullanıcılardan kişi ekleyebilir; gruplardaki kullanıcıları görerek bilgi alabilir ve grup değişikliği yapabilir; grup adı ve grupla ilgili kısa açıklama bilgisini güncelleyebilir ve mevcut bir grubu silebilir. Ancak platformda tanımlı üç temel grup yani istemci, yayımcı ve yönetici grupları silinemez. Bunlar dışında tanımlanan gruplar silinebilmektedir. Şekil 5.5’te grup yöneticisi modülünün işleyişini anlatan akış diagramı görülmektedir.



Şekil 5.5 LCDM platformu “grup yöneticisi” modülü akış diagramı.

5.2.3 Kullanıcı yöneticisi modülü

Sadece yönetici grubundaki kullanıcıların erişimine açık olan bu modül sayesinde, grup bazında tanımlı olan kullanıcılar listelenebilir, seçim ve güncellemeler yapılabilir. Kullanıcıların sisteme son giriş tarihleri görülebilir. Kullanıcı adı, şifresi, e-posta, güvenlik sorusu gibi bilgiler alınarak bir gruba yeni kullanıcı eklenebilmektedir. Kullanıcı bilgilerinin güncellenmesi ve kayıtlı bir kullanıcının sistemden silinmesi sağlanabilir. Kayıtlı kullanıcı yayımcı grubundan ise vereceği dersler, istemci grubundan ise alacağı derslerin seçilmesi yine bu modül altında gerçekleştirilebilecek işlemlerdir. Şekil 5.6’da bu modülün çalışmasını gösteren akış diagramı verilmiştir.



Şekil 5.6 LCDM platformu “kullanıcı yöneticisi” modülü akış diagramı.

5.2.4 Dosya yöneticisi modülü

Bu modüle sadece yayımcı ve yönetici grubundaki kullanıcılar erişebilirler. Sisteme URL tanımlama dosya yükleme, özizleme, silme, dizin yapısında değişiklik yapabilme, isim değiştirme gibi işlevleri yerine getirebilmektedirler. Yönetici grubundaki kullanıcıların ise bu modül ile ilgili olarak, hangi yayımcının platforma hangi kaynak dosyaları, ne zaman yüklediği gibi bilgileri görme ve o kullanıcıya ait dizin yapısına müdahale etme hakkı vardır (Şekil 5.7).



Şekil 5.7 LDCM platformundaki “dosya yöneticisi” modülü arayüzü

Yayımcılar, bir seferde tek bir dosyayı platforma yükleyebilirler. Ancak “html” dosyalarının içindeki her referans otomatik olarak taranmakta ve buldukları yerler gösterilip, arama alanı açılarak bu dosyaların da yüklenmesi sağlanmaktadır. Bu dinamik kontrol için “DBauer.Web.UI.WebControls.DynamicControlsPlaceholder.dll” kütüphanesi kullanılmıştır. Ayrıca, “html” dosyalarının yüklenmesi sırasında, içerisindeki html tagları arasında dolaşmayı kolaylaştırmak amacıyla “MILHTMLParser.dll” kütüphanesi kullanılmıştır.

Bir “html” dosyanın referansları olan diğer dosyalar ve buldukları dizinin adı veritabanında tabloda saklanmaktadır. Tüm dosyalara tekil bir isim verilir ve sunucuda tek dizin altında tutulurlar. Böylece aynı dosyanın birden farklı kopyası tutulabilir ve aşağı doğru karmaşık dizin yapısı olmaz.

Yayımcıların sisteme yükleyecekleri dosya boyutu en fazla 50MB ile olabilir. Yayımcıların sayısının ve hazırlanacak eğitsel içerik miktarındaki artış düşünüldüğünde, platforma daha büyük boyutta dosya yüklenmesi disk alanının harcanmasına neden olacaktır. Kullanılan sunucunun kapasitesi, desteklenen kullanıcı sayısı ve Internet bağlantı kalitesi gibi etkenlere göre bu limit değişebilir. Bu nedenlerle, yüklenecek dosya boyutunun artması yükleme süresinin artmasına neden olabilir.

Kullanıcının sisteme yüklediği dosya adı aynı kalmayarak, GUID yani “Globally Unique Identifier” ile farklılaştırılarak saklanır. GUID yapısında dosya uzantıları olmaz ve GUID kullanılması ile dosyaların eski sürümlerinin saklanması sağlanmıştır. Gerçek dosya bilgileri ve hiyerarşik yapı tabloda tutulur. Değişik içerik paketleri oluştururken, aynı dosyayı sisteme tekrar tekrar yüklenmek yerine mevcut dosya kullanılabilir. Bir yayımcı aynı zamanda platform üzerinde URL tanımlaması yapabilir. Bu sebeple sistemde bir HTML sayfası oluşturularak ilgili adrese bu sayfadan bağlantı verilmektedir.

Yayımcı, dosya yükleyip, URL tanımlarken aynı zamanda platform üzerinde kendisine ayrılan alanda dizin de yaratabilir. Yüklenen dosyalar ve tanımlanan URL adresleri hiyerarşik yapıda sanal dizinler oluşturularak bu dizinler altına taşınabilir. Dosyaların yükleneceği ve URL tanımlamalarının yapılacağı ana dizinin adı *DosyaDeposu* olarak belirlenmiştir. Bu dizin Windows işletim sisteminde “inetpub\wwwroot” dizini altında görünmeyeceğinden URL girilerek erişim sağlanamaz.

Dosya yöneticisi sayfasında, listelenen dizin yapısındaki bir dosya ya da URL tıklanırsa ilgili dosya ayrı bir pencerede açılarak önizleme sağlanabilmektedir. Eğer seçilen dosya “html” formatında ise, “html” kod içerisinden başka dosyalara olan referanslar taranarak bulunan dosyalar ana dosya ile OnIzlemeDosya dizinine dizin yapısı korunarak kopyalanır ve ilgili dosya ayrı bir pencerede gösterilir. Önizleme

sırasında bir html belgesinin referans verdiği dosyalar için, bulunmaları gereken izin ve ilgili diğer bilgiler veritabanında tablodan çekilerek okunmaktadır.

Dizin ya da izin içerisindeki dosya ve URL tanımlamalarının listelendiği tablodaki *İşlemler* sütunundan, ilgili komut tıklanarak dosya /URL tanımlaması ya da izin silme işlemi gerçekleştirilebilir. Yanlış işlem yapmamak amacıyla, kullanıcıdan “doğrulama” istenmektedir. Silinmesi istenilen izin/dosya/URL’nin herhangi bir projeye dahil olup olmadığı kontrol edilir ve dahil değil ise silinir. Dosya html ise, silinirken aynı kontrol her dosya için yapılır. Eğer dosya bir projeye dahilse, kullanıcıya dosyanın bir projede kullanıldığını ve silinemeyeceğini belirten bir açıklama mesajı görüntülenir. Eğer silinmek istenilen izin de, proje içerisinde bir organizasyonda kullanıldıysa aynı şekilde uyarı mesajı görüntülenir.

Ayrıca, kullanıcı sistemdeki bir izin, dosya ya da URL tanımlamasına ait isim değişikliği yapabilmektedir. Dizin/dosya ismi değiştirilirken, ilgili dosyanın bir projede kullanılıp kullanılmadığı bilgisi kontrol edilmez. Çünkü, dosyalar için ilgili projelerde dosya ismi ile değil *dosyaID*’si ile referans verilmektedir. Dizinler ise sadece dosya yöneticisi modülünde sanal saklama alanları olarak kullanılır, projelerle ilgileri yoktur ve bunların isimlerinin değişmesi proje oluşturma aşamasında önem taşımamaktadır. Şekil 5.8’de dosya yöneticisi modülünün genel işleyiş mantığını açıklayan akış diagramı görülmektedir.

Bu modülde ve diğer modüllerde dosya, paket, proje listelemek gibi işlemler için kullanılan ızgaralı yapıda sütun isimlerine tıklanıldığında ilgili sütuna bağlı olarak listelenen içerik sıralanmaktadır. Bu ızgaralı yapı için “RadGrid.Net2.dll” kütüphanesi kullanılmıştır.

5.2.5 Proje yöneticisi modülü

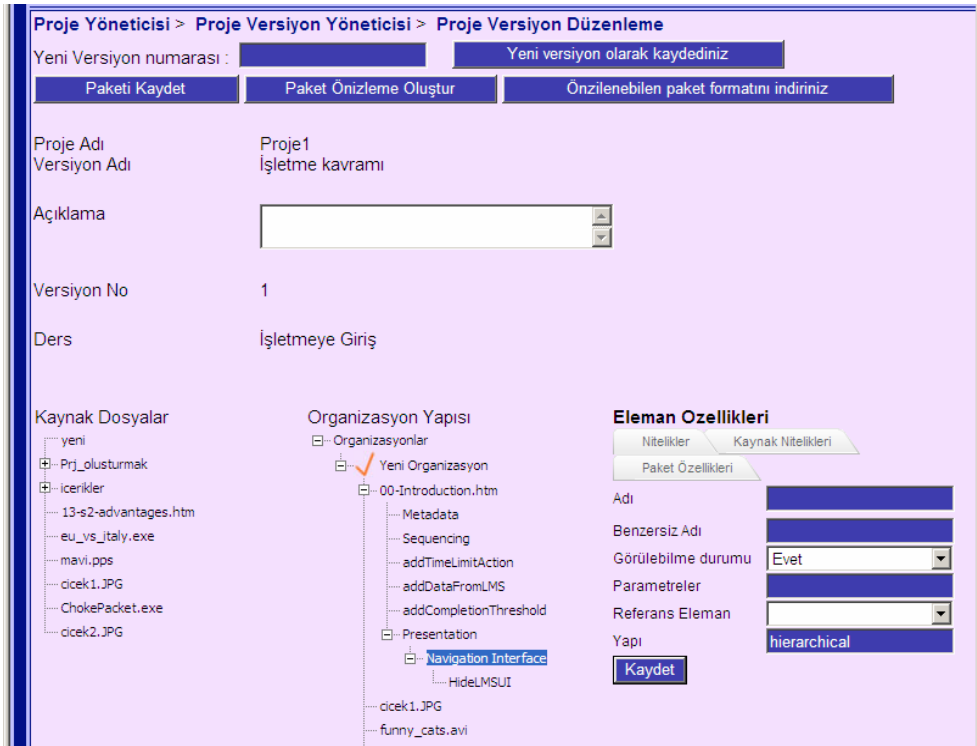
Çok genel bir ifade ile buradaki *proje* kavramı, seçilen bir ders ile ilişkili olarak ve sisteme yüklenen kaynak dosyalarının kullanılması ile öğrencinin göreceği bir içerik organizasyonu yapısının oluşturulması ve oluşturulan yapının bir içerik paketi olarak kaydedilmesi sürecini ifade etmektedir.

Bu süreç içerisinde, içerik yapısında kaynak dosya, öğrencinin göreceği içerik içerisindeki bir başlık ya da içeriğinin kendisi gibi farklı kademelerde üst-veri ve sıralama bilgilerinin tanımlanması da Proje Yöneticisi Modülü içerisinde yapılabilecek işlemlerdir. Ancak üstveri tanımlamaları ve sıralama bilgilerinin girilmesi ayrı modüller olduğundan farklı başlıklar altında ele alınmıştır. Eğitsel içerik oluşturulduktan sonra, alınıp farklı yazarlık araçları arasında çalışabilmesi ve farklı LMS’lerde kullanılabilmesi, içerik havuzları arasında taşınabilmesi için, eğitsel içeriğin paketlenmesi gerekmektedir.

LCDM platformunda projeler, klasör bazında tutulmazlar yani fiziksel olarak bir yer işgal etmezler. Proje bilgileri ve bir projenin farklı sürümleri veritabanında tablolarda tutulur.

Proje yönetimi modülünde oluşturulacak olan eğitsel içerik yapısı yani içerik organizasyonu, tamamen “Dosya Yöneticisi” modülündeki dosyalar ve URL tanımlamaları kullanarak oluşturulmaktadır ve bu dosyalarla ilgili bilgiler veritabanından çekilerek elde edilir. Kullanıcının içerik organizasyonu oluşturduğu arayüzde, iki ayrı ağaç yapı görüntülenmektedir. Bunlardan biri, sistemdeki kaynak dosyalarının bulunduğu, diğeri ise bu kaynak dosyalarının sürüklenip bırakılarak yeni içerik organizasyonunun oluşturulduğu ağaç yapısıdır. Ağaç yapısı sürükle-bırak metodu ile ilgili herşey tek bir ekrana toplanarak, kullanım kolaylığı sağlanmıştır. Organizasyonun oluşturulduğu ağaç yapısı üzerinde

fare sağ tuş fonksiyonunun çalışması sağlanmış ve ilgili her organizasyon, organizasyon altında tanımlı her öge ve o ögeye bağlı olan kaynak dosyalar için yapılabilecek olan işlemler tanımlanmıştır. Bu ağaç yapının oluşturulabilmesi için, RadTreeView.Net2.dll kütüphanesi kullanılmıştır. Şekil 5.9'da LCDM platformunda içerik organizasyonu oluşturulmak için kullanılan ekran görüntülenmektedir.



Şekil 5.9 LCDM platformundaki “proje yöneticisi” modülüne ait arayüz.

“Organizasyonlar” ağacı, üzerinde yeni organizasyon eklemek, eklenen her yeni organizasyon için sağ tuş menüsünde, organizasyonun silinmesi, altına yeni item eklenmesi, varsayılan organizasyon yapılması, üst-veri bilgilerinin eklenmesi, sıralama kurallarının tanımlanması ile

ilgili işlemler tanımlanmıştır. Bir organizasyon altına eklenen her öge için ise sağ tuş menüsünde, yukarıda anlatılan “TimeLimitAction”, “DataFromLMS”, “CompletionThreshold”, “Presentation” bilgilerinin girilmesi için başlıklar, ayrıca ilgili item’ın silinmesi, üstveri tanımlamalarının yapılması, sıralama kurallarını eklenmesi ve o item’ın refere ettiği kaynak dosyaya ait üst-veri tanımlamalarını girilebilmesi için ilgili başlıklar tanımlanmıştır. Yayımcı isterse, seçilen işleme göre ağaç yapının sağ tarafında açılan bir form üzerinden paket özellikleri, nitelikler, kaynak dosya nitelikleri gibi alanlara veri girişi yapılabilmektedir.

İlgili adımlar tamamlandıktan sonra, oluşturulan eğitsel içeriğin proje halinden çıkarak LCDM platformunda paket halinde saklanabilmesi için kaydedilmesi gerekmektedir. “Paketi Kaydet” denildiğinde, projede oluşturulan organizasyon arka planda, “imsmanifest.xml” dosyası içerisinde ilgili kısımlara yerleştirilmekte, kullanılan tüm kaynak dosyalar, ilgili “schema” dosyaları paketin adını taşıyan bir dizin altında bir araya getirilip sıkıştırılarak zip dosyası halinde kaydedilmektedir. Paketlerin sıkıştırılmış dosyaya dönüştürülebilmesi için “ICSharpCode.SharpZipLib.dll” kütüphanesi kullanılmıştır. dosyasının Bu aşamadan sonra kaydedilen zip dosyaları ile ilgili işlemler Paket Yönetim modülünde gerçekleştirilir.

LCDM sisteminde, sadece yayımcılar aktif olarak proje oluşturma hakkına sahiptir. Yayımcı grubundan bir kullanıcı proje yönetimi sayfasına girdiğinde kendi verdiği dersler bazında mevcut projelerinin listesini görebilir. Listelenen projeler ile ilgili Proje-ID numarası, Proje dosyasının adı, açıklaması, projenin eklenme tarihi gibi bilgilere erişebilir. Yayımcının yeni bir proje oluşturması için öncelikle projenin bağlı olacağı dersi seçerek, proje adı, proje hakkında kısa açıklama gibi proje bilgilerini sisteme girmesi gerekmektedir. Bu bilgilerin

girilmesinden sonra kullanıcı, o proje ile ilgili bir sürüm oluşturmak zorundadır. Sürüm oluşturma, SCORM spesifikasyonunda tanımlı bir özellik değil, tamamen LDCM platformunda içerisinde bulunması gereken bir özellik olarak düşünülmüştür. Bir projeye ait birden fazla sürüm oluşturulması ve saklanması mümkündür. Sistemde proje ve sürüm bilgilerini oluşturmadan içerik organizasyonu oluşturmak mümkün değildir. Bir projede birden fazla organizasyon oluşturabilir fakat bunlardan biri “varsayılan” olarak işaretlenmek zorundadır. Bu seçim yapılmazsa, oluşturulan ilk organizasyon varsayılan olarak işaretlenir. “Organizasyon”, öğrencinin içeriği nasıl göreceğini gösteren bir yapıdır ve bir kitaptaki “İçindekiler” listesi gibi düşünülebilir.

