



## e-Öğrenme standartlarında yeni yönelimler

Doç.Dr. Mehmet Emin MUTLU<sup>a</sup>

<sup>a</sup> Anadolu Üniversitesi, Açıköğretim Fakültesi, Eskişehir

### Özet

2000’li yılların başında tasarlanan ve bütün dünyada yaygın kabul gören SCORM modeli e-Öğrenme ders içeriklerinin paketlenmesi, öğrenme yönetim sistemlerine yüklenmesi ve öğrenenin derse ait bileşenlerdeki temel davranışlarının kaydedilmesinde bir standart haline gelmiştir. 2010’lu yıllara gelindiğinde öğrenme teknolojilerinde yaşanan gelişmeler ve öğrenenlerin öğrenme yönetim sistemleri dışındaki deneyimlerinin de önem kazanması SCORM modelinin yetersiz kalmasına neden olmuş ve yeni gereksinimleri karşılayacak mimarilerin geliştirilmesine başlanmıştır. Bu çalışmada, ADL kurumunun başlattığı ve SCORM sonrası gereksinimlere yanıt vermesi öngörülen “Yetiştirme ve Öğrenme Mimarisi”, “Sanal Dünya Çerçevesi” ve “Öğrenme için Kişisel Yardımcı” projeleri incelenecek ve e-Öğrenmenin 2020’li yıllardaki yapısı tartışılacaktır. Çalışmada ayrıca, “Yetiştirme ve Öğrenme Mimarisi” nin bir parçası olan ve deneyim izleme amacıyla geliştirilen “Experience API” ve “Öğrenme Kayıtları Deposu” yapılarının ayrıntıları incelenerek, bu alanda uygulama geliştirecek olanlara yol gösterilmiştir.

**Anahtar Sözcükler:** e-Öğrenme standartları, SCORM, Yetiştirme ve öğrenme mimarisi, Deneyim izleme, Experience API, LRS, Öğrenen profilleri, Yetkinlik ağları, İçerik aracılığı, Sanal dünya çerçevesi, Öğrenme için kişisel yardımcı.

### Abstract

SCORM model, which is design in the beginning of 2000s and is accepted worldwide, is become a standard for wrapping e-learning course contents, uploading them to the learning management systems, and recording fundamental behaviors of the learner in course components. When it comes to 2010s, developments in learning technologies and increasing importance of experiences of the learners out of the learning management systems caused SCORM model to fall behind the expectations and new architecture which can satisfy the new necessities has been developed. In this study, projects, which are expected to meet the needs after SCORM, started by ADL named as “Training and Learning Architecture”, “Virtual World Framework” and “Personal Assistant for Learning” are examined and the structure of e-learning in 2020s will be discussed. Also, in this study, details of configurations such as “Experience API” which is developed for following the experiences and a part of “Training and Learning Architecture” and “Learning Record Store” will be examined in details, in order to guide prospective application developers in this field.

**Keywords:** e-Learning standarts, SCORM, Training and learning architecture, Experience tracking, Experience API, Learner profiles, Competency networks, Content brokering, Virtual world framework, Personal assistant for learning

## Giriş

SCORM (Sharable Content Object Reference Model) öğrenme yönetim sistemlerine (ÖYS) yüklenecek e-Öğrenme içeriğinin paketlenmesi için genel kabul görmüş bir e-Öğrenme paketleme standardıdır. 90'lı yıllarda internetin öğrenme amacıyla kullanımına başlanması ve bunun sonucunda şekillenen e-öğrenme süreçlerinde, ders içeriklerinin taşınabilirliği, farklı öğrenme yönetim sistemlerine kurulabilmesi, öğrenenin ders kapsamındaki etkinliklerinin ve başarı durumunun izlenebilmesi amacıyla, 1999'da ABD Savunma Bakanlığının bir kuruluşu olan Advanced Distributed Learning (ADL) başkanlığında ilgili kurumlar biraraya gelerek, SCORM girişimini başlatmışlardır. Çalışmaların sonucunda Ocak 2001'de SCORM 1.1, Ekim 2001'de SCORM 1.2, Ocak 2004'de SCORM 2004 Edition 1, Temmuz 2004'de SCORM 2004 Edition 2, Ekim 2006'da SCORM 2004 Edition 3 ve son olarak da Mart 2009'da SCORM 2004 Edition 4 yayınlanmıştır (ADL, 2015g). SCORM standardı, ADL'nin AICC (Aviation Industry Computer-Based Training Committee), IMS Global (IMS Global Learning Consortium), IEEE (Institute of Electrical and Electronics Engineers), ve Ariadne Vakfı kurumlarıyla yaptığı işbirliği ile, bu kurumlara ait CMI Data Model, LOM Metadata, AICC Packaging ve IMS Simple Sequencing gibi yaklaşımların biraraya getirilmesiyle geliştirilmiş ve dünya çapında yaygın kabul görmüştür (Hruska ve Johnson, 2012; Poltrack, Hruska, Johnson, ve Haag, 2012). e-Öğrenme içeriği tasarlamak ve geliştirmek amacıyla kullanılan Adobe Captivate, Articulate Presenter, iSpring Presenter gibi başlıca yazarlık sistemleriyle tasarlanan içerikler SCORM paketi olarak kaydedebilmektedir (ADL, 2015d). Bağımsızca içerik geliştirenler XML tabanlı paylaşılabilir içerik nesnelere oluşturmak amacıyla ADL'nin yayınladığı Reload Content Editor'ü kullanabilirler. Ayrıca Blackboard, Moodle, Desire2Learn ve Canvas gibi yaygın olarak kullanılan öğrenme yönetim sistemleri de e-öğrenme içeriklerinin SCORM paketleri halinde yüklenmesine olanak sağlamaktadırlar (SoftwareInsider, 2015). SCORM kapsamında e-Öğrenme içeriğinin paketlenmesi ve taşınmasını sağlayan kuralların varlığının yanısıra, öğrencinin dersi tamamlama durumunu, tamamlama süresi, başarı durumunu ve başarı notunu ÖYS içerisinde izleme olanağını sağlayan kurallar da bulunmaktadır.

SCORM artık olgunluğa erişmiş ve 2000'lerin ortalarından itibaren yaygınlaşan yeni teknolojik dalgalara ve öğrenme yaklaşımlarına uyum sağlamamaya başlamıştır. Bu teknolojik dalgalarda ve öğrenme yaklaşımları arasında mobil teknolojiler ve yaygın kablosuz erişim, sosyal ağlar, ciddi oyunlar, açık kaynak hareketi, sanal dünyalar ve artırılmış gerçeklik, web teknolojilerindeki gelişmeler, biçimsel olmayan öğrenme, kişisel öğrenme ortamları ve ağları,

takım halinde öğrenme ve grupla öğrenme bulunmaktadır (Hruska ve Johnson, 2012). ADL tarafından 2010'lu yıllarda “yeni nesil öğrenme ortamı” oluşturmak için Yetiştirme ve Öğrenme Mimarisi (Training and Learning Architecture - TLA), Sanal Dünya Çerçevesi (Virtual World Framework - VWF) ve Öğrenme için Kişisel Yardımcı (Personal Assistant for Learning - PAL) araştırma projeleri başlatılmıştır (Regan, 2013; Durlach ve Johnson, 2014). Anadolu Üniversitesinde 4-6 Şubat 2015 tarihleri arasında düzenlenen Akademik Bilişim'15 konferansında “Açık ve Uzaktan Öğrenmede Yeni Teknolojilerin Kullanımı” isimli panelde bildiri olarak sunulan bu çalışmanın izleyen bölümlerinde ADL'nin yeni projeleri incelenecektir. Çalışmanın sonuç bölümünde incelenen e-Öğrenme standartlarının yaygın olarak kabul görmesi durumunda 2020'li yıllarda e-öğrenmenin sahip olacağı özellikler tartışılacaktır.

### **Yetiştirme ve Öğrenme Mimarisi (Training and Learning Architecture - TLA)**

TLA yetiştirme ve öğrenme için zengin ortam oluşturmak amacıyla tasarlanmış açık kaynak yazılımlar ve standartlaştırılmış bir dizi web hizmetlerini kapsamaktadır. TLA ile ADL'nin yeni nesil öğrenme ortamı araştırma ve geliştirme stratejisini desteklemek ve gelecekteki Kişisel Öğrenme Yardımcısının (PAL) temellerini oluşturmak amaçlanmaktadır. SCORM gibi TLA da kendi başına bir sistem değil, öğrenme için sistemler ve uygulamaların geliştirilmesine olanak sağlayan bir mimaridir. Bu özelliği ile TLA SCORM'un yerine geçen değil, onu kapsayan bir üst küme olarak tasarlanmaktadır. SCORM sadece ÖYS içerisindeki öğrenme deneyimlerini basitçe tamamlanma ve başarı durumu boyutuyla izleyebilirken TLA'da cihaz bağımsız modern öğrenme senaryolarına olanak sağlanması amaçlanmaktadır. TLA “Deneyim İzleme”, “İçerik Aracılığı”, “Öğrenen Profilleri” ve “Yetkinlik Ağları” olmak üzere dört bileşenden oluşmaktadır. Her TLA bileşeninde geriye uyumluluk ve SCORM'dan kolay geçiş için en iyi uygulamalara yer verilmektedir. TLA kapsamında “Deneyim İzleme” bileşeni “Tin Can Projesi” sonucunda hayata geçirilmiş ve “İçerik Aracılığı” bileşeni kapsamında bir dizi açık kaynak uygulama geliştirilmiş iken diğer bileşenler üzerinde çalışmalar devam etmektedir (ADL, 2015e; Durlach ve Johnson, 2014).

