PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Maziar AboualizadehBehbahani

Entitled
Proposing a New System Architecture for Next Generation Learning Environments

For the degree of Master of Science in Electrical and Computer Engineering

Is approved by the final examining committee:

Brian King
Co-chair

Ali Jafari
Co-chair

Huanmei Wu

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy of Integrity in Research" and the use of copyright material.

Approved by Major Professor(s): Brian King, Ali Jafari

Approved by: Brian King   4/7/2016
Head of the Departmental Graduate Program   Date

PROPOSING A NEW SYSTEM ARCHITECTURE FOR NEXT GENERATION

LEARNING ENVIRONMENTS

A Thesis

Submitted to the Faculty

of

Purdue University

by

Maziar AboualizadehBehbahani

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Electrical and Computer Engineering

May 2016

Purdue University

Indianapolis, Indiana

For my mom and dad, who always support me and to whom I owe all my life.

And for my wife because of her priceless patience and encouragement.

ACKNOWLEDGMENTS

I would like to express my appreciation to all who have supported and helped me throughout this research. I especially want to thank Professor Ali Jafari for providing opportunity to do this research on CourseNetworking (which is in production from March 2012), for his guidance through every step of my Master's Program, and for sharing years of experience with me. I gratefully thank Dr. Brian King for his continuous support and advising throughout this research and the Master's Program and also sharing his pure knowledge during algorithm and cryptography classes. I would also like to thank Dr. Huanmei Wu for her guidance in many aspects of my academic success including her priceless comments regarding my thesis. In addition, her database class makes me able to achieve a deep understanding of database design and modern databases.

I owe my deepest gratitude to Dr. Charles Severance, the inventor of Learning Tools Interoperability standard, because of his support, sharing original knowledge and ideas and revising my research plan and make it more eye-catcher specially in job market. Special thanks to Mr. David Mills who helped me resolve ambiguous points in technical parts of my thesis in early steps.

I wish to express my thanks to IUPUI CyberLab and CourseNetworking company for their human resources, and facilities. I would also like to thank Indiana University-Purdue University Indianapolis (IUPUI) for the opportunities and the support. Additionally, I would like to express my gratitude to my colleagues at CyberLab, Ms. Mengyuan Zhao (Project Coordinator), Mr. Joseph Hofmeisterand (Graphical Designer) and Ms. Allison Wigginton (Testing intern) and CourseNetworking Mr. Elson Wu (Lead Developer) and Mr. King Chung (Front-end Developer). I want to thank Ms. Valerie Lim Diemer, and Ms. Sherrie Tucker for their support and guidance.

Finally, for her priceless patience and encouragement of my academic pursuits, I would like to thank my wife, Golnoush.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

AboualizadehBehbahani, Maziar. M.S.E.C.E., Purdue University, May 2016. Proposing a New System Architecture for Next Generation Learning Environments. Major Professors: Brian King, Ali Jafari.

The emergence of information exchange and act of offering features through external interfaces is a vast but immensely valuable challenge, and essential elements of learning environments cannot be excluded. Nowadays, there are a lot of different service providers working in the learning systems market and each of them has their own advantages. On that premise, in today's world even large learning management systems are trying to cooperate with each other in order to be best. For instance, Instructure is a substantial company and can easily employ a dedicated team tasked with the development of a video conferencing functionality, but it chooses to use an open source alternative instead: The BigBlueButton. Unfortunately, different learning system manufacturers are using different technologies for various reasons, making integration that much harder.

Standards in learning environments have come to resolve problems regarding exchanging information, providing and consuming functionalities externally and simultaneously minimizing the amount of effort needed to integrate systems. In addition to defining and simplifying these standards, careful consideration is essential when designing new, comprehensive and useful systems, as well as adding interoperability to existing systems, all which subsequently took part in this research.

In this research I have reviewed most of the standards and protocols for integration in learning environments and proposed a revised approach for app stores in learning environments. Finally, as a case study, a learning tool has been developed to avail essential functionalities of a social educational learning management system

integrated with other learning management systems. This tool supports the dominant and most popular standards for interoperability and can be added to learning management systems within seconds.

1. INTRODUCTION

1.1 Overview

In today's world, different technologies make it easier for software manufacturers to choose the best option from wide variety of databases, technologies and programming languages based on their resources, such as human resources, time, money and the nature of product. This creates an environment that consists of different software applications (specifically web applications) with different underlying technologies. These systems need to interact and integrate and exchange information with each other. The ideal solution for these stipulations should be independent of underlying technologies. Research has shown, in order to produce these kind of integrations, the use of RESTful APIs can proficiently handle interaction of mobile interfaces and backend developments, and similarly other services and backed (both through RESTful API).

The problems associated with integration through RESTful API is the need for developers to customize or in some cases, create new interfaces based on RESTful API architecture in order to integrate a system with another system, or make their API very dynamic and flexible which is, at the very least, an annoyance that can lead to the waste of time and money. Learning domain applications present similar obstacles when it comes to integration and interoperability complications, however, defining domain specific standards can help utilize common attributes between systems from the same domain, which can potentially resolve the issues. Subsequently, if a service provider and service consumer support the standard, the integration and interoperability will be done seamlessly with almost no effort from the developers.

Today, dominant learning management system manufacturers are developing standards, supported by IMS Global Consortium. Although this consortium is in charge of

the development process of these standards, they also always listen to their users and will update standards and protocols based on users' feedback. These standards are not open to immediate modification but are open for modification based on feedback given on every version release. This research, after careful review and categorizing different standards in this area (Chapter 2), proposed a new or revised approach to one of the standards in the area of learning systems (Chapter 3).

In the learning domain different services can be offered as tools, and one of the proposed models from IMS Global Consortium is the Community App Store Architecture (CASA), which is basically a proposed architecture designed to bring several external tools to one place to better distribution systems. Through this research, we have suggested the adaptation of a standard tool container to existing CASA architecture and we further explained how it would benefit CASA. Finally, we have combined the collected information from the review of existing approaches and the ongoing project (CourseNetworking a social educational learning management system) and presented it as case study. The result was an external tool, which avails the main features taken from CourseNetworking, such as posting tool, gradebook and analytics, any compliant tool consumer.

1.2   Purpose of Study

The main goals of the conducted research can be concluded by the following:

1. Comprehensive overview of standards regarding interoperability and easy integration between systems in learning environment.

2. A categorization of different standards and protocols (group by application) regarding information exchange and interoperability between learning systems in chronological order in each category.

3. Proposing a new or revised architecture to facilitate and improve the user experience in CASA.

4. Development of a LTI compliant external tool and find out challenges regarding development and adaptation into CourseNetworking.

1.3   Statement of Problem

The main idea and motivation behind this research is to make the existing learning management system able to be easily integrated in order to interoperate and exchange information. Therefore, after studying the existing approaches carefully, the initial statement of problem narrowed to the categorization of technologies and the approaches concerning the interoperability and data exchange between learning management systems. The main problem or question remains: to find the best approach to achieve a learning management system, which seamlessly interoperates and exchanges data with its peers. Extensive research on this area opened two more opportunities:

First, improve one of the standards in learning environments. The standard architecture is subject to improvement is the Community App Store Architecture (CASA) because of the detected lack of information about the usage of tool by consumers somewhere central.

Second, develop a new tool in the existing learning management system to offer its functionality as external tool to be consumed by compliant learning management systems, or, generally, tool consumers.

1.4   Significance of the Study

Due to lack of updated information about interoperability approaches in learning environments, we tried to equip a learning management system with the capability of exchanging information with its peers within a learning environment. Obviously, a study on methods of interoperability and data exchange between learning systems would bring benefits like interoperability and integration to users. Benefits such as reducing development time by ease of integration, and reusing existing services in

learning domain beside opportunity to use services from other providers by a consumer (e.g. Personal Learning Environment or Learning Systems) could do just that.

Another section of this study proposes the addition of an extra capability (a data container) to conventionalize external learning tool app stores. This improvement will allow app store users to retain information regarding their service consumption in a separate place, which would benefit the service consumers in two ways. First, it would store the related data in another place, which makes it more reliable and invulnerable to failure from the service provider side. Secondly, this information can be stored in a datacenter, which is under the full control of consumer institution (e.g. Universities or other organizations).

The last section of this research is associated with a case study on the development process of a service oriented learning tool. The output of this case study is a tool named "CN Post" which brings the capability of social interaction into the courses in a conventional learning management system. This tool significantly improved the quality of student-to-student and student-to-instructor communication and, subsequently, collaboration.

1.5   Definitions

Framework: A work environment specially designed to solve common and complex problems in different domains [1].

SOA: The Service Oriented Architecture (SOA) is a software engineering approach that provides a separation between the interface of a service, and its underlying implementation [2, 3].

Learning Management System: A Learning Management System (LMS) is a software application being used to manage learning content. It is facilitating teaching and learning by offering other services such as course catalog support, launching courses, registering students, delivering instructional material, assessment activities management, tracking students' progress and doing some analytics [4].

Learning Object: "A small modular block of learning material serving a single learning objective, independent of other learning materials" [4].

Personal Learning Environment: A set of tools and services being used by learners in one place but originally provided by different institutions [5] [6]

Learning Tool Interoperability: Generally created to equip learning management systems with capability of getting connected with external service tools in standard and seamless way [7].

LTI Provider: A tool or application that is providing a feature and/or service which can be used remotely through other compliant learning management systems [8]

LTI Consumer: An application able to connect to LTI tool providers and consume their provided features in a seamless manner [8]

Metadata: "Data that provides information about other data" [9].

Widget: A small and portable element that can be run in any HTML context [10].

Cartridge: A collection of complementary material produced by publishers to accompany a textbook [4].

Common Cartridge: "Standards for organization, publishing, distribution, delivery, search and authorization of a wide variety of collections of digital learning content, applications, and associated online discussion forums used as the basis for or in sup- port of online learning of any type" [4].

LTI-App Store: An app store (digital distribution platform) to offer tools with capability of being added to LTI compliant learning management systems. The unique characteristic of these app stores is managing consumer's information in the app store instead of provider.

1.6 Novelty

The first section of the research is only collecting and regrouping the new/historical information in the area of interoperability in learning environments, which is needed for the rest of research, even valuable by itself. However, the main novelty of this

research may be found in chapters 2 and 3. The refinement of CASA architecture, with an already designed container to improve its availability in chapter 3, and later in chapter 4, is designed and developed as an external tool to utilize the social features from CourseNetworking. In conventional LMS are two main novel ideas in this research.

1.7 Research Method

The research method used for this research is mainly divided into a few steps that can briefly be mentioned as follows:

The first step was based on the need to equip an existing learning management system with the capability to seamlessly interact and exchange information with other learning systems. This step was actually the initial step and formed the problem and question of this research. The second step has to do with extensive research on existing standards, technologies and solutions that assist learning management systems in seamless integration and interoperability. The third step was based on the knowledge collected in the second step an improvement, and proposed one of the standard architectures increase the quality of user experience. The forth step was designed and developed as an external tool in a case study based on theoretical knowledge taken from the second step and part of third step.

2. EXISTING APPROACH

2.1 Importance of Communications in Learning Systems

Similar to other industries in eLearning, there is wide variety of different learning platforms, repositories, tools and types [11]. Thus, there is necessity to invent an efficient way to communicate between these elements and improve it to provide seamless integration. Service oriented architecture is the most common and accepted approach in information exchange and integration between different information systems. Therefore, in order to have the same for learning management systems, learning system experts and integration engineers need to team up.

2.2 Personal Learning Environments

Personal Learning Environments (PLE) come to define a new concept: to support different tools in one place. Thus, in order to satisfy all or most of the student's necessities it has to interoperate with other systems. These systems can be LMS, portfolios, content repositories and so on [11]. This, aside from equipping the traditional classrooms to social networks, can be another aspect of PLEs [12]. PLE is a concept that can be executed by different institutions in different ways [6]. It will offer free personalization lead by the student and the capability of being managed by organization.

PLE is a new concept, which is defined based on learning management systems. This demand, from the institutional users, motivate IMS to come up with a convenient standard and at the same time, force LMS manufacturers to follow that standard.

In [13] some of the limitations and problems mentioned have been resolved over time, thanks to interoperability standards. These problems are listed quickly as followed:

1. Student-centered learning (also known as web 2.0) can cause misusage of itself by student which decreases the returned value after using PLE. Solution: Similar to other approaches PLE should be able adapt, evaluate and refine based on criteria.

2. Technological problems, most of which are resolved today.

3. Poorly supported by learning management systems, which can be resolved thanks to LTI compliant tool providers and tool consumers.

2.3   Learning Systems Interoperability

Similar to other systems, learning systems and other nodes in learning environments require working closely with each other. To facilitate the procedures related to this, such as information exchange, integration and interoperability, learning management systems should be designed (or re-engineered). Interoperability and integration have two different aspects. The first aspect would be the technical one, which is handled mostly by using initiatives of service-oriented architecture. The second aspect is providing a standard approach based on service- oriented architecture in order to minimize effort for interoperability and maximize productivity, specifically for learning systems.

Service oriented architecture (SOA) is a software engineering approach to separate the interface of functionality and underlying technology from each other and to also make them independent. SOA is an approach to achieve two main goals [11]:

1. Make functionalities available as services to be used externally.

2. Make consumer technologies independent of underlying technology (or minimize technology dependencies in an application).

These two make us able to define standards based on SOA, to make functionalities available for external use. In eLearning context, we have a few general scenarios for interoperability [11] [14]:

1. Export a functionality from an institution to another environment (such as PLE) [14].

2. Add an external tool to an institution's learning management system.

3. Add external tools to PLE and recover information from LMS (with or without evaluation support).

In order to meet requirements to handle the above scenarios, we have to add interoperability capabilities to learning management systems; system design and SOA would be the right tools to develop the technical aspect of said interoperability and integration.

Another step needed to facilitate the interoperability between LMSs and external tools to create a set of external tools in PLE, [11] which is specific usage SOA in learning context. The learning system is an 18 year old concept but today representation of SOA is much newer. Thus, we have to reconsider and reengineer the classic LMS concept to make it much more convenient and able to open interoperate with other parties. This reengineering procedure process and equipping of the old system to SOA will answer the following needs [11]:

1. Additional features offered to students.

2. Need for interaction between LMS and other apps like mobile apps.

3. LMS needs SOA to import/export information to/from it.

Another good point to look at when facilitating integration between the LMSs is the use of some interoperability specifications aside from reducing the cost of integration. The first step needs a lot of time and effort [14]. Student and teacher cannot find out they are using something from outside of LMS [15].

2.4   Adapting LMS to SOA

Before adapting an LMS with SOA, we need to consider which elements from LMS need to be SOA compliant. In other words, we should improve classic the LMS in different ways and because it would be a big change, it could be done in different phases, for existing learning management systems. There are a few starting points that would make it easier to figure out how to adapt the classic learning management systems to modern service oriented architecture.

1. Adaptation of some/all parts of LMS to Mobile [16] would be a very useful option because it would force system designers to separate functionalities and different layers, which would also force parts of software to have different technologies. Therefore, different layers should be independent enough, in terms of underlying technology.