Bir proje oluşturulduktan sonra aynı proje altından birden fazla sürüm oluşturulabilir. Bunun için iki metot kullanılabilir. İlki, projenin sürümlerinin listelendiği ekranda “Versiyon Ekle” butonunu tıklayarak ilerlemek; ikincisi ise projeye ait bir sürümdeki “Organizasyonlar”ı seçerek eğitsel içerik organizasyonu yapısında, üstveri ve sıralama bilgilerinde vs. istenilen değişiklikleri yapıp son halini aynı proje altında yeni bir sürüm numarası ile kaydetmek şeklindedir (Bkz. Şekil 5.9). Proje ile ilgili her şey veritabanında tutulduğundan projenin farklı sürümleri için yeni dizin yaratılmaz ve kullanılan dosyaların gereksiz yere kopyaları alınmayacağından sistem kapasitesi boşa harcanmaz. Bu şekilde sürüm oluşturma, mevcut sıkıştırılmış haldeki bir eğitsel içerik paketinin açılıp değişiklik yapılmasından daha mantıklı ve pratiktir. Yayımcı, bir proje ile ilgili sürümler bağlantısı tıklandığında o proje altında oluşturulmuş tüm sürümleri “adı, açıklaması, sürüm numarası, eklenme tarihi ve organizasyon bilgileri”ni içeren bir tabloda sıralanmaktadır. Burada sürümün silinmesi ya da güncelleme yapılması mümkündür. Eğer kullanıcı yaptığı değişikliği yeni bir sürüm numarası ile kaydetme yolunu seçmez ise yapılan güncellemeler önceden

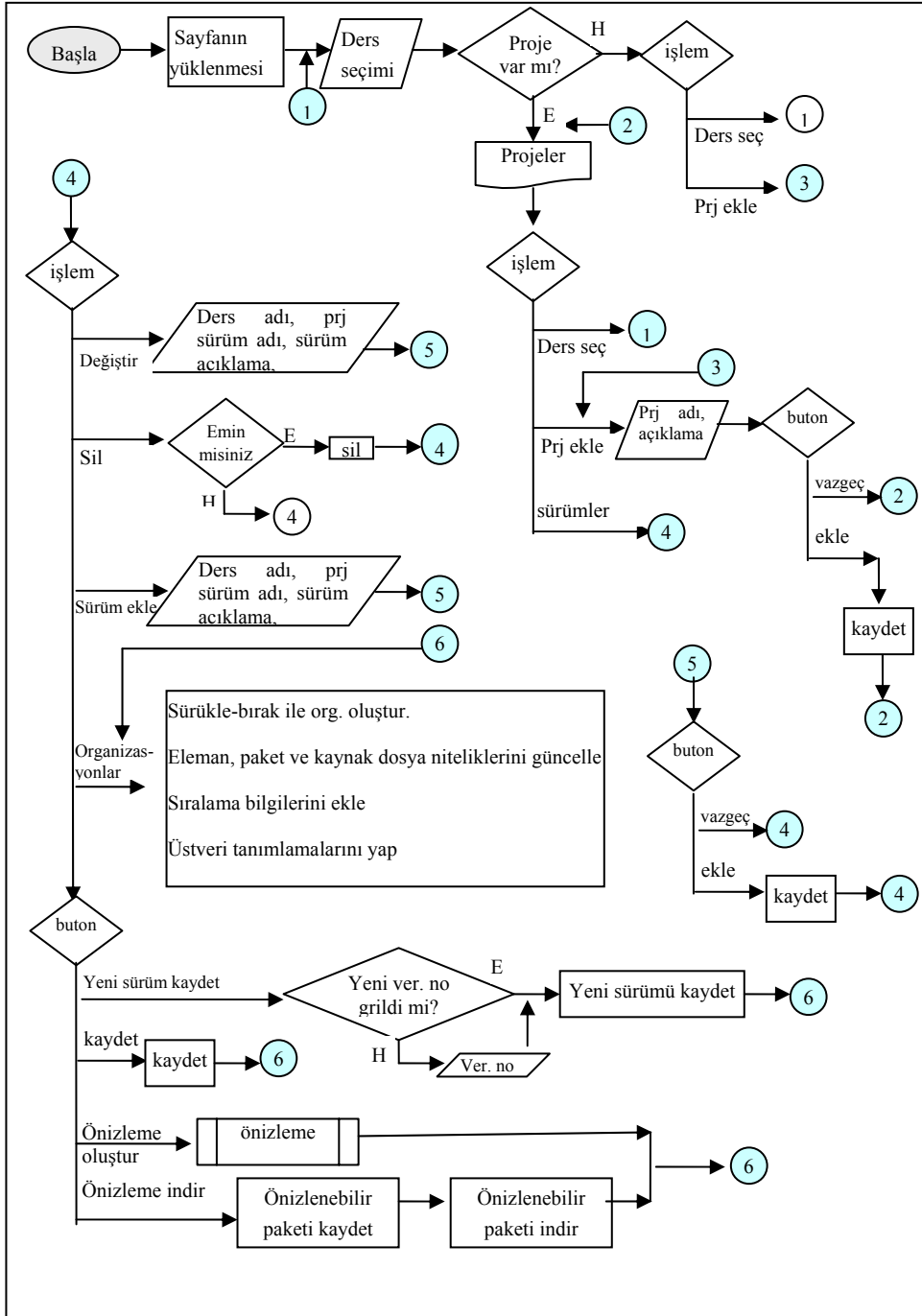
kaydedilmiş olan proje adı üzerine yazılır. Projeyi paket halinde kaydetmek için “Paketi Kaydet” butonu kullanılmaktadır. Hazırlanan içerik, ilgili ders altında “projeadı_versiyonadı_versiyonnumarası” şeklinde bir isim verilerek sıkıştırılır ve paket olarak kaydedilir. Şekil 5.10’da proje yöneticisi modülü genel akış diagramı verilmiştir.

İçerik paketleme ile dersi oluşturan eğitim kaynaklarının, dersin bir LMS’e taşınabilmesi için bir zip dosyasında sıkıştırılması gerekmektedir. LCDM platformunda, SCORM 2004 spesifikasyonuna göre, içerik paketleri PIF (Package Interchange File) formatında oluşturulmaktadır. PIF dosyası, imsmanifest.xml dosyası, tüm kontrol dosyaları, schema dosyalarını ve içeriğe refere edilen kaynakları içerir. Eğer içerik paketinin gösteriminde PIF kullanılacaksa SCORM, RFC1951’e uyumlu bir PIF gerektirir. Arşiv formatı ise PKZip v2.04g olmalıdır. SCORM İçerik Paketleme, *IMS İçerik Paketleme Belirtimine (IMS Content Packaging Specification)* bağlı kalınarak geliştirilmiştir ancak bazı ek kılavuzlar içermektedir. Üstveri etiket organizasyonu ve içerik paketlemede SCORM dışında kullanılan diğer bazı standartlar ise EAD (Encoded Archival Description), TEI (Text Encoding Initiative), MARC (Machine –Readable Cataloguing), Dublin Core, IMS (Instructional Management Systems) standartlarıdır (Balcı ve İnceoğlu, 2005).

İçerik paketleme, ayrıca, bir öğrenme deneyiminin nasıl sunulacağını yani kaynakların hangi sırada kullanılacağını belirten İçerik Yapısı’nı (*Content Structure*) da tanımlayabilir.

5.2.6 Metadeta modülü

LCDM platformunda, üstveri tanımlamalarını sadece yayımcı grubundaki kullanıcılar yapabilirler. Üstveri tanımlamaları ayrı bir modül olarak tasarlanmıştır, fakat projenin oluşturulması aşamasında yapılabileceğinden, Proje Yöneticisi modülü içerisinde çağırılmaktadır.



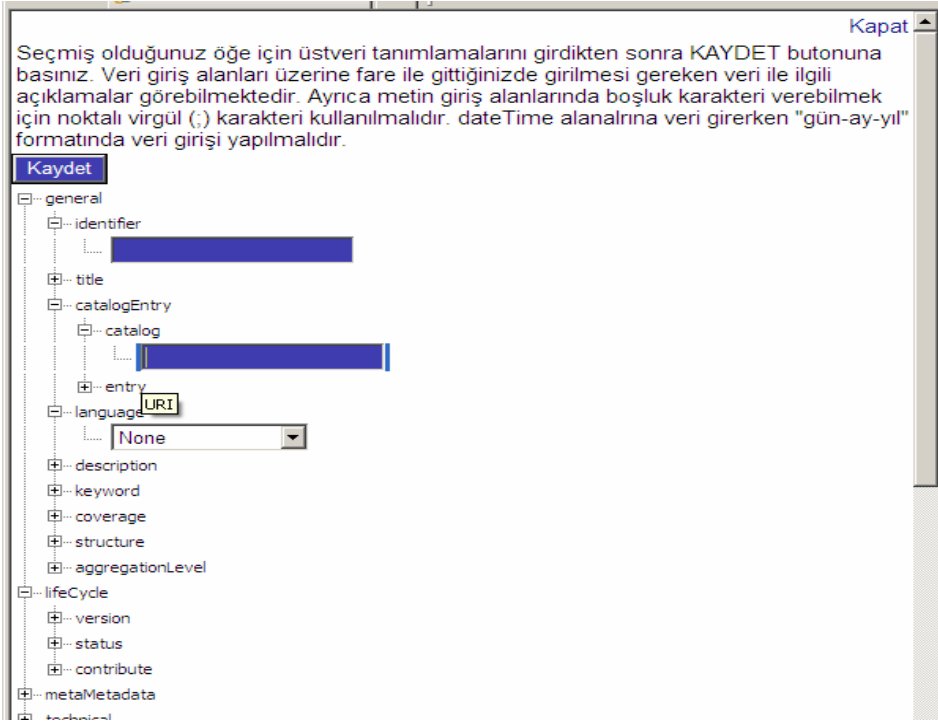
Şekil 5.10 LCDM platformu “proje yöneticisi” modülü genel akış diagramı.

LCDM sisteminde “metadata modülü” sayesinde SCORM 2004’e uygun olarak, eğitsel içerik oluştururken kaynak dosya, içerik organizasyonundaki her bir eleman, organizasyon ve proje bazında olmak üzere dört farklı seviyede üstveri tanımlaması yapılabilmektedir. Üst-veri bilgileri <metadata>...</metadata> etiketleri arasına gömülmektedir. Ancak, Şekil 3.4’te önceden görüldüğü üzere, manifest dosyası <manifest> elemanı altında tanımlı ve içerik kümesinin bir bütün olarak üstverisini veren <metadata>...</metadata> bloğu, ÖN üst-verisi ile karıştırılmamalıdır. Bu, içerik paketini bir bütün olarak tanımlayan üst-veri bilginin tanımlandığı bloktur.

SCORM’a göre üstveri bilgilerinin tanımlanması, biri “imsmanifest.xml” dosyasında ilgili kısımlara tanımlanan bilgilerin yerleştirilmesi, diğeri ise bağımsız “xml” dosyaları şeklinde kaydedilip manifest dosyasında ilgili yerlerden referans verilmek şeklinde tanımlanabilmektedir. LCDM platformunda ise hem karmaşıklığı önlemek hem de dosya ve paket havuzu olmak üzere iki ayrı havuz elde ederek dosya havuzunda da arama yapabilmek amacıyla, üstveri tanımlamalarının kaydedilmesinde farklı bir kriter belirlenmiştir. Paket üstverisi, organizasyon ve organizasyon altındaki item’lar için tanımlanan açıklayıcı bilgiler içerik paketinin imsmanifest.xml dosyasında ilgili etiketlerin arasına gömülmektedir. Fakat kaynak dosyalara ait üstveri tanımlamaları dosya havuzunda arama yapılabilmesini sağlayabilmek amacıyla, kaynakdosyaadı.xml şeklinde isimlendirilerek bağımsız birer xml dosyası halinde kaydedilmektedirler. Ayrıca, üstveri tanımlaması yapılan kaynak dosya başka bir yerde tekrar kullanılırsa üstveri tanımlaması kendisiyle birlikte çağırılmaktadır.

LCDM platformunda üstveri bilgilerinin kolay girilebilmesini sağlamak için, form yapısında bir arayüz değil, 9 kategori ve altındaki 69 üstveri elemanının tek bir ekranda görülebilmesini sağlayan ağaç

yapısında bir arayüz hazırlanmıştır. Bu arayüz Şekil 5.11’de görülmektedir. Üstveri tanımlamalarının girişi için kullanılan ağaç yapı, “RadTreeView.Net2.dll”i kullanılarak sağlanmıştır. Bu ağaç yapının kullanıldığı her yerde, adı geçen “RadTreeView.Net2.dll”i kullanılmaktadır. Üstveri bilgileri girildikten ya da güncellemeler yapıldıktan sonra, Kaydet komutu verilmezse, girilen bilgiler saklanmayacaktır. Kaydetme işleminden sonra, organizasyon oluşturulan ekrana dönüldüğünde, projenin paket olarak kayıt edilmesi işlemi sırasında tüm üstveri bilgileri, sırası önemli olmadan, tek tek kaydedilmektedir.



Şekil 5.11 LCDM platformunda üstveri girişi ekranı

5.2.7 Sequencing modülü

LCDM platformunda geliştirilen içerik paketleri, SCORM 2004 uyumlu diğer yazarlık araçları ve LMS'ler üzerine taşınabilir ve kullanılabilir şekilde tasarlandığından, LMS üzerinde kullanılmaları durumunda dikkate alınacak olan sıralama kurallarının da ilgili adımlarda paketi hazırlayan kullanıcı tarafından tanımlanması sağlanmıştır. Sıralama kurallarının tanımlanması bir modül altında gerçekleştirilmektedir. Fakat, sıralama bilgileri içerik organizasyonunun oluşturulması aşamasında tanımlanmalıdır ve bu bilgiler sadece oluşturulan organizasyon ve bu organizasyonlar altındaki birinci seviye öğeler (item) bazında tanımlanabileceğinden, bu modül proje yönetim modülü içerisinde ve ağaç yapıda ilgili yerde sağ tuş tıklandığında gelen menüden çağırılmaktadır. Sıralama bilgilerinin tanımlanmasında, bölüm 3.2.3'te anlatılan kurallara göre tasarım yapılmıştır ve bu amaçla kullanılan arayüzlerden biri Şekil 5.12'de görülmektedir.

Kontrol Modu	Açıklama	Var
Choice	"Bu aktivitenin çocuğunu hedef alan seçim sıralamasına izin ver"	<input type="checkbox"/>
Choice Exit	"Bu aktivitenin aktif olan çocuğuna, eğer Choice sıralama talebi işleme alınırsa sonlandırma hakkı ver"	<input type="checkbox"/>
Flow	"Bu aktivitenin çocuğuna Flow alt işlemlerinin uygulanmasına izin ver"	<input type="checkbox"/>
Forward Only	"Bu aktivitenin çocuğu için geriye dönük hedeflere izin verilmes"	<input type="checkbox"/>
Use Current Attempt Objective Info	"Bu aktivitenin çocuğu için objective progress bilgisi, eğer bu bilgi aktivitenin o anki girişiminde kaydedildi ise sadece kural değerlendirmeleri ve rollup sırasında kullanılacaktır"	<input type="checkbox"/>
Use Current Attempt Progress Info	"Bu Aktivitenin çocuğuiçin attempt progress bilgisi, eğer bu bilgi aktivitenin o anki girişiminde kaydedildiyse sadece kural değerlendirmeleri ve rollup sırasında kullanılacaktır"	<input type="checkbox"/>

Kaydet

Şekil 5.12 Sıralama bilgilerinin tanımlanması için bir ekran görüntüsü.

Arayüzde sıralama kurallarına bağlı kalınarak, veri girişi toplam 9 sekme altında organize edilmiştir. Bu sekmeli yapı için, “RadTabStrip.Net2.dll ” kütüphanesi kullanılmıştır. Kullanılan “Rad” kontrollerine “ajax” desteği kazandırmak için ise “RadAjax.Net2.dll” kütüphanesi kullanılmıştır. Ayrıca, bu tanımlamalar organizasyon oluşturulurken yapıldığından burada ayrıca akış diagramı verilmemiştir.

5.2.8 Paket yöneticisi modülü

Oluşturulan projeler, son halini aldıktan sonra sıkıştırılarak zip formatında bu modülde saklanırlar. Her projenin hatta her sürümünün fiziksel olarak saklanan bir son hali bulunur. Bu zip dosyaları orjinal isimleriyle PaketDeposu klasöründe saklanırlar. Bu anlamda, bu modül aslında bir “eğitsel içerik paketi havuzu” görevi görmektedir.

Yayımcı, yönetici ve istemci grubundaki kullanıcıların hepsinin bu modüle erişim hakkı vardır. Şekil 5.13’de de görüldüğü gibi, yayımcı sisteme giriş yaptığında paket yöneticisi modülünde, kendi oluşturduğu paketlerin listesini, paket hakkında kısa açıklama, sisteme eklenme tarihi bilgileri ve paketi kendi diskine indirebileceği, paketin önzilemesini LCDM platformu üzerinden görebileceği, paketin önizleme formatını kendi diskine indirebileceği ve paketi silebileceği bağlantıları görebilir. Ayrıca bu modül içerisinde “arama” modülünü de çağırılarak, ders bazında ve farklı kriterlere göre eğitsel içerik paketleri aranabilmektedir.

Diğer taraftan, yönetici grubundan bir kullanıcı sisteme dahil olduğunda, Paket Yöneticisi modülünde, sistemde tanımlı yayımcıları bir aşağı açılan listeden seçerek ve o kullanıcının verdiği dersleri de başka bir aşağı açılan listeden seçerek, hangi yayımcının hangi ders altında ne zaman hangi paketleri oluşturduğu gibi bilgilere erişebilmektedir .