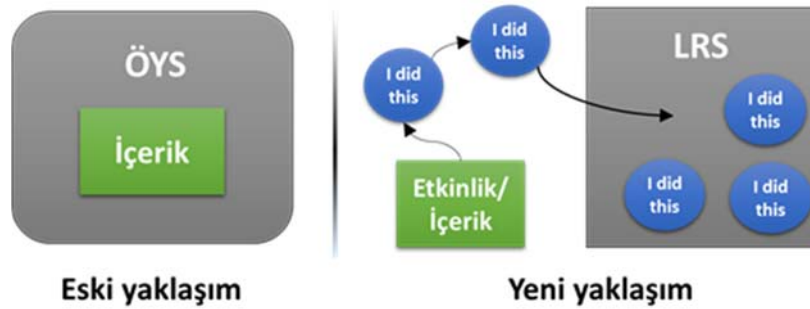
### **Deneyim İzleme: Experience API ve Öğrenme Kayıtları Deposu (LRS)**

“Deneyim İzleme” bileşeni, öğrenenlerin öğrenme deneyimleri esnasındaki etkileşimlerinin izlenmesi ve bir veritabanına kaydedilmesi amacıyla kullanılan kurallar ve yazılımları kapsamaktadır. Öğrenme deneyimleri biçimsel öğrenme kapsamındaki derslerden

olduğu kadar web siteleri, videolar, oyunlar ve sosyal medya etkileşimleri gibi biçimsel olmayan öğrenme bağlamlarından da derlenebilmektedir. Bu bileşenin önemli bir özelliği kaydedilen verinin diğer yetkili sistemlerin kullanımına da açık olmasına olanak sağlamasıdır.

Deneyim izleme projesinde iki alt bileşen bulunmaktadır: Experience API (xAPI) olarak adlandırılan web hizmeti tanımlamaları ve Learning Record Store (LRS) olarak adlandırılan açık kaynak kodlu yazılım bileşeni. API terimi Application Programming Interface ifadesinin kısaltmasıdır ve günümüzde yazılım geliştirme ortamlarında yararlanılan hazır yazılım kitaplıklarını tanımlamak için kullanılır. API'ler bir programlama diliyle kullanılabilen belirli bir alana yönelik nesnelere, yöntemler, özellikler, alanlar, sabitler ve yapıları barındıran hazır yazılım kitaplıklarıdır (ADL, 2015e; Poltrack vd., 2012).

SCORM döneminde içerik bir ÖYS içerisinde tutulmakta ve öğrenenin bu içerik üzerindeki etkinlikleri izlenmekteyken, SCORM sonrası dönemde öğrenenin herhangi bir içerik üzerindeki etkinliği yakalanarak bir öğrenme kayıtları deposunda saklanabilmektedir (Grafik 1). “Deneyim izleme” sürecinin ayrıntılarına çalışmanın sonraki bölümlerinde yer verilecektir.



Grafik 1. Experience API Öncesi ve Sonrası (Rustici Software, 2015a)

### İçerik Aracılığı

“İçerik Aracılığı” bileşeni anında (just-in time) öğrenmeyi desteklemek ve bir sonraki mantıksal etkinliğin seçimine olanak sağlamak amacıyla içeriğin yönetilmesi, araştırılması ve anlaşılması için geliştirilmiş kurallar ve yazılımları kapsamaktadır.

Bir öğrenme içeriğinin modüller halinde tasarlandığı, her modülde belirli sayıda problemin yer aldığı ve öğrenenin problemi çözme davranışlarına yönelik yönlendirici geri beslemelerin bulunduğu varsayılabilir. Bu modelde öğrenenin her davranışında problemin tamamlanıp tamamlanmadığı denetlenir, problem tamamlanmadıysa öğrenene geri besleme

desteđi verilir. Eđer problem tamamlandıysa bir mikro dizilimden yararlanarak izleyen probleme geçilir. Eđer öğrenen yeterince problem çözümlenirse modülde ustalaştıysa bir makro dizilimden yararlanarak bir sonraki modüle geçilir. Bu modeldeki mikro ve makro dizilimlerin, öğrenme amaçlarına karşı gelecek şekilde oluşturulmasını, isimlendirilmesini ve depolanmasını sağlayacak genel kabul görmüş standartların varlığı içeriklerin birer akıllı içerik haline gelmesini sağlayacaktır.

Yukarıda basitleştirilerek tanıtılan yapıyı geliştirmeyi amaçlayan içerik aracılığı projesi üç açık kaynak kodlu yazılım bileşeni ile bir üstveri standartlaştırma girişimini içermektedir (Regan, 2013; Durlach ve Johnson, 2014):

### **Açık kaynak kodlu yazılım bileşenleri**

İçerik Aracılığı bileşeninde RUSSEL, Learning Registry ve 3D-Repository açık kaynak yazılımları bulunmaktadır:

E-Öğrenme için Tekrar Kullanılabilirlik Destek Sistemi (Re-Usability Support System for E-Learning - RUSSEL): RUSSEL dijital öğrenme içeriđi, ilgili üstveriler ve SCORM paketlerinin yönetimi amacıyla geliştirilmiş tamamen işlevsel bir dijital kütüphane sistemidir. RUSSEL'in dijital kütüphane sistemi Learning Registry ve 3D-Repository ile bütünleşik çalışabilmektedir (RUSSEL, 2015).

Öğrenme Tescili (Learning Registry): Öğrenme Tescili eğitimciler ve öğrencilerin giderek genişleyen dijital evrendeki zengin dijital içeriđe erişmelerini kolaylaştırmayı amaçlayan, çevrimiçi kullanılabilir öğrenme kaynaklarına ait verinin yakalanması, birbiriyle bağlanması ve paylaşılması için yeni bir yaklaşım getirmektedir. Böylece yayıncılar öğrenme kaynaklarına ait tanımlamalarını öğrenme amaçlarıyla ilişkilendirerek Learning Registry sunucularına yükleyebilir, öğrenci ve öğretmenler bu kaynakları sorgulayabilirler, Learning Registry izleyen bölümde incelenen LRMI üstveri yapısını kullanmaktadır (LearningRegistry, 2015).

3B Deposu (3D-Repository): ADL'nin 3B Deposu 3 boyutlu modellerin yüklenmesi, aranması ve indirilmesi için kullanılabilir bir web sitesidir (3DR, 2015).

### **Öğrenme kaynakları üstverisi girişimi**

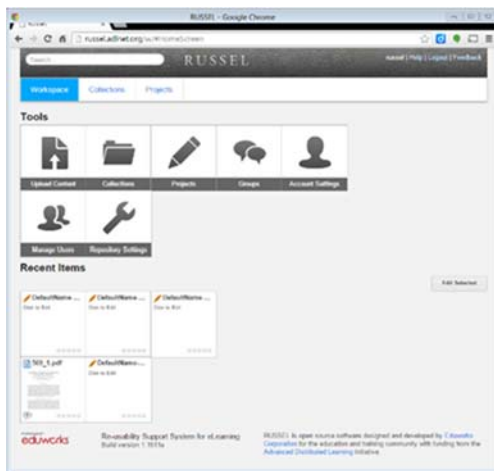
Öğrenme Kaynakları Üstverisi Girişimi (Learning Resources Metadata Initiative - LRMI) web ortamındaki öğrenme kaynaklarının tanımlanması veya etiketlenmesi için geliştirilmiş ortak bir üst veri çerçevesidir. Bu çerçeveye öğrenme kaynađının künyesi, türü, formatı, kullanım biçimi, hedef kitlesi, karşı gelen öğrenme standartları, içeriđi tanımlayan etiketler, çalışma süresi, etkileşim türü vb. çok sayıda özelliđin etiketlenmesi için standart

yapılar tanımlanarak eğitimciler ve öğrencilerin daha zengin ve verimli bir arama deneyimi sahip olmaları amaçlanmaktadır (LRMI, 2015). LRMI 1.1 tanımlaması 23.10.2014’de RDF biçiminde Schema.org’da yayınlanmıştır (DublinCore, 2015).