2. Add SOA semantic search interface to the LMS. Actually, the semantic search function, based on domain and needs, can be interchanged with any other agent, intelligent agent or analytic tool.

3. Integration between different systems would assist the old system (or in some cases both of systems) in reevaluating their interfaces and even help them come up with a standard.

4. Creating "Domain Specific Standard (LTI) and Interoperability Specification". There is a tradeoff between having a very common framework and having a specific standard. interoperability standard in learning context that can be developed by different vendors with their own choices for underlying technology.

5. Unidirectional Interoperability can be another requirement, in addition to interoperability standards basics, which can be added into the next phases of the development of these standards. For instance, LTI 1.0 (basic LTI launch) was the starter; unidirectional interoperability has been added to its next version, LTI 1.1 and similarly for RESTful API.

2.5   General Concept of LMS / SOA

To summarize, all of the discussions on migration from classic LMS to SOA-equipped LMS it can be described in a few sentences. SOA has been added to LMS to make it able to adapt into the other systems. LMS should be a place to gather all of the different services together. (LTI as Canvas did nicely). Client application can connect to the LMS through SOA, RESTful API. Figure 2.1 is one of the first reasonable examples of adding a SOA as a middle interface or as an extra layer to handle interoperability between LMS and other elements in learning context .
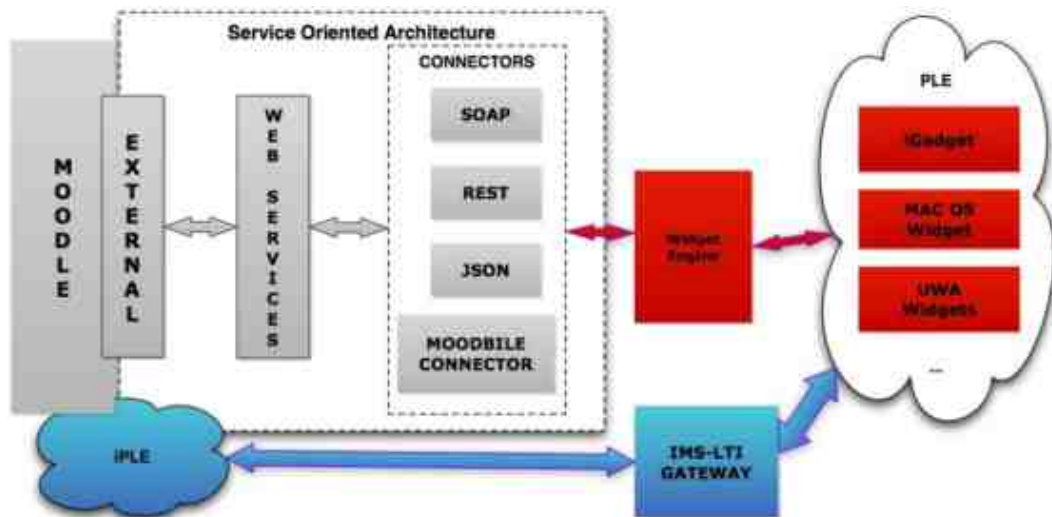
Fig. 2.1.: One of the Proposals for Applying SOA to Moodle [13]

Because the Moodle source code has a good PHP API, it can be connected to a wrapper to provide external web service.

[Moodle PHP API] + [wrapper] = [External/Web Service]

Conceptual aspects of service-oriented architecture have matured more and more over time and platforms become ready for an upgrade to another level of service-oriented infrastructure. In addition to this level up and the popularity of learning management systems, a number of successful SOA initiatives were selected for mod-

ification and found new service oriented standards. These standards are specifically for learning context. Thus begun a new era for Learning Management Systems.

2.6 Learning Interoperability Standards and SOA Initiatives

Data exchange between different e-learning systems has always been a big challenge [2]. Learning frameworks, standards and specification in eLearning domains have been invented to bring several advantages to classic learning management systems, such as reusing existing functionalities, integration cost reduction and one-time implementation of common functionalities.

2.6.1 eLearning Frameworks

A framework is a work environment specially designed to solve common and complex problems in different domains [1]. Frameworks have two portions of code, a frozen one and a hot one (also known as frozen-spot and hotspot). The frozen spot is a developed by institutions behind the framework or lead by developers who initiate the framework and users would not usually modify the frozen spot. The other part of framework, known as hotspot, contains codes, which are usually overridden by users as a pattern of development. eLearning frameworks support LMS development by utilize several layers of services [1]. There are two different kinds of frameworks within the mentioned portions. The types are abstract and concrete frameworks.

2.6.1.1 Abstract Frameworks

Abstract frameworks are about creation of specifications, recommendations and best practices for the development of eLearning systems.

2.6.1.1.1   IEEE Learning Technology Systems Architecture [1]

This abstract framework contains processes, stores and flows as you can see in Figure 2.2.
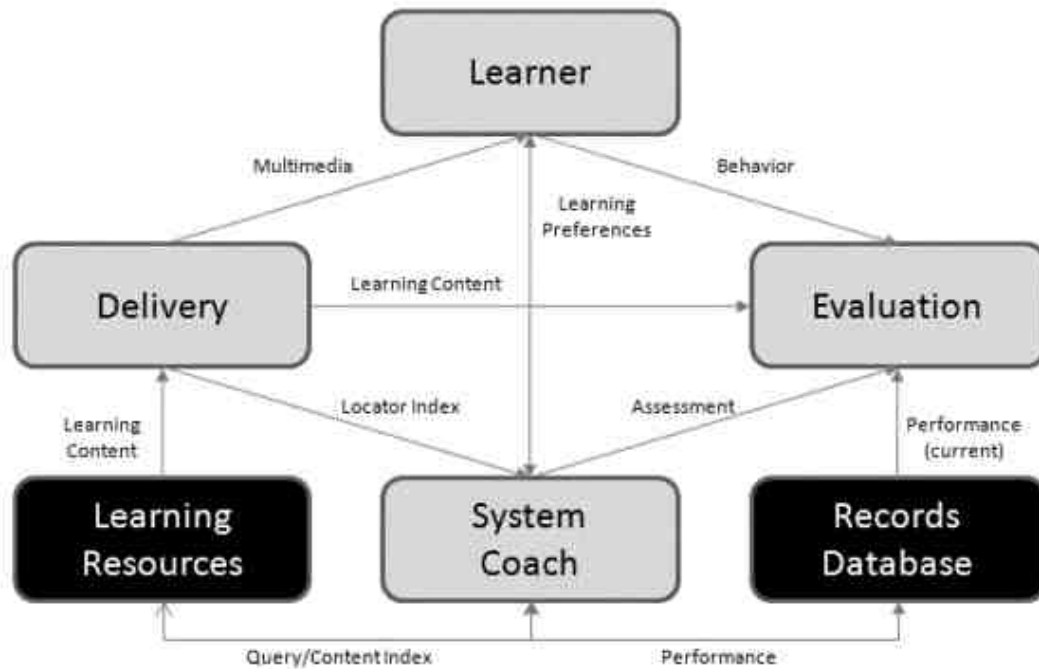
Fig. 2.2.: Learning Technology System Architecture [1]

2.6.1.1.2   Open Knowledge Initiative (OKI OSIDs) [2]

The OKI OSIDs provides definitions for interfaces in a standard manner (to provide a service). Other parties are following the same standard in order to avail data, functionality or external tools, known as OSID consumer [1] [11]. This scenario is very close to LTI Basic Launch. An architectural view of OKI is represented precisely in Figure 2.3:

OKI is invented, mainly, to create a standard architecture. This makes learning software systems able to share popular services. Authentication and authorization are the most popular and common services shared [2]. OKI has developed a series of
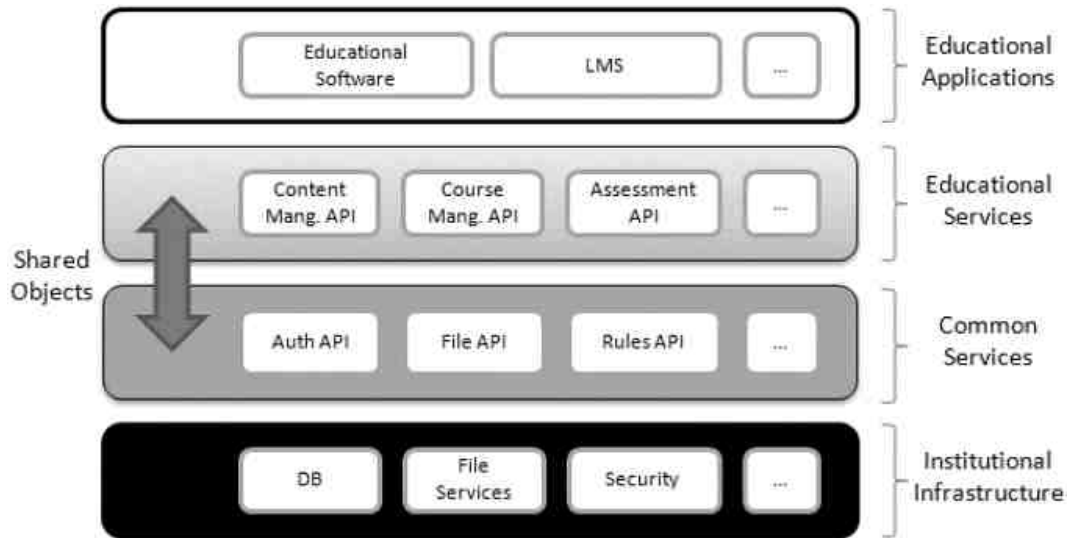
Fig. 2.3.: OKI Architecture [1]

interfaces known as Open Service Interface Definition. An OSID describes details of communication between providers and consumers (service definition).

Thanks to service oriented architecture, the OKI compliant software is capable of use across a wider range of domains. The main advantages of OKI are listed as follows:

- Ease of developed software: reusing the already developed common functions.

- Common service factory.

- Reduce integration cost (not unique, just like its other alternatives).

- Produced software is a cross-environment thanks to SOA.

- The use of OKI OSIDs guarantees the portability of the architecture and the bidirectional work [2].

2.6.1.1.3   IMS Global Learning Consortium initiatives for interoperability in learning systems [2] [1]

The IMS Abstract Framework (IAF) enables IMS to define and compile descriptions of standard specifications in learning context. This framework will describe the underlying technology and approach regarding specifications. A quick layer model of IMS Abstract Framework is represented in Figure 2.4:
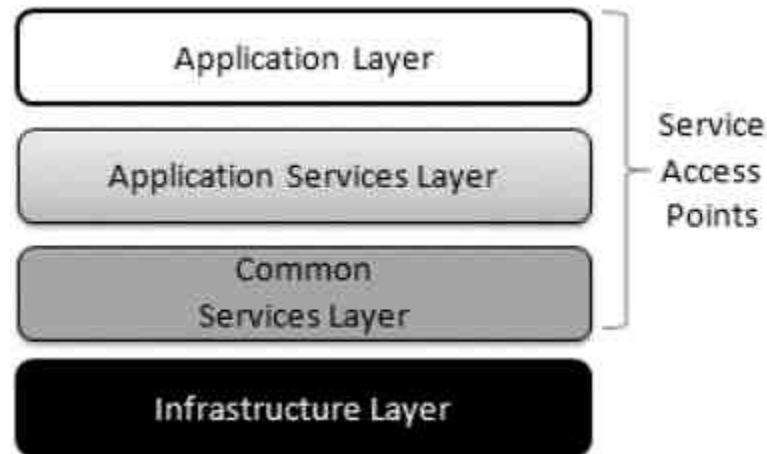
Fig. 2.4.: IAF Layered Model [1]

Gonzales describes different layers of IAF concisely as follows [2]:

- Application layer a set of tools and elements that provide part of the eLearning functionality.

- Application services layer a set of services that provide specific eLearning functionality to the applications.

- Common services layer a set of services that provide generic services used by the application services.

- Infrastructure layer a set of services that enable the exchange of data structures in terms of physical communications.

IMS LTI has been defined to ensure that all of necessary features and characteristics of OKI OSID are ported to new standard of developed focuses on the process

which describes how a remote service can be installed on a web based learning system [LTI website] and how to bring all the important capabilities from OKI into one place. The use of IMS LTI guarantees the portability of the architecture and the bidirectional work [2].

TSUGI is a framework used to facilitate and simplify LTI tools' development and deployment [15] as one. As Dr. Charles Severance [15] mentioned TSUGI, "The overall goal behind this line of work is to create a learning ecosystem that spans all the LMS systems including Sakai, Moodle, Blackboard, Desire2Learn, Canvas, Coursera, EdX, NovoEd, and perhaps even Google Classroom. It is time to move away from the one-off LTI implementations and move towards a shared hosting container for learning tools. It is time to move away from the one-off LTI implementations and move towards a shared hosting container for learning tools". The framework and new approach supporting the above statement could be vital to LTI's popularity.

2.6.1.2   Concrete Frameworks

Concrete Frameworks extend the goals of abstract frameworks by providing complete service designs and/or components that can be integrated into actual implementations of artifacts [1].

2.6.1.2.1   Open University Support System

Open University Support System (OpenUSS) is a series of specifications for Learning Management Systems to be implemented. Most of its implementations have been done in Java (J2EE and related technologies). The base version is relatively discontinued but some forks are still being developed.

According to their website, OpenUSS is based on the Application Service Provider model, meaning different institutions can work together by becoming connected and working with a single instance of OpenUSS. Information can be accessed through the OpenUSS via a variety of methods and devices such as internet, mobile, etc. [17].

2.6.1.2.2 Schools Interoperability Framework

Schools Interoperability Framework (SIF) is an interoperability framework that links schools and other corresponding systems together. Similar to other frameworks in this field, it was invented to facilitate integration between different systems in a more efficient, secure and economic manner without creating dependency between consumers/providers underlying technologies [18].

This standard is supported by the A4L community (SIF Association). They provide a variety of consultants and other services to organizations that use SIF.

SIF specification generally consist of two main components. The two components are Data Model and Infrastructure guidelines, which define the details for data packets and body of messages, and information exchange channel between different components.

Over the time this standard has become more mature and equipped to handle different situations and scenarios like supporting REST and it also plays nicer with scalability requirements [19].

2.6.1.2.3 E-Framework

This framework counts as an international effort to form a new approach for the development and integration of different learning and research systems together. In other words, E-framework is a service-oriented factor of main services, required by different e-learning systems or general web applications. These services are considered RESTful, like services [20]. Services designed by this framework can be divided into two main categories in terms of their application. The first category contains things that are related to e-Learning directly, and the second category contains services which can be used generally in e-Learning context or any other service oriented web application. The services mentioned before are shown as a whole in Figure 2.5:

A framework should be flexible, portable, best match to requirements and reusable. Thus, new frameworks and newer versions of existing frameworks are working to improve these characteristics [11]. In any framework or service oriented approach we

have to identify the institutional context, user context and communication channel between these two and that of external systems. The institutional contexts can be the learning environment in an institution, in which students carry out their related activities [14].