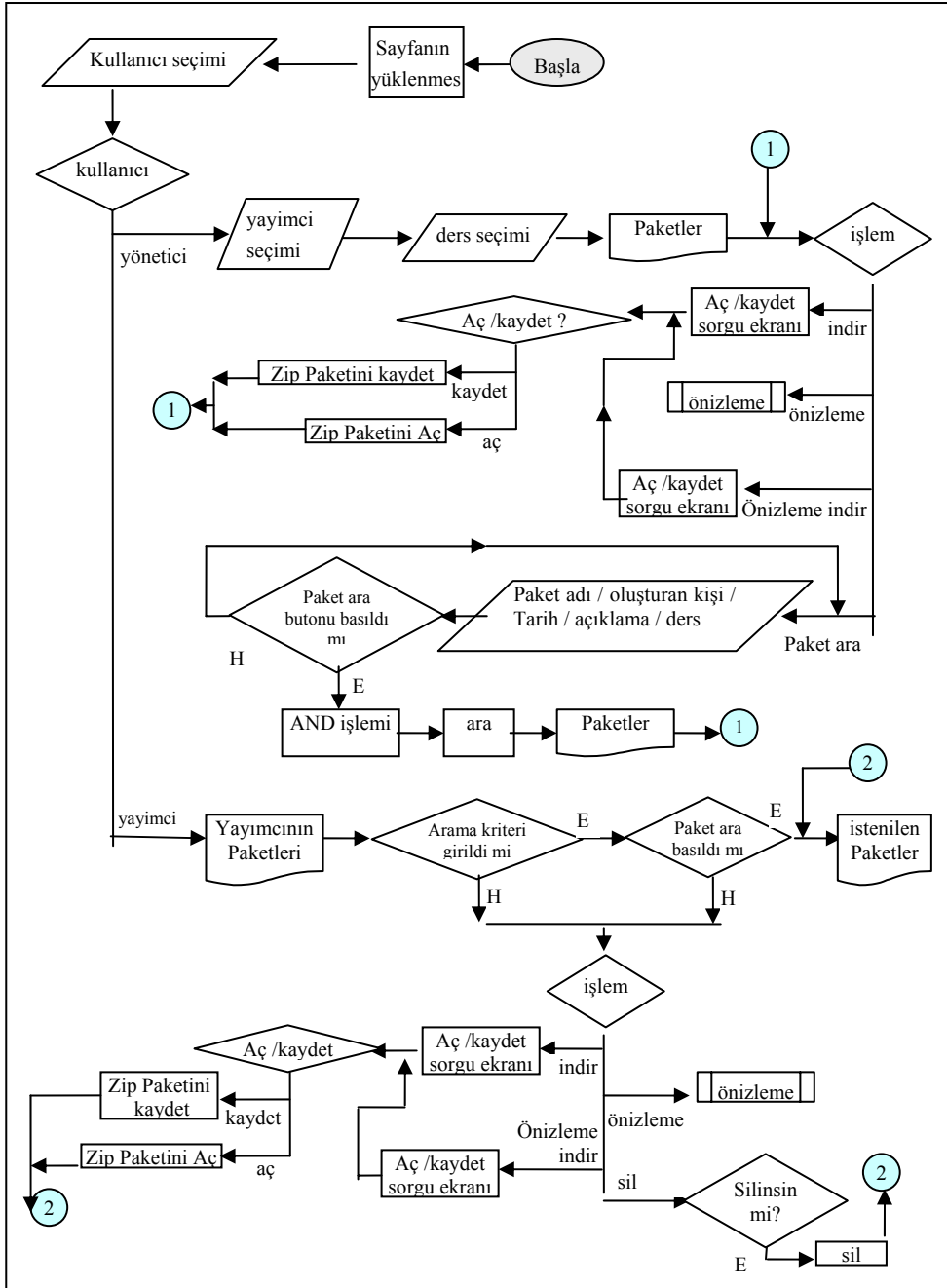
Paket Adı	Paket Açıklama	Paket Ekleme Trh.	Ekleyen
ReloadanUpload_test1_1		07.09.2007 10:24:43	bBozkurt İndir Önzilebilir formatını indir Sil Önzileme
ReloadanUpload_test1_2		07.09.2007 10:34:21	bBozkurt İndir Önzilebilir formatını indir Sil Önzileme
Proje1_İşletme kavramı_1		19.09.2007 15:00:05	bBozkurt İndir Önzilebilir formatını indir Sil Önzileme

Şekil 5.13 LCDM Paket Yöneticisi ekranı: Yayımcı girişi.

İstemci grubundan bir kullanıcı sisteme girdiği zaman ise, paket yöneticisi modülü altında, sistemde aldığı dersleri seçebileceği bir aşağı açılan liste görecektir. Buradan ders seçimi yaptığı zaman bir tablo halinde o ders altında sistemde kayıtlı kullanıcıların hazırlamış oldukları içerik paketlerini görebilir. Öğrencilerin bu tabloda içerik paketini silme hakkı yoktur. İstemciler sadece içerik paketinin önzilemesini görebirler ve kendi makinelerine kaydedebilirler.

LCDM platformunda, dışarıda başka araçlar kullanılarak hazırlanmış içerik paketlerinin yüklenmesi ve çalışır hale getirilmesi sağlandığından, paket yöneticisi modülünde bu paketler de listelenmektedir. Şekil 5.14'te ise modülün işleyişi akış diagramı şeklinde gösterilmiştir.

Paket yöneticisi modülünde paket arama işlemi ile ilgili detaylı bilgi, arama modülünde verilmiştir.



Şekil 5.14 "Paket yöneticisi" modülü genel akış diagramı.

5.2.9 Arama modülü

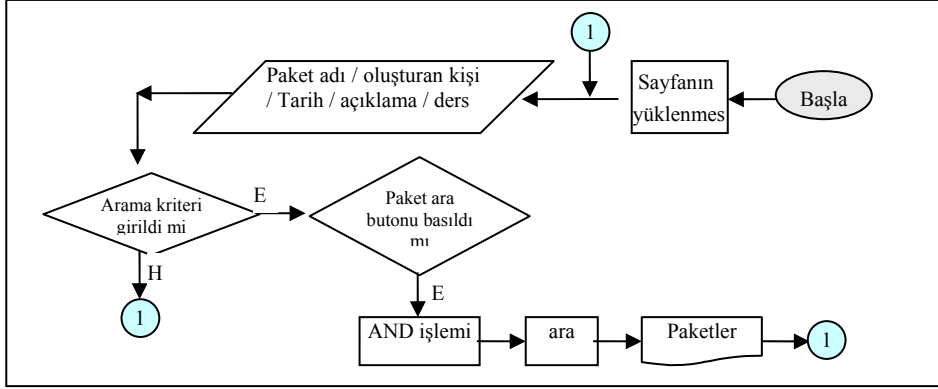
Arama modülü hem DosyaYöneticisi modülü içinde hem de Paket yöneticisi modülü içinde dosya ve içerik paketi arama amaçlı olarak kullanılabilir. Arama işleminin nasıl gerçekleştirildiği, “dosya yöneticisi” ve “paket yöneticisi” modüllerinde akış diagramları üzerinde gösterilmiştir. Dosya yöneticisi modülünde dosya arama işlemi sırasında, dosya adı, dosya tipi, dosyayı sisteme yükleyen yayımcı, dosyanın platforma eklenme tarihi gibi kriterlere göre arama işlemi yapılabilir.

Paket Yöneticisi modülü altında çalıştırılan arama modülünde ise, paket adı, paketi oluşturan yayımcı yani eğitmen, paketin oluşturulma tarihi, paketi ifade eden bir açıklama metni gibi kriterlere göre paket arama işlemi gerçekleştirilmektedir. Yönetici ve yayımcı grubundan bir kullanıcı sisteme dahil olmuş ise bu modülde bir de derslere göre arama işlemi sağlanabilir.

Arama işlemleri veri tabanındaki kayıtlar üzerinden yapılmaktadır. Özellikle paket arama işleminin platformda saklanan sıkıştırılmış formattaki içerik paketleri üzerinden yapılması, paketlerin açılarak eski haline getirilmesi ve sıkıştırılmış haldeki içerikten arama işleminin gerçekleştirilmesini gerektireceğinden, arama işlemi veri tabanındaki kayıtlı proje bilgileri üzerinden daha hızlı bir şekilde yapılmaktadır.

Gerek dosya ve gerekse paket arama işlemi sırasında kullanılan kriterler temel bilgileri içeren kriterlerdir. Dosya/Paket Adı, Dosya/Paket Tipi (MIME Type), Oluşturan Kişi ve Oluşturulma Tarihi kriterlerine göre arama yapılabilmesi mümkündür. Oluşturulma tarihine göre arama yapılabilmesi için iki tarih girilmesi gerekmektedir. Tarih kontrolü işlemi için “RadCalendar.Net2.dll” kütüphanesi kullanılmıştır. Birden fazla arama kriteri verilirse, bunlar “VE” (AND) işlemine alınarak arama

işlemi gerçekleştirilir. Arama işlemi, üstveri etiketlerinde gerekli görülen alanlara göre genişletilebilir. Şekil 5.15'te paket arama işlemine ait akış diagramı gösterilmiştir.



Şekil 5.15 Paket arama işlemi akış diagramı.

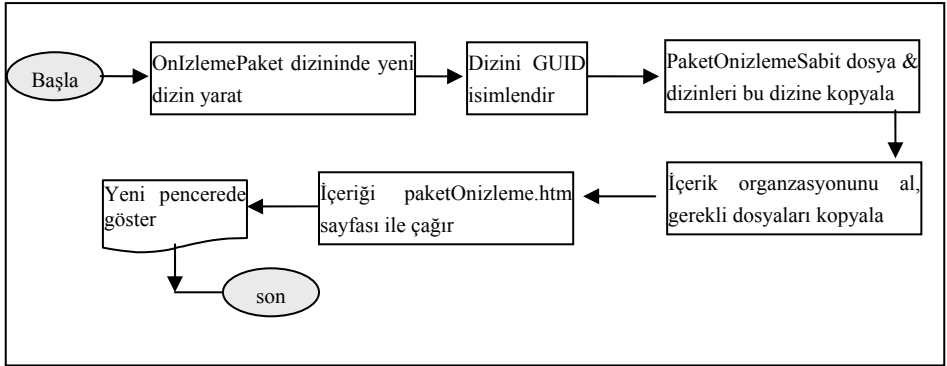
5.2.10 Önizleme modülü

LCDM platformunda, “Dosya Yönetici” modülünde sisteme yüklenen dosyalar bazında ve hazırlanan içerik paketleri bazında olmak üzere iki ayrı şekilde önizleme sağlanabilmektedir.

Dosya yöneticisi modülünde listelenen dosyaların üzerine tıklanması, ilgili dosyanın yeni bir tarayıcı penceresinde açılmasını sağlamaktadır. Uygulama dosyaları seçilirse, sistem kullanıcıdan dosyayı kaydetmek ya da açmak için onay ister. Uygulama dosyaları dışındaki dosyaların kullanıcının yerel diskinde kayıt edilebilmesi için, yeni tarayıcı penceresinde dosya görüntüledikten sonra “Dosya” menüsünden “Farklı kaydet” işlemi yapılmalıdır.

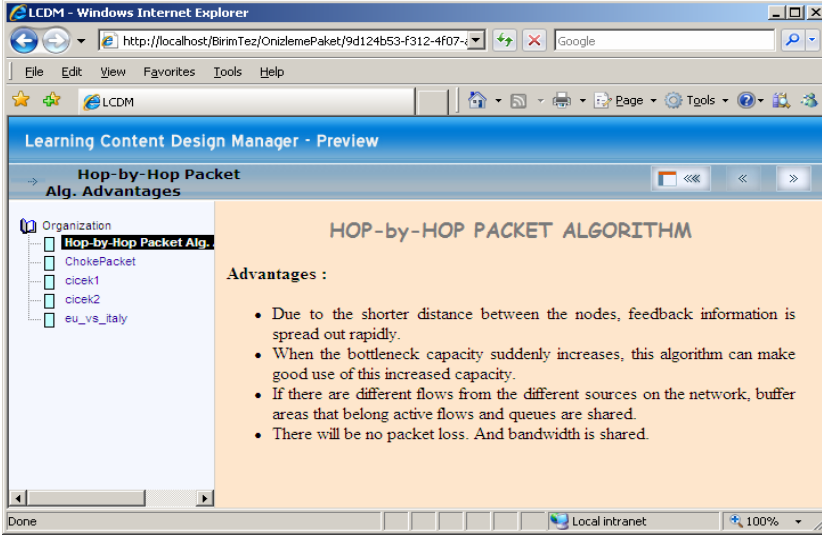
Paket yöneticisi modülündeki, “paket önizleme” ise daha karmaşıktır. Önizleme işlevini başlatacak olan bir “htm” dosyası şablon dosya olarak tutulmaktadır. Bu dosya içinde kullanılan sabit dosyalar da

aynı lokasyonda bir dizin altında organize edilmişlerdir. Bu sabit dosyalar, içeriğin görüntüleneceği arayüz, alanlar, ağaç yapısındaki menude kullanılan resimler, “script” dosyaları ve “css” dosyalarıdır. Bir paketin önizlemesi görüntülenmek istendiğinde, ilgili yerde, GUID ile isimlendirilerek yeni bir klasör yaratılır. Bu yeni klasör içerisine PaketOnIzlemeSabitDosyaları dizini ve PaketOnIzleme.htm başlangıç sayfası ve ayrıca izlenmek istenen pakette kullanılan ilgili dosyalar kopyalanır. Ve içerik paketOnizleme.htm sayfasından çağırılarak görüntülenir. Bu sayede proje içeriğinin LMS’e gerek kalmadan kullanıcının bilgisayarında yüklü olan Internet tarayıcı üzerinden görüntülenmesi sağlanmış olur. Şekil 5.16’da paket önizleme işlemi basamakları görülmektedir.



Şekil 5.16 LCDM platformu “paket önizleme” işlemi temel akış diagramı.

Şekil 5.17’de ise, paket önizleme sayfası üç pencereden oluştuğu görülmektedir. Üst pencerede sistem adı, paket adı bilgileri ve ileri-geri ilerleme butonları; sol pencerede, öğrencinin göreceği ve “içindekiler” tablosuna benzeyen eğitsel içeriğin nasıl sıralandığını gösteren bir ağaç yapı bulunur. Bu ağaç yapı üzerinde ilgili başlıklar tıkladığında ya da ok tuşları ile ilerlendiğinde sağ pencerede içerik görüntülenmektedir.



Şekil 5.17 LCDM Paket önizleme.

Ayrıca yayımcı grubundan bir kullanıcının eğitsel içeriği hazırlarken de önizleme yapması mümkündür. Bu, Proje Yöneticisi modülünde sağlanmaktadır. Eğitmen hazırladığı içerik yapısını paket olarak kaydetmeden de, kullanıcı arayüzündeki ilgili buton sayesinde her aşamada içerik önizleme yapabilir ve içeriğini buna göre şekillendirebilir.

İstemci ve yönetici grubundaki kullanıcılar da paket arama işlemi sonucunda görüntülenen tablo üzerinden ilgili bağlantıları tıklayarak paket önizlemesini görebilirler.

5.2.11 Hazır paket yükleme modülü

LCDM platformuna, başka uygulamalar / platformlar kullanılarak hazırlanan, SCORM 2004 uyumlu içerik paketlerini yüklenebilir. Bir yayımcı başka bir ortamda hazırlanan paketini sisteme yüklediğinde, eklenen paket kendisine ait içerik paketleri arasında görülmektedir. Bir paket yükleme işleminin özünde, bir projenin yüklenmesi değil, LCDM platformunda tanımlı olan bir projenin altına ilgili sürümlerinin

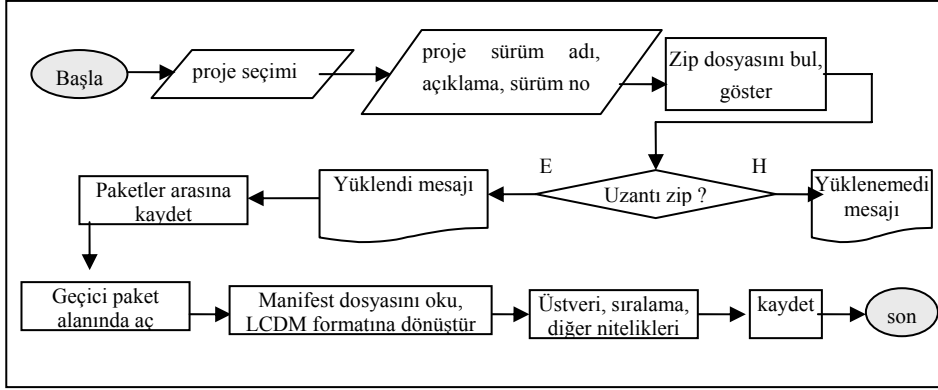
yüklenmesi mantığı vardır. Paket yüklenmek istenildiğinde, Şekil 5.18’de da görüldüğü üzere, kullanıcıdan paketin hangi proje kapsamına dahil edileceği bilgisi, kayıt altına alınacağı sürüm adı, sürüm ile ilgili bir açıklama ve sürüm numarası bilgileri istenir. Eğer ilgili paketi kapsayabilecek bir proje yoksa, paket yükleme işleminden önce, istenilen ders altında bir proje yaratılması gerekmektedir. “Zip” dosyası olarak yüklenen paketler, açılarak veritabanındaki tablolara işlenir.

Şekil 5.18 LCDM Hazır paket yükleme.

Sisteme hazır bir paket yüklendikten sonra, o paket bilgileri ve ilgili organizasyon üzerinde değişiklik yapılarak farklı sürüm numarası altında kaydedilmesi mümkündür. Bu sayede yeniden kullanılabilirlik artar ve eğitimlere kullanım kolaylığı sunulur.

ReLOAD uygulamasıyla oluşturulan içerik paketlerinin sisteme yüklenmesi ve çalışır hale getirilmesi testleri yapılmıştır. Platforma hazır

paket yüklenmesi için gerekli adımlar Şekil 5.19'daki akış diagramında görülmektedir.