LRMI standartları oluşturulurken diğer öğrenme kaynağı üst verisi oluşturmayla ilgili projelerden yararlanılmaktadır:

- IEEE Learning Object Metadata
- Dublin Core Metadata Initiative
- IMS Global Learning Consortium Learning Resource Metadata
- ARIADNE Educational Metadata Recommendation
- SCORM
- ISO/IEC Metadata for Learning Resources

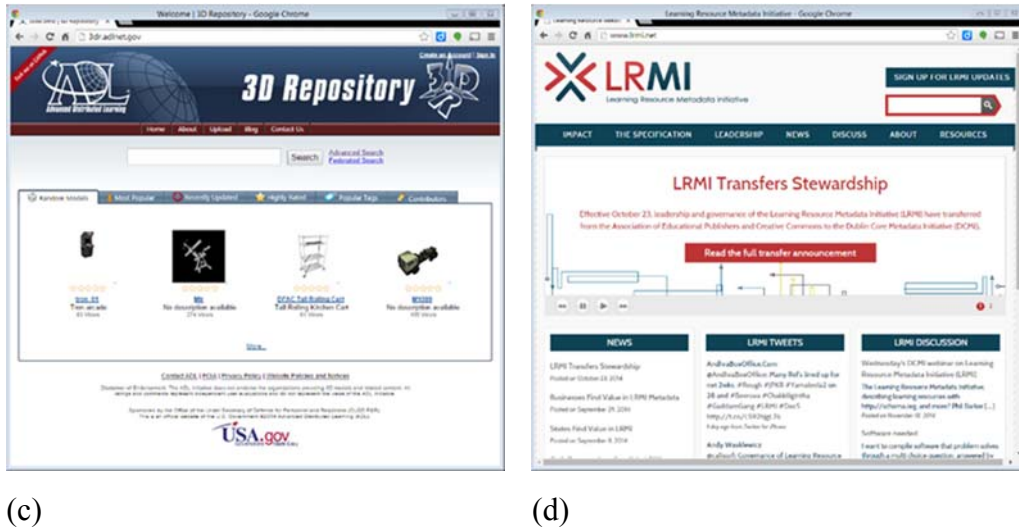
ADL’nin “İçerik Aracılığı” projesinde içeriğin yönetimi için RUSSEL ile 3D Repository’den yararlanılacak; içeriğin araştırılması için Learning Registry kullanılacak; içeriğin anlaşılmasını ise LRMI üstlenecektir.



(a)



(b)



Grafik 2. (a) RUSSEL, (b) Learning Registry, (c) 3D-Repository, (d) LRMI siteleri

RUSSEL, LearningRegistry ve 3D-Repository için açık kaynak yazılımları geliştirilmiş ve yayınlanmıştır. Diğer bir deyişle araştırmacılar bu yazılımları indirerek kendilerine ait RUSSEL (GitHub, 2015d), Learning Registry (GitHub, 2015g) ve 3D-Repository (GitHub, 2015c) sunucularını kurabilirler, hatta kendi sunucularını geliştirebilirler.

### Yetkinlik Ağları ve Öğrenen Profilleri

TLA'nın "Yetkinlik Ağları" bileşeni e-öğrenme uygulamaları tarafından kullanılacak olan öğrenme amaçları ve yetkinlik tanımları ile bunlar arasında uygun ilişkileri kuracak sistemlerin tasarlanmasını sağlayacak kurallar ve yazılımları kapsamaktadır. Bu ilişkiler içerik/dersler ile amaçlar/yetkinlikleri eşleştirebilmeyi sağlayacaklardır (Durlach ve Johnson, 2014).

ADL bu projede aşağıdaki kurumların sahip olduğu yetkinlik çerçeveleri deneyimlerinden yararlanmayı öngörmektedir (Regan, 2013):

- MedBiquitous Competency Framework: Sağlık eğitimi alanındaki yetkinlik çerçevelerinin gösterimi amacıyla ANSI tarafından akredite edilmiş teknik standart.
- Achievement Standards Network (ASN): K-12 standartlarıyla birlikte kullanılan açık tanımlama.
- Integrating Learning Outcomes and Competences (inLOC): 2013 yılında tamamlanmış bir Avrupa Birliği projesi.

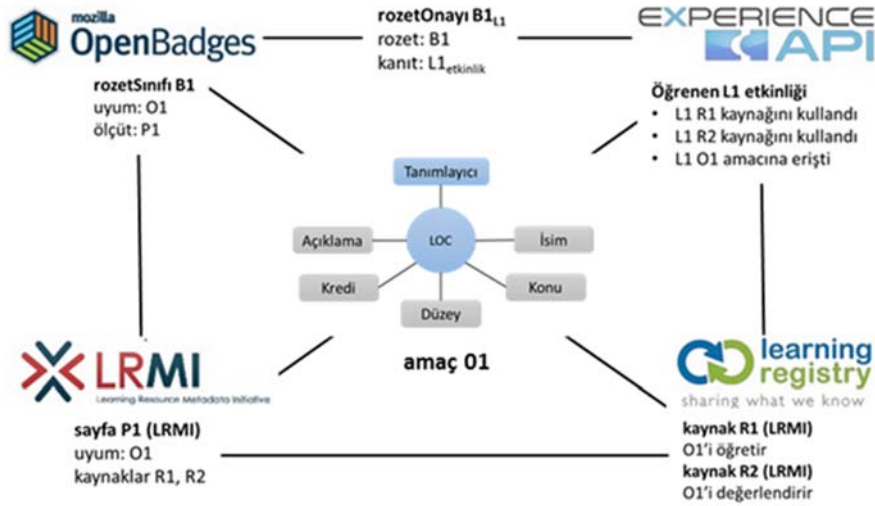


TLA'nın "Öğrenen Profilleri" bileşeni, öğrenen tercihleri ve ders geçmişi gibi öğrenene ait veriye erişmek amacıyla kullanılan kurallar ve yazılımları kapsamaktadır. Profilde bulunan enformasyon gelecekte bireyin belirli bir konuda en uygun stildeki öğrenme deneyiminin tasarlanması amacıyla kullanılabilir. Öğrenen profilleri için yetkinlikleri tanımlayan yapılardan yararlanılacaktır. Örneğin öğrenen için bir yetkinliğe ait ayrıntılar üzerinde "yeterli", "devam etmekte", "planlanan" gibi etiketler kullanılarak, o yetkinliğe ne ölçüde sahip olduğu izlenebilir. Model, bir öğrenenin öğrenci, çalışan, kişisel merak, sosyal ilişkiler vb. sahip olduğu birden fazla role ait profil verilerini işleyebilecektir. Öğrenen kendi durumunu "görselleştiriciler" yardımıyla grubun bütünüyle karşılaştırarak değerlendirebilecektir. ADL öğrenen profilleri bileşenini şekillendirmek için Mozilla OpenBadges projesini izlemeye almıştır (Regan, 2013).

Deneyim izleme bileşeni ve içerik aracılığı için kullanılabilir tanımlamalar ve paylaşılabilir kodlar geliştirilmişken öğrenen profilleri ve yetkinlik ağları bileşenleri 2015 yılından henüz düşünce ve taslak düzeyinde bulunmaktadır.

### Resmin Bütünü

TLA mimarisi tamamlandığında mimariyi oluşturan bileşenler arasında Grafik 3'te özetlenen ilişkiler kurulabilecektir:



Grafik 3. TLA'daki Veri İlişkileri (Regan, 2013)

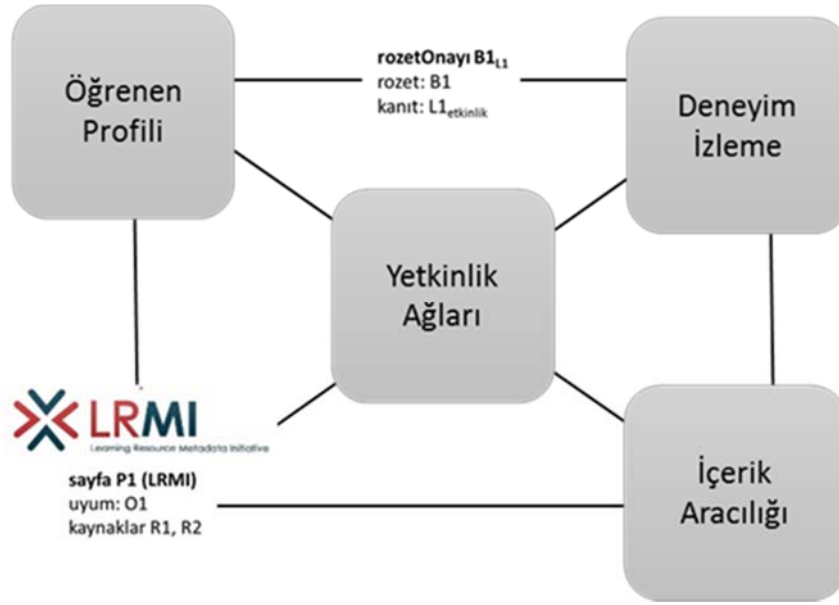


Grafik 3'teki iş akışına göre; öğrenme amaçlarının tutulduğu bir havuzda O1 amacı tanımlayıcısı, adı, açıklaması, konusu, kredisi ve düzeyi ile ayrıntılı olarak bir LOC (Learning Outcome and Competency) nesnesi olarak tanımlanmıştır. O1 amacına erişmek için R1 ve R2 kaynaklarının kullanılabilceği LearningRegistry'deki bilgi tabanında üst veri halinde tutulmaktadır. Buna göre R1 kaynağı O1'i öğretmekte, R2 kaynağı ise O1'i değerlendirmektedir.

- L1 isimli öğrenen, R1 ve R2 kaynaklarını kullanarak O1 amacına erişebileceğine dair veriyi LearningRegistry'den edinir.
- Öğrenen L1, R1 ve R2'yi kullanarak O1 amacına erişir ve bununla ilgili olarak gerçekleşen “L1 R1'i kullandı”, “L1 R2'yi kullandı”, “L1 O1 amacına erişti” etkinliklerini Experience API ile LRS'ye kaydeder.
- Öğrenen gerçekleştirmiş olduğu L1 etkinliklerini kanıt olarak kullanarak OpenBadge'e B1 rozetine sahip olması gerektiği savında bulunur.
- OpenBadge L1'e ait etkinlikleri kanıt olarak değerlendirerek LRMI'dan O1 amacı ve R1, R2 kaynaklarının hangi öğrenme standartlarıyla uyduğunu sorgular ve öğrenene P1 ölçütüne göre O1 yetkinliğine sahip olduğunu belirten B1L1 rozetini kullanma hakkını verir.