2.6.2   Learning Objects

2.6.2.1   Definitions

A good definition for Learning Object was offered in the 1999 Cisco Systems Reusable Information Object Strategy: "a collection of content items, practice items, and assessment items that are combined based on a single learning objective" [21]. In later references, an alternate definition for learning object was created, which is more compatible with today's technology. Chiappe defined Learning Objects as: "A digital self-contained and reusable entity, with a clear educational purpose, with at least three internal and editable components: content, learning activities and elements of context. The learning objects must have an external structure of information to facilitate their identification, storage and retrieval: the metadata" [22]. There are several proposed and defined standards and specifications to help learning object providers and consumers create and consume ideas more efficiently.

2.6.2.2   Characteristics

Regardless of specific standards, learning objects have a few characteristics and are categorized as follows:

- **Self-contained:** can be used independently.

- **Reusability:** each learning object can be used by different consumers and each update will affect all of them.

- **Hierarchical:** Similar to files and folders, learning objects can contain other learning objects (by reference). This capability makes them able to simulate the traditional course structure and make it easier to transform the existing courses into the baked learning objects.

- **Meta-Data:** Every learning object has descriptive information, such as keywords and tags [23].

A quick look at the history of learning objects' specifications split the time into the two main intervals: First interval was from initial release of SCORM from 2000 to 2010 and the second one is from 2008 to today. SCORM was the first step in answering the above characteristics, and Common Cartridge introduced the SCORM to new features and improved its existing requirements [4].

2.6.2.3 SCORM: 1st Generation Reference Model

SCORM (Sharable Content Object Reference Model) is one of the most popular reference models for learning objects has been created by ADL (Advance Distributed Learning) in 1999 and released SCORM 1.0 in 2000. It was created to take advantage of the standard web technologies and learning technology specifications to enable Department of Defense to create interoperable and web-based e-learning content. SCORM, as a set of specifications, was created to address the challenges associated with the following characteristics [24]:

- **Interoperability:** SCORM has defined the common data model and the API for eLearning content. Based on this definition, each learning management system or any SCORM player such as "Reload SCORM" or "SCORM Pool" can develop an appropriate interface to communicate with the specific SCORM object.

- **Portability:** In order to meet the portability requirement SCORM provides standard methods for packaging content to prepare a portable ZIP file, Package Interchange Format (PIF).

- **Reusability:** SCORM reference model describes the components in a learning experience and descriptive meta-data about them. Learning management systems and learning object repositories can use this information to offer search and discovery capabilities on SCORM objects. Thus, thanks to SCORM's portability and what is mentioned as SCORM's reusability, learning objects with this reference model can be used by any SCORM compliant learning management system. This came to reduce cost by sharing the already created contents [4].

- **Sequencing:** Navigation and progress tracking for a learning object is one of the most critical requirements. For instance, a learning object is a quiz or type of course material that can be read and can receive a feedback from the user. The reading progress and feedback (can be stored as results and/or grade) will be preserved in the learning object and can be passed back to the consumer. In addition to maintaining the progress, SCORM can provide a hierarchical structure over the materials in a SCORM package.

As shown in Figure 2.6, after 2010 another group (Rustici Software) took lead of SCORM's development and released the first stable release of that in 2013 under title of Tin Can API, as the next generation of SCORM (a few unstable versions released between 2010 and 2013).

2.6.2.4   Common Cartridge: New Generation standard

Several years after SCORM, IMS Global Consortium proposed the Common Cartridge to improve the SCORM in aspects like flexibility and interactivity. In order to achieve these goals, IMS Global has introduced the following features to SCORM (which were necessary but were not addressed in the SCORM): content authoriza-

tion, collaborative forums, outcome reporting and accessibility. Common Cartridge has been developed by companies (such as LMS companies) to meet the missed features in the SCORM [4].

The main idea behind the Common Cartridge is to give learning objects the ability to interact with external services. IMS Global proposed this environment and the other standards from this consortium to facilitate this interaction. We will discuss more about the Common Cartridge later in this chapter under the title of IMS Global Learning Consortium's standards. One of the today's approaches regarding the learning content is to provide learning content as LTI tools instead of learning objects [8].

2.7 IMS Global and Interoperability Standards

IMS Global Learning Consortium, nonprofit organization, was founded to assist learning technologies grow and be impactful. The main goal and effort of IMS is interoperability standards (design, development and support) and improving these standards to adapt to distributed learnings.

From the engineering point of view, these standards will be essential and save a lot of time and money for the vendors. Today, most of the pioneer LMS manufacturers support LTI and other standards from IMS Global. This good support makes it much easier to interoperate and integrate between different systems in learning environments. In addition, several meetings by IMS Global allows vendors to take part in the development of standards based on their needs, and also keep IMS standards fresh and convenient. Throughout the rest of this chapter, standards and specifications from IMS Global will be reviewed.

2.7.1 Learning Tools Interoperability

Learning Tools Interoperability (LTI) is one of the most popular specifications created by IMS Global under Dr. Charles Severance supervision. Created to standardize

integration process between tool providers and tool consumers learning context. Although learning management systems can sometimes offer tools, generally they are the Tool Consumers. On the other side, Tool Providers deliver remotely hosted applications accessed by consumers. In the other words, LTI was created to make LMS consume and to make Tool providers provide the tools through a seamless connection over the web [7]. From a similar point of view, LTI can be introduced as a proxy tool to avail tools externally to LMS users [2]. In this scenario, all of the corresponding data to courses and students will be saved in the tool provider. This allows tool providers to provide the functionality as service to several consumers (mostly LMS) through a hosted tool [2].

IMS Global is supporting this specification by providing different types of information and guidelines including specification documents, best practice guidelines, tutorials, presentations, sample codes and conformance tools to certify the tools plus series of quarterly and annual meetings in order to improve the standard.
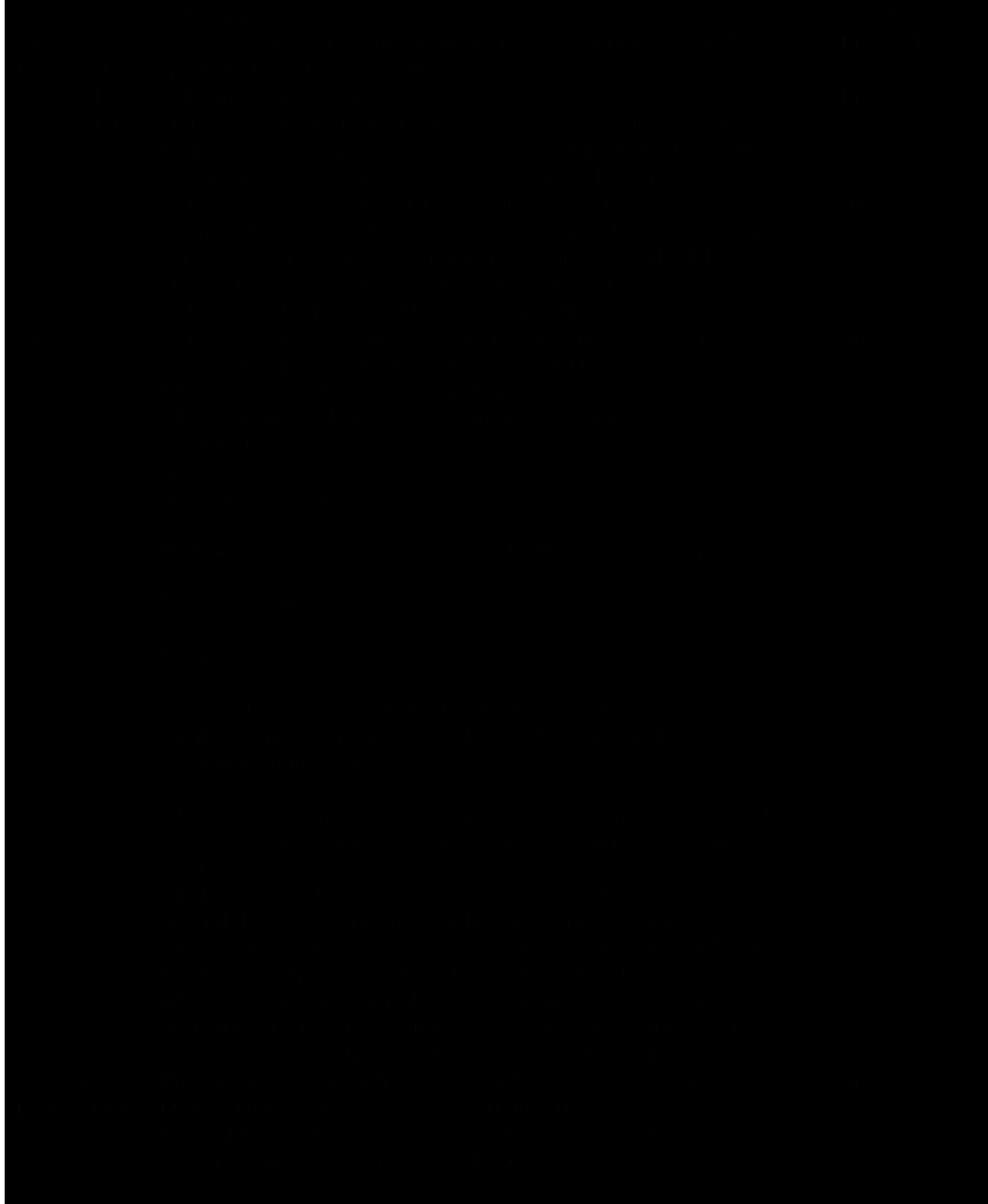
As vendors started to develop necessary modules in their systems for LTI compliancy it becomes more popular and new requirements are discovered and thus, different versions of LTI standard released. We can have a quick comparison between different versions of LTI standard over time in Table 2.1.

In LTI 2.0, on the launch step a lot of information from consumers come to the provider side, which allows it to provide different services such as single-sign-on, grading and dynamic view based on input parameters [2].

2.7.2   Advantages of LTI

Currently learning management systems are mature and stable enough to answer to the classic requirements [25] [15]. In order to separate functionalities of LMSs and offer them as provided services to external systems, service-oriented approach proposed. As Dr. Charles Severance mentioned in his paper, "The IMS Learning Tools Interoperability [IMS LTI] is a standard under development that standardizes

Table 2.1: Learning Tools Interoperability Comparison Table

the protocols between the LMS system and the external tools allowing external tools to function as if the tool were a native tool inside the LMS" [26]. The main goal of the LTI is to standardize the process for building links between learning tools and the LMS [27].

Undoubtedly, similar to other domains we need to reduce cost of this integration in both sides of integration (LMS vendors and external tool providers). One of the most popular standards today is LTI, which is offered by IMS Global and the main advantage its development comes from LMS vendors and publishers (IMS will get feedback four times a year). Before LTI became more popular between manufacturers, its lack of implementation was one its important disadvantages [11]. Another advantage is its other related standards from IMS such as LIS and CASA, which work very well with LTI and support it to handle other requirements. For instance, LIS is used to interact and exchange information between student information systems and learning management systems. IMS has created different standards with a compatible data schema. The compliancy between different standards will reduce overheads, cost of developments and make it easier to use them together.

To summarize, the win-win scenario offered by IMS, an LTI developer is a winner because he is developing a single and reusable product only by providing the connection information to consumer (less support cost). The teacher, as other party in this scenario, only needs to add the external tool one-time and all other upgrades and maintenance will be done on provider side, with zero amount of effort from teacher (consumer) [8]. In addition to this, there are also several customization options regarding the connection between the consumer and external tool, these include privacy controls, which make consumer able to customize usage of the external tool. Figure 2.7 shows how the LTI benefits the tool consumer and tool provider and other related stakeholders.

2.7.3   IMS Basic LTI to Integrate a Tool

IMS Basic LTI was invented to add external tools into the LMS in simplest way and with minimal problems, and to expand LMS's functionalities [28]. Therefore, it can achieve a noticeable acceptance from LMS providers [11]. This simplicity will lead us to some limitation, which should be handled consciously and find out a break-even point.

IMS Basic LTI, is a very easy plug-n-play standard for soft integration. Soft Integration can be defined as integration of a tool in a LMS through a hyperlink. Hard integration is a special kind of soft integration with additional control from the consumer on the provided tool. These controls involve several issues such as keeping relation of users and tool, transferring the consumer information to provider, controlling the users' permissions and single-sign-on authentication [28].

IMS Basic LTI is the simplest option of the LTI standard and has its own advantages and disadvantages [28]. In order to offer more complex and useful specifications to handle related requirements, IMS Global offered other standards, such as higher versions of LTI and Content Item. In order to have the ease of use and flexibility of RESTful API we can use JSON format [29] for custom fields of LTI, therefore it would remain convenient for JSON fans.

2.7.4   LTI Content Item

Content-Item is a standard message format that forces Tool Provider and Tool Consumer to interact with each other. Content-Item is part of LTI standard and is used by the Tool Provider and the Tool Consumer as a message passing standard. One of the best practices using Content-Item (content item basically created to be used to facilitate this mechanism) is to allow users a to interact with the Tool Provider and make a selection of (or multiple) instances of Content-Item and insert those into the Tool Consumer [30]. As IMS described, common usage of this standard can be in a basic LTI launch with the following scenario: user clicks on a hyperlink to

open an iframe or a new window and from that iframe can select one or more tools (conceptually very close to common meaning of an App-Store) then the Content-Items corresponding to selected items will post back to the Tool Consumer. Those items can be included in the Tool Consumer as new content or even as new LTI tool.

In addition to creator description about LTI Content Item it may be worth it to take a look at how Moodle is using Content Item. In development documents of support for Content Item it has been mentioned as a new capability which make teachers able to add new hyperlinks to the activity feed of their courses. These links would be configured by the self-configurable tool and ready to be used.

2.7.5 Community App Sharing Architecture

Community App Sharing Architecture (CASA) is a proposed method from IMS Global to help tool providers publish their tools in different catalogs and include necessary information and technical parameters in the application package [31]. CASA can be deployed as a single server or a hierarchical set of servers.

CASA is using URI Sharing Environment protocol for sharing the apps and related information including meta data and configuration parameters, which is an open source project. The CASA architecture can be divided into three different components and are listed as follows [31]:

1. CASA Engine: is querying other CASA engines (its peers based on predefined configuration) in order to discover new apps and publishing apps. Suppose two CASA Engine peers, when one of them is trying to discover apps, its peer will play the role as an app publisher.

2. CASA Engine Manger: provides a admin panel to manage CASA Engines and configuration.

3. CASA Storefront: offers the App Store for LTI tools as a web application (or at least as widget) and user can have access to the app repository which is pro-

vided by CASA Engine module. This module is not necessarily one instance. The CASA StoreFront module will provide all of the common services for an App store such as rating, search (based on tags), categorizing and so on. CASA StoreFront can be easily added to learning management systems or other consumers.

The key point about CASA is that it doesn't keep any data related to consuming the provided tool such as grades or any other additional information. That is why the inventor of LTI initiated a new project entitled "TSUGI" to overcome this weakness [15]. We will talk more about these weaknesses in Chapter 3.

2.7.6 Caliper Analytic Framework

Caliper framework is a standard way to collect different information from entities with a learning environment and use the information to offer services to educators. These services can act as parameters to measure quality and effectiveness of learning content and interactions. Similar to other standards offered by IMS it is compliant to other standards. Capiler framework makes it easier to equip the learning environment with a superior analytics feature by offering the following [32]:

- Improve learning analytics by defining parameters to measure learning activity.