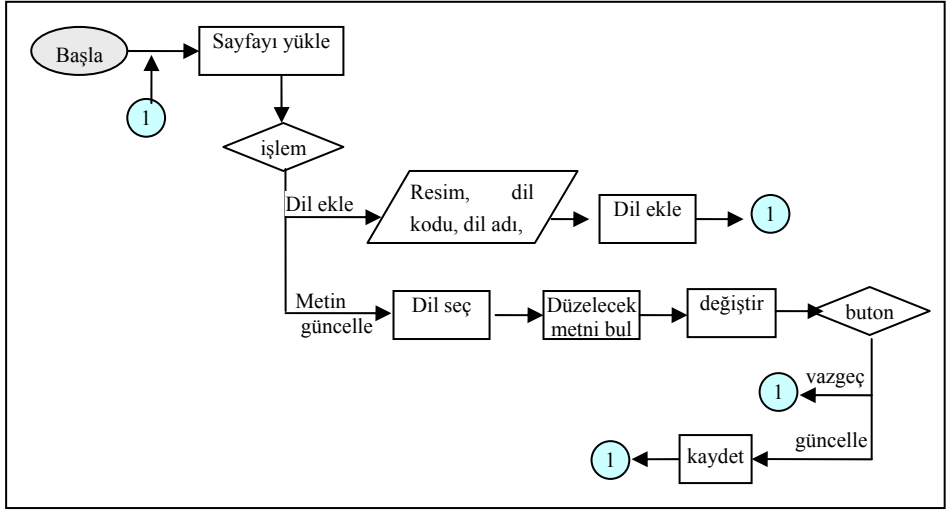
Şekil 5.19 “Hazır paket yükleme” modülü temel akış diagramı.

5.2.12 Dil yöneticisi modülü

Bu modül yönetici grubundaki kullanıcıların erişimine açık bir modüldür. Oluşturulan arayüz sayesinde, platforma yeni bir dil ekleme ve ekranlardaki birden fazla dil için kodlanan açıklamaların bu dilde tanımlamalarının yapılması işlemlerini yönetici grubundaki kullanıcılar, veri tabanını açıp kayıt girmeden direk arayüz üzerinden yapabilmektedirler. Platform üzerindeki ekranlardaki tüm açıklama metinleri ve düğmeler birer kod ile isimlendirilmektedir ve bu kodların yine sistemde tanımlı olan dillerdeki karşılıkları veritabanında tutulmaktadır. Dil yönetim modülünde, seçilen dil ile ilgili tanımlamaların yapılabilmesi için sistemde Türkçe karşılığı girilmiş olan tüm kodlar listelenir. Seçilen dildeki karşılıklarının girilmesi ya da metin güncelleme işlemlerinin yapılmasıyla işlem onaylanır ve arka planda veri tabanının güncellenmesi işlemi gerçekleştirilir.

Bu modül tıklandığında, arayüz üzerinde, platformda tanımlı diller görülebilmektedir. Dil seçimi yapıldığında, bir tablo üzerinde tüm dil

tanımlamaları için, kod, açıklama metni, seçilen dildeki karşılığı ve “düzenle” bağlantısı görüntülenir. Bu bağlantı tıklandıktan sonra, ilgili kodun ekranda görüntülenecek olan dil karşılığı güncellenebilmektedir. Platformda yeni bir dil tanımlaması yapılmak istenirse, tanımlanacak dil kodu, dil adı bilgileri girilir ve o dil ile ilişkilendirilecek olan resim bulunarak seçilir ve dil ekleme işlemi tamamlanmış olur. Bu işlemler genel olarak, Şekil 5.20’deki akışdiagramında gösterilmiştir.



Şekil 5.20 “Dil yöneticisi” modülünün işleyişini gösteren akış diagramı

6. LCDM İLE BENZER İŞLEVLERİ OLAN DİĞER UYGULAMALARIN KARŞILAŞTIRILMASI

Burada, üçüncü bölümde dayandırıldığı standartlar hakkında bilgi verilen, detaylı olarak işlevlerinin açıklandığı, bu tez çalışması kapsamında tasarlanan LCDM platformu ve benzer işlevleri yerine getiren diğer uygulamalar karşılaştırılmakta ve sonuçlar sunulmaktadır.

Önceki çalışmada (Balcı ve İnceoğlu, 2005), eğitsel içerik, öğrenme nesnelere şeklinde ve SCORM uyumlu olarak hazırlanmıştır. Bunu takiben, tasarlanan ÖN içerik paketi, Ege Üniversitesi Mühendislik fakültesi dördüncü sınıf öğrencileri üzerinde denenerek, sistemi kullanan öğrencilerden öğrenme nesnelere kullanım kolaylığı, konunun anlaşılmasında faydalı olduğu, ilgi çekici olduğu ve öğrencilerin ÖN'lerini kullanmaya devam etmek istedikleri konularında olumlu geribildirimler alınmıştır (Balcı ve İnceoğlu, 2006b).

Doktora tezi kapsamında yapılan bu çalışmada ise LCDM, Learning Content Design Manager, adı verilen ve eğitsel içerik hazırlama ve üstveri tanımlamalarının yapılmasını sağlayan, SCORM'un son sürümü olan SCORM 2004 standardının üçüncü sürümünü destekleyen ve bu sayede farklı LMS ve yazarlık araçları arasında paylaşılabilen içerikler elde edilmesini sağlayan bir platform tasarlanmıştır. Bu platformun, birinci bölümde incelenen LMS, yazarlık araçları ve üstveri editörleri ile benzer ve farklı yönleri aşağıda ele alınmıştır. İlk olarak LCDM platformu ve incelenen LMS'ler karşılaştırılmıştır. Sonuçlar Çizelge 6.1 üzerinde görülmektedir. Çizelgede veri içermeyen hücreleri tanımlayan özellikler ya ilişkili platform tarafından desteklenmemektedir, ya da incelenen kaynaklarda ilgili konuda bilgiye rastlanmamıştır.

1. LCDM ile Moodle (Moodle, 2007) eğitim yönetim sistemini karşılaştırıldığında, her ikisi de nesne tabanlı modüller

uygulamalardır. Fakat Moodle Linux üzerinde Apache, PHP ve MySQL kullanılarak geliştirilmişken, LCDM Microsoft Windows üzerinde SQL server ve C# programlama dili kullanılarak geliştirilmiştir. Moodle, zaman içinde geliştirilerek Türkçe de dahil olmak üzere 60 dil sürümüne sahip olmuştur. Diğer taraftan LCDM multilanguage olarak tasarlanmıştır ve şu anda Türkçe ve İngilizce olmak üzere iki dili desteklemektedir. Moodle sisteminde site yönetiminden sorumlu, kurs oluşturmak ve bunlara öğretmen atamak gibi işlemleri gerçekleştiren *administrator* ve kurs oluşturmak ve öğretmekle yükümlü *course creator* olmak üzere 2 rol vardır. LCDM platformunda ise dosya, proje, paket yönetimi, üstveri, sıralama, hazır paket yükleme işlemlerinden sorumlu aynı zamanda önizleme ve arama işlemlerini de yerine getirebilen *yayımcı*; grup, kullanıcı, ders ve dil yönetiminden sorumlu ve buna ek olarak dosya ve paket yönetimi modüllerine erişerek denetim yapma hakkına sahip yönetici; son olarak ise önizleme ve arama hakkına sahip olan istemci olmak üzere 3 kullanıcı grubu vardır. Ayrıca LCDM’de farklı kullanıcı grupları da oluşturulabilmektedir. Moodle’da bir seferde en fazla 2MB büyüklüğüne tek bir dosya yüklenebilirken, LCDM platformunda en fazla 50MB büyüklüğünde tek bir dosya yüklenebilmektedir. Ayrıca, LCDM platformunda paket bazında sürüm oluşturulabilmektedir.

2. dotLRN (dotLRN, 2007) ile LCDM karşılaştırıldığında, dotLRN’nin Unix ve Linux işletim sistemlerinde çalıştığı; bir LMS olmayan LCDM’in ise Microsoft Windows üzerinde SQL server ve C# programlama dili kullanılarak geliştirildiği görülmektedir. Hem dotLRN’de hem de LCDM’de kurs ekleme, kurs üstveri girişi,

Çizelge 6.1 LCDM ile incelenen LMS'lerin karşılaştırılması.

	LCDM	Moodle	dotLRN	Claroline	Learnwise	Teknical's Virtual C.	Atutor	WebCT
Açık k. kodu		✓	✓	✓			✓	
Prog. dili	C#	PHP		PHP			PHP	
Veri tabanı	SQL	MySQL		MySQL	SQL		MySQL	
İşletim sistemi	Windows	Windows, Mac, Linux	Unix, Linux		Win NT /2000			
Dil desteği	Türkçe,İngilizce (multilanguage)	60 dil		28 dil				
Kullanıcılar	Yönetici, yayımcı, istemci (eklenebilir)	Yönetici, kurs yaratıcısı	Yönetici, eğitmen, öğrenci	Yönetici, öğretmen, öğrenci				
Yüklenen tek dosya boyutu	50 MB	2MB		10MB			1MB	

Çizelge 6.1 LCDM ile incelenen LMS'lerin karşılaştırılması (devam).

	LCDM	Moodle	dotLRN	Claroline	Learnwise	Teknical's Virtual C.	Atutor	WebCT
Üstveri	IEEE LOM		✓		Ims, IEEE LOM	IMS, SCORM		
LO hazırlama (kurs oluşturma)	✓	✓	✓		Çevrim dışı olarak	✓	✓	
Sıralama bilgisi	✓							
İç. paketleme	✓		✓	✓	✓	✓	✓	
Hazır paket yükleme	✓		✓		✓			✓
Dosya havuzu	✓							
Dosya paylaşımı	✓							
İçerik paketi havuzu	✓	✓	✓			✓	✓ çevrimdışı erişim	

Çizelge 6.1 LCDM ile incelenen LMS'lerin karşılaştırılması (devam).

	LCDM	Moodle	dotLRN	Claroline	Learnwise	Teknical's Virtual C.	Atutor	WebCT
Önizleme	✓		✓		✓	✓		
Arama	Dosya, paket	✓	nesne					
Versiyonlama	Paket		dosya					
Ders yönetimi	✓							
Kullanıcı yönetimi	✓							
Grup yönetimi	✓							
Dil yönetimi	✓							

klasörlere öge ekleme, ögelere üstveri ekleme, ögelere kaynaklar ekleme, öğrenme nesnelerini araştırma, kursu bir zip dosyasına çevirmek gibi içerik hazırlamak ile ilgili işlemlerle birlikte ÖN havuzundan arama yapmak ve önizleme yapabilmek mümkündür. Ancak LCDM'in önizleme fonksiyonu tamamen internet tarayıcı üzerinde çalışmaktadır. Ayrıca LDCM'de kaynak dosya bazında üstveri girişi, içerik paketlerini SCORM 2004 standardına göre hazırlanması, sıralama kurallarının eklenmesi, başka yazarlık araçları kullanılarak geliştirilen içerik paketlerinin platforma yüklenmesi gibi özellikler de vardır. dotLRN platformuna da sıkıştırılmış dosya olarak SCORM içerik paketleri eklemek mümkündür. Hem dotLRN'de hem de LCDM platformunda üç tip kullanıcı tanımlanmıştır. dotLRN platformuna yüklenen tüm dosyalar birer sürüm numarası aldığından eski sürümler saklanabilmektedir. LCDM platformunda ise sürüm oluşturma, içerik paketleri için geçerlidir. Böylece içerik paketi hazırlamak için harcanan zaman ve emekten tasarruf sağlanmış olunur.

3. CLAROLINE (CLAROLINE, 2007) ise, 28 dile çevrilmiş açık kaynak kodlu bir öğrenme ortamıdır. PHP ve MySQL kullanılarak tasarlanmıştır. Her iki platformda da yönetici, öğretmen ve öğrenci olarak 3 tip kullanıcı vardır. Claroline'da sisteme eklenebilecek dosya boyutu 10MB ile dosyaların toplam boyutu ise 30MB ile sınırlandırılmıştır. LCDM'de ise tek seferde en fazla 50MB dosya upload edilebilmektedir. Claroline'da LCDM gibi SCORM desteğine sahiptir. Ancak sistemde öğrenim nesnesi hazırlamak ile ilgili bir kısım bulunmamaktadır. Aynı bir araç kullanılarak SCORM içeriği yüklenebilmektedir. LCDM ise içerik oluşturma işlemini bağımsız bir modül ile gerçekleştirir.

4. Granada Learning tarafından oluşturulan “LEARNWISE” (LEARNWISE, 2005) da bir LMS’tir. IMS içerik paketleme, IMS Metadata/IEEE LOM desteği vardır. Çevrim dışı olarak hazırlanan içerik sonradan sisteme yüklenebilmektedir. LCDM ise bir LMS olarak nitelendirilemez. Daha çok içerik oluşturma ve paketleme işlevlerine yöneliktir. Ve eğitsel içerik organizasyonu çevrimiçi durumda iken hazırlanır.
5. Bir başka LMS, “Teknical's Virtual Campus” (Teknical's Virtual Campus, 2005), ile LCDM’i karşılaştırdığımızda her ikisinde de öğrenme nesneleri oluşturma ve üstveri tanımlamaları yapma işlevlerinin olduğu görülmektedir. Fakat Teknical's Virtual Campus’te içerik oluşturma işlemi Microsoft Word kelime işlemci programı ile yapılırken, LCDM’de farklı araçlarda hazırlanmış içerik sayfaları platforma yüklenerek, çevrimiçi durumda iken bir araya getirilmektedirler. Her ikisinde de nesne üstverileri birer havuzda tutulmaktadır.
6. Açık kaynak kodlu olan “ATUTOR” (ATUTOR, 2007) ise LCDM platformundan farklı olarak PHP ile ve MySQL veritabanı kullanılarak yazılmış başka bir eğitim yönetim sistemidir. ÖN havuzu vardır ancak çevrim dışı iken çalışmaktadır. Bu havuzdaki içerik paketleri yerel saklama alanlarına indirilip görülebilir ya da direk ATutor’a aktarılabilir. LCDM’de ise çevrim içi durumda nesne havuzuna erişilip paket arama ve indirme işlemleri yapılabilmektedir. Her iki sistem de IMS/SCORM içerik paketleme spesifikasyonlarına uyumludur. Atutor’da eğitsel içerikleri HTML ya da text formatı olarak yerel bir editörden yüklenebileceği gibi direk çevrimiçi olarak da girilebilir. LCDM’de ise farklı formattaki kaynak dosyalar sisteme yükledikten sonra oluşan paylaşılabilir havuzdan seçilen dosyalarla içerik organizasyonları oluşturulmaktadır. Atutor’da sisteme

yüklenebilecek dosya boyutu en fazla 1MB ile sınırlandırılmışken LCDM'de bu çok daha fazladır.

7. WebCT'de (WebCT, 2005) ise, IMS içeriğini alıp vermek mümkündür. İçerik yönetim ünitesi ReLOAD (ReLOAD, 2007), içerik paketleriyle çalışır. Bunlar manifest içinde üstveri elemanı içerirler. LCDM'de ise içerik paketleri kendi bünyesinde oluşturulabildiği gibi, başka yazarlık araçlarıyla hazırlanan içerik paketleri sistemde sorunsuz çalışmakta ve aynı zamanda LCDM platformunda hazırlananlar da uyumlu diğer yazarlık araçlarında sorunsuz çalışmaktadırlar.

LMS'ler ile yapılan karşılaştırmadan ayrı olarak, bağımsız olarak kullanılan yazarlık araçları ile de LCDM platformunu karşılaştırılabilir. Bunlar web tabanlı olmayan uygulamalar olduklarından kullanıcı, grup, ders yönetimi gibi işlevler içermemektedirler. Aşağıda karşılaştırmalar yapılarak sonuçlar açıklanmış ve Çizelge 6.2'de sonuçlar gösterilmiştir. Çizelgede veri içermeyen hücreleri tanımlayan özellikler ya ilişkili platform tarafından desteklenmemektedir, ya da incelenen kaynaklarda ilgili konuda bilgiye rastlanmamıştır.

1. LCDM platformu tasarlanırken ağırlıklı olarak başvuru örnek ReLOAD editor olmasına rağmen, LCDM, ReLOAD'a göre çok fazla sayıda artışı olan bir platform olarak karşımıza çıkmaktadır. Bir içerik paketi oluşturma ve üstveri editörü olan ReLOAD (ReLOAD, 2007) açık kaynak kodlu bir java uygulamasıdır. Diğer araçlar ile oluşturulmuş olan içeriğin paketleme, mevcut içeriğin tekrar amaçlandırılması, ve son kullanıcıya içeriğin önizleme ile dağıtılması fonksiyonlarını yerine getirmektedir. LCDM de aynı fonksiyonları yerine getirir. Ancak LCDM web tabanlı bir uygulamadır ve bu nedenle biraz daha kapsamlı olarak tasarlanmıştır.

Çizelge 6.2 LCDM ile içerik paketleme ve üstveri araçlarını karşılaştırmak.

	LCDM	RELOAD ver 2.5.4	LOAT	ELO	WeLOAD	IMS package editor
Açık k. kodu		✓		✓	✓	✓
Prog. dili	C#	Java	C++		PHP	Java
Veri tabanı	SQL		Access		MySQL	
İşletim sistemi	Windows	Windows, Macintosh OSX, Linux	Windows			
Dil desteği	Türkçe, İngilizce (multilanguage)	İngilizce	Çince	İngilizce	Almanca	İngilizce
Kullanıcı grupları	Yönetici, yayımcı, istemci (eklenebilir)		Sistem geliştirici, içerik sağlayıcı		Öğretmen, öğrenci	
Üstveri	IEEE LOM	IMS, IEEE LOM	✓	IMS		IMS
LO hazırlama	✓	✓	✓		✓	
Sıralama bilgisi	✓	✓				
İçerik paketleme	✓	✓	✓	✓	✓	✓

Çizelge 6.2 LCDM ile içerik paketleme ve üstveri araçlarının karşılaştırılması (devam).