Görüldüğü gibi öğrenme verileri yetkinlik tanımlamaları etrafında dönerek hareket etmektedir.

TLA bu iş akışında öğrenenin sahip olduğu rozetleri “Öğrenen Profili”, amaçların havuzunu “Yetkinlik Ağları”, içeriğin amaçlarla ilişkilendirilmesini “İçerik Aracılığı”, öğrenenin kaynakları kullanarak amaçlara eriştiği verisini ise “Deneyim İzleme” bileşeni ile izlemeyi amaçlamaktadır (Grafik 4).



Grafik 4. TLA Hizmetlerinin Birbirine Bağlanması (Regan, 2013)

### Experience API

Bu bölümde ADL'nin TLA projesinin en fazla ilerleme kaydedilmiş aşaması olan "Deneyim İzleme" bileşenine ait ayrıntılara yer verilecektir.

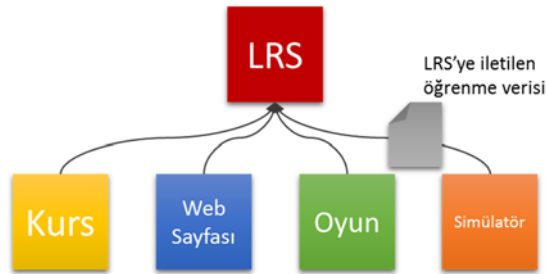
"Deneyim İzleme" bileşeni kapsamında Experience API isimli web hizmeti tanımlamaları ile "Öğrenme Kayıtları Deposu" isimli açık kaynak kodlu bir yazılım yayınlanmıştır. Açık kaynak topluluğu tarafından geliştirilen Experience API biçimsel olmayan öğrenme, gerçek dünya, sanal dünya, mobil ve oyun deneyimlerini izleyebilme ve izleme verilerini bir öğrenme kayıtları deposuna kaydedebilme olanağı sağlamaktadır. Experience API'nin 2010 yılında başlayan geliştirme sürecinin tarihçesinde aşağıdaki aşamalar bulunmaktadır (GitHub, 2015f):

- Ekim 2010 - Yeni nesil SCORM yaratılması için bir çalışmanın başlatılması amacıyla topluluğun görüşlerinin derlenmesi ve araştırma yapılmasına yönelik proje Rustici Software firmasına Tin Can Projesi adıyla ihale edildi.
- Eylül 2011 - Yeni nesil SCORM için Tin Can API adıyla başlangıç taslağı yayınlandı.
- Mart 2012 - Tin Can API'nin .9 sürümü yayınlandı.
- Ağustos 2012 - Tin Can API'nin .95 sürümü yayınlandı.
- Nisan 2013 - Tin Can API'nin 1.0.0 sürümü yayınlandı. (Projenin adı "Experience API" - xAPI olarak değiştirildi.)
- Ekim 2013 - Experience API'nin 1.0.1 sürümü yayınlandı.

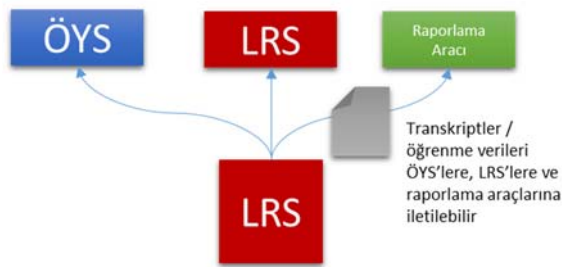
## Öğrenme Kaydı Deposu (LRS)

Experience API ile öğrenene ait deneyim verisi bir Öğrenme Kaydı Deposuna kaydedilebilir ve kaydedilmiş deneyimler sorgulanabilir. Experience API deneyim verisinin nasıl tanımlanacağını, kaydedileceğini ve sorgulanabileceğine ait tanımlamalar içermektedir. Bir öğrenme kaydı deposu Experience API'nin bu tanımlamalarına uygun olacak biçimde geliştirilebilir. ADL kullanıcılara ve geliştiricilere örnek olacak bir açık kaynak LRS geliştirerek yayınlamıştır (GitHub, 2015b).

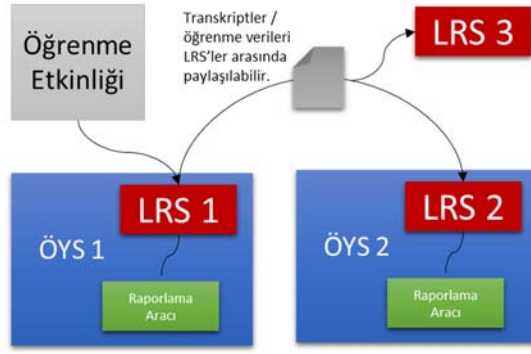
Experience API ile bir LRS'ye öğrenenin kurs, web sayfası, oyun ya da benzetim gibi değişik ortamlarda gerçekleştirdiği öğrenme deneyimlerine ait öğrenme verisi yazılabilir (Grafik 5a). LRS içeriğinde bulunan öğrenene ait verileri öğrenenin ilişkili olduğu değişik ÖYS'lere, LRS'lere ya da raporlama araçlarına iletebilir (Grafik 5b). LRS'ler ayrıca ÖYS'lerin içerisinde de bulunabilirler. Böylece öğrenenin hem ÖYS içinde hem de ÖYS dışındaki öğrenme etkinliklerini kaydedilebilir. Bu etkinlik verileri ÖYS içerisinde raporlanabileceği gibi diğer bir ÖYS içerisindeki ya da bağımsız bir LRS'ye aktarılarak orada raporlanabilir (Grafik 5c).



(a) LRS'ye Girdi Sağlayanlar.



(b) LRS'den Veri Alanlar.



(c) LRS ve ÖYS İlişkisi.

Grafik 5. LRS'nin Yapısı (TinCanAPI, 2015c).

### Experience API'nin Sağladığı Olanaklar

Experience API'nin sağladığı olanaklar beş grupta toplanabilir (TinCanAPI, 2015b).

Experience API izleme seçenekleri sınırsızdır: Experience API ile mobil öğrenme, ciddi oyunlar, benzetimler, biçimsel olmayan öğrenme ve gerçek dünya performansı gibi ortamlardaki deneyimler izlenebilir.

Experience API her yerde çalışır: Experience API bağlı olmayan ya da ara sıra bağlanan ortamlarda, herhangi bir sistemde (örneğin akıllı telefon, denizaltı sonar sistemi), herhangi bir sunucuda (kendi içeriğinizin olduğu yerde bir LRS barındırılabilir), web tarayıcısı dışında (örneğin, iPhone uygulaması, f-16 uçuş simülatörü) ve ÖYS dışındaki içeriğin bulunduğu ortamda çalıştırabilir.

Experience API SCORM ile kolay ya da mümkün olmayan şeyler yapabilir: İçeriğin dağıtımını ve kullanıcı deneyimi üzerindeki bütün denetimi düzenleyebilir, ayrık içerikler üzerinde öğrenenin serbestçe dolanımını izleyebilir, kullanıcıları korsanlardan koruma amaçlı güvenlik desteği sunabilir.

Experience API ile daha zengin öğrenme deneyimleri tasarlanabilir: Etkileşimli ve uyarlamalı öğrenme deneyimleri, çoklu kipli öğrenme deneyimleri, harmanlanmış öğrenme deneyimleri ve uzun zamanlı dönemlere dağılmış öğrenme deneyimleri izlenebilir.

SCORM ile mümkün olmayan raporlar elde edilebilir: Takım tabanlı yetiştirme, bir ders için ön test ve son test gibi çoklu puanlama, birden fazla girişi denetleme, ayrıntılı test sonuçları ve daha fazlası raporlanabilir.