- Suggesting a common format for the meta-data related to learning data.

- Suggesting some measures for learning activities which make it easier to compare them.

- Capability to connect other systems (such as smart agents) to complete analytics.

- Recommend best practices on system architecture.

- Simplifying the data gathering from LMSs.

- Track student activity to help vendors to map their LMS specific measures to something common between other alternatives and therefore it makes them all able to compare this data and conduct more effective analysis.

Figure 2.8 is an overview on the Calipar Analytic framework, which explains the proposed system design.

2.7.7   Common Cartridge

Similar to its alternative, SCORM from ADL, Common Cartridge was invented to standardize the packaging process of learning materials and grading in order to facilitate information exchange between different IMS Conformance Certified learning environments. Using Common Cartridge learning objects can easily be generated and used by learning management systems or similar consumers. Common Cartridge is created to provide a popular and standard method of packaging digital course materials (and their assessment) so they can be used as a guideline for both already created materials for CC compliancy and used as a publishing model for online courses or different course publisher applications [33]. The IMS Global website provides an excellent representation of information exchange between LMS and Learning Object Repositories (LOR) via Common Cartridge's (Figure 2.9).

According to [4], SCORM Common Cartridge has advantages over other alternatives and they are listed as follows:

- Flexible

- Cover new needs

- Content Authorization

- Collaborative Forum

- Outcome Report

- Accessibility

2.7.8   OneRoster and Learning Information Services

One of the best applications of integration and interoperability is to integrate Learning Management Systems with Human Resource systems and specifically Student Information System (SIS). In order the facilitate this process, IMS Global created a set of standards under title of Learning Information Services and those standards work mostly with information exchange based on SOAP and REST protocols. It covers information in different categories such as personal info, memberships, courses and grades. In addition to proposing the system design and a best practice of information exchange process, defined under LIS, OneRoster (as subset of LIS) it also provides a standard file format and data schema for CSV files which are exchanged between different systems which is specifically is being used for roster and grade exchange [34]. Obviously, this interface and standard can be used by other systems other than SIS but the main intention behind this is to facilitate integration between SIS and LMS. Figure 2.10 shows the proposed system architecture benefit from OneRoster.

2.7.9   Other Standards and Specifications from IMS

There a few more standards offered by IMS Global; these standards are not useful enough to be discussed in detail, so they will be mentioned quickly. The standards are listed as follows:

- **OpenVideo:** This standard is mainly for capturing videos, which technically is an XML interface, and it provides services such as: parallel capturing, offline recording and background upload, video editing and text search based on meta-data (for the presentations slides). This content type is useful because OpenVideo compliant consumers can easily add content and offer a rich playback experience for users.

- **Question and Test Interoperability (QTI) and Accessible Portable Item Protocol (APIP):** QTI as a specification was invented to offer interoperability for grading of test objects (quizzes, etc). It provides related data schema for questions, answers and results by specifying a proposed structure for maintaining the data corresponding to a test, which can be questions, tests and result evaluations. This data is easily exchangeable to other consumers such as learning management systems, student information systems, etc. [35]. APIP is a specification that provides a standard file format and data model for exchanging information data units, which contains questions and assessment corresponding information. There are two sub-standards under APIP that can handle both transferring test data from one repository (such as test bank) to another repository. The second one focuses on delivering required info and resources to improve the tests in order to pass the accessibility check list and be useful for students with disabilities or special conditions [36].
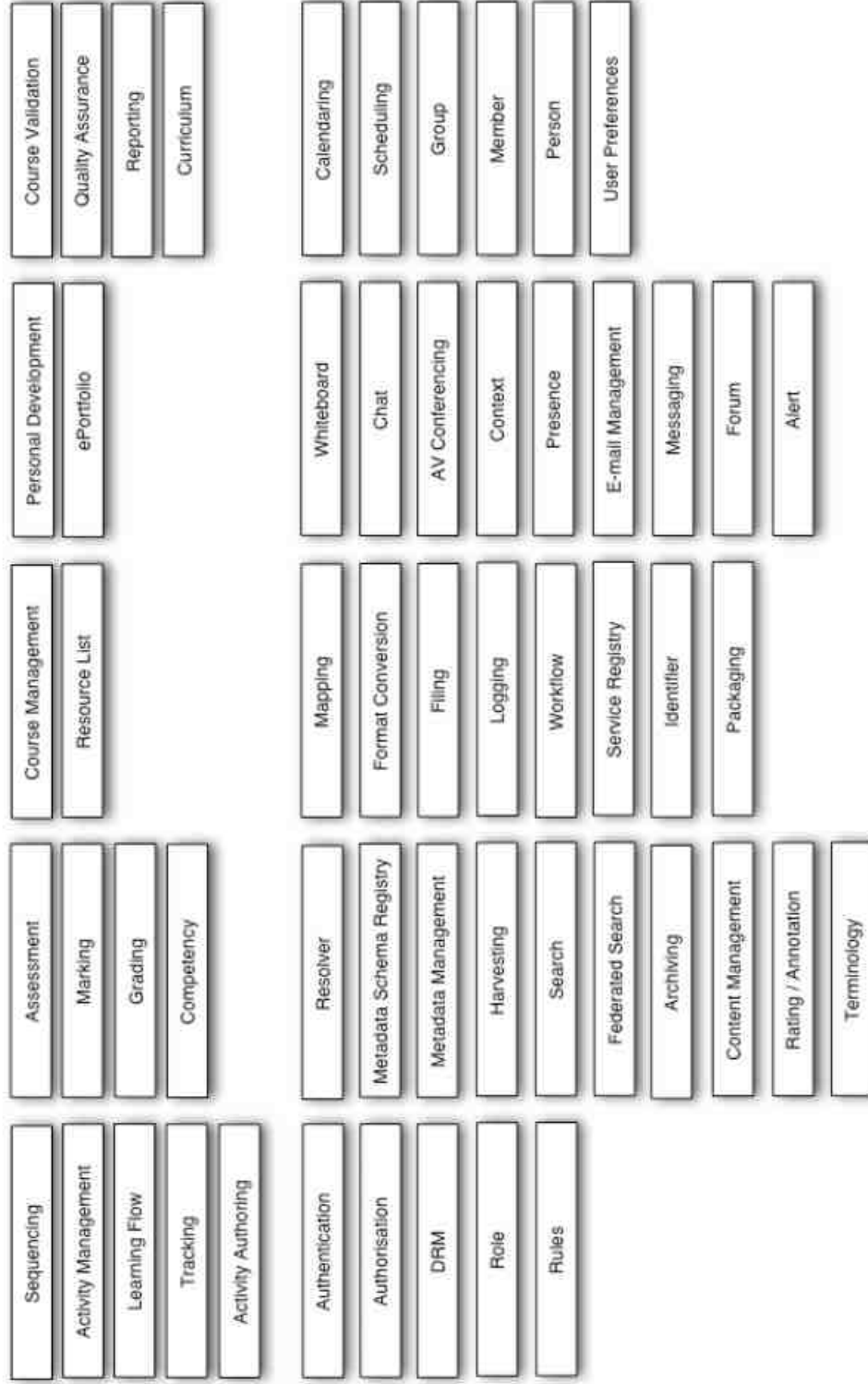
Course Validation | Quality Assurance | Reporting | Curriculum

Calendaring | Scheduling | Group | Member | Person | User Preferences

Personal Development | ePortfolio

Whiteboard | Chat | AV Conferencing | Context | Presence | E-mail Management | Messaging | Forum | Alert

Course Management | Resource List

Mapping | Format Conversion | Filing | Logging | Workflow | Service Registry | Identifier | Packaging

Assessment | Marking | Grading | Competency

Resolver | Metadata Schema Registry | Metadata Management | Harvesting | Search | Federated Search | Archiving | Content Management | Rating / Annotation | Terminology

Sequencing | Activity Management | Learning Flow | Tracking | Activity Authoring

Authentication | Authorisation | DRM | Role | Rules

Fig. 2.5.: e-Learning Framework Services [18]

Fig. 2.6.: Chronological Trend of SCORM
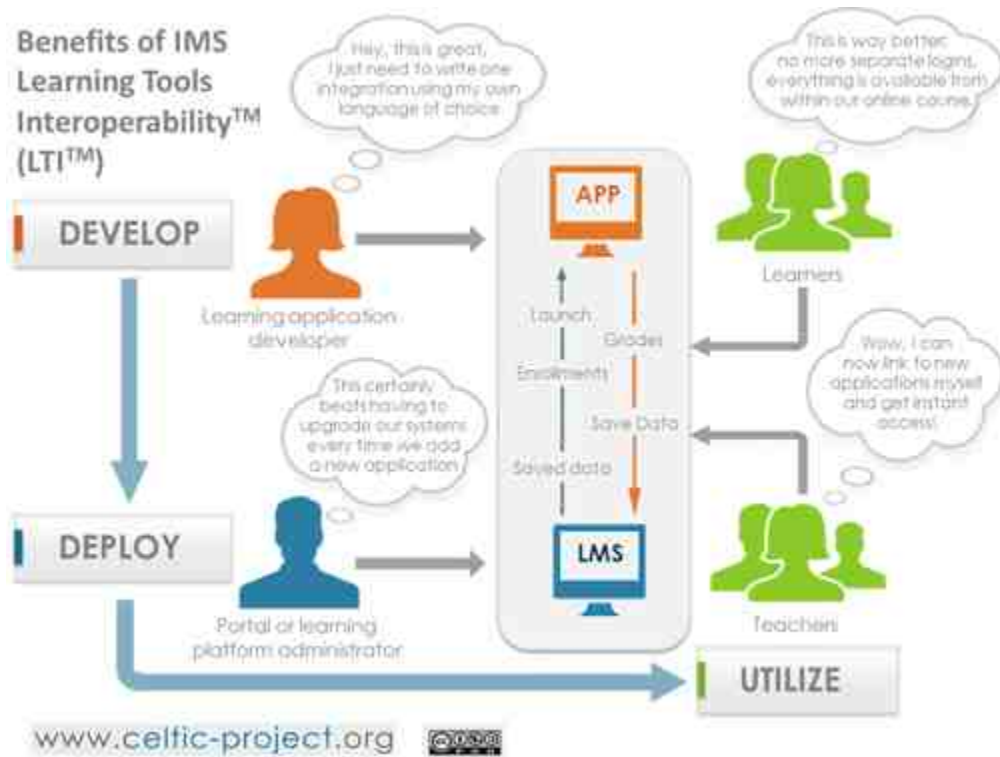
Fig. 2.7.: A LTI Successful Scenario

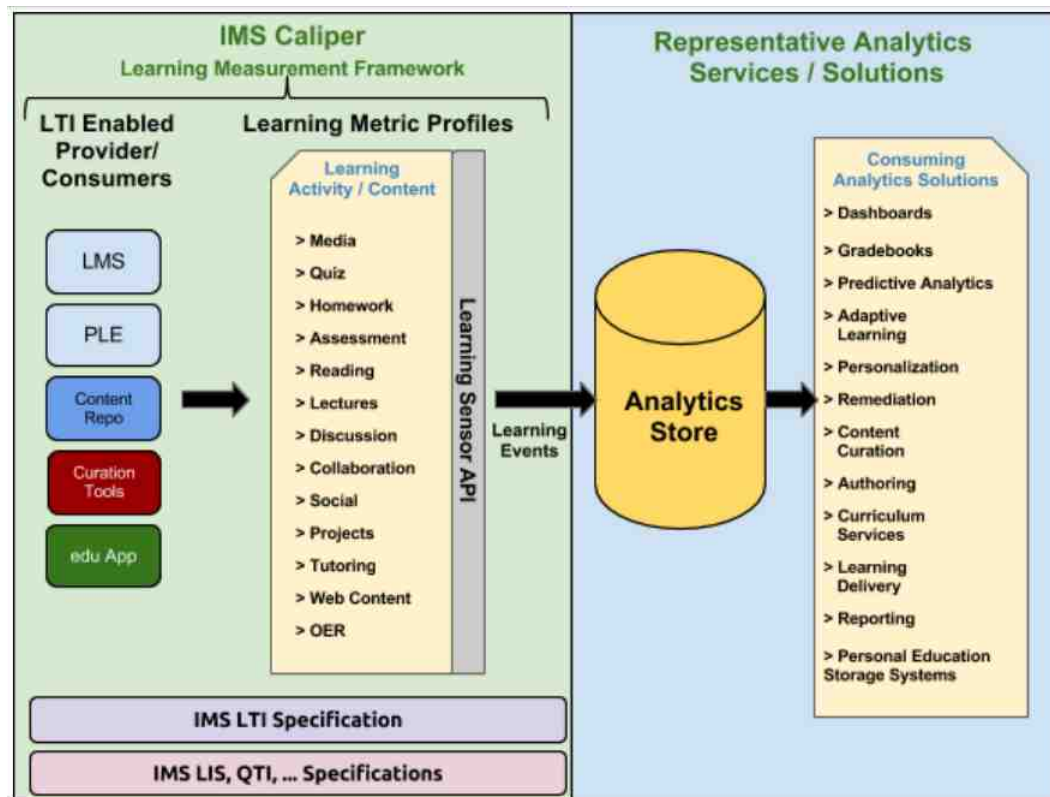Fig. 2.8.: IMS Caliper's Architecture Overview [32]

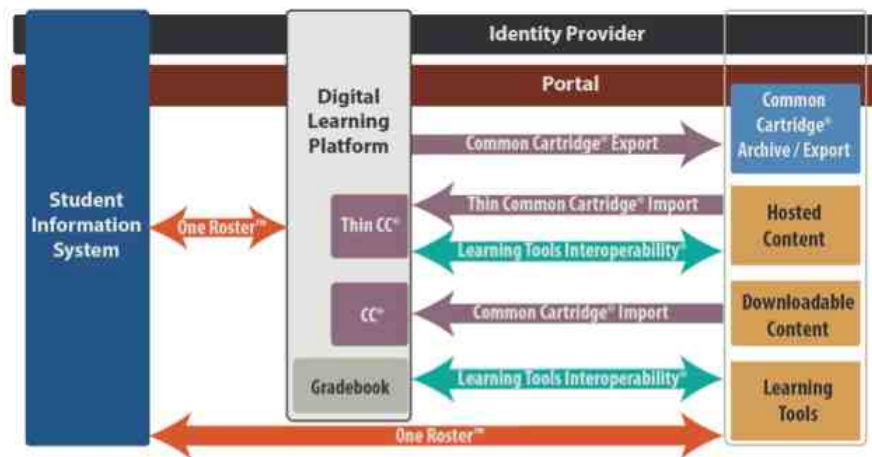Fig. 2.9.: Common Cartridge Role [33]

Fig. 2.10.: How OneRoster works in Learning Context [34]

3. LTI APP-STORE: A NEW APPROACH

3.1   Background

Today's App Stores have inherited their main idea from an old App Store from 1991, the Electronic AppWrapper. The AppWrapper was the electronic version of AppWrapper application catalogue for distribution of apps and multimedia content. The AppWrapper, at first, was only a printed catalogue, which demonstrates the concept of App Store regardless of its format. Apparently, this App Store was developed by Paget Press, Inc. in Objective-C on the NexT platform. These details, while interesting, aren't the main focus, however, they do show that electronic distribution, encryption and providing digital rights all on a certain environment can be considered core characteristics and they are still important in designing of the App Stores as they've shown in last 20 years. However, other requirements such as rating and categorization came to the scene because of the massive number of applications nowadays. Figure 3.1 is a screenshot from the first app store:
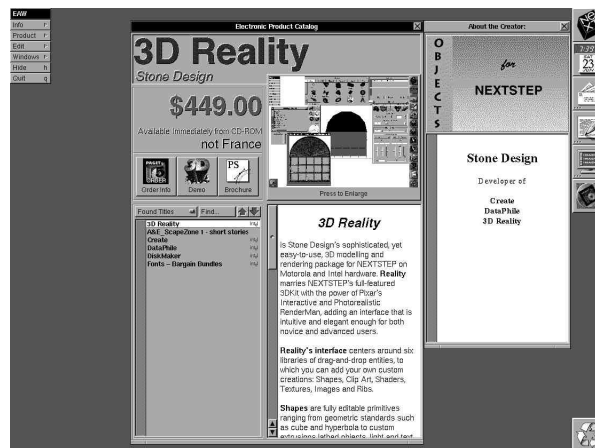

Fig. 3.1.: Electronic AppWrapper (first app store)

App Store is basically a platform for the distribution of applications. This platform can provide its service to different environments. For instance, Apple's App-Store or Google Play Store or Samsung App all provide their services in different environments with different conceptual definition. Learning systems can be recognized as another environments, which need App Store. Therefore, there is need for a new App Store architecture for LTI apps (tools) to be proposed. There is also a need to adapt the App Store concept to Learning Environment by improving the App Store existing architecture in a few ways to enhance the quality of learning tools.