	LCDM	RELOAD ver 2.5.4	LOAT	ELO	WeLOAD	IMS Package Editor
Farklı araçlarla hazırlanan paketleri yükleme /açma	Yüklenebilir	Açılabilir				
Dosya havuzu	✓		✓			
Dosya paylaşımı	✓					
İçerik paketi havuzu	✓		✓			✓
Önizleme	✓	✓	✓		✓	
Arama	Dosya, paket					
Versiyonlama	Paket					
Ders yönetimi	✓					
Kullanıcı yönetimi	✓					
Grup yönetimi	✓					
Dil yönetimi	✓					

Her iki uygulamada da dosya ve paket bazında önizleme mümkündür. ReLOAD Editor'de, dosyalar için de üstveri tanımlamaları yapılabilmektedir. Bu tanımlamalar ya bağımsız “.xml” dosyaları olarak kaydedilir ya da paket içindeki manifest dosyasına gömülür. LCDM'de ise kaynak dosya bir proje içinde kullanılıyorsa üstveri tanımlamaları girilebilir ve bunlar ayrı bir yerde tutulmaktadır. Aynı kaynak dosya farklı organizasyonlarda kullanıldığında, üstveri tanımlamaları ile birlikte alınır. Reload Editor, üstveri girişi için form/ ağaç/hem form hem ağaç yapı alternatiflerinin olduğu bir arayüz sunar. LCDM'de ise kullanımının kolay olması amacıyla üstveri girişini ağaç yapıda bir arayüz üzerinden sağlanmaktadır.

ReLOAD'da üstveri için IMS haricinde farklı profiller de seçilebilmektedir. LCDM'de ise, eğitsel içerik hazırlarken kullanılan profil olan ve SCORM 2004 standardının bir gereği olan IEEE LOM seçilmiştir. LCDM'deki gibi fakülte, bölüm ve ders bazında içerik oluşturmak ReLOAD'da mümkün değildir. Ayrıca LCDM platformunda paket bazında sürüm oluşturulurken ReLOAD'da yapılamaz.

LCDM birden fazla dil desteği olacak şekilde tasarlanmıştır ve Türkçe, İngilizce olmak üzere iki dili desteklemektedir ve bir dil yönetim arayüzü geliştirilmiştir. Oysa ReLOAD'da birden fazla dil desteği yoktur. LCDM web tabanlı çalıştığından, Grup, kullanıcı yönetiminden bahsedilebilir. Yine oluşturulan içerik paketlerinin aranmalarının daha kolay olması açısından paketler, sistemde tanımlı fakülte-bölüm-dersler kriterlerine bağlı kalınarak oluşturulmakta ve saklanmaktadır. Oysa ReLOAD'da böyle bir sınıflandırma söz konusu değildir.

2. LOAT (Liu, Huang, Chao, 2005) ise Çince tasarlanmış bir öğrenme nesnesi yazarlık aracıdır. LCDM gibi web tabanlıdır, modüler

yapıdadır ve Windows işletim sistemi üzerinde tasarlanmıştır. C++ ve Microsoft Access veritabanı kullanılmıştır. LCDM ise C# dili ile geliştirilmiş, ve Access'ten daha fonksiyonel ve güvenli olan SQL veritabanı kullanılmıştır. Yine LCDM platformundaki gibi SCORM uyumlu öğrenme nesneleri yaratmak, içerik paketi oluşturmak, ve sıkıştırmak ve önlizleme gibi işlevleri için modüllere sahiptir. LOAT'ta "system developer" ve "content provider" olmak üzere 2 kullanıcı tanımlanmışken, LCDM'de 3 kullanıcı grubu vardır. Her iki uygulamada da kaynak havuzu vardır.

3. ELO eğitsel içerik hazırlama aracı ise LCDM'den farklı olarak, IMS, LOM Metadata, Alliance of Remote Instructional Authoring and Distribution Networks for Europe (ARIADNE, 2005), DCMI (DCMI, 2007) gibi farklı üstveri şemaları arasında çeviri yapılabilmesine izin vermektedir (Liliana at all, 2003) ve farklı xml teknolojilerini desteklemektedir. LCDM ise SCORM'a bağlı olarak IEEE LOM standardına bağlıdır. LCDM'deki gibi fakülte, bölüm ve ders bazında içerik oluşturmak, sürüm oluşturmak, ELO'da mümkün değildir.
4. WeLOAD (WeLOAD, 2007), PHP ve MySQL ile geliştirilmektedir. WeLOAD, sisteme dosya yükleme, içerik paketi oluşturulması ve bunların arşivlenmesi, sıkıştırılmış dosyalarının indirilmesi gibi işlemleri yapabilmektedir fakat, üstveri girişi ile ilgili bir uygulama boyutu henüz yoktur. Oysa LCDM platformunda bunlar yapılabildiği gibi arşivleme, üstveri ekleme modulu de aktif olarak çalışmaktadır. Her iki uygulamada da dosya ve paket bazında önlizleme mümkündür. Ancak LCDM, sürüm oluşturmak, SCORM 2004 desteği ve sıralama bilgilerinin girilmesini desteklerken WeLOAD desteklemez. LCDM birden fazla dil desteği olacak biçimde tasarlanmıştır ancak WeLOAD'da böyle bir destek yoktur. Ayrıca WeLOAD'da içerik paketi havuzu olmadığından bir arama işlemi de gerçekleşemez.

LCDM'deki gibi fakülte, bölüm ve ders bazında içerik oluşturmak WeLOAD'da mümkün değildir.

5. İçerik paketleme amaçlı kullanılacak başka bir editor olan "IMS Packet Editor" de (IMS Packet Editor, 2007), LCDM gibi, öğrenim nesnelerinin bir ders paketi (sıkıştırılmış zip dosyaları şeklinde) içine eklenmesi, organize edilmesi ve depolanmasına olanak sağlar. Üstveri tanımlamalarında IMS LOM tanımlayıcı bilgileri temel alınmıştır. LCDM de ise IEEE LOM bilgileri temel alınmıştır. IMS Paket Editor, web tabanlı değildir fakat LCDM web tabanlıdır ve geniş bir kullanıcı kitlesine ulaşma, içerik paylaşma gibi olanaklar sunmaktadır. Fakat LCDM, IMS Package Editor gibi açık kaynak kodlu değildir.

Üstveri tanımlamalarına izin veren LCDM platformu, bölüm 2'de adı geçen üstveri editörleri ile de karşılaştırılabilir. Ancak bu editörlerin eğitsel içerik paketleme ve SCORM standardına uyum özellikleri yoktur. Karşılaştırma sonuçları aşağıda açıklanmış ve Çizelge 6.3'te gösterilmiştir. Çizelgede veri içermeyen hücreleri tanımlayan özellikleri ya ilişkili platform desteklememektedir, ya da incelenen kaynaklarda ilgili konuda bilgiye rastlanmamıştır.

1. Imse Vimse açık kaynak kodlu ve IMS tabanlı bir ÖN üstveri tanımlama editörüdür (Nilsson & Palmér, 2002). Oysa LCDM, Scorm standardı desteğinden dolayı üstveri girişini IEEE LOM spesifikasyonuna uygun olarak yapmaktadır. Ayrıca LCDM, ImseVimse gibi tek bir amaca hizmet etmemektedir. Üstveri tanımlama haricinde organizasyon oluşturma ve içerik paketleme gibi farklı fonksiyonlara da sahiptir.

2. Learning Object Metadata (LOM) Editör (Technische Universität Darmstadt, 2001) de sadece üstveri tanımlamalarını yapabilmek için kullanılmaktadır ve tek bir amaca hizmet etmektedir.
3. EM² (Sampson et. all, 2002), LCDM gibi Windows ortamında geliştirilmiştir, fakat C# ile değil java ile yazılmıştır. Ayrıca EM², üstveri standartları arasında dönüşümü desteklemektedir. LCDM'den farklı olarak IMS LOM standardını destekler. Her iki uygulamada da 3 kullanıcı grubu tanımlanmıştır. Hem EM² hem de LCDM'de, XML dosyaları havuzunda tutulan her üstveri dosyası aynı zamanda bir veritabanında da tutulmaktadır.
4. CLOMAT (Malaxa, 2003) ise LCDM platformu gibi, web tabanlı olarak ASP.NET, C# ve SQL veritabanı kullanılarak geliştirilmiştir. Ancak LCDM sadece SCORM üstveri modelini desteklerken, CLOMAT ise yeni üstveri modeli tanımlanmasına imkan verir. Her iki platformun da birden fazla dil desteği vardır. Ve yine her iki platformda da üstveri girişi ağaç yapı görüntüsünde bir arayüz üzerinden gerçekleştirilir.
5. Web üzerinden, "html" dosyaları ile üstveri oluşturmada kullanılan "LOMGen" Java programlama dili kullanılarak yazılmıştır ve oluşturulan üstveri dosyaları CANLOM üstveri havuzunda tutulmaktadır (Singh, et.all, 2004). LCDM ise üstveri dosyalarını saklarken aynı zamanda veritabanına da kaydetmektedir. Ancak LCDM'de tutulan üstveri dosyaları GUID olarak isimlendirilir.
6. Bu grup altında incelenen son editor olan Tree View LOM Editor (TreeLOM, 2006), web tabanlı bir editördür ve erişilebilen mevcut sürümü beta1.0'dır. Bu da LCDM gibi .NET platformunda, Windows sunucu ve IIS (Internet Information Server) üzerinde geliştirilmiştir.

Çizelge 6.3 LCDM platformunun üstveri editörleri ile karşılaştırılması.

	LCDM	Imsevimse	LOM Editor	EM²	Clomat	LOMGEN	TreeLOM
Açık k. kodu		✓	✓	✓		✓	✓
Prog. dili	C#	Java	Java	Java	ASP.NET, C#	Java	ASP.NET C#
Veri tabanı	SQL				SQL		
İşletim sistemi	Windows		Win, Unix, Linux	Windows NT	Windows		windows
Dil desteği	Türkçe,ingilizce (multilanguage)	İngilizce	İngilizce	İngilizce	İngilizce	İngilizce	İngilizce
Üstveri	IEEE LOM	IMS	IEEE LOM	IMS, IEEE LOM	IEEE LOM	✓	IEEE LOM

Her iki çalışmada da üstveri girişi ağaç görünümdeki bir arayüz ile sağlanmaktadır. TreeView ile sadece kaynak dosyalar için üstveri tanımlaması yapılabilirken, LCDM ile kaynak dosya, içerik organizasyonu ve bu organizasyon içindeki farklı kademeler ve içerik paketinin kendisini tanımlayan üstveri girişi yapılabilmektedir.

Bu karşılaştırmaların dışında, LCDM platformu, bir içerik paketi deposuna sahip olduğundan dolayı, AtaNesA (AtaNesA, 2007) nesne ambarı ile de karşılaştırılabilir. Her ikisi de doktora tezi kapsamında geliştirilmiştir. Fakat, LCDM sadece bir nesne ambarı olarak değerlendirilemez. AtaNesA ise sadece hazırlanan nesnelerin sisteme yüklenebildiği ve sistemdeki nesnelerin kullanıcıların erişimine açıldığı, sorgulama sonucunda istenilen nesnelerin görüntülenebildiği ve kayıtlı kullanıcılar tarafından yerel diske kayıt edilebildiği bir nesne ambarıdır. LCDM ise, hem bir kaynak dosya deposu hem de bir ÖN içerik paketi deposu olarak da hizmet verebilmesinin dışında, aynı zamanda eğitsel içerik oluşturmak için de kullanılan bir platformdur. LCDM'deki, paketlerin farklı sürümlerini oluşturma, birden fazla dil desteği, grup, kullanıcı, ders, dil yönetimi gibi fonksiyonlar AtaNesA'da bulunmamaktadır. LCDM platformunda kayıtlı içerik paketleri, ders ve paketi hazırlayan kullanıcılara göre ayrılabilindiğinden platform düzgün ve kolay bir sorgulama sağlar. AtaNeSa, web arayüzü üzerinden nesne kaydına izin vermektedir. LCDM platformunda ise tanımlı tüm fonksiyonlar web arayüzü üzerinden gerçekleştirilmektedir.

Bu bölümde LCDM platformu, bölüm 2'de incelenen içerik paketleme ve üstveri oluşturma araçları ile detaylı olarak karşılaştırılmıştır. Bölüm 7'de ise LCDM platformu farklı açılardan değerlendirilmektedir.

7. LCDM PLATFORMUNUN FARKLI AÇILARDAN DEĞERLENDİRİLMESİ

Bu bölümde LCDM platformu, öğrenme modelleri açısından, bilgi yönetimi, iletişim araçlarının kapasitesi ve öğretim yazılımlarının özellikleri açısından değerlendirilmekte ve sonuçlar sunulmaktadır.

7.1 Öğrenme Modelleri Açısından Değerlendirilmesi

LCDM platformu öğrenme modelleri açısından değerlendirildiğinde aşağıdaki sonuçlar görülür.

- Tasarlanan eğitsel içeriğin tümü, öğrencilere web üzerinden aktarılmaktadır. Bu, öğrencinin algıladıklarını basitleştirmek için öncelikle önceki deneyimlerine başvurmak zorunda kalmasını sağlayabilir. Bu bağlamda sistem “Gestalt” öğrenme teorisini (Leflore, 2000) desteklemektedir .
- Sistemde eğitsel içerik öğrenciye sunulur ancak öğrenme işlemini öğrenci kendisi gerçekleştirir. Yani bilgi, öğrenen kişi tarafından oluşturulur. Bu bakımdan sistem “constructivist” (yapılandırmacı) öğrenim modelini (Huerta and Igarria, 2003; Leidner and Jarvenpa, 1995) destekler. Çünkü bu model, öğrenme işlemini “öğrenen kişi tarafından bilginin oluşturulması” olarak görür (Leidner and Jarvenpa, 1995).
- Yapılandırmacı teoriye göre öğrenme, öğrencilerin önceden edindikleri bilgilere dayanarak yeni fikirler kavramlar oluşturdukları yapılandırdıkları aktif bir süreçtir (Constructivism, 2005). LCDM platformunda, öğrenmenin gerçekleşebilmesi için öğrencinin istekli katılımını gerektirmektedir ve öğrenci kendi öğrenme sürecinin kendisi kontrol eder.

- Ayrıca yapılandırmacı teoride eğitimler öğrenmek için kolaylaştırıcı, cesaretlendirici rol oynarlar. LCDM’de öğretmen eğitsel içeriği öğrenmeyi kolaylaştırıcı şekilde görsel materyallerle destekleyip, içeriği öğrencilere farklı organizasyonlar halinde sunabilmek hakkına sahiptir. Sistem, öğrencinin kendi kendine çalışma, zamanını programlama, araştırma yapan aktif öğrenciler haline getirecek şekilde tasarlanmıştır.
- Yapılandırmacı öğrenmedeki bazı temeller, yeniden kullanılabilir ÖN tasarımı için de geçerlidir. Öncelikle ÖN’nin yaratıcısı, öğrencilere açıklayacağı bir durum seçer. Bu duruma bir başlık verir ve öğrenme işlemini tanımlar. Bu durum, öğrenenden beklenenleri ve kendi anlamlarını nasıl yaratacaklarını içermelidir. Öğrencinin bildikleri ile, öğrenmelerinin istendiği şeyler arasında bir köprü yaratılır (Gagnon and Collay, 2005; Bloom, 1956; Gagne, 1970). Gerçek yaşamı etkileyen içerikteki anlamları yapılandırmak için birey diğer bireylerle iletişime geçer.
- Tasarlanan sistem ayrıca “objectivist” (nesnel) öğrenim modelini (Huerta and Igarria, 2003; Leidner and Jarvenpa, 1995) de desteklemektedir. Çünkü proje, hazırlanan eğitsel içeriğin istenilen kriterlere göre arama yapmaya açık olmasından dolayı içeriğin eğiticiden öğrenciye transfer edilmesini sağladığından nesnelidir.
- “Minimalist” (minimal) tasarım ve ayrıntıcılık da LCDM platformundaki özellikler arasında sıralanabilir. ÖN’leri kitaplarla karşılaştırıldığında öğrencinin daha kısa sürede öğrenmesini sağlamaktadır. ÖN içerik paketleri, minimal büyüklüklerde yeniden kullanılabilen kaynaklar bir araya getirilerek oluşturulmuş eğitsel içeriklerdir. Diğer taraftan ÖN’lerinin içeriği genelde kolaydan zora, basitten karmaşığa doğru şekillenmektedir. Yüksek öğretimde pek çok alanda kullanılan “ayrıntıcı teori”nin

(elaboration) temelinde de içeriğin basitten karmaşık yapıya doğru organize edilmesi yatmaktadır. İçlerindeki animasyon ve simülasyonlar da Öğrenme nesnelere daha etkili ve detaylandırıcı kılmaktadır. Öğrenme nesnelere ve ÖN içerik paketlerini oluştururken öğrenme teorilerinin temel prensipleri dikkate alınmalıdır (Balcı, İnceoğlu, 2006-a).