## Experience API Deyimleri

Experience API kısaca “Javascript object notation (JSON)” ile oluşturulmuş deyimlerin iletilmesini sağlayan bir “Representational state transfer (REST)” web hizmetidir. Deyimler Facebook, Microsoft, Google gibi firmalar tarafından sosyal ağ akışlarında kullanılan “Activity Streams” Modeli kullanılarak oluşturulmuştur (ActivityStreams, 2015). Basit deyimler <Actor Verb Object> terimleri kullanılarak <I did this> yapısında oluşturulabilir. Bir deyim oluşturulurken bu üç terimin kullanımı zorunludur. Basit deneyim deyimlerinin örnek sözel ifadeleri:

- Mehmet **completed** Ders101
- Ayşe **attended** Meeting10
- Ali **experienced** “http://inet-tr.org.tr/inetconf19”

Bu deyimlerin bir LRS tarafından anlaşılabilmesi için JSON notasyonu ile oluşturulması gerekmektedir (GitHub, 2015f). Grafik 6’da sözel olarak ifade edilmiş “Ayşe Peri Mutlu **deneyimledi** 'http://www.anadolu.edu.tr/'” deneyime ait deyim JSON notasyonu ile oluşturulmuş hali verilmiştir.

```
{
  "id": "71e4874e-d884-4e42-a43a-035bddd8b238",
  "actor": {
    "name": "Ayşe Peri Mutlu",
    "mbox": "mailto:aperi@anadolu.edu.tr",
    "objectType": "Agent"
  },
  "verb": {
    "id": "http://adlnet.gov/expapi/verbs/experienced",
    "display": {
      "en-US": "deneyimledi"
    }
  },
  "timestamp": "2014-10-09T14:38:38.263Z",
  "stored": "2014-10-09T14:38:38.263Z",
  "authority": {
    "name": "Mehmet Emin Mutlu",
    "mbox": "mailto:memutlu@anadolu.edu.tr",
    "objectType": "Agent"
  },
  "version": "1.0.1",
  "object": {
    "id": "http://www.anadolu.edu.tr/",
    "objectType": "Activity"
  }
}
```

Grafik 6. Örnek Bir Deneyime Ait JSON Gösterimi

## Karmaşık deyimler

Kullanımı zorunlu olan Actor, Verb ve Object terimlerinin dışında kullanımı tercihe bağlı olan ID, Result, Context, Timestamp, Store, Authority, Version ve Attachements terimleri de bulunmaktadır. Böylece en kapsamlı bir deneyim deyimini 11 adet <ID Actor Verb Object Result Context Timestamp Store Authority Version Attachements> terimini barındırır. Karmaşık deyimler Grafik 7’deki “Deyim Üretici” ile kolayca oluşturulabilir.

The screenshot shows the Tin Can Statement Generator & Validator web application. The form includes the following fields and options:

- Actor Email:**
- Name:**
- Preferred Verbs:**
- Verb ID:**
- Verb Display:**
- Activity ID:**
- Language:**
- Activity Name:**
- Activity Description:**
- Endpoint:**
- Username:**
- Password:**

Buttons: 1. Generate JSON, 2. Validate JSON, 3. Post JSON, 4. View Public URI

Your Tin Can API Statement:

```
{
  "Actor": {
    "Email": "mehmet@webduzleri.com",
    "Name": "Mehmet Emre Yildiz",
    "Preferred Verbs": "experience"
  },
  "Verb": {
    "ID": "http://www.experienceapi.com/experience",
    "Display": "experience"
  },
  "Activity": {
    "ID": "http://www.webduzleri.com",
    "Language": "en-US",
    "Name": "Experience Activity",
    "Description": "Experience Activity Object"
  },
  "Endpoint": "http://www.experienceapi.com/experienceapi/TinCanAPI/submit",
  "Username": "Tuz",
  "Password": "*****"
}
```

Grafik 7. Experience API Deyimleri Oluşturucusu (TinCanAPI, 2015e).

## Fiiller ve Nesnelere

Experience API’de kullanılan Fiiller ve Nesnelere önceden belirlenmiş ve ADL’nin sitesinde birer URI olarak JSON örnekleriyle yayınlanmıştır. Kullanıcılar sadece bu fiilleri ve nesnelere kullanabilirler. Topluluk yeni bir fiil ya da nesneyi Experience API komisyonuna önerebilir ve kabul edildiğinde listeye eklenir (Tablo 1).

Tablo 1.

*Experience API Fiilleri ve Nesneleri (ADL, 2015b; ADL 2015c).*

Fiiller	Nesneler
<ul style="list-style-type: none"> <li>• Answered (yanıtladı)</li> <li>• Asked (sordu)</li> <li>• Attempted (girişimde bulundu)</li> <li>• Attended (katıldı)</li> <li>• Commented (yorumladı)</li> <li>• Completed (tamamladı)</li> <li>• Exited (ayrıldı)</li> <li>• Experienced (deneyimledi)</li> <li>• Failed (başarısız oldu)</li> <li>• Imported (içe aktardı)</li> <li>• Initialized (başlattı)</li> <li>• Interacted (etkileşimde bulundu)</li> <li>• Launched (çalıştırdı)</li> <li>• Mastered (ustalaştı)</li> <li>• Passed (geçti)</li> <li>• Preferred (tercih etti)</li> <li>• Progressed (ilerledi)</li> <li>• Registered (kayıt oldu)</li> <li>• Responded (tepki verdi)</li> <li>• Resumed (yeniden başladı)</li> <li>• Scored (puan aldı)</li> <li>• Shared (paylaştı)</li> <li>• Suspended (durdurdu)</li> <li>• Terminated (sonlandırdı)</li> <li>• Voided (iptal etti)</li> </ul>	<ul style="list-style-type: none"> <li>• Assessment (değerlendirme)</li> <li>• Course (kurs)</li> <li>• File (dosya)</li> <li>• Interaction (etkileşim)</li> <li>• Lesson (ders)</li> <li>• Link (bağlantı)</li> <li>• Media (medya)</li> <li>• Meeting (toplantı)</li> <li>• Module (modül)</li> <li>• Objective (amaç)</li> <li>• Performance (başarım)</li> <li>• Question (soru)</li> <li>• Simulation (benzetim)</li> </ul>

Kullanıcılar asılları İngilizce olan bu fiil ve nesneleri kullanırken kendi dillerindeki karşılıklarını da belirterek raporları anlaşılır hale getirebilirler.

### **Açık Kaynak Kitaplıklar**

Rustici Software firması Experience API'nin değişik programlama dilleriyle kullanılabilmesi amacıyla Javascript (TinCanJS), .NET, Objective C, PHP, Python, Java kitaplıklarını açık kaynak olarak yayınlamıştır (GitHub, 2015i).



TinCan API kitaplığını kullanarak bir deneyim kaydeden C# ile geliştirilmiş bir kod örneği Grafik 8’de verilmiştir. Bu örnekte bir LRS’ye “Mehmet Emin” isimli öğrenenin GitHub sitesindeki TinCan.Net isimli bir yazılımı tamamlamış olduğu verisi gönderilmektedir.

```
using System;
using TinCan;
using TinCan.LRSResponses;

var lrs = new RemoteLRS(
    "https://cloud.scorm.com/tc/JMUW510QI5/",
    "memutlu@anadolu.edu.tr",
    "██████"
);

var actor = new Agent();
actor.name = "Mehmet Emin";
actor.mbox = "mailto:info@tincanapi.com";

var verb = new Verb();
verb.id = new Uri ("http://adlnet.gov/expapi/verbs/completed");
verb.display = new LanguageMap();
verb.display.Add("en-US", "completed");

var activity = new Activity();
activity.id = new Uri ("http://rusticisoftware.github.io/TinCan.NET");

var statement = new Statement();
statement.actor = actor;
statement.verb = verb;
statement.target = activity;
statement.timestamp = DateTime.ParseExact("2015-08-29 07:42:10Z", "u",
    System.Globalization.CultureInfo.InvariantCulture);

StatementLRSResponse lrsResponse = lrs.SaveStatement(statement);
if (lrsResponse.success)
{
    // Updated 'statement' here, now with id
    respondTextBox.Text = "Respond: " + "Save statement: " +
        lrsResponse.content.id + " "
        + lrsResponse.content.actor + " "
        + lrsResponse.content.verb + " "
        + lrsResponse.content.target;
}
else
{
    // Do something with failure
    respondTextBox.Text = "Failure";
}
```

Grafik 8. LRS’ye Deneyim Kaydetme Örneği (Rustici Software (2015b)’den uyarlanmıştır).

TinCan API kitaplığı kullanarak LRS'yi sorgulayan bir C# örneği Grafik 9'da verilmiştir. Buna göre "29 Ağustos 2013" tarihinden itibaren kaç deneyim kaydedildiği öğrenilmek istenmektedir.

```

using System;
using TinCan;
using TinCan.LRSResponses;

var lrs = new RemoteLRS(
    "https://cloud.scorm.com/tc/JMUW510QI5/",
    "memutlu@anadolu.edu.tr",
    "██████"
);

var query = new StatementsQuery();
query.since = DateTime.ParseExact("2013-08-29 07:42:10Z", "u",
    System.Globalization.CultureInfo.InvariantCulture);
query.limit = 10;

StatementsResultLRSResponse lrsResponse = lrs.QueryStatements(query);
if (lrsResponse.success)
{
    // List of statements available
    respondTextBox.Text = "Count of statements: "
        + lrsResponse.content.statements.Count;
}
else
{
    // Do something with failure
    respondTextBox.Text = "Failure";
}

```

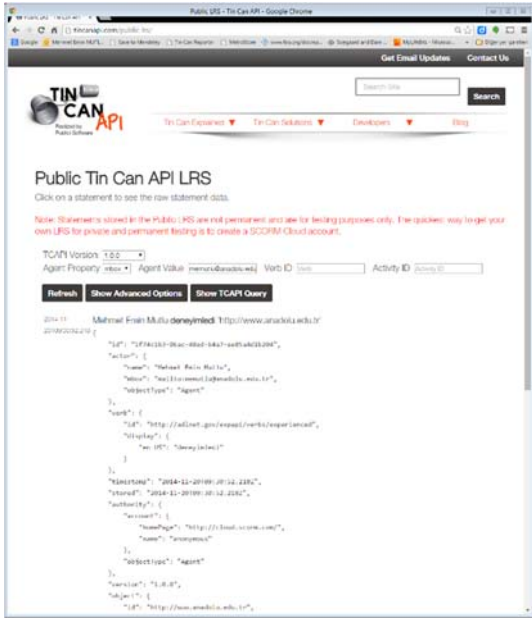
Grafik 9. LRS'yi Sorgulama Örneği (Rustici Software (2015b)'den uyarlanmıştır).