3.2 Examples of LTI App Store

As we discussed before in Chapter 2, Community App Sharing Architecture (CASA) was proposed as a white paper [31] in 2013, as a suggested architecture for sharing applications, which is called App Store in market. Figure 3.2 briefly shows the CASA architectures and how CASA interacts with other systems.

One of the existing examples of an LTI app store, based on CASA, is Eduappcenter (www.eduappcenter.com). Eduappcenter is up and running, created by Instructure, the company who also created Canvas. Eduappcenter is a good example of app store that offers LTI external tools (applications) in different categories and other LTI compliant apps; users can add these apps easily to their courses. This app store also provides a RESTful API, which provides a list of applications, single applications and a review feature for each application. Vendors can add their LTI applications into Eduappcenter and make it easier for tool providers and consumers to manage these applications. We have designed and developed the CN Post, an LTI application that offers a subset of functionalities from CourseNetworking and we published it into Eduappcenter. It is available to every Canvas site all around the world. Eduappcenter has also provided the capability of easy configuration to providers and consumers by offering their services as different extensions; those extensions are supported by Canvas. CN Post is an IMS certified LTI tool and has been approved by EduAppCenter.

Fig. 3.2.: IMS-GLC Initiative CASA [31]

This tool will be reviewed and explained in detail in chapter 4. Figure 3.3 shows the CN Post in EduAppCenter.

3.3 Need for App Store and Something More

In learning environments, similar to other environment applications and external tools, can bring new functionalities or content to consumers with no development. Recently, these tools (from LTI compliant providers) became popular and thanks to IMS, and the LTI providers can now easily develop LTI tools or make existing tools and services LTI compliant. App stores, designed to distribute applications in a certain platform and for LTI compliant external tools LTI App Stores, should provide
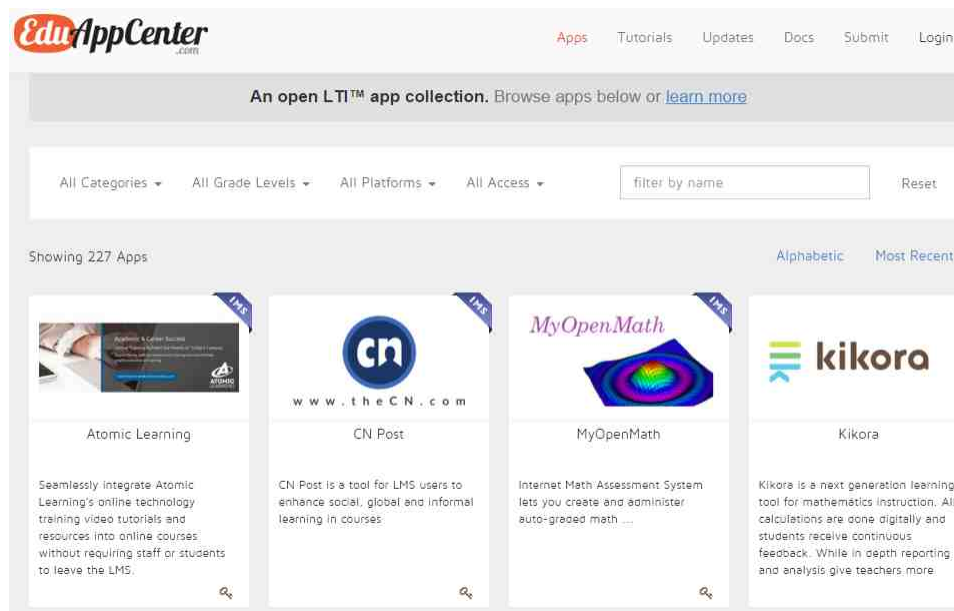
Fig. 3.3.: CN Post in EduApps

the enhancements in categorization and installation of the apps. Eduappcenter is now providing features for LTI apps, mentioned briefly as follows:

1. Admin panel for LTI providers to manage their apps and configurations: In this section providers can manage their organization, add other people to their organization and maintain or modify related information. In addition, they can also modify and update their submitted LTI tools and configurations.

2. App publishing tool: Tool providers can submit their tool (app) using the publishing tool, which is basically an input form. In this form, tool providers can register their LTI tool and specify different attributes about the tool including: xml configuration file, visibility, authentication procedure for consumers, basic information about provider, installation and testing instruction, logo, supported platform, category and Canvas specific configuration (extension).

3. Advanced configuration file builder for LTI apps: One of the options regarding xml configuration is use of the Eduappcenter's built in xml configuration creator. This widget helps users create a very clean and error-free xml configuration file (Figure 3.4).

4. App categorization (based on different characteristics): One of the inputs in the publishing tool is the application category, which allows users to put their application with one or more categories such as Assessment, Community, Content, Math, Media, Open Content, Science, Study Helps, Textbooks/eBooks, Web 2.0, Completely Free, Beta. However, this categorization is only usable inside of Eduappcenter and will not offer to external consumer of its RESTful API.

5. App Review: In Eduappcenter, users can write reviews about each tool and/or rate the tools by giving them stars on a scale of 1 to 5. However, in new version of Eduappcenter the Instructor company has disabled it.

6. RESTful API

   a. List Apps: This API third party application can list all of the available tools in Eduappcenter. The output is JSON and the sample URL is:

   "https://www.eduappcenter.com/api/v2/lti_apps (Last Date Accessed: April 1, 2016)"

   b. Get details for an individual app: Similar to the tool listing API, another one will provide the ability to retrieve details for a certain tool. This API is accessible from "https://www.eduappcenter.com/api/v1/lti_apps/:id (Last Date Accessed: April 1, 2016)" and the id is the unique identifier that is retrieved using tool listing API.

   c. Submit and retrieve reviews for an app: Every single tool can have its own collection of reviews. Eduappcenter's API allows third party applications to

submit and retrieve reviews. Similar to retrieval of tools list, but it provides required end to handle the POST request to submit new review.

7. Easy installation: Eduappcenter and Canvas are using the LTI content-item from IMS and this protocol as message format to enable them to install the tool into the LMS seamlessly.
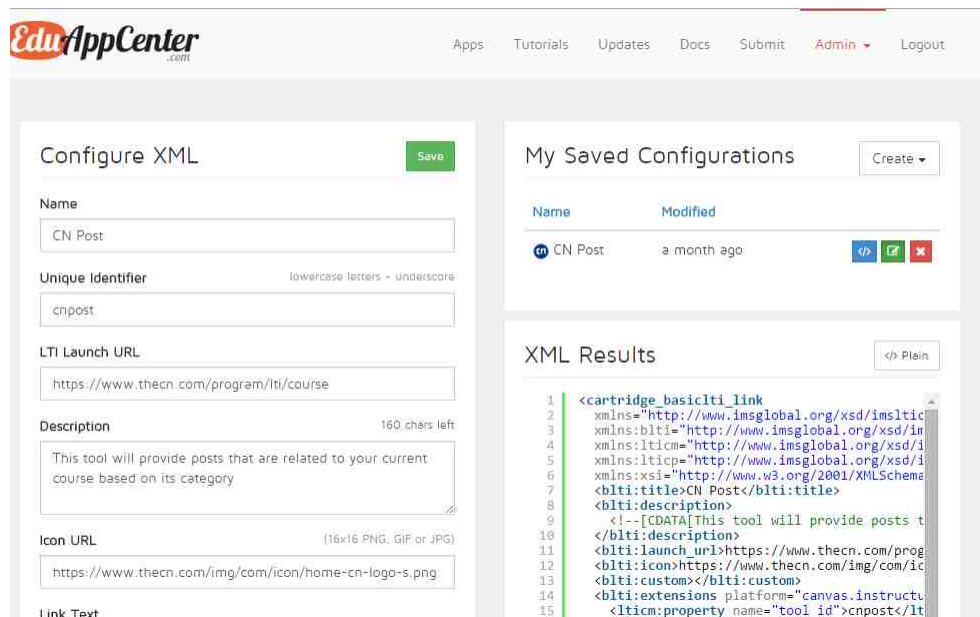

Fig. 3.4.: XML Configuration Creator with GUI in Eduappcenter

The above features and facilities, provided by EduAppCenter, make life easier and serve as a nice showcase for LTI applications, but imagination and creativity are endless. Actually, these features have not conflict with each other, and EduAppCenter is one of the most successful examples (probably dominant in the market) of LTI app stores. However, if we look at this product from a different point of view with a bit more of a futuristic mindset, we can conclude the need to equip current definition of LTI app store to more options.

For instance, EduAppCenter has nothing to do with consumer key and secret generation for LTI applications and they only provide an external link to vendor's

website, mostly because they ask users to fill out a form or send an email to generate the key. There are a few missing characteristics with this approach.

The first disappointing point is the lack of a "get" or "purchase" button to provide the capability generating a consumer key and a secret key for a chosen LTI app. In the current version, users have to go to the tool provider's website or email them to get the consumer key and secret. This inconvenience probably decreases user interest and motivation in these applications. The importance of this solution would be more tangible if the provider just wants consumers to have consumer key in order to maintain their data more efficiently, but not for charging them. Therefore, the procedure could be very straightforward and automated behind the scenes. A suggested approach is a RESTful message passing between LTI app store (e.g. EduAppCenter) and the provided endpoint by tool providers to generate consumer key and secret and finalize the installation process.

The second procedure that needs to be refined, is the automatic installation of tools which needs to be much more standard. In this version of EduAppCenter, a tool can be added to learning management system as a web-service and the LMS provides the GUI by itself and behind the scenes, calls the EduAppCenter's RESTful API to provide listings, retrieval for app's details and reviews. A more convenient and reasonable way would be to use LTI Content-Item message passing to provide ease of installation. In addition, for each consumer, the app store could have a series of configurations such as hidden-apps-list and other preferred options. We will discuss self-configurable LTI compliant external tools and vender's app stores in detail later in this chapter.

The third, and final, concern regarding EduAppCenter is a feature often missed by other alternatives and results in a lack of LTI launch information. This lack of information presents after the installation of an external tool on the consumer side; no data regarding this relationship is held or shown in the app store. This data can be helpful for future users and could also create more meaningful statistics. In addition, institutions, as consumers, prefer to host the tool on their own servers. Another

alternative for these consumers is to utilize a third party provider for external tool use, but at the same time preserve this tool on their own server.

Failure of the server may be another issue, which could happen after an LTI provider has stopped offering its service but consumer need for data is still there. For instance, suppose that TurnItIn went bankrupt or they made decision to shut their service down, then what would happen to the university's data? We will discuss TSUGI, as a new framework or even platform for LTI compliant tool/app development later on this chapter and especially how TSUGI can benefit the conventional LTI app stores.

3.4   LTI App Store with New Characteristics

3.4.1   Overview

By considering all of suggested solutions and improvements, in order to enhance the existing LTI app stores we find that we have to come up with a more detailed proposed solution. In this model we have tried to introduce a new composition of CASA architecture, TSUGI and a few suggested refinements to make the LTI app store more reliable, usable and convenient.

3.4.2   Self-configurable LTI tool

The basic idea behind the self-configurable LTI tools is to enable vendors or LTI provider developers to develop their app stores (conceptually) by taking advantage of content-item. LTI tools can provide services like TurnItIn or content similar to Pearson or YouTube. Therefore, providers can bring together all of the different services or content in the form of applications and then the consumer (the institution) would only need to add one application (LTI compliant external tool) from each vendor. The first application from a certain vendor could install other applications from that vendor and behave similar to an app store.

LTI tool providers could bring a few of their general services like a consumer key or a secret generator module as a tool into the self-configurable tool (app store). After the consumer key and secret for a certain consumer are generated, the following tool installations can be done with little to no effort by using the same consumer key and secret. Subsequently, services can be broken-down into different modules and that means a categorization by vendor.

CASA has already been designed to reasonably resolve about 80 percent of the above problems, but still, the problem with hosting LTI apps remains. This actually isn't a problem; the solution is simple, CASA can be equipped with powerful platforms like TSUGI to increase effectiveness in learning environments. Figure 3.5 shows the CASA's role in distribution of an app when the consumer tries to add an LTI tool through CASA using content-item message protocol.
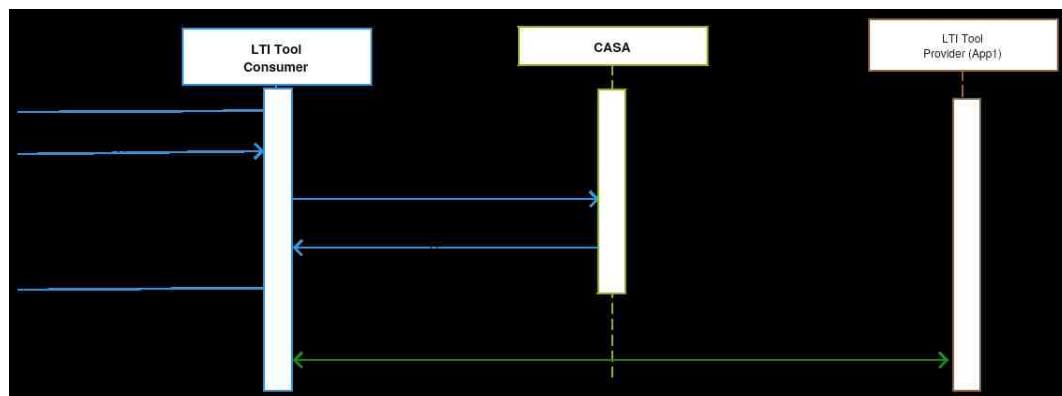
Fig. 3.5.: Add an Application to LTI Consumer through CASA

In the past, CASA has been developed by different people and organizations. One of its best implementations, which helps users better understand the concept behind the CASA, was Sakai under supervision of Dr. Charles Severance. Actually, UX in Sakai is not very user friendly due to several different options given to administrators by default help developers get some ideas about how CASA works. In addition, the content-item message standard is not yet finalized by IMS and the implementation of CASA in Sakai is not yet certified by IMS.