7.2 Bilgi Yönetimi Açısından Değerlendirilmesi

Web tabanlı eğitimde bilgi işleme süreci, bilginin oluşturulması, depolanması ve erişilmesi, bilginin transferi ve uygulama olmak üzere dörde ayrılmaktadır (Alkan, 1997). Bilgi yönetiminin esas amacı organizasyonda oluşturulan bilginin yeniden kullanımınıdır. Bilgiler “Bilgi Ambarları” (Knowledge Repositories) adı verilen veritabanlarında tutulur. İki tür bilgi veritabanı vardır. Birincisi öğrencinin oluşturduğu ve öğrencinin kullandığı veritabanları, ikincisi ise öğretmenin oluşturup öğrencinin kullandığı veritabanlarıdır. Bilgi yönetimi açısından incelendiğinde, bu çalışmada tasarlanan LCDM platformu aşağıdaki dört noktayı desteklemektedir.

- LCDM platformunda öğretmenlerin sisteme yüklenen dosyaları kullanarak SCORM standardına göre ÖN ve ÖN içerik paketleri oluşturmaları, bilgi yönetiminde “bilginin oluşturulması” adımına karşılık gelmektedir.
- Sistemdeki kaynak ve içerik paketi havuzları ve bunlar içinden arama yapılabilmesi, bilgi yönetiminde “bilginin depolanması ve erişilebilir olması” adımlarına karşılık gelmektedir.
- Kullanıcıların istedikleri içerikleri kendi makinelerine indirebilmesi veya sistem üzerinden takip edebilmesi, bilgi yönetiminde

“bilginin transferi” ve “bilginin uygulanması” adımları desteklemektedir.

- LCDM platformunun SCORM standardını desteklemesi ve web-tabanlı olarak tasarlanması ise bilgi yönetiminde bilginin yapısı ve kolay erişilebilir olması açısından, platformun değerini artırmaktadır.

7.3 İletişim Araçlarının Kapasitesi Açısından Değerlendirilmesi

Web-tabanlı eğitimde öğretmen-öğrenci iletişimi bilgisayar tabanlıdır. Kullanılan iletişim aracının kapasitesini ölçmede “ortamın zenginliği (media richness)” ve “sosyal etki (social influence)” olarak 2 temel model vardır. Yüksek seviyedeki uygulamalar için daha zengin bir iletişim aracı gereklidir. Tasarlanan sistemde eğitsel içeriğin farklı platformlardaki en küçük eğitsel kaynaklar yani varlıklar ve SCO’lar halinde hazırlanmasıyla içeriğin zorluk derecesine göre metin editöründen videoya, Flash animasyona kadar hem göz hem kulağa hitap ederek öğrenmeyi daha kalıcı kılmak mümkün olabilmektedir. Sosyal etki modelinde ise iletişim aracı algıları sosyal olarak inşa edilmiştir. Webster ve Trevina’ya (1995) göre araç seçiminde rasyonel faktörler kadar sosyal faktörler de önemlidir. Eğitimciler öğrenciye kullanacağı iletişim aracını seçmekte özgür bırakmalıdır. Ya da farklı işlerde farklı araçların kullanımını sağlamalıdır (sohbet, videokonferans vs). Teknik bilgi de araç seçiminde önemlidir (Huerta, E. and Igarria, M., 2003).

7.4 Öğretim Yazılımlarının Özelliklerine Göre Değerlendirme

Daha önceki çalışmada (Balcı ve İnceoğlu, 2006b) öğrenme nesneleri öğrencilerin ilgisini çekmiş, konuyu daha kolay öğrenmelerini

sağlamış ve öğrenciler öğrenme nesnelere ile çalışmaktan memnun olduklarını belirtmişlerdir. LCDM de bu bakımdan incelendiğinde;

- Bir ÖN içerik paketi oluşturma ve gösterme sistemi olduğuna göre, öğrencilerde ilgi çekici, öğrenme isteğini artırıcı, öğrenmeye motive edici, öğrenmenin devamını sağlayıcı nitelikte olacaktır.
- Bir eğitim yazılımı, bilgisayar teknolojisinin gelişmiş multimedya (çoklu ortam; metin, görüntü, resim, video, animasyonlar, ses, vb.) olanaklarından mümkün olduğunca çok yararlanmalıdır. LCDM’de de ÖN içeriği organizasyonu oluştururken kullanılan kaynaklar farklı farklı multimedya ortamlarını barındırabilirler.
- Tasarım açısından eğitim yazılımlarında dikkat edilmesi gereken kriterler oldukça fazladır. Kullanıcı arayüzündeki ikon, mönü dizaynı, bağlantılar, metin formatı gibi kriterler açısından bakılırsa, tasarlanan sistemdeki tüm arayüz sayfalarının aynı formatı desteklemesine, kullanıcının zihinsel yükünü artırmamasına dikkat edilmiştir. Bağlantılar, kullanıcının program içinde kaybolma riskini azaltacak şekilde verilmiştir.
- Pareja-Flores ve Velázquez-Iturbide’nin (2003) dediği üzere bir animasyon 3 ile 7 arasında renk içermelidir. Çünkü çalışmalar göstermiş ki, 20 sn’lik kısa bellek 5 kelime ya da şekil, 6 harf, 7 renk ve 8 digit tutabilmektedir. LCDM platformundaki arayüzlerin tasarımında da buna dikkat edilmiş, hatta hepsinde aynı renkler kullanılarak bütünlük sağlanmış, kullanıcının sistemde dolaşması kolay hale getirilmeye çalışılmıştır

Bu bölümde, LCDM platformu öğrenme modelleri, bilgi yönetimi, iletişim araçlarının kapasitesi ve öğretim yazılımlarının özellikleri açısından incelenmiştir. Bölüm 8’de ise sonuçlar açıklanmaktadır.

8. SONUÇ

Bölüm 1 ve 2’de anlatılan öğrenme nesneleri, ÖN içerik paketleri, ÖN havuzlarının kullanımı ile eğitsel içerik hazırlama ve bu içeriğe ulaşma aşamalarındaki zaman kaybı azaltılmakta, mekan bağımlılığı ortadan kalkmaktadır. İçeriğin öğrenme nesneleri denilen küçük, aranabilir ve yeniden kullanılabilir birimler halinde tasarlanması bilgi paylaşımını artırır. Hazırlanan içeriğin, aynı standardı destekleyen bir eğitim yönetim sisteminde çalışabilir olması da içeriğin platform bağımsız olmasını desteklemektedir. Uyarlanabilir ve birlikte kullanılabilir olması ÖN’lerinin farklı kişiler tarafından ve farklı ortamlarda da çalışabilir olmasını sağlamaktadır.

Doktora tezi kapsamında yapılan bu çalışmada, LCDM, “Learning Content Design Manager”, adı verilen, eğitsel içerik hazırlama ve üstveri tanımlama işlevlerini gerçekleştiren bir platform tasarlanmıştır. Platform, SCORM 2004 standardını desteklediğinden farklı yazarlık araçları ve eğitim yönetim sistemleri arasında taşınabilir olan içerikler oluşturmayı mümkün kılmaktadır. Aşağıda kısaca platformun özellikleri verilerek sonuçlar üzerinde durulmuştur.

- İnternet tabanlı, ders ve kullanıcı bazında eğitsel içerik oluşturulmasına izin veren bir eğitsel içerik tasarım yöneticisidir.
- Mekan bağımsız olarak kullanıcıların dahil olup eğitsel içerikleri takip edebileceği bir platformdur.
- ASP.NET, C# programlama dili ve Microsoft SQL Server veritabanı kullanılarak tasarlanmıştır.
- Kullanıcıların sisteme girmeleri kullanıcı adı ve şifre doğrulamasını gerektirmektedir.

- Sistemde yayımcı, istemci ve yönetici olmak üzere üç farklı kullanıcı grubu tanımlanmıştır. İstenilirse yeni kullanıcı grupları yaratılabilmektedir.
- “İyi teknoloji, modüler olan teknolojidir” (Hashim and Mustapha, 2005) mantığı ile sistemde yapılacak ana işler de modüler şekilde tasarlanmıştır. LCDM platformu toplam 12 modülden oluşmaktadır.
- Sistemde, tüm yayımcıların erişimine açık olan bir kaynak havuzu bulunmaktadır. Ancak yayımcı, istediği dosyaları paylaşım açma, istemediklerini açmama hakkına sahiptir. Kaynak dosyaların önlizemesinin görüntülenmesi, arama, güncelleme vs. gibi işlemlerin yapılması mümkündür.
- Platformda ayrıca bir içerik paketi havuzu oluşturulmuş ve bu havuzdan arama, önlizleme yapılabilmesi sağlanmıştır. Eğitsel içerik havuzu Huerta ve diğerlerinin (2003) de tanımladığı üzere, eğitmenlerin hazırlayıp öğrencilerin ve eğitmenlerin kullanımına açık olan bir nesne havuzudur.
- SCORM 2004 şartnamesini destekleyen başka bir uygulama ile hazırlanan eğitsel içerik paketlerinin platforma yüklenmesi, çalışır hale getirilmesi, üzerinde değişiklik yapılarak saklanması mümkün kılınmıştır. Eklenen bu paketlerle gelen kaynak dosyalar işaretlenmekte, bu da yönetsel açıdan yapılan işlemlerin takibini kolaylaştırmaktadır.
- Sisteme kayıtlı öğrenciler, içerik paketi havuzuna erişerek, aldığı derslerle ilgili paketler arasından arama, izleme ve kendi sistemine indirme hakkına sahiptir. Böylece öğrenciler bir LMS’e gerek kalmadan, zaman ve mekandan bağımsız olarak, hatta aynı konuları farklı içerik sağlayıcıların hazırladıkları içeriklerden takip ederek

bireysel öğrenme gerçekleştirebileceklerdir. Öğrencinin durumuna göre dersin ayarlanması (Wenger, 1987) diye bir sorun kalmayacak, öğrenci bildiği konuları atlayarak ilerleyebilecektir.

- Öğrencilerde içerikleri takip etme isteği uyandırabilmesi için mümkün olduğu kadar basit ve sade ekranlar tasarlanması uygun görülmüştür.
- LCDM platformu, öğretim yazılımlarında olması gereken özellikler gözönünde bulundurularak hazırlanmış bir eğitim platformudur. İçerik sağlayıcıların daha hızlı ve daha etkili şekilde eğitsel içerik oluşturmaları için kullanışlı, birbirleri arasında basit geçişler içeren ekran tasarımlarına gidilmiştir. Sistem, kullanım sırasında ne içerik sağlayıcılar ne de öğrencilerin çok fazla teknik detay bilmesine gerek kalmayacak kolaylıkta tasarlanmıştır. Ekranlarda aşırıya kaçmamak koşuluyla açıklama metinleri girilmiştir (Kearsley, 1986, Merja et al., 2004).
- Tasarlanan eğitsel içeriklerde sürüm oluşturmak sayesinde hazırlanan eğitsel içerik üzerinde güncelleme yapılması durumunda eski sürümlerin saklanması, ve bunların yöneticiler tarafından izlenmesi sağlanmıştır.
- LCDM üstveri bilgileri ağaç yapı içeren bir arayüz üzerinden girilmektedir. Bu sayede bilgi girişi kolaylaşmakta ve kullanıcı ihtiyacı olmayan alanları saklayabilmektedir.
- Ayrıca, Hashim ve Mustapha'nın (2005) "iyi teknoloji" tanımında kullandıkları kriterler göz önünde bulundurulduğunda, tasarlanan sistem *çevrimiçi ve her zaman erişilebilir* olarak tasarlanmıştır. En azından bir tarayıcıya sahip ve standart özelliklerde bir bilgisayar ile ilave bir yazılım yüklemeye gerek kalmadan sisteme kayıtlı her

kullanıcı istediği zaman eğitsel içerik hazırlayabilir ya da içeriği görebilir.

- Platform üzerinde öğrenci içeriği istediği kadar tekrar edebilme ve kendi makinesine indirebilme hakkına sahiptir.
- ÖN içerik paketlerine simulasyon ve animasyonların eklenebilmesi ile öğrenciye daha hareketli ve interaktif içerikler sunulabilir.
- LCDM sayesinde, içeriğin küçük konulara bölünerek, öğrenciye daha kısa sürede ve daha sistemli şekilde aktarımı sağlanmış olur.
- LCDM, şu anda Türkçe ve İngilizce ile sınırlı olmakla birlikte, birden fazla dili destekleyecek şekilde tasarlanmıştır. Platformda bir “Dil Yönetimi Modülü” oluşturulmuştur. İlgili arayüz üzerinde, tanımlı diller görülebilmektedir. Dil eklemek, dil seçimi yaparak tüm dil tanımlamaları için, kod, açıklama metni, seçilen dildeki karşılığı gibi bilgiler görüntülenebilir ve güncellemeler yapılabilir.
- Yönetici grubundaki kullanıcılar sisteme dahil olduklarında, platforma yeni dil ekleyebilecekleri ve mevcut dillerdeki açıklamaları değiştirebilecekleri bir arayüze erişecek istedikleri işlemleri yapabilmektedirler. Bu sayede yönetici dil güncellemeleri için veritabanını açmak zorunda kalmadan, daha hızlı şekilde işlemlerini yapabilmektedir.
- LCDM platformu, bölüm 7’de de açıklandığı üzere, “Gestalt”, yapılandırmacı, nesnel, minimal ve ayrıntıcı öğrenme modellerini desteklemektedir.
- LCDM platformu, bilginin oluşturulması, depolanması ve erişilebilir olması, bilginin transferi ve kolay erişilebilirlik açısından bilgi yönetimi basamakları ile uyumludur.

- LCDM, bir eğitim yönetim sistemi değildir. Ancak kullanıcı, grup, ders, dil yönetimi, önlzleme gibi modüller içermektedir. Önlzleme modülü sayesinde, hazırlanan içerikler bir LMS'e gerek kalmadan takip edilebilmektedir.
- LCDM, bir eğitim yönetim sistemi değildir. Ancak, LCDM platformu üzerinde hazırlanan içerik paketleri, SCORM 2004 standardını destekleyen LMS ve yazarlık araçları arasında paylaşılabilir.
- LCDM platformuna yüklenebilecek dosya boyutu, incelenen diğer eğitim yönetim sistemlerindeki kullanılan dosya boyutundan çok daha büyük olarak belirlenmiştir. Bu da, platforma video dosyası gibi eğitsel kaynakların yüklenebilmesini kolaylaştırmaktadır.
- LCDM, incelenen literatürden anlaşıldığı üzere, Türkiye'deki bu fonksiyonları yerine getiren kapsamlı ilk uygulamadır.

Öğrenme nesneleri, eğitsel tasarım aşamasında önemli yeri olan ve kullanımı yaygınlaşan bir konudur. Web tabanlı eğitimde ÖN'lerinin kullanılması ve yeniden kullanılabilir olma özellikleri sayesinde eğitim zamandan ve mekandan bağımsız hale gelmektedir. ÖN'lerinin yaratılması zaman almaktadır ancak küçük ÖN'leri oluşturmak ve bunları diğerleri ile birleştirmek genel olarak zaman tasarrufu sağlamaktadır. LCDM platformu, web-tabanlı eğitimde kullanılacak eğitsel içeriklerin hazırlanması ve dışarıdan sisteme bütünleştirilmesini sağlayan, içerik tasarımında zaman tasarrufu, içeriğe erişimde ise mekan bağımsızlığı sunan, paylaşılabilir web-tabanlı bir platformdur.

İleriki çalışmalar kapsamında, SCORM 2004 standardına uyumlu bir LMS geliştirilerek, LCDM geliştirilen bu LMS'e bütünleştirilebilir. LCDM platformu sadece IEEE Learning Object Metadata şartnamesini desteklemektedir. Yeni bir araştırma olarak, IEEE LOM dışındaki diğer

üstveri şemaları incelenip LCDM platformuna farklı üstveri şemalarının eklenmesi ve aralarındaki çevirilerin yapılması düşünülebilir. LCDM platformunun yaygın olarak kullanılması sonrasında alınacak geribildirimler ve Türk eğitim sistemi de gözönünde bulundurularak, yeni bir üstveri tanımlama şeması oluşturulması ve bunun LCDM platformuna özgü bir üstveri tanımlaması olarak kabulü sağlanabilir. Oluşturulan eğitsel içeriğin farklı LMS'lerde yeniden kullanılabilir olması için bir veri modeli tanımlanmalı ve sadece o veri modelindeki elemanlar kullanılmalıdır. Veri modeli, LMS ile kaynak arasında iletilen, kaynağın durumu gibi kaynakla ilgili bilgileri içeren standart veri elemanları kümesidir. Farklı bir araştırma olarak, SCORM çalışma ortamının kullanacağı yeni bir veri modeli üzerinde çalışılabilir.

EK AÇIKLAMALAR A

PROJEDE KULLANILAN VERİTABANI

Projede Microsoft SQL Server veritabanı kullanılmıştır. Bu bölümde, veritabanındaki tablolar gösterilmekte ve kısaca açıklanmaktadır. Ayrıca tablolar arasındaki ilişkiler de bir “diagram” olarak verilmektedir.

LCDM platformunun birden fazla dili desteklemesi için, Şekil A.1’de görülen iki tablo kullanılmıştır. Dil tanımlamaları için “KTPHN_DILLER” tablosu, tanımlı dillerdeki metin karşılıkları için ise “KTPHN_METINLER” tablosu oluşturulmuştur. “KTPHN_DILLER” tablosunda, otomatikolarak artan D_ID sütunu dili tanımlamak için; D_KODU dil kodu için; D_ADI ise dilin adı için kullanılmaktadır. Ayrıca platformda her dil bir resim ile gösterilmektedir. Bunun için ise D_RESIM_YOLU alanı kullanılmıştır.