### Kullanılabilir LRS Sunucuları

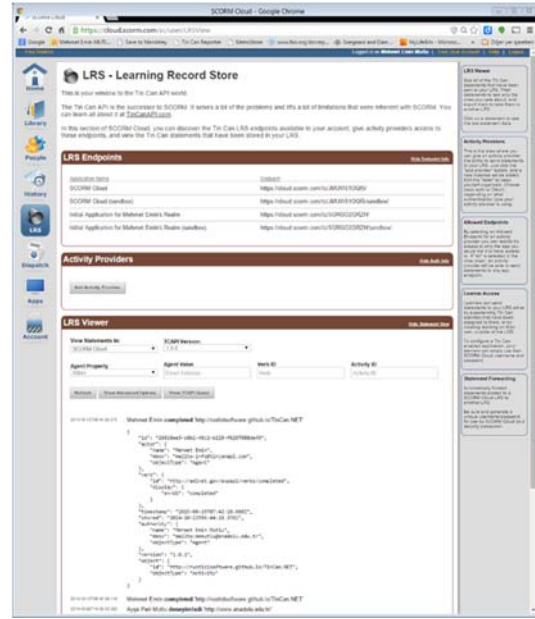
Bir LRS, Experience API kullanarak oluşturulan ve kendisine gönderilen deneyim deyimlerini kabul eden ve depolayan bir veritabanına sahip bir web hizmetidir. Diğer bir deyişle deneyimlere ait JSON dosyaları bir deneyim sağlayıcısı (Activity Provider) yazılımı tarafından LRS'ye HTTP üzerinden POST komutuyla gönderilmektedir. Aynı LRS bir raporlama yazılımı tarafından HTTP üzerinden GET komutuyla gönderilen sorguları da yanıtlayabilir. Sorgularda kullanılan başlıca parametreler aktör, fiil, nesne olabilir. Ayrıca verilen bir zamandan önce ya da verilen bir zamandan sonraki veriler süzülebilir.

Geliştiriciler Experience API tanımlamalarına uyarak kendi LRS'lerini yazabilirler. ADL örnek olması amacıyla açık kaynak kodlu bir LRS yayınlamaktadır. Ayrıca ücretsiz olarak

kullanıma açık LRS'ler de bulunmaktadır. Örneğin Rustici Software firması tarafından testler için kayıt yaptırmaya gerek olmadan kullanılabilir bir halka açık LRS TinCanApi.com adresinde hizmet vermektedir (Grafik 10a). ADL (ADL, 2015f) ve Rustici Software firmasının (Grafik 10b) geliştiricilerin üye olarak yararlanabilecekleri birer tam işlevli LRS'leri bulunmaktadır. Ayrıca Anadolu Üniversitesi bünyesindeki Öğretim Teknolojileri Araştırma Geliştirme Birimi'nde deneme amaçlı bir LRS sunucusu hizmeti bulunmaktadır.



(a) Halka açık bir LRS (TinCanAPI, 2015d)



(b) Kayıt yaptırılarak kullanılan bir LRS (SCORM.COM, 2015)

### Grafik 10. LRS Sunucuları

ADL ve Rustici Yazılım firması geliştiriciler için açık kaynak kodlu örnekler yayımlamaktadırlar. Uygulayıcılar bu örnekler yardımıyla kısa sürede deneyim izleme olanağına sahip e-öğrenme içerikleri hazırlamaya başlayabilirler. Experience API'nin kullanımına yönelik örnekler için Mutlu (2014)'dan, LRS kurulumu ve Experience API örnekleri için Yılmazel ve Mutlu (2015)'dan yararlanılabilir.

Experience API içerisinde LRS raporlama için hazır bir hizmet bulunmamaktadır. Kullanıcılar Experience API ile LRS'yi sorgulayarak, veritabanından gelen sonuçları kendileri raporlayabilirler. Bu amaçla ADL tarafından açık kaynak kodlu uygulama örnekleri yayımlanmıştır (GitHub, 2015j). LRS sorguları yetkiler dahilinde sadece bir öğrenenle ilgili

olabileceği gibi, takım ya da sınıf gibi gruplar üzerinde ya da LRS’deki verilerin tümü üzerinde de olabilir.

### **SCORM’dan xAPI’ye Geçiş İçin Yol Haritası**

ADL kullanıcıların SCORM’dan xAPI ve TLA’ya geçişi için dört aşamalı bir yol haritası tanımlamıştır. Birinci aşama kullanıcının SCORM kullanmasıdır. İkinci aşama ÖYS merkezli olarak SCORM ve xAPI kullanılmasıdır. Üçüncü aşamada LRS merkezli SCORM ve xAPI kullanımına geçiş öngörülmektedir. Dördüncü aşama ise xAPI ve TLA’ya bütünüyle geçiştir (GitHub, 2015a). Başlıca yazarlık sistemleri e-öğrenme içeriğini xAPI uyumlu paketler halinde yayınlamaya başlamışlar, başlıca öğrenme yönetim sistemleri de xAPI uyumlu e-Öğrenme içeriklerini kabul etmeye başlamışlardır (TinCanAPI, 2015a).

### **Diğer ADL Projeleri**

ADL tarafından TLA’nın yanı sıra (Virtual World Framework - VWF) ve Öğrenme için Kişisel Yardımcı (Personal Assistant for Learning – PAL) projeleri yürütülmektedir.

#### **Sanal Dünya Çerçevesi (VWF) Projesi**

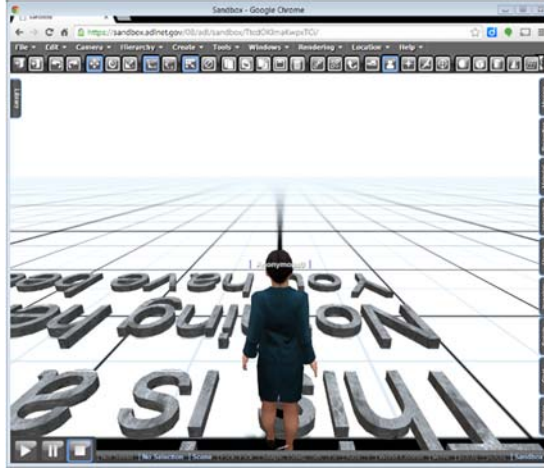
Birden fazla öğrenenin ortak öğrenme amaçları için bir araya gelmelerini sağlayan etkili yöntemlerden birisi de sanal dünyalardır. Bu gereksinime yanıt vermesi için geliştirilen “Sanal Dünya Çerçevesi” işbirlikli ve ölçeklendirilebilir web tabanlı uygulamalar geliştirmek için geliştirilmiş bir mimaridir. Açık kaynak kodlu bu mimariyle bir web tarayıcısında çalışabilen kalıcı sanal ortamlar oluşturmak mümkündür (VWF, 2015a).

VWF sanal ortam oluşturmak için gerekli üç boyutlu şekil oluşturma, ışık ve kamera özelliklerini kullanma, üç boyutlu şekilleri canlandırma, kullanıcı davranışlarını yakalayıp kullanıcıların üç boyutlu nesnelere etkileşime girmelerini sağlama, canlı sohbet arayüzü oluşturma, ses, video ve grafik ekleme vb. araçlara sahiptir. Bu araçları kullanarak oluşturulan ve web ortamında çalışan bir sanal dünyaya diğer kullanıcıları da davet etmek ve diğer kullanıcılarla bu dünyada etkileşime geçmek mümkün olmaktadır. Böylece bir e-öğrenme içeriğine ait bir web sayfasında aynı anda birden fazla öğrenen birbirleriyle bir sanal dünyada etkileşime geçebilmektedir (Burns, Easter, Chadwick, Smith, ve Rosengrant, 2014; Easter ve Chadwick, 2013).

Projenin açık kaynak sitesinden (GitHub, 2015h) VWF sunucusu indirilip kurulduktan sonra geliştiriciler JavaScript ile VWF uygulamaları yazarak bütün kullanıcıların eşzamanlı

olarak dahil olabildikleri bütünüyle üç boyutlu bir ortam oluşturabilirler. İki kullanıcının kendi bilgisayarlarından aynı anda erişerek oynayabilecekleri bir üç boyutlu Pong uygulaması 200 satırdan az komutla ile yazılabilmektedir (VWF, 2015c). VWF ile kullanılacak çok sayıda hazır model bulunmaktadır. Ayrıca VWF ile oluşturulmuş hazır örnekler proje sitesinde yayınlanmaktadır (VWF, 2015b). VWF ile Experience API uyumlu uygulamalar geliştirmek mümkündür. Böylece öğrenenlerin VWF ile geliştirilmiş sanal dünyalardaki etkinlikleri de LRS'lere kaydedilebilmektedir.

Geliştirici olmayan tasarımcılar da eğitim ya da eğlence amacıyla VWF tabanlı sürükleyici 3B ortamlar geliştirme ve dağıtım platformu olan Virtual World Sandbox'ı kullanabilirler (Grafik 11). Virtual World Sandbox ile oluşturulan ortamlar açık kaynak kodlu Virtual World Sandbox sunucusu üzerinden yayınlanabilirler (GitHub, 2015e).