3.4.3 App Store Support in CourseNetworking

Learning management systems, similar to other software systems, will benefit from seamless integration, which will in turn increase the LTI. Generally, IMS compliancy will enhance and facilitate this integration and minimize the required effort for the integration. In Figure 3.6 the interaction between CourseNetworking and TSUGI-assisted app store is shown. Although, the two main goals of the TSUGI is to provide a platform where LTI tools can be developed easier and faster while also maintaining data in reference to the instance of an app. This is used by consumers, but it can also improve user experience by equipping its interface with self-configurable LTI apps to act like a small app store. Technically there is no limit to the number of applications the app store can have, which is provided by a single TSUGI instance. Also, this TSUGI instance can provide applications from different groups and companies in a single instance. However, the recommended approach, and probably the best practice of TSUGI is to carefully choose the applications based on each instance. For example, a single company can provide all of the apps or the consumers associated with that instance can be in a common geographical segment.

The conceptual approach in CourseNetworking, regarding management of the self-configurable LTI tools, is to simply put them on the left hand side navigation menu and present them as a basic launch (LTI v1.0) or as a basic launch with an outcome push back (LTI v1.1). Later on, CourseNetworking could add the support for Canvas extensions such as homework submission, file-picker and other useful LTI tool categories as long as there is no conflict with the concept behind the CourseNetworking and its essential entities as a social educational learning management tool.

CourseNetworking has the capability of adding LTI tools as basic launch hyperlinks to its course navigation on the left. These LTI compliant tools can be just like normal LTI external tools or they can be app stores. In the current version of CourseNetworking, V4 Mar 07 2016, it is only supported by adding LTI tools as basic launch to a course by a course instructor. The next phase of improvement,
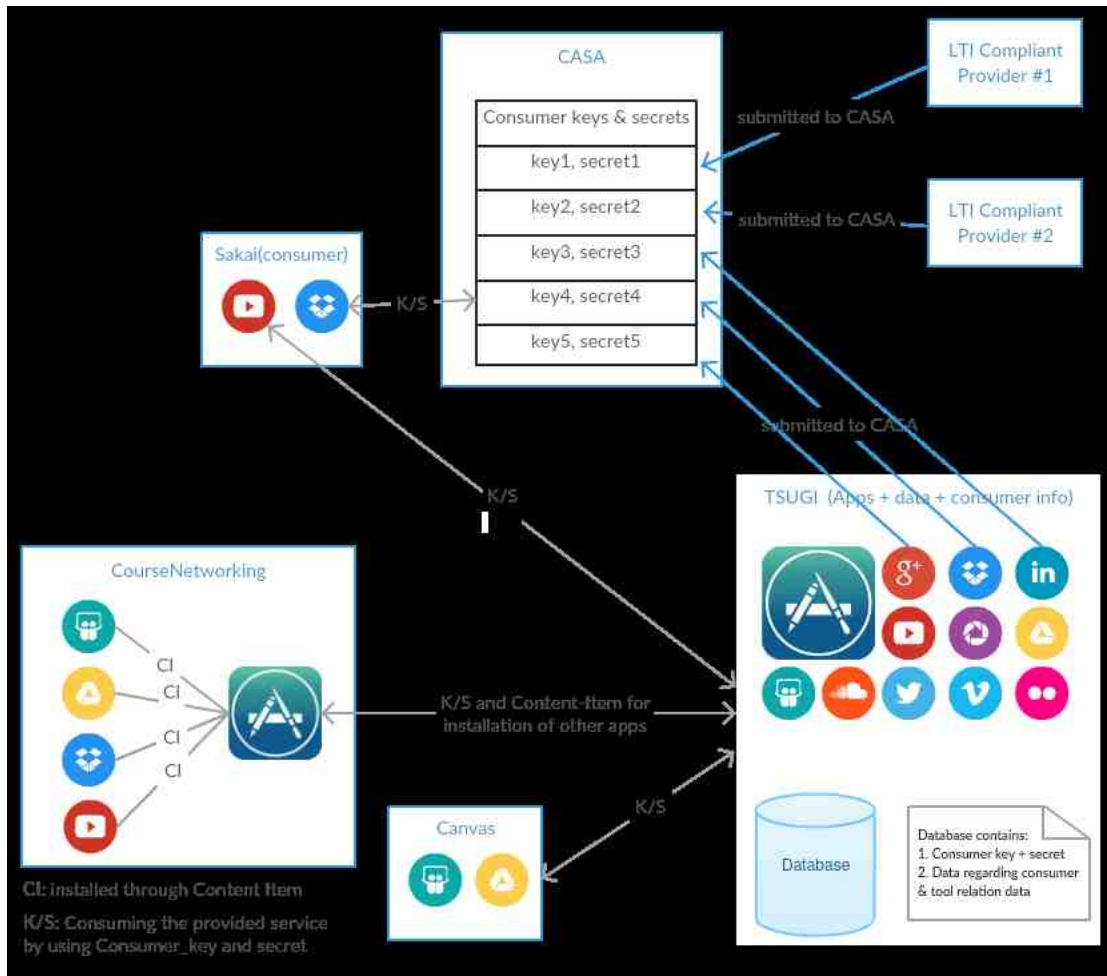
Fig. 3.6.: CourseNetworking, CASA and TSUGI Interaction

in terms of LTI compliancy for the CourseNetworking as a LTI consumer, is to add LTI app store (self-configurable) support. Following the best practices from IMS-GLC's certified learning management systems and adapting their used solution to CourseNetworking will save a lot of time in the technical aspects and most of the time can be dedicated to designing the user experience. The first step is to add the capability of adding an LTI tool in three different modes:

1. System wide: means the LTI tool (whether self-configurable or not) that has been added in this mode will be available in all courses around CourseNetworking. The visibility of these courses can be managed by a system administrator.

2. Institution wide: means the LTI tool will be installed on all of the courses in an institution. Instructors should be able to enable/disable the tool for each course.

3. Course wide: means adding a LTI tool by instructor to his course and the tool will be accessible only for the course members.

In above scenarios, the system administrator has the highest permission and he or she can overwrite everyone else's preferences. In a hierarchical form, instructors of the independent courses and institutional administrators have their own preferences, which can be only overwritten by master administrator. The lowest level of permissions is that of the instructors of the courses at an institution. Both a system administrator and an institution administrator can overwrite their preferences.

The second step will be adding the capability of adding LTI tools from another tool, LTI tool, which is defined as self-configurable LTI tool or LTI app store. In order to do this, we have to pass a specific URL to the LTI app store while launching and that URL will be used to by the app store to pass back the details regarding the new LTI tool to be installed to CourseNetworking. A preview from the technical perspective can be found on Figure 3.5 and the LTI Content-Item, the protocol and message format, which is being used to handle new tool configuration/installation.

The third step is to add the capability of mapping passed back grade (outcome) from LTI providers to CourseNetworking, as consumer, and bind that capability to the gradebook item in CourseNetworking. This could be a little bit more complicated because in a social educational learning management system the definition of gradebook item, in some cases, is different from the conventional meaning of gradebook item. Therefore, it should be adapted to a new environment and it will need careful analysis and consideration to avoid any conflict with existing definitions and entities in CourseNetworking.

An optional step can be adding the capability of adding EduAppCenter, which is ran by Instructure, and will open a new window for adding high quality LTI tools to the courses. They offer a very clean and useful RESTful API in order to access their LTI applications.

3.4.4   TSUGI: A Container for Installed Applications

TSUGI was initiated in Sep 15, 2013 by Dr. Charles Severance, inventor of LTI, as a standard framework to make simplify the development of LTI compliant tools. Similar to the general definition of framework, TSUGI provides a simpler API for LTI tool developers to compare to working directly with IMS standards. The developed LTI tool will be exactly the same as conventional LTI tools from outside companies and it would be usable by LTI compliant consumers, and can be certified by IMS Global conformance tools. TSUGI is an active open-source project, which was recently ported to Java as well, enabling Java developers to create LTI tools based on TSUGI in Java.

According to TSUGI's definition, it can benefit tool providers by separating the LTI tool's data and making them independent (standalone). The next step, looks like a dream at the moment, is to choose TSUGI as the standard proposed platform to be used in CASA for developing tools by vendors (or at least a de facto). In this case each vendor can provide its own service or content as an LTI compliant external tool and offer their tool as a self-configurable tool or as a conceptually small app store. This can simplify and even automate the procedure of LTI tool development for the content-based LTI providers (e.g. publishers and Wiki pages in institutions) to develop and advance complicated LTI tools to save time. This will bring TSUGI's benefits to the conventional LTI app store architecture, the CASA and the capability of gathering all related information in one place (hosted by consumer). For instance, analogy about the advantage of TSUGI is the smart phone. In smart phones, there are excellent app stores (Apple Appstore and Google Play Store) that host the applications, but

all of the critical or necessary data is hosted on the phone. In the LTI tools' world, sometimes the need to store the critical information somewhere other than provider or consumer side and independent to all of them arises. TSUGI will take care of everything. Figure 3.7 is a brief demonstration of what we have when TSUGI placed between the CASA and the consumers.
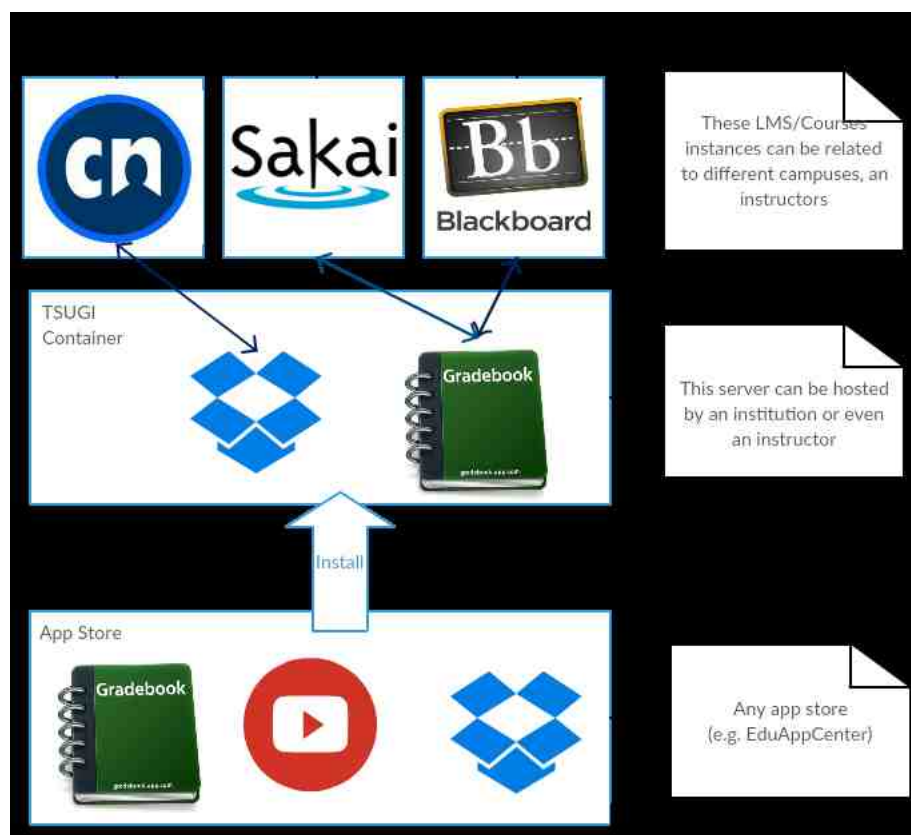

Fig. 3.7.: TSUGI Connects Different Courses

Finally, after introducing the new architecture and putting CASA and TSUGI together as a case study based on the research done in chapters 2 and 3, we have developed an LTI tool to bring a subset of features from CourseNetworking to conventional LMSs such as Canvas and Moodle. This tool can be seamlessly integrated into any LTI compliant consumer and can equip them with new social features. The conceptual design and development details of this LTI tool are explained in Chapter 4.

4. CN POST: A CASE STUDY

4.1 A Glance at CourseNetworking

Classic learning systems are offering their services with a focus on content delivery, making life easier for students and instructors, compared to traditional classroom. Users' collaboration and exposure to social elements in today's world can be considered the main features in eLearning contexts [18]. CourseNetworking (CN) introduced a new concept in 2012, in addition to classic course management, the use of social networking as a leverage to upgrade the user experience was founded. In CourseNetworking, learning has been built based on social networking to introduce new concepts, entities and methods for teaching and learning. This approach allows CourseNetworking to be scalable both conceptually and technically [37].

CoursNetworking ports and adapts useful characteristics from social networks into conventional CMS/LMS to enable their users to collaborate and connect with each other more effectively and efficiently. CourseNetworking offers features such as Posting Tools (authoring tool), reward system/gamification, personalized social networking and course categorization. These features assist users in involvement within different types of learning process such as socially mediated learning, reflective learning, rewarding learning, intercultural learning, etc. [37].

Learning processes are always changing and we need a LMS like CourseNetworking, which adapts to these changes. Institutions are ready socialized learning but learning management systems have failed to do that. We summarized the effective factors as follows:

1. Different specs are being resolved thanks to IMS Global and its standards, along with LMSs who are supporting specs from IMS Global.

2. There are not any social features in old LMSs (totally new concept then it is hard to map old concept and content to new form). For instance, collaboration on an assignment in social context needs a new form of evaluation.

3. Old ones focus on the institution and courses instead of the user.

Aside from the factors mentioned above, a system like this is needed to utilize lessons learned from older LMSs, so we may not reinvent the wheel, so to speak. By using those lessons, we can also use easy and inexpensive interoperability and integration with even the usable parts of old systems.

4.2 Overview

CourseNetworking offers a number of new features, which provide new learning opportunities for institutional users, in addition to traditional features. This shows that CourseNetworking is not only comprehensive enough to handle all institutional requirements, but also that the institutions are already using traditional learning/course management systems and that work. Therefore, it would be exceedingly hard to convince them to use CourseNetworking as a modern, social-educational environment. In order to resolve this, CourseNetworking offers a subset of its services as an external tool, which can be easily added into any LTI 1.0 compliant learning environment. CN Post was created to connect traditional LMS users to each other, from different institutions, and courses all over the world. This newfound connection opens up opportunities like Reflective Learning and Socially mediated learning. In the next subsection excellent features from CN post will be reviewed.