TBL_KTPHN_DILLER				
	Column Name	Data Type	Length	Allow Nulls
PK	D_ID	smallint	2	
	D_KODU	varchar	5	✓
	D_ADI	varchar	50	✓
	D_RESIM_YOLU	varchar	20	✓

(a)

TBL_KTPHN_METINLER	
PK	M_ID
	M_DID
	M_KODU
	M_METIN
	M_ACIKLAMA
	SISTEM_DEGERI_MI

(b)

Şekil A.1 Dil tanımlamaları ile ilgili tablolar. a) Diller b) Dillerdeki metin karşılıkları

Sistemdeki kullanıcı bilgileri “aspnet_users” tablosunda tutulmaktadır. Her kullanıcının benzersiz bir kullanıcı tanımlayıcısı vardır ve UserId sütun adı ile belirtilmiştir. Bu, pekçok tablo ile ilişki kurmak için kullanılan bir alandır. Şekil A.2’de “aspnet_users” tablosu görülmektedir.

aspnet_Users				
	Column Name	Data Type	Length	Allow Nulls
	ApplicationId	uniqueidentifi	16	
⚡	UserId	uniqueidentifi	16	
	UserName	nvarchar	256	
	LoweredUserName	nvarchar	256	
	MobileAlias	nvarchar	16	✓
	IsAnonymous	bit	1	
	LastActivityDate	datetime	8	

Şekil A.2 Kullanıcıların tanımlandığı tablo

Platformda tanımlı dersler ve bu derslerin kullanıcılar ile ilişkilendirilmesini gösteren tablolar ise Şekil A.3’de gösterilmektedir. “TBL_DERSLER” tablosunda ders kodu, adı, açıklama, aktif olup olmadığı, eklenme tarihi ve dersin kim tarafından eklendiği bilgileri; “TBL_KULLANICILAR_DERSLER” tablosunda ise hangi kullanıcının hangi ders ile ilişkisi olduğu bilgisi tutulmaktadır.

TBL_DERSLER				
	Column Name	Data Type	Length	Allow Nulls
⚡	DRS_ID	int	4	
	DRS_ADI	varchar	200	✓
	DSY_EKLENME_TRI	datetime	8	✓
	DSY_EKLEYEN_KISI	uniqueidentifi	16	✓
	AKTIF_MI	bit	1	✓
	ACIKLAMA	nvarchar	200	✓
	KOD	varchar	10	✓
	FK_BOLLUMID	int	4	✓

TBL_KULLANICILAR_DERSLER				
	Column Name	Data Type	Length	Allow Null
	KULLANICI_ID	uniqueidentifi	16	✓
	DRS_ID	int	4	✓
	KULLANICI_ROL_ID	uniqueidentifi	16	✓

(a)

(b)

Şekil A.3 Dersler ve Kullanıcılar ile ilişkilendirildikleri tablolar a) Derslere ait bilgiler
b) Kullanıcılar ile dersler arasındaki ilişkiler

LCDM platformunda dizin yapısı hiyerarşiktir, yani iç içe dizinler oluşturulabilmektedir. Yayımcıların oluşturdukları dizinlerin tutulduğu tablo olan “TBL_DIZINLER” tablosunun işlevi hangi dizinin hangi dizinin altında olduğunu bilebilmektir. Şekil A.4’te bu tablo

görülmektedir. Buradaki DZ_ID değeri dizin numarasını, DZ_UST_ID değeri dizinin bağlı olduğu üst dizin numarasını, DZ_DOSYA_ADI adını, DZ_EKLENME_TRH eklenme tarihini, DZ_EKLEYEN_KISI ise dizinin hangi kullanıcı tarafından eklendiğini göstermek için tanımlanmıştır.

TBL_DIZINLER				
	Column Name	Data Type	Length	Allow Nulls
🔑	DZ_ID	smallint	2	
	DZ_UST_ID	smallint	2	✓
	DZ_DOSYA_ADI	varchar	500	✓
	DZ_EKLENME_TRH	datetime	8	✓
	DZ_EKLEYEN_KISI	uniqueidentifi	16	✓

Şekil A.4 Dizinler tablosu

Platforma yüklenecek dosyalar ve ilişkili bilgileri tutmak amacıyla “TBL_DOSYALAR” tablosu oluşturulmuştur. Şekil A.5’te görülen tabloda DSY_COCUK_VAR_MI sütunu, içerdiği değere göre dosyanın içinde başka dosya, referans, içerip içermediğini gösterir. Aldığı değer 1 ise dosya html tarzında bir dosyadır. Dosyayı barındıran dizin, DSY_DIZIN_ID ile belirtilerek, “TBL_DIZINLER” tablosu ile ilişkilendirilmektedir. Dosyanın gerçek adı DSY_G_DOSYA_ADI, GUID’e göre belirlenen adı ise DSY_DOSYA_ADI alanlarında tutulur. Ayrıca bu tabloda dosyanın uzunluğu, bulunduğu yer yani dosya yolu, eklenme tarihi, hangi kullanıcı tarafından eklendiği bilgileri için alanlar, sıralama bilgisi için DSY_SEQUENS sütunu, dosyanın paylaşılıp paylaşılmadığını gösteren DSY_ISSHARED sütunu, üstveri dosyası adı, ayrıca dosyanın o kullanıcının dosyası olup olmadığını belirlemek için DSY_IS_MY_FILE alanı bulunmaktadır.

Ayrıca, eğitsel içerik oluşturmak için yayımcılar platforma web sayfası adresi de ekleyebilirler. Eklenen bu web adresleri veritabanındaki “TBL_LINKLER” isimli tabloya (Şekil A.6) kaydedilir. Burada, URL’nin

hangi dizin altına eklendiği, açıklaması, eklenme tarihi ve kimin tarafından eklendiği, aktif olup olmadığı gibi bilgiler bulunur.

TBL_DOSYALAR			
	Column Name	Data Type	Length
🔑	DSY_ID	int	4
	DSY_COÇUK_VAR_MI	smallint	2
	DSY_ANADOSYA_ID	int	4
	DSY_DIZIN_ID	smallint	2
	DSY_DOSYA_ADI	varchar	50
	DSY_UZUNLUGU	varchar	50
	DSY_DOSYA_MIME_TIPI	varchar	50
	DSY_G_DOSYA_ADI	varchar	500
	DSY_G_DOSYA_YOLU	varchar	1000
	DSY_EKLENME_TRH	datetime	8
	DSY_EKLEYEN_KISI	unique	16
	AKTIF_MI	bit	1
	DSY_SEQUENS	int	4
	DSY_ISSHARED	bit	1
	DSY_METADATA_FILE_NAME	unique	16
	DSY_IS_MY_FILE	bit	1

Şekil A.5 Dosyalar tablosu

TBL_LINKLER				
	Column Name	Data Type	Length	Allow Nulls
🔑	L_ID	int	4	
	L_DIZIN_ID	smallint	2	✓
	L_METIN	varchar	500	✓
	L_URL	varchar	1000	✓
	L_EKLENME_TRH	datetime	8	✓
	L_EKLEYEN_KISI	uniqueidentifier	16	✓
	AKTIF_MI	bit	1	✓

Şekil A.6 Web adresleri tablosu

Platforma eklenen dosya ve URL adresleri kullanılarak oluşturulacak olan eğitsel içerikler ile ilgili bilgiler “TBL_PROJELER” tablosunda tutulmaktadır. Şekil A.7’de de görülen tabloda, otomatik olarak değer alan proje tanımlayıcısı PRJ_ID alanı, proje adı,

açıklaması, eklenme tarihi, ekleyen kişi, projenin bağlı olduğu ders gibi bilgiler saklanmaktadır.

TBL_PROJELER				
	Column Name	Data Type	Length	Allow Nulls
🔑	PRJ_ID	int	4	
	PRJ_DOSYA_ADI	varchar	50	✓
	PRJ_ACIKLAMA	varchar	500	✓
	PRJ_EKLENME_TRH	datetime	8	✓
	PRJ_EKLEYEN_KISI	uniqueidentifi	16	✓
	AKTIF_MI	smallint	2	✓
	DRS_ID	int	4	✓

Şekil A.7 Projeler tablosu

Platformda oluşturulan projeler için yaratılan sürümlerle ilgili bilgiler “TBL_PRJ_VERSIYONLAR” isimli tabloda tutulmaktadır. Şekil A.8’de bu tablo görülmektedir. PRJV_ID alanı, projeye ait sürüm tanımlayıcısını, PRJV_PRJ_ID alanı sürümün bağlı olduğu proje tanımlayıcı bilgisini, PRJV_ADI alanı projeye ait sürüm adını tutmaktadır. Diğer alanlar sırayla sürüm ile ilgili açıklama, sürüm numarası, projeye sürümü ekleyen kişi ve ekleme tarihi bilgileri ile ilgilidir.

TBL_PRJ_VERSIYONLAR				
	Column Name	Data Type	Length	Allow Nulls
🔑	PRJV_ID	int	4	
	PRJV_PRJ_ID	int	4	✓
	PRJV_ADI	nvarchar	50	✓
	PRJV_ACIKLAMA	nvarchar	500	✓
	PRJV_NO	int	4	✓
	PRJV_EKLENME_TRH	datetime	8	✓
	PRJV_EKLEYEN_KISI	uniqueidentifi	16	✓

Şekil A.8 Proje ile ilişkili versiyonlar tablosu

LCDM platformunda eğitsel içerik paketi oluştururken bir sonraki adım, içerik organizasyonunu oluşturmaktır. Oluşturulan

organizasyonlarla ilgili bilgiler “TBL_ORGANIZASYONLAR” tablosunda tutulmaktadır. Buradaki Organizasyon_ID organizasyonun tanımlayıcısını, ORG_PRJ_ID organizasyonun hangi projeye ait olduğunu gösterir. Şekil A.9’deki diğer alanlar ise sırayla proje versiyon numarası, başlık bilgisi, organizasyon sırası, açıklama, eklenme tarihi ve ekleyen kişi, eğer silinmiş ise silinme tarihi ve hangi kullanıcı tarafından silindiği, organizasyona ait olan xml verisi ve organizasyonunun varsayılan olarak işaretlenip işaretlenmediği bilgilerini içerir.

TBL_ORGANIZASYONLAR				
	Column Name	Data Type	Length	Allow Nulls
PK	ORG_ID	int	4	
	ORG_PRJ_ID	int	4	✓
	ORG_PRJ_V_ID	int	4	
	ORG_BASLIK	nvarchar	500	✓
	ORG_SIRA	smallint	2	✓
	ORG_ACIKLAMA	nvarchar	500	✓
	ORG_EKLENME_TRH	datetime	8	✓
	ORG_EKLEYEN_KISI	uniqueidentifier	16	✓
	ORG_SILME_TRH	datetime	8	✓
	ORG_SILME_KISI	uniqueidentifier	16	✓
	ORG_XML	ntext	16	✓
	ORG_IS_DEFAULT	bit	1	✓

Şekil A.9 Organizasyonlar tablosu

Proje yönetim kısmında oluşturulacak organizasyonlarla ilgili olarak detay bilgilerin tutulacağı tablo ise “TBL_ORG_DETAYLARI” tablosu (Şekil A.10) olarak isimlendirilmiştir.

TBL_ORG_DETAYLARI				
	Column Name	Data Type	Length	Allow Nulls
PK	OD_ID	int	4	
	ORG_ID	int	4	
	OD_BASLIK	nvarchar	50	✓
	OD_UST_ODID	int	4	✓
	OD_SIRA	smallint	2	✓
	OD_GORUNURMU	bit	1	✓
	OD_DETAY_ID	int	4	✓
	OD_DETAY_TIP	tinyint	1	✓

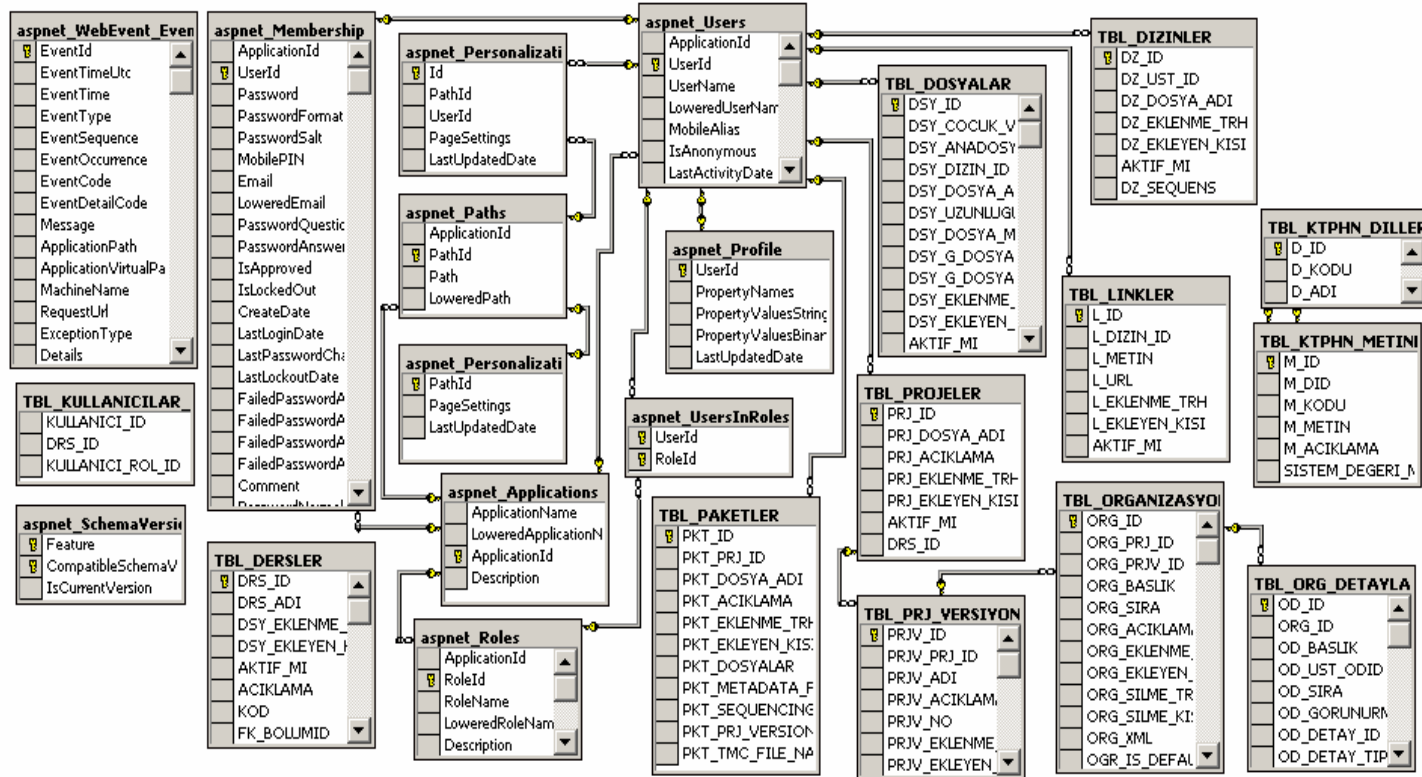
Şekil A.10 Organizasyon detayları tablosu

Oluşturulan organizasyon paket olarak kaydedildiğinde ise ilgili bilgiler “TBL_PAKETLER” tablosunda tutulmaktadır. Şekil A.11’de görülen tabloda sırayla paket tanımlayıcı bilgisi, paketin dahil olduğu proje numarası, paket adı, açıklaması, eklenme tarihi, ekleyen kişi, paketteki dosyalar, üstverisi ve sıralama kuralları ve paketin dahil olduğu projenin sürüm numarası tanımlanmıştır.

TBL_PAKETLER				
	Column Name	Data Type	Length	Allow Null
PKT_ID	int	4		
PKT_PRJ_ID	int	4		
PKT_DOSYA_ADI	varchar	50		✓
PKT_ACIKLAMA	varchar	500		✓
PKT_EKLENME_TRH	datetime	8		✓
PKT_EKLEYEN_KISI	uniqueidentifi	16		✓
PKT_DOSYALAR	ntext	16		✓
PKT_METADATA_FILE	uniqueidentifi	16		✓
PKT_SEQUENCING_FILE	uniqueidentifi	16		✓
PKT_PRJ_VERSION_ID	int	4		✓

Şekil A.11 Paketler tablosu

Aşağıda, Şekil A.12’de ise LCDM platformu için oluşturulan veritabanında tablolar arasındaki ilişkiler diagram halinde gösterilmektedir.



Şekil A.12 Tablolar arası ilişkiler

KAYNAKLAR DİZİNİ

Advanced Distributed Learning Network (ADL), 2006a, SCORM 2004 3rd Edition Content Aggregation Model, 245p.

Advanced Distributed Learning Network (ADL), 2006b, SCORM 2004 3rd Edition Sequencing and Navigation, 250p.

Advanced Distributed Learning Network (ADL), 2006c, SCORM 2004 3rd Edition Run-Time Environment, 220p.

ADLNet, 2007, <http://www.adlnet.org>, Son erişim tarihi: Ağustos 2007.

Aggarwal, A.K., 2003, A Guide To Ecourse Management :The Stakeholders' Perspectives, 1-23, Web Based Education: Learning From Experience, A.K.Aggarwal (Ed.), Hershey, Pa,Usa : Idea Group Inc., 472p.

Alkan, C., 1997, Eğitim Teknolojisi, Anı yayıncılık, Ankara, 225s.

Anido, L., Rodríguez, J., Caeiro, M., and Santos, J. M., 2003, High-level brokerage services for the elearning domain, *Computer Standards & Interfaces*, 25(4), p303-327.

ARIADNE, 2005, <http://ariadne.unil.ch>, Son erişim tarihi: Ocak 2005.