*Grafik 11.* Virtual World Sandbox Uygulaması

### **Öğrenme İçin Kişisel Yardımcı (PAL) Projesi**

Kişisel öğrenme yardımcısı ADL'nin gelecek 10-15 yıl boyunca uzun süreli odaklandığı; öğrenmenin yapay zekâ ile desteklendiği bir ARGE girişimidir. Bu araştırmanın amacı öğrenenin öğrenme gereksinimlerini önceden tahmin edebilme, bunu var olan enformasyonla zahmetsizce birleştirebilme, etkili ve kişiselleştirilmiş içeriğe ve iş verimliliğine yardımcı araçlara her yerden ve her zaman erişebilme yeteneklerine sahip bir akıllı yardımcının geliştirilmesidir. PAL, öğrenenin öğrenme ve yetiştirme performanslarıyla ilişkili deneyimlerinin farklı cihazlar ve formatlar boyunca izlenmesi, içeriğin uygun biçimde özelleştirilmesi ve bireyin öğrenme stillerine uyarlanması, kullanıcının öğrenmesinin

değerlendirilmesi ve gelişiminin izlenmesi süreçlerini içermektedir (Durlach ve Johnson, 2014).

Taslak halindeki PAL mimarisinde üç katman bulunmaktadır. Birinci katman olan ön yüzde öğrenen podcast'ler, sanal ortamlar, algısal öğrenme modülleri ve akıllı öğretici sistemler gibi değişik öğrenme yöntemleriyle temas halindedir. Bu katmanda ayrıca öğrenen çeşitli uygulamalar, giriş – çıkış cihazları ve algılayıcılar yardımıyla PAL ile iletişim kurar. Mimarinin ikinci katmanı arka yüzdür. Burada öğrenenle ilgili yetkinlikler/konular, öğrenme nesnelere, üst veri ve öğrenen profili verilerinin saklandığı bir veri deposu ve aynı zamanda bu verileri değerlendiren PAL hizmetleri bileşeni bulunur. Pal hizmetleri arasında öneri hizmetleri, hedefe yönelik iş akışları, bağlam duyarlılığı ve kişiselleştirme hizmetleri bulunmaktadır. PAL hizmetleri gerçekleştirdiği çıkarımlarla ön yüzdeki öğrenme yöntemlerinin işleyişine yön verir. Mimaride ayrıca dış kaynaklara erişim sağlayan bir katman daha vardır. Bu katmanda da ağırlıklı olarak TLA ya ait bileşenler olan yetkinlik ağları, içerik aracılığı, öğrenen profilleri ve deneyim izleme hizmetlerine erişim sağlanmaktadır. Arka yüz katmanının erişebileceği takvimler, haritalar ve sosyal ağlar gibi ticari uygulamalar ve diğer PAL uygulamaları da dış kaynaklar katmanında yer alırlar. Bu katmandaki hizmetler arka yüzdeki veri deposu ve PAL hizmetlerine gerekli verileri sağlar (Regan, 2013; Regan, Raybourn, ve Durlach, 2013).

ADL tarafından 2014'ün Eylül ayında PAL'ın geliştirilmesi amacıyla bir ihale açılmıştır (ADL, 2014). İhale metninde PAL'ın birden çok alanda araştırma ve geliştirme yapmayı gerektiren geniş kapsamlı bir proje olduğu vurgulanmaktadır. İhale kapsamında araştırılacak alanlar şunlardır:

- Kişiye özel içeriği, şeffaf kullanıcı arayüzünü, kalıcı internet erişimini ve kablolu / kablosuz güç seçeneklerini destekleyecek “cihaz özellikleri”;
- Öğrenen deneyimlerini izlemeyi, içeriği uyarlamayı ve sosyal medya ile bütünleştirmeyi destekleyecek “PAL mimarisi”;
- Ortam boyunca paylaşılabilen, her zaman ve her yerde erişilebilir, yenilikçi öğretim ve öğrenen profili yönetimini destekleyebilecek “bilgi ve enformasyon”;
- İsmarlama ve kişiselleştirilmiş deneyim oluşturmak amacıyla kullanıcı profilini, öğrenme özelliklerini ve yetkinlikleri çözümleyebilecek; bilişsel uyumu ve duygusal esnekliği destekleyebilecek, PAL'a eklenmiş bir “yapay zekâ”;
- “Sanal ortamlar”, PAL ve yapay zekânın bir karma gerçeklik öğrenme kümesi içinde fiziksel dünya ile nasıl bütünleşebileceği;



- PAL ile donatılmış “gelecek nesil öğrenenler” için öğretim tasarımı ve değerlendirme kuralları.

Bu ihale sonucunda gerçekleştirilecek çalışmalar yayınlandığında PAL yol haritası topluluk tarafından bilinir hale gelecektir.

### Sonuçlar

ADL, SCORM projesinde, 1999’da bu alanda çalışan önemli kuruluşların kaynaklarının bir araya getirilmesi ve uygulanabilir bir standardın elde edilmesi için liderlik rolünü üstlenmiştir. Benzer yöntemin TLA, VWF ve PAL projelerinde de uygulandığı görülmektedir. Bu projelerde de ADL bu alanlara ait etkili kuruluşların çabalarını ve kaynaklarını bir araya getirerek süreci hızlandırmakta ve her bir kurumun kendi başına üstesinden gelemeyeceği büyüklükteki projelerin geliştirilmesine ve proje çıktılarının açık kaynak olarak dağıtılmasına öncülük etmektedir.

TLA, VWF ve PAL projeleri tamamlandığında 2020’lerde öğrenmenin aşağıdaki özelliklere sahip olması beklenebilir:

- Her yerdeki ve her andaki öğrenme deneyimlerinin yakalanabilmesi.
- Öğrenenin yaşam boyu öğrenme deneyimlerinin topluca değerlendirilebilmesi.
- Öğrenme içeriklerinin akıllı içerikler haline gelmesi.
- Öğrenenlerin sahip olduğu yetkinliklerin deneyimlere dayalı kanıtlarının kullanılabilmesi.
- Sanal dünyalarda ortak öğrenme deneyimlerinin yaygınlaşması.
- Öğrenenlere yaşam boyu öğrenme süreçlerinde akıllı yardımcının yol göstermesi.

Projelerin son aşaması olan Öğrenme için Kişisel Yardımcı projesi sonuçlandığında bireylerin öğrenme süreçlerini yakından izleyen, elde ettiği verileri değerlendirerek öğrenmeyi yönlendiren, öğrenme ortamlarına ve içeriğine erişimi her an her yerde mümkün kılarak öğrenme etkinliklerine ait yükü önemli ölçüde üstlenen, uzun dönemli öğrenme planlarının uygulanmasında etkili olan bir kişisel yapay zekâ ortaya çıkacaktır. Böyle bir akıllı yardımcı bireylerin yaşam boyu öğrenme süreçlerini daha önce hiç yaşanmamış bir boyuta taşıyabilecektir.



## Öneriler

Bu çalışmada ele alınan projelerin tümünün birer açık kaynak çıktılara sahip olduğu görülmektedir. Projelerin ürünlerinin bir-iki yılda bir yeni sürümleri yayınlanmakta ve projelerin yol haritaları doğrultusunda ürünler daha gelişmiş ve birbirleriyle daha tümleşmiş hale gelmektedir. Öğrenme teknolojilerine odaklanan bir birimin bu çalışmada ele alınan projelerdeki açık kaynak yazılımlarını kurması, denemesi, birlikte çalıştırması ve yeni sürümlerini izlemesi hem bu teknolojilerin uygulanması hem de geliştirilmesinde önemli bir rol üstlenmesini sağlayacaktır. Bu amaçla, Anadolu Üniversitesi Öğrenme Teknolojileri Ar-GE Biriminde bir Öğrenme Kaydı Deposu sunucusu kurulmuş ve özgün olarak geliştirilen Etkinlik Sağlayıcısı yazılımlarıyla test edilmeye başlanmıştır. Bir sonraki adım olarak RUSSEL, LearningRegistry, 3D-Repository gibi İçerik Dağıtıcılığı sunucuları ile VWF ve Virtual World Sandbox gibi Sanal Dünya Çerçevesi sunucuları kurularak denemelere başlanacaktır. Ayrıca Öğrenen Profilleri, Yetkinlik Ağları ve Öğrenme İçin Kişisel Yardımcı projeleri yakından izlenerek, bu projelere ait açık kaynaklar yayınlandıkça önceki çalışmalara eklenilecektir. Gerçekleştirilen denemelerin bulgularının düzenli olarak yayınlanmasıyla eğitim teknolojisi alan yazınına katkı sağlanması beklenmektedir.