4.3 Characteristic

4.3.1 Overview

CN Post, as a simplified version of CourseNetworking, is providing a useful set of features to conventional LMS users. These features and services mostly come to add social and engagement capabilities, are categorized and explained as follows:

- **Posts:** Post is one the simplest but most powerful concepts in social networking which is being used in CN in order to create content and share that content easily with other people in your network, class or global classroom and it can be reviewed and rated. This content unit can be text, photo, video, any attachments or even sharable content object reference (SCORM). Posts can be reviewed, liked and reflected on by people who are meet visibility settings set up by a user.

- **Polls:** Polls are similar to Posts in terms of visibility settings and reviewability but the content is different. Polls are used to survey opinions, review knowledge, gather data, or to conduct course evaluations.

- **Events:** Events have been defined based on their traditional concept and have different attributes such as time and location but they are inherited from posts and have visibility settings and reviewability options. Events can also be added to the Course Calendar and by doing this they take advantage of the CN notification system and CN Calendar (which is accessible from full CN).

- **Hashtags:** All content types including Posts, Polls and Events can be tagged simply and conceptually; tags are equivalent to course discussion topics, which can be created by instructors or students. Hashtags allow students to search and pull up contents from hot discussion topics with a single click.

- **Global Classroom:** Each course in CN can belong to a certain category based on its content and relevancy to other courses. Courses with relevant character-

istics can form a new concept, called Global Classroom. This allows learners in each of the courses in a Global Classroom to post (Posts, Events and Polls) to their Global Classmates, as well as course classmates. This is one of the most important characteristics brought to traditional LMS by CN Post.

- **Anar seeds:** CN measures user's participation and this measurement is called Anar seeds (Anar is a Persian word which means pomegranate). This measurement shows the participation of users in a certain course, or generally in CN. The number of Anar seeds can be used to show student's engagement and participation. In addition to the benefits of Anar seeds as an index of participation, it is also gamification, which encourages student involvement psychologically.

- **Engagement Analytics:** CN offers different types of analytics based on Anar seeds and other factors such as login time. The analytics are mostly a visual representation of collected information from users' participation. Analytics on participation in CN can be prepared based on different indexes such as logging in, number of collected Anar seeds, number of posts, number of reflections (comments) and number of likes all within the course. Aside from participation analytics, CN offers analytics for gradebook and course content. The Word Cloud Analytics uses a Word Cloud graphic to visually represent popular discussion topics emerging in course postings.

Although CN Post has the capability of offering other features such as Tasks, Syllabus, GradeBook, DropBox and Quizzes, in order to avoid conflict with these features on the host LMS (the LTI compliant environment which is consuming the CN Post's service) CN Post eliminates the features which already are available on host LMS.

In the next section we will go through the CN Post's main features to showcase how this modern tool can benefit traditional learning management systems.

4.3.2   Enhancements to Learning Management Systems

CN Post can be easily added to any LTI 1.0 compliant consumer, which means it can be added as a Basic LTI Launch tool. In this case study, we have deployed and tested CN Post on both Moodle and Canvas. In Moodle, it appeared as an External Tool in the list of different types of activity. In Canvas it automatically appeared on left hand side of course in the navigation menu, thanks to Canvas' LTI extension, which very popular, yet still unofficial in IMS.

After CN Post was successfully added to LMS, users could automatically obtain a CN account by clicking the CN Post from a LMS and it would provide a simplified version of CourseNetworking to the user. This click breaks the limitations of traditional classroom and users enter a global academic network. This tool opens as an iframe in middle of the consumer or can be opened in a new browser tab. In this tab, the user (registered on consumer, e.g. Canvas) is automatically logged in to the CN and the course corresponding to the one on consumer side has been created in CN, and the news feed of that course allows students and instructor to connect. In the "Features in detail" section, all of the features are described. Figures 4.1 and 4.2 show the CN Post in Canvas and Moodle, two of the most popular open-source LMSs.

CN Post also allows students and instructors from any LMS have access to the full CourseNetworking website and its mobile app. This product also offers a premium feature that allows institutions control over the courses, users and users' permissions, and it includes several customization options.
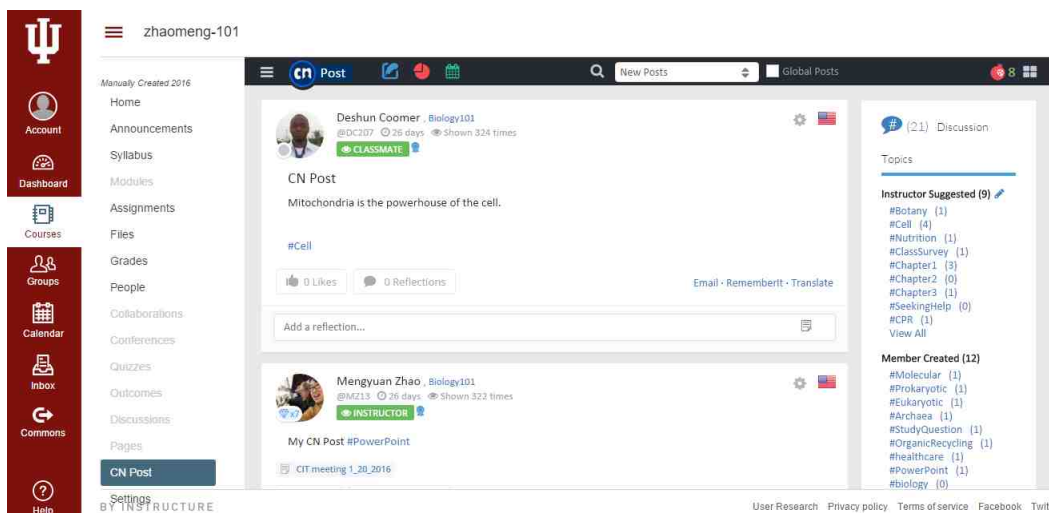
Fig. 4.1.: A Quick Look at CN Post in Canvas
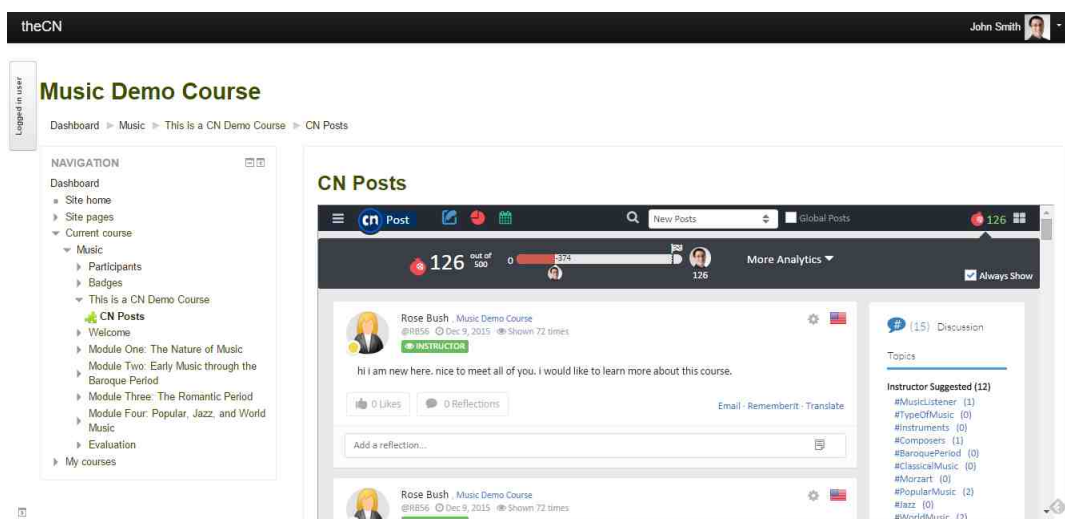
Fig. 4.2.: A Quick Look at CN Post in Moodle

4.3.3 Features in Detail

In this section all of the features from CourseNetworking in CN Post are reviewed in detail. This set of features can make the user able to generate the content and add enough semantic additional information to the content. This additional information will improve the quality of personalized content delivery significantly.

4.3.3.1 Posts

As we mentioned in the overview section, Post tool is the most important and essential content type in CN Post, which transforms passive content into active content by allowing posts to be liked and reviewed. Post has a variety of capabilities. For instance, post content can be a combination of text, image, YouTube video, external link, any attached file (downloadable) or the sharable content object reference (SCORM - learning object) files. For SCORM files, the post author uploads zip files (with SCORM structure) and CN will take care of parsing the file and provide a nice platform that enables users to go through the different sections in the SCORM file (Figure 4.3, 4.5).
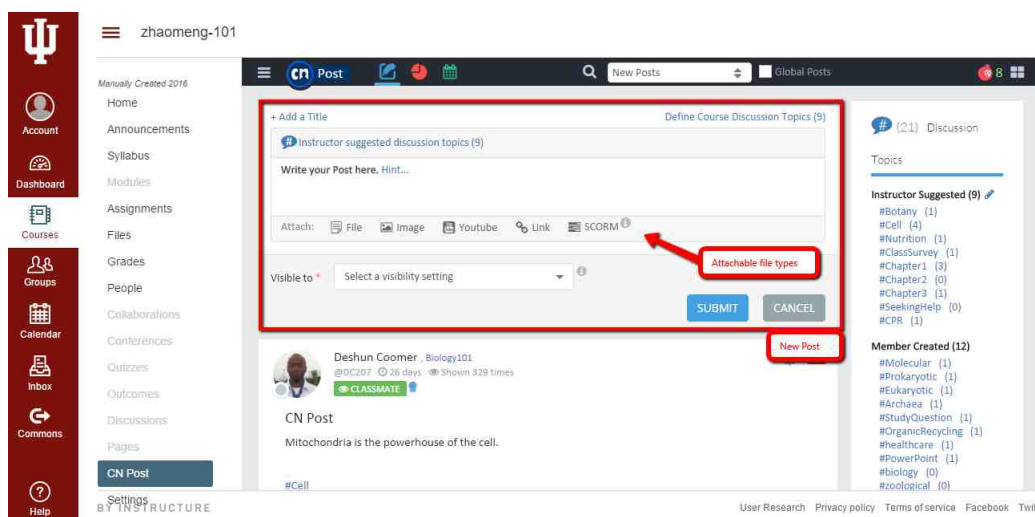
Fig. 4.3.: New Post View and Attachable File Types

Text in a Post can be a simple text or it can have several hashtags (will be discussed separately in this section) or it can mention or tag CN members. These hashtags are created by users (student or instructor) or suggested to them based on predefined hashtags created by instructors (Figure 4.4).
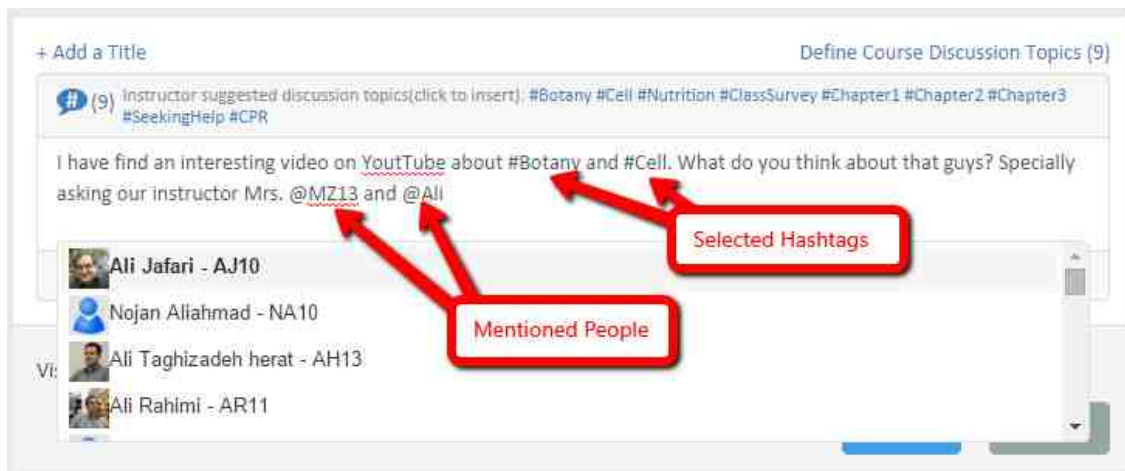
Fig. 4.4.: A New Post with Hashtags and Mentioned People
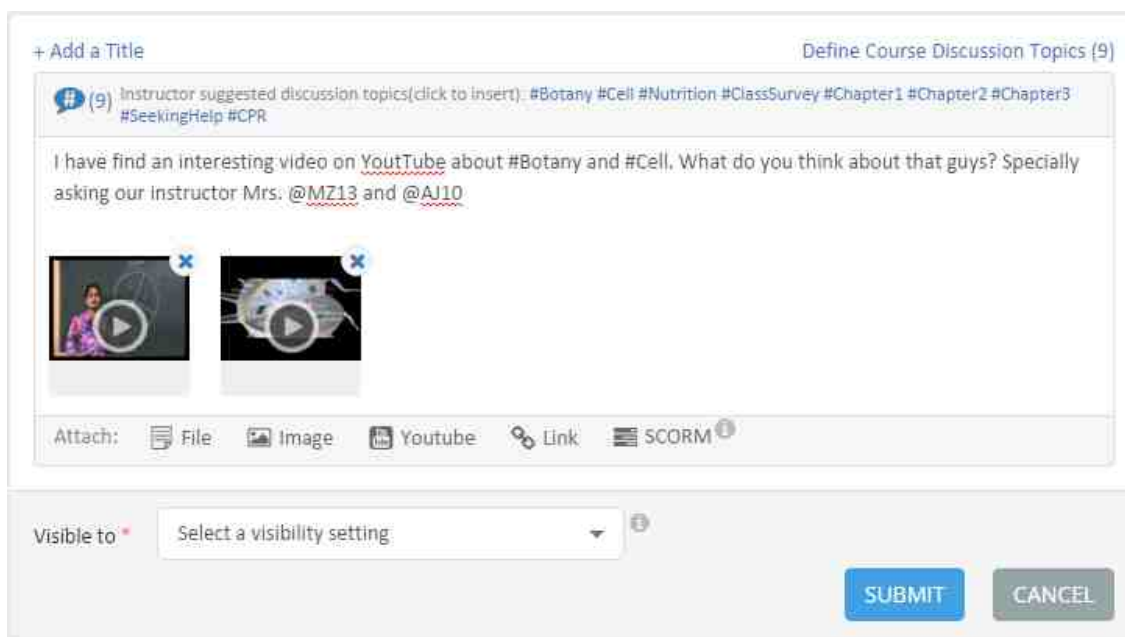
Fig. 4.5.: A New Post with Attached YouTube Videos

Visibility setting is another feature provided on Post and makes it more customizable. Posts can be reviewed, liked and reflected on by people who meet the visibility restrictions set by the post author. Visibility settings provide the options that are listed as follows (Figure 4.6):

- Visible to This Course: Only members of this Course will see the Post.

- Visible to This Course and Global Classmates: Members of this Course and the Global Classroom (others who are in similar courses on CN) will see the Post.

- Visible to This Course and My Other Courses: Members of this Course and any other Courses you are a member of can see the Post, based on which Courses you choose.

- Visible to Course Groups: Members of the Course Groups will see the Post.