Aslantürk, O., 2002, Web tabanlı Uzaktan eğitim sistemi tasarlanması ve gerçekleştirilmesi, Yüksek Lisans Tezi, Hacettepe Üniversitesi Fen Bilimleri Enstitüsü.

KAYNAKLAR (devam)

- Aşkar, P.**, 2003, Eğitim Teknolojisi İçin Yeni Bir Kavram: Öğrenme Nesneleri, 12.Eğitim Bilimleri Kongresi.
http://www.ebit.hacettepe.edu.tr/Ogretim_Elemanlari/Petek_Askar/gazikongre.pdf
- AtaNesA**, Karaman,S., 2007, <http://atanesa.atauni.edu.tr/Default.aspx>,
Son erişim tarihi: Ağustos 2007.
- ATUTOR**, 2007, <http://www.atutor.ca/>, Son erişim tarihi: Ağustos 2007.
- Balcı,B. ve İnceoğlu M.**, 2004, Tıkanıklık Kontrolü Konusu İçin İnternet Tabanlı Bir Uzaktan Eğitim Uygulaması, Bilişim Teknolojileri Işığında Eğitim Konferansı Bildiriler Kitabı, Ankara, 150s.
- Balcı, B. ve İnceoğlu M.**, 2005, Eğitim teknolojilerinde Yeni Bir Kavram: Öğrenme Nesneleri ve Bir Uygulama, 1th International Vocational and Technical Education Technologies Congress, İstanbul, Türkiye.
- Balcı, B. ve İnceoğlu M.**, 2006a, How to Make Educational Content Reusable?, 6th International Educational Technology Conference – IETC 2006 Proceedings- Volume I. p:196-204. North Cyprus.
- Balcı, B. and İnceoğlu M.**, 2006b , Reusable Learning Objects (RLOs) for Computer Science Students, p:373-382, ICCSA2006 proceedings, LNCS 3980, Garilova, M.(Ed.), Glasgow.

KAYNAKLAR (devam)

- Başkömürçü, G. ve Öztürk, Y.**, 1996, Uzaktan Eğitim Sistemlerinin Tasarımı, 1. Türkiye Uluslararası Uzaktan Eğitim Sempozyumu Bildiriler Kitabı, Ankara.
- Bloom, B.** (Ed.), 1956, Taxonomy of educational objectives. Handbook I: Cognitive domain. New York: David McKay.
- Blum, H. and Sachs, D.**, 1999, An Asynchronous Distance-Learning Course in Data Communications and Networks, *SIGCSE Bulletin*, 31(4): 52-55.
- Careo**, 2007, <http://www.careo.org/>, Son erişim tarihi: Ağustos 2007.
- Cebeci, Z.**, 2003, Öğrenim Nesnelere Giriş, Elektronik Sosyal Bilimler Dergisi, 2(6):1-6.
- Cebeci, Z. ve Cimilli, C.**, 2004, Öğrenim İçeriklerinin Paketlenmesi: I. Bir Çevrim-İçi Ders Yaratma, Akademik Bilişim 2004 Konferansı (AB'04), Trabzon.
- CEN/ISS**, 2007, www.cenorm.be/iss, Son erişim tarihi: Ağustos 2007.
- CLAROLINE**, 2007, <http://www.claroline.net>, Son erişim tarihi: Ağustos 2007.
- Constructivism**, 2005, Constructivism as a Paradigm for Teaching and Learning, Son erişim tarihi: Ağustos 2005, <http://www.thirteen.org/edonline/concept2class/constructivism/index.html>

KAYNAKLAR (devam)

- Çağiltay, E. N.**, 2001, Türkçe Merhaba Eğitim Sitesi.
<http://www.princeton.edu/~turkish/practice/tlepss.html>, Son erişim tarihi: Şubat 2002.
- Çağiltay, N.E. ve Çağiltay K.**, 2002, Tekrar Kullanılabilen Öğrenme Nesneleri (TEKÖN) ve Örnek bir çalışma, 19th Turkish Informatics Society Conference, Istanbul.
- Çağiltay, N. ve Çağiltay, K.**, 2003, Sayısal Kütüphanelerin Tasarımında Sayısal Nesneler: Eğitimi nasıl destekleyebilirler?, *Türk Kütüphaneciliği*, 17(1):139-154.
- DCMI**, 2007, www.dublincore.org, Son erişim tarihi: Ağustos 2007.
- Demirel, Ö., Seferoğlu, S. ve Yağcı, E.**, 2001, Öğretim Teknolojileri ve Materyal Geliştirme, Pegem Yayıncılık, Ankara.
- DLESE**, 2007, <http://www.dlese.org>, Son erişim tarihi: Ağustos 2007.
- dotLRN**, 2007, <http://www.dotlrn.org/>, Son erişim tarihi: Ağustos 2007.
- Dzakira, H. and Idrus, R. M.**, 2003, Theacher-Learner Interactions in Distance Education: A Case of Two Malaysian Universities, *Turkish Online Journal of Distance Education- TOJDE*, 4(3).
- Ertürk, S.**, 1975, Eğitimde Program Geliştirme, Yelkentepe Yayınları, 4, Ankara.
- EduSource**, 2007, <http://www.edusource.ca>, Son erişim tarihi: Ağustos 2007.

KAYNAKLAR (devam)

ESCOT, 2007, <http://www.escot.org>, Son erişim tarihi: Ağustos 2007.

Gagne, R., 1970, *The conditions of learning*, New York: Holt, Rinehart, and Winston.

Gagnon, G.W., and Collay M., 2005, *Constructivist Learning Design*, <http://www.prainbow.com/cld/cldp.html>, Son erişim tarihi: Ağustos 2005.

Hashim, H.R.Hj., Mustapha, W.N., 2005, *Computers: Educational Technology Paradox?*, *The Turkish Online Journal of Educational Technology – TOJET*, ISSN: 1303-6521 4(3).

Huerta, E., Ryan T., and Igbaria, M., 2003, *A Comprehensive Web – Based Learning Framework : Toward Theoretical Diversity*, 24-35, *Web-Based Education: Learning from Experience*, A.K.Aggarwal (Ed.), Hershey, PA, USA : Idea Group Inc.

IDDL, 2006, *An Introduction to SCORM*.
<http://www.iddl.vt.edu>, Son erişim tarihi: Şubat 2006.

IEEE, 2002, 1484.12.1 IEEE Standart for Learning Object Metadata.

IEEE LOM, 2000, *Learning Object Metadata working draft v4.1*

IEEE LOM, 2004, *IEEE Standard for Learning Object Metadata*.

IEEE LTSC, 2007, <http://ieeeltsc.org>, Son erişim tarihi: Ağustos 2007.

IMS, 2003, *IMS Simple Sequencing Behavior and Information Model v1.0 Final Specification*.

KAYNAKLAR (devam)

IMS, 2004, Instructional Management Systems Global Learning Consortium, Inc.

IMS, 2007, <http://www.imsproject.org/>, Son erişim tarihi: Ağustos 2007.

IMS Package Editor, 2005, <http://www.met.ed.ac.uk/pac-man/editor/>,
Son erişim tarihi: Şubat 2005.

Indiana University, 2001, Variations Project,
<http://www.dlib.indiana.edu/variations/> , Son erişim tarihi: Şubat 2005.

Indiana University, 2002, USSteel Project, Indiana University USSteel photograph collection, Son erişim tarihi: Ağustos 2002,
<http://www.dlib.indiana.edu/collections/steel/>

ISO, 2007, www.iso.ch, Son erişim tarihi: Ağustos 2007.

Karaman, S., 2004, Öğrenme Nesnesi Nedir?,
<http://atanesa.atauni.edu.tr>, Son erişim tarihi Aralık 2004.

Karaman, S., 2005, Öğrenme Nesnelere Dayalı Bir İçerik Geliştirme Sisteminin Hazırlanması ve Öğretmen Adaylarının Nesne Yaklaşımı İle İçerik Geliştirme Profillerinin Belirlenmesi, Doktora Tezi, Atatürk Üniversitesi Sosyal Bilimler Enstitüsü, 142s (yayımlanmamış).

Kaya, İ. ve Bingöl, H., 2004, Üniversitelere Yönelik Bir Eğitim Yönetim Sistemi Uygulaması, Bilişim Teknolojileri Işığında Eğitim Konferansı ve Sergisi – BTIE 2004 Bildir Kitabı. ODTÜ, Ankara, 51-58.

KAYNAKLAR (devam)

- Kearsley, G.**, 1986, *Authoring A Guide to The Design of Instructional Software*, Addison-Wesley Publishing Company Inc, U.S.A.
- Klassen, J. and Vogel, D.**, 2003, *New Issues Arising From E-Education*, 36-48, *Web Based Education: Learning From Experience*, A. K. Aggarwal (Ed.) Hershey, Pa, Usa : Idea Group Inc. 472p.
- LEARNWISE**, 2005, <http://www.learnwise.com>, Son erişim tarihi: Şubat 2005.
- Leflore, D.**, 2000, *Theory supporting design guidelines for Web-based instruction*, *Instructional and Cognitive Impacts of Web-based Education*, B. Abbey (Ed.), Hershey , Pa: Idea Grup Publishing.
- Leidner, D.E. and Jarvenpa, S.L.**, 1995, *The use of information technology to enhance management school education : a theoretical view*, *MIS Quartly* (19:5), 265-291.
- Liliana P., Santacruz-Valencia, Ignacio A. and Carlos D.**, 2003, *Designing Learning Objects with the ELO-Tool*, *Proceedings of the The 3rd IEEE International Conference on Advanced Learning Technologies (ICALT'03)*, 272-273p.
- Liu J-H., Huang B-S. and Chao M.**, 2005, *The Design of Learning Object Authoring Tool Based on SCORM*, *Proceedings of the Fifth IEEE International Conference on Advanced Learning Technologies (ICALT'05)*, 778-782p.

KAYNAKLAR (devam)

- Malaxa, E. V.**, 2003, Customizable Learning Object Metadata Authoring, College Of Arts and Sciences, M.Sc. Thesis, Florida State University, 78p (unpublished).
http://etd.lib.fsu.edu/theses/available/etd-08282003-172442/unrestricted/01_evm_thesis.pdf , Son erişim tarihi: Ocak 2006.
- MARCOPOLO**, 2005, www.wcom.com/marcopolo, Son erişim tarihi: Ocak 2005.
- Merja Ranta-aho, M., Äijö, R. And Köykkä, M.**, 2004, User-Centered User Interface Design of Web-based Learning Environments, www.comlab.hut.fi/hft/IHM-HCI2001Workshop2%20.pdf, Son erişim tarihi: Temmuz 2004.
- Merlot**, 2007, <http://www.merlot.org> , Son erişim tarihi: Ağustos 2007.
- MOODLE**, 2007, <http://moodle.org> , Son erişim tarihi: Ağustos 2007.
- New Jersey Department of Education**, 1998, Educational Technology, New Jersey Science Curriculum Framework Chapter 6 Part B: Educational Technology, Son erişim tarihi: Ekim 2004,
<http://www.State.Nj.Us/Njded/Frameworks/Science/Chap6b.Pdf>
- Nilsson, M. and Palmér, M.**, 2002, ImseVimse: The IMS Editor Vimse. URL:<http://kmr.nada.kth.se/imsevimse>, Son erişim tarihi: Ocak 2006, <http://www.sourceforge.net/projects/imsevimse>

KAYNAKLAR (devam)

- Pareja-Flores, C. and Velázquez-Iturbide, J.Á.**, 2003, Program Execution and Visualization on the Web, 236-259, Web Based Education: learning from experience. Anil K. Aggarwal (Ed.), Hershey, PA, USA : Idea Group Inc.
- POOL**, 2005, www.eduplash.net, Son erişim tarihi: Ocak 2005.
- ReLOAD**, 2007, <http://www.reload.ac.uk>, Son erişim tarihi: Ağustos 2007.
- Robson, R.**, 2000, Learning Object Tutorial, Eduworks web site, <http://www.eduworks.com/LOTT/tutorial/index.html>, Son erişim tarihi: Şubat 2005.
- Sampson, D., Papaioannou, V. and Karadimitriou, P.**, 2002, Educational Technology & Society, 5(4). http://ifets.ieee.org/periodical/vol_4_2002/sampson_papaioannou.html, Erişim tarihi: Ocak 2006.
- SCHOOLNET**, 2007, www.schoolnet.ca, Son erişim tarihi: Ağustos 2007.
- Shepherd, C.**, 2000, Objects of Interest, <http://www.fastrak-consulting.co.uk/tactix/features/objects/objects.htm>, Son erişim tarihi: Aralık 2004.
- Silveira R. A., Gomes, E.R. and Vicari R.**, 2006, Intelligent Learning Objects: An Agent Approach to Create Interoperable Learning Objects, Education for the 21st Century Proceedings, 440, Chile, 411-415p.

KAYNAKLAR (devam)

- Singh, A., Boley, H., and Bhavsar, V.C.**, 2004, A learning object metadata generator applied to computer science terminology, Presentation in *Learning Objects Summit*, Fredericton, CA.
<http://www.cs.unb.ca/~boley/treesimilarity/LOMGen-29Mar2004.pdf> ,
Son erişim tarihi: Ocak 2006.
- SMETE**, 2007, <http://alpha.smete.org>, Son erişim tarihi: Ağustos 2007.
- Sönmezer, İ.B. ve Bingöl H.**, 2004, E-öğrenim için SCORM uyumlu elektronik içerik, Bilişim Teknolojileri Işığında Eğitim Konferansı ve Sergisi – BTIE 2004 Bildiri Kitabı, Ankara, 150s.
- Teknical's Virtual Campus**, 2005, Son erişim tarihi: Şubat 2005.
http://www.teknical.com/Products/virtual_campus.htm
- TILE**, 2007, <http://tile.atutor.ca/tile/servlet/search>, Son erişim tarihi: Ağustos, 2007.
- Technische Universität Darmstadt**, 2001, LOM Editor version 1.0, Multibook website <http://www.multibook.de/lom/en/index.html>,
Son erişim tarihi: Ağustos 2007.
- TheGateway**, 2007, <http://www.thegateway.org>, Son erişim tarihi: Ağustos 2007.
- TreeLOM**, 2006, <http://cebeciz.cukurova.edu.tr/tools/treelom>, Son erişim tarihi: Mart 2006
- vCard and vCalender**, 2007, <http://www.imc.org/pdi/>, Son erişim tarihi: Ağustos 2007.

KAYNAKLAR (devam)

- Verbert, K., Klerkx, J., Meire, M., Najjar, J. and Duval, E.,** 2004, Towards a Global Component Architecture for Learning Objects: an Ontology Based Approach, On The Move To Meaningful Internet Systems 2004: Otm 2004 Workshops, Proceedings Lecture Notes In Computer Science 3292: 713-722.
- WebCT,** 2005, <http://www.webct.com>, Son erişim tarihi: Ocak 2005.
- Webster,J. and Trevino,L.K.,** 1995, Rational and social theories as complementary explanations of communication media choices :two policy-capturing studies, Academy of Management Journal, 38(6), 1544-1572.
- WeLOAD,** 2007, <http://weload.lernnetz.de>, Son erişim tarihi: Ağustos 2007.
- Wenger, E.,** 1987, Artificial Intelligence and Tutoring Systems. Morgan Kaufmann, Los Altos, ABD.
- Wiley, David.A.,** 2000, Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy, 35s, The Instructional Use of Learning Objects, D.Wiley (Ed.), Online Book. <http://reusability.org/read/> , Son erişim tarihi: Aralık 2004
- Yalvaç, M. ve Bayraktutan, F.,** 2004, Elektronik Öğrenme Nesnelere Erişim ve Metadata, First International Conference on Innovations in Learning for the Future: e-learning, Istanbul, Turkey, Proceedings, 643-656.

KAYNAKLAR (devam)

Yaman, E. ve Hamedođlu, M. A., 2002, Bilgisayarlı Öğretim, Sakarya Üniversitesi II. Uluslararası Eğitim Teknolojileri Sempozyumu Bildirileri, http://www.ef.sakarya.edu.tr/sayfa/bildiri/index_b.htm

Zhigang Li, J., Nesbit, J.C. and Richards G., 2004, Crossing Boundaries with Web-Based Tools for Learning Object Evaluation, Advances in Web-Based Learning-ICWL 2004 Lecture Notes in Computer Science 3143: 286-292p.

ÖZGEÇMİŞ

Birim BALCI, 1975 yılında İstanbul Kadıköy’de doğdu. Lise öğrenimini İstanbul Kadıköy Anadolu Kız Meslek Lisesi Bilgisayar bölümünde tamamladı. 1994 yılında kazandığı Marmara Üniversitesi Teknik Eğitim Fakültesi Elektronik-Bilgisayar Eğitimi bölümünü 1999 yılında bitirdi. Marmara Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar-Kontrol Eğitimi Yüksek Lisans Programından 2001 bahar döneminde mezun oldu.

Lise eğitimi sırasında Yıldız Teknik Üniversitesi, Bilgisayar Mühendisliği Bölümü stajı, üniversite eğitimi sırasında Türk Telekom ve Almanya-Berlin Max-Delbrück Centrum stajlarından sonraki ilk iş deneyimini 1999 yılında Ericsson firması Bilgi İşlem Bölümünde tamamladı. Aynı yıl Marmara Üniversitesi Teknik Eğitim Fakültesi Bilgisayar Eğitimi Bölümünde ve bir yıl sonra ise Maltepe Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümünde araştırma görevlisi kadrosuna katıldı. Halen Maltepe Üniversitesinde öğretim görevlisi olarak çalışmaktadır.