### Kaynakça

- 3DR (2015). *ADL 3D-repository*. <http://3dr.adlnet.gov/>, 20.3.2015 tarihinde erişildi.
- ActivityStreams (2015). *Etkinlik akışları*. <http://activitystrea.ms/>, 20.3.2015 tarihinde erişildi.
- ADL (2014). *ADL initiative BAA – New submission window open for Personal Assistant for Learning (PAL) proposals*. <http://www.adlnet.gov/adl-initiative-baa-new-submission-window-open-for-personal-assistant-for-learning-pal-proposals-2/>, 20.3.2015 tarihinde erişildi.
- ADL (2015a). *Virtual world framework*. <http://vwf.adlnet.gov/>, 20.3.2015 tarihinde erişildi.
- ADL (2015b). *Experience API activities*. <http://www.adlnet.gov/expapi/activities/index.html>, 20.3.2015 tarihinde erişildi.
- ADL (2015c). *Experience API verbs*. <http://www.adlnet.gov/expapi/verbs/index.html>, 20.3.2015 tarihinde erişildi.
- ADL (2015d). *SCORM adopters*. <http://www.adlnet.gov/wp-content/uploads/2014/09/SCORMAdoptersLocked.xlsx>, 20.3.2015 tarihinde erişildi.
- ADL (2015e). *Training and learning architecture*. <http://www.adlnet.gov/tla/>, 20.3.2015 tarihinde erişildi.
- ADL (2015f). *ADL learning record store*. <https://lrs.adlnet.gov/xapi/>, 20.3.2015 tarihinde erişildi.
- ADL (2015g). *SCORM Project*. <http://www.adlnet.gov/scorm/>, 20.3.2015 tarihinde erişildi.
- Burns, E., Easter, D., Chadwick, R., Smith, D. A., & Rosengrant, C. (2014). The virtual world framework: collaborative virtual environments on the web. *Virtual Reality (VR), 2014 iEEE* (pp. 165-166). IEEE.
- DublinCore (2014). *LRMI version 1.1*. <http://dublincore.org/dcx/lrmi-terms/1.1/>, 20.3.2015 tarihinde erişildi.
- Durlach, P., & Johnson, A. (2014). Advanced distributed learning initiatives 2014. *Journal of Advanced Distributed Learning Technology*, 2(6), 35-48.
- Easter, D., & Chadwick, R. (2013). *The virtual world framework: implementing a web based client side simulator*. Interservice/Industry Training, Simulation, and Education Conference (IITSEC) 2013.
- GitHub (2015a). *SCORM to TLA roadmap*. <http://adlnet.github.io/SCORM-to-TLA-Roadmap/index.html>, 20.3.2015 tarihinde erişildi.
- GitHub (2015b). *Advanced distributed learning (ADL) Open Source Repository*. <https://github.com/adlnet>, 20.3.2015 tarihinde erişildi.

- GitHub (2015c). *3D-Repository*. <https://github.com/adlnet/3D-Repository>, 20.3.2015 tarihinde erişildi.
- GitHub (2015d). *Re-Usability Support System for eLearning (RUSSEL)*. <https://github.com/adlnet/RUSSEL>, 20.3.2015 tarihinde erişildi.
- GitHub (2015e). *Virtual world sandbox*. <https://github.com/adlnet/Sandbox>, 20.3.2015 tarihinde erişildi.
- GitHub (2015f). *Experience API specification*. <https://github.com/adlnet/xAPI-Spec/blob/1.0.1/xAPI.md>, 20.3.2015 tarihinde erişildi.
- GitHub (2015g). *Learning registry*. <https://github.com/LearningRegistry/LearningRegistry>, 20.3.2015 tarihinde erişildi.
- GitHub (2015h). *Virtual world framework*. <https://github.com/virtual-world-framework/vwf>, 20.3.2015 tarihinde erişildi.
- GitHub (2015i). *Rustici software open source projects*. <https://github.com/RusticiSoftware>, 20.3.2015 tarihinde erişildi.
- GitHub (2015j). *xAPI dashboard*. <http://adlnet.github.io/xAPI-Dashboard/>, 20.3.2015 tarihinde erişildi.
- Hruska, N., & Johnson, A. (2012). *The experience API: origin and capabilities*, ADL. <http://www.adlnet.gov/wp-content/uploads/2012/10/Experience-API-Webinar-v2.1-Final-hw.pdf>, 20.3.2015 tarihinde erişildi.
- LearningRegistry (2015). *Learning registry*. <http://learningregistry.org/>, 20.3.2015 tarihinde erişildi.
- LRMI (2015). *Learning resource metadata initiative*. <http://lrmi.net>, 20.3.2015 tarihinde erişildi.
- Mutlu, M. E. (2014). *Experience API ile yaşam günlüğüne dayalı öğrenme deneyimlerinin kaydedilmesi*. 19. Türkiye’de İnternet Konferansı – İnet-Tr’14, Yaşar Üniversitesi, İzmir. <http://inet-tr.org.tr/inetconf19/bildiri/52.pdf>, 20.3.2015 tarihinde erişildi.
- Poltrack, J., Hruska, N., Johnson, A., & Haag, J. (2012). The next generation of scorm: innovation for the global force. *The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)*.
- Regan, D. A. (2013). The Training and learning architecture: infrastructure for the future of learning. In *Invited Keynote International Symposium on Information Technology and Communication in Education (SINTICE), Madrid, Spain*. [http://www.adlnet.gov/wp-content/uploads/2013/09/regan\\_tla\\_infrastructure\\_for\\_future\\_of\\_learning\\_sintice\\_2013.pdf](http://www.adlnet.gov/wp-content/uploads/2013/09/regan_tla_infrastructure_for_future_of_learning_sintice_2013.pdf), 20.3.2015 tarihinde erişildi.

- Regan, D., Raybourn, E. M., & Durlach, P. J. (2013). Personalized Assistant for Learning (PAL). *Design Recommendations for Intelligent Tutoring Systems: Volume 1-Learner Modeling, 1*, 217.
- RUSSEL (2015). *Re-Usability Support System for eLearning (RUSSEL)*. <http://russe.adlnet.org/>, 20.3.2015 tarihinde erişildi.
- Rustici Software (2015a). *Project TinCan Phase 3*. <http://scorm.com/project-tin-can-phase-3-distribution-of-contentcross-domain/>, 20.3.2015 tarihinde erişildi.
- Rustici Software (2015b). *TinCan.NET - C#.NET library for Tin Can API*. <http://rusticisoftware.github.io/TinCan.NET/>, 20.3.2015 tarihinde erişildi.
- SCORM.COM (2015). *Bulut tabanlı SCORM sunucusu (Kayıt yaptırmak gerekiyor)*. <http://cloud.Scorm.com>, 20.3.2015 tarihinde erişildi.
- SoftwareInsider (2015). *Compare learning management systems (LMS) – SCORM compliant*. <http://lms.softwareinsider.com/>, 20.3.2015 tarihinde erişildi.
- TinCanAPI (2015a). *TinCanAPI adopters*. <http://tincanapi.com/adopters/>, 20.3.2015 tarihinde erişildi.
- TinCanAPI (2015b). *Layer 1: A modernized version of SCORM*. <http://tincanapi.com/layer-1-freeing-us-from-the-constructs-of-old>, 20.3.2015 tarihinde erişildi.
- TinCanAPI (2015c). *What is a Learning Record Store (LRS)?*. <http://tincanapi.com/learning-record-store/>, 20.3.2015 tarihinde erişildi.
- TinCanAPI (2015d). *Public Tin Can API LRS*. <http://tincanapi.com/public-lrs/>, 20.3.2015 tarihinde erişildi.
- TinCanAPI (2015e). *Statement generator*. <http://tincanapi.com/statement-generator/>, 20.3.2015 tarihinde erişildi.
- VWF (2015a). *Virtual World Framework*. <https://virtual.wf/>, 20.3.2015 tarihinde erişildi.
- VWF (2015b). *Virtual World Framework Demos*. <https://virtual.wf/demos.html>, 20.3.2015 tarihinde erişildi.
- VWF (2015c). *Virtual World Framework Tutorial*. [https://virtual.wf/getting\\_started.html](https://virtual.wf/getting_started.html), 20.3.2015 tarihinde erişildi.
- Wikipedia (2015a). *Learning management systems*. [http://en.wikipedia.org/wiki/Learning\\_management\\_system](http://en.wikipedia.org/wiki/Learning_management_system), 20.3.2015 tarihinde erişildi.
- Wikipedia (2015b). *List of learning management systems*. [http://en.wikipedia.org/wiki/List\\_of\\_learning\\_management\\_systems](http://en.wikipedia.org/wiki/List_of_learning_management_systems), 20.3.2015 tarihinde erişildi.

Yılmazel, S. ve Mutlu, M. E. (2015). *Experience API ve Öğrenme Kayıt Deposu (Learning Record Store-LRS) ile uygulama örnekleri*. XVII. Akademik Bilişim Konferansı, 4-6, Anadolu Üniversitesi, Eskişehir.

## Yazar Hakkında

### Mehmet Emin MUTLU



Mehmet Emin Mutlu Anadolu Üniversitesi Açıköğretim Fakültesinde öğretim üyesi olarak görev yapmaktadır. Lisans derecesini Matematik Mühendisliği, Yüksek Lisans derecesini Endüstri Mühendisliği, Doktora derecesini ise Yöneylem Araştırması alanında almıştır. Araştırma alanları e-Öğrenme, açık ve uzaktan öğrenme, yaşam boyu öğrenme, kişisel öğrenme ortamları, kişisel bilgi yönetimi ve yaşam günlüğü'dür.

Posta adresi:

Tel (İş):

+90-222-3350580/2435

GSM:

+90-505-7330725

Eposta:

[memutlu@anadolu.edu.tr](mailto:memutlu@anadolu.edu.tr)

URL:

<http://home.anadolu.edu.tr/~memutlu/>