- Only Me (Edit Later): No one will see the Post until you change the visibility setting.

Post's hashtag and visibility settings are two essential features from the Post tool and are inherited by other tools like Polls and events, which share exactly the same logic.
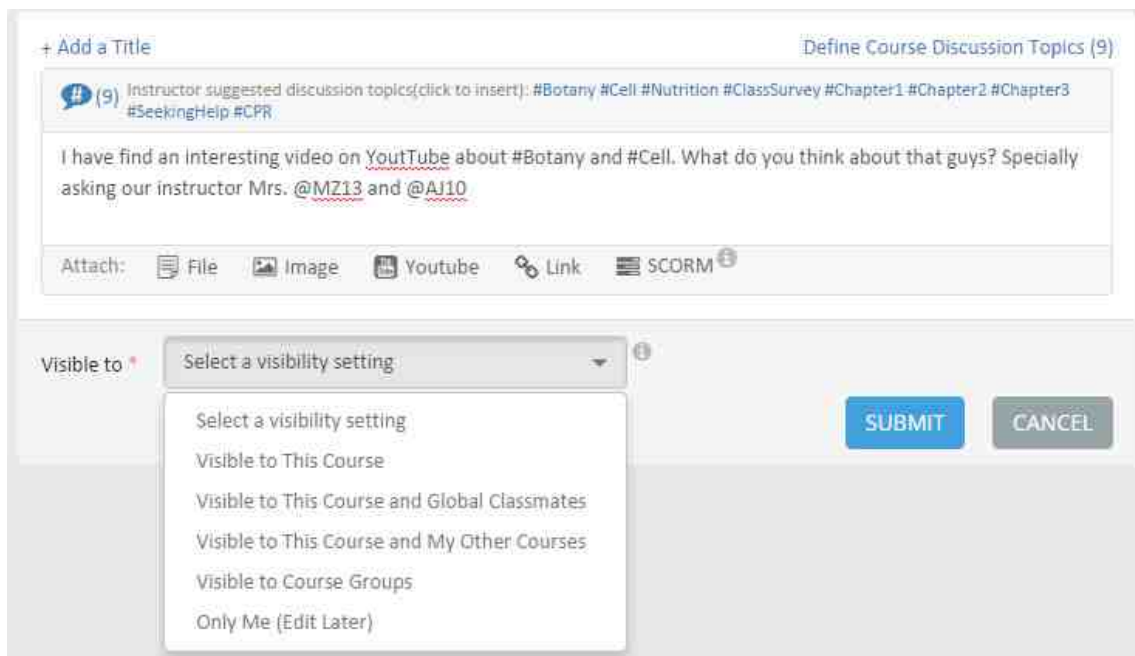
Fig. 4.6.: Visibility Options For a New Post

Search and Filtering

Post Search and Filter is another feature ported from the full CourseNetworking and are easily accessible from the top navigation bar. Posts in CN can be sorted based on most recently created or reflected, most commented/liked, or most visited.

The filter function uses different kinds of inputs to filter out the posts. These inputs can be the popularity of a post (in CN it is defined as Post of the Week, which means a post with a lot of popularity), post type (poll, post, event), the author of post (user's own posts, from instructors or from classmates) and highlighted posts by instructor or CN Admin (Figure 4.7)
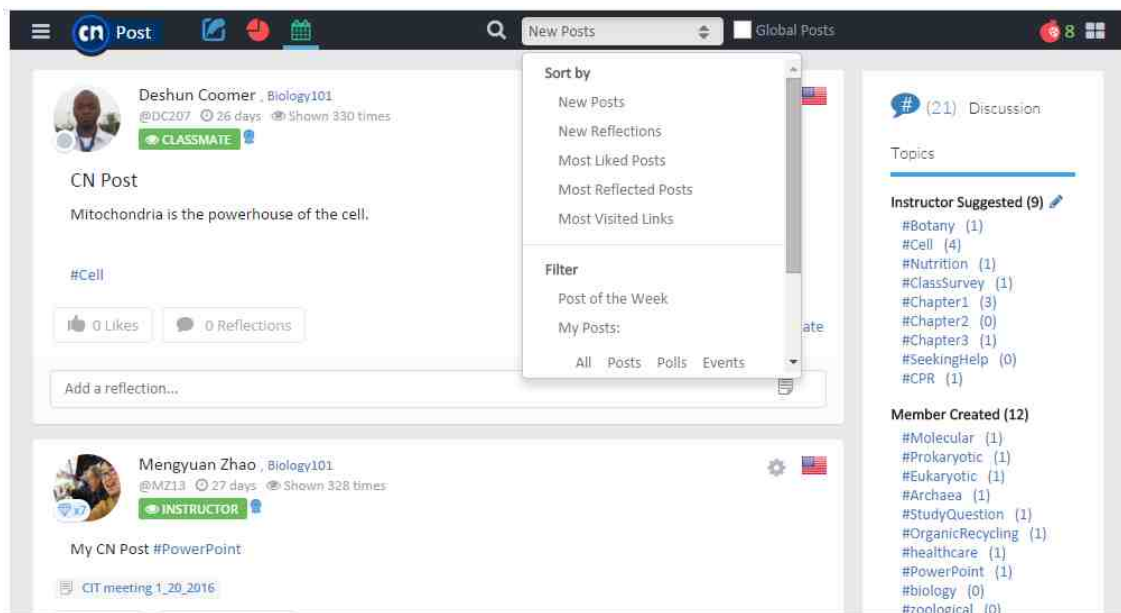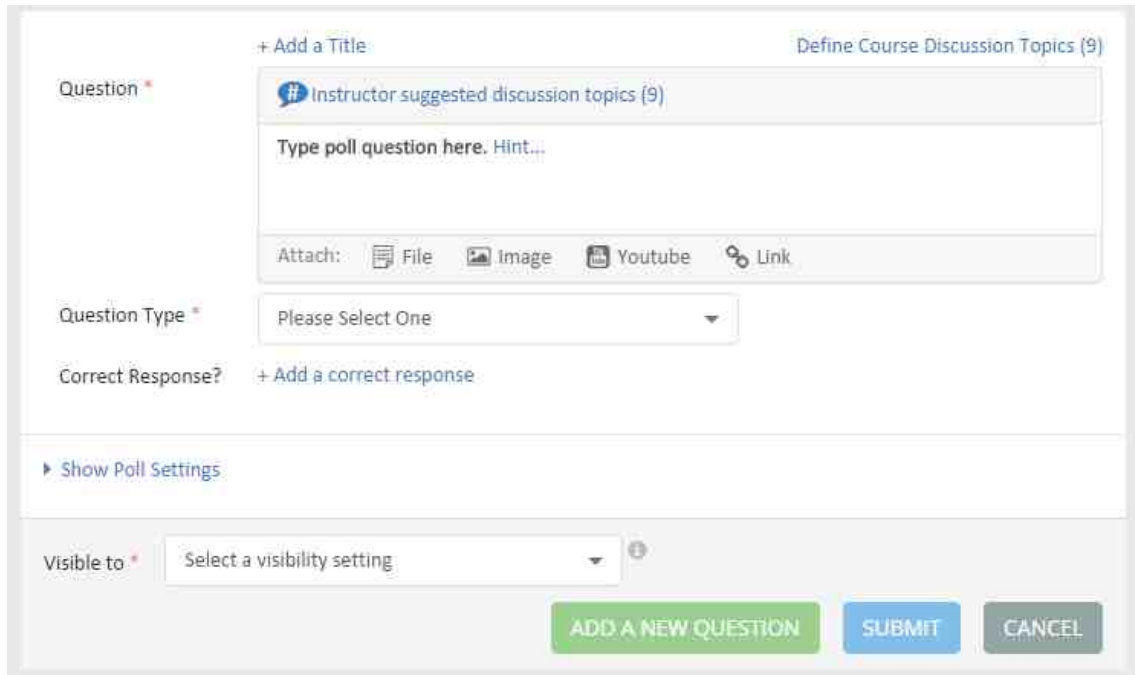
Fig. 4.7.: CN Post Filtering Feature

4.3.3.2   Polls

As mentioned previously, survey opinions, reviewing knowledge, gathering data and conducting course evaluations are the main capabilities provided by Polls. Poll, in terms of visibility, is exactly the same as post. For this reason, we can skip over the visibility feature for Polls.

Technically, Poll is like a quiz but the question types and purpose are different. The content of poll types of questions is similar to Post (text and attachments) and are used to explain the questions. Polls support hashtags just like Post but they do not support the mentioning or tagging of CN members (Figure 4.8).

Fig. 4.8.: A New Poll

There are different types of questions a Poll can include and the creator of the Poll can add correct answers for any questions provided. Poll question types include: Multiple Choice, Short Answer, True/False, Yes/No, Agree/Disagree (with a variety of opinions), scale from 1 to 5 or scale of 1 to 10 (Figure 4.9).

There are a few additional settings for each Poll. A poll creator may: set a delayed release date for poll, set an automatic closure date, set result release options or select a visibility setting for respondent list (Figure 4.10).

Fig. 4.9.: Different Question Types in a Poll

Fig. 4.10.: Poll Settings

4.3.3.3   Events

Events have a few mandatory fields to collect information as well. These include: Title, Location, Date/Time and other optional fields. The only optional field is the event's description, which is similar to the Polls description field. The description portion is inherited from the Post's content text box and users may add hashtags or attachments, similar to Poll (Figure 4.11).

Fig. 4.11.: A New Event Creation Dialog

Visibility settings are exactly the same as Post and Polls and we will not repeat their characteristics because they were previously mentioned. The only new option is the check box below the Date and Time, which gives users the option to add the created event to Course Calendar.

4.3.3.4   Hashtags

As mentioned in Post, Poll and Event all of the textual descriptions can use Hashtags to make it easier for users to search and filter. The hashtags sidebar, placed in the right hand side of the CN Post, and can filter the posts. These hashtags can be defined by instructors (conceptually proposed discussion topics by instructor), or can be created by any other course member (Figure 4.12).
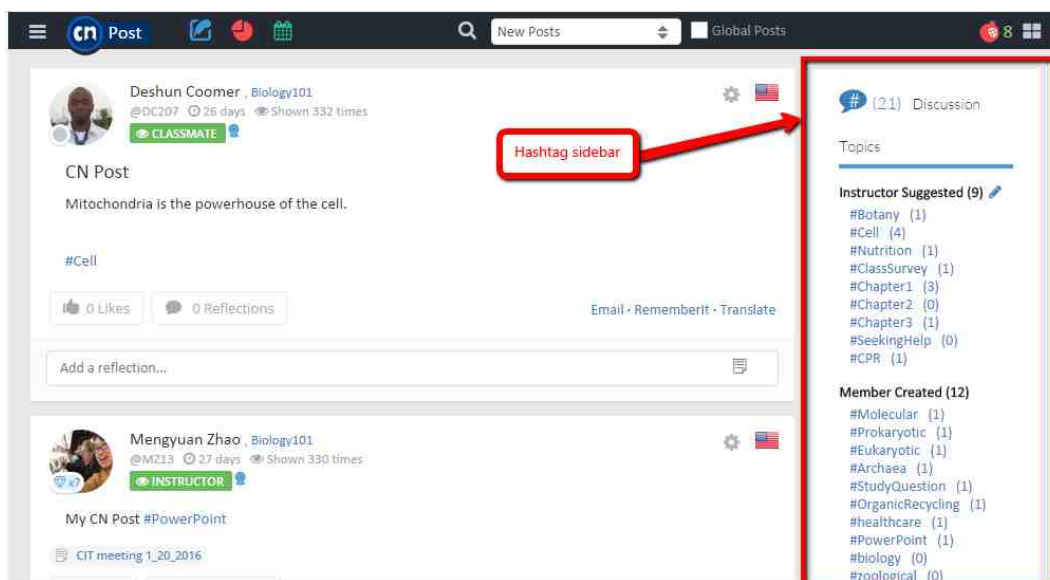
Fig. 4.12.: Hashtag Sidebar in CN Post

Hashtags give students ability to search and pull up discussion topics easily. Students can easily create new hashtags by starting the word with # sign and instructors can create or edit lists of suggested hashtags (Figure 4.13).

4.3.3.5   Global Classroom

In CN, each course can belong to a category that best describes the course. All the courses from the same category form a new logical concept called Global Classroom. In the visibility settings students can set the post to be visible to their global classmates.
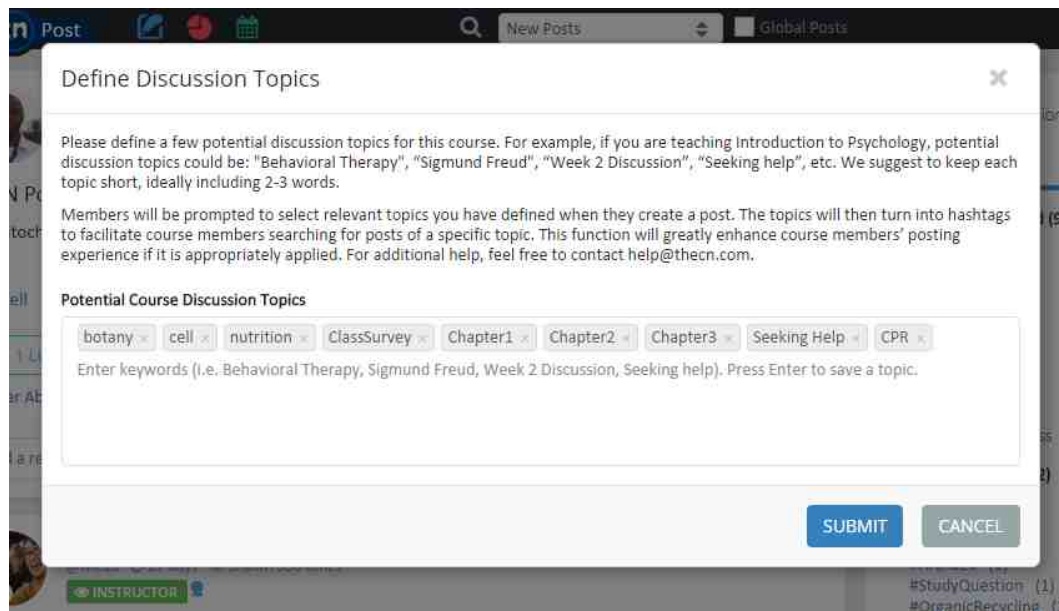
Fig. 4.13.: Modify the Discussion Topics by Instructor

In order to access the posts from the Global Classmates users can easily click on the checkbox entitled "Global Post" in top bar, then their post feed will be updated with posts from the Global Classmates. In addition to post feed, the hashtag sidebar will be updated to correspond with global posts defined hashtags (Figure 4.14).

4.3.3.6 Anar seeds

Anar seeds are a form of measurement used to track quality and quantity of students' participation in a course and in the whole CN site. The etymology behind the Anar refers to the Persian word Anar, which means pomegranate. This is a gamification symbol put in place for users to receive Anar seeds as reward for their participation.

Students like the idea of Anar seeds because on the CN course page and in CN Post there is a bar under to top navigation bar, called the Anar bar which provides a representation of student progress compared to the average participation rate in the course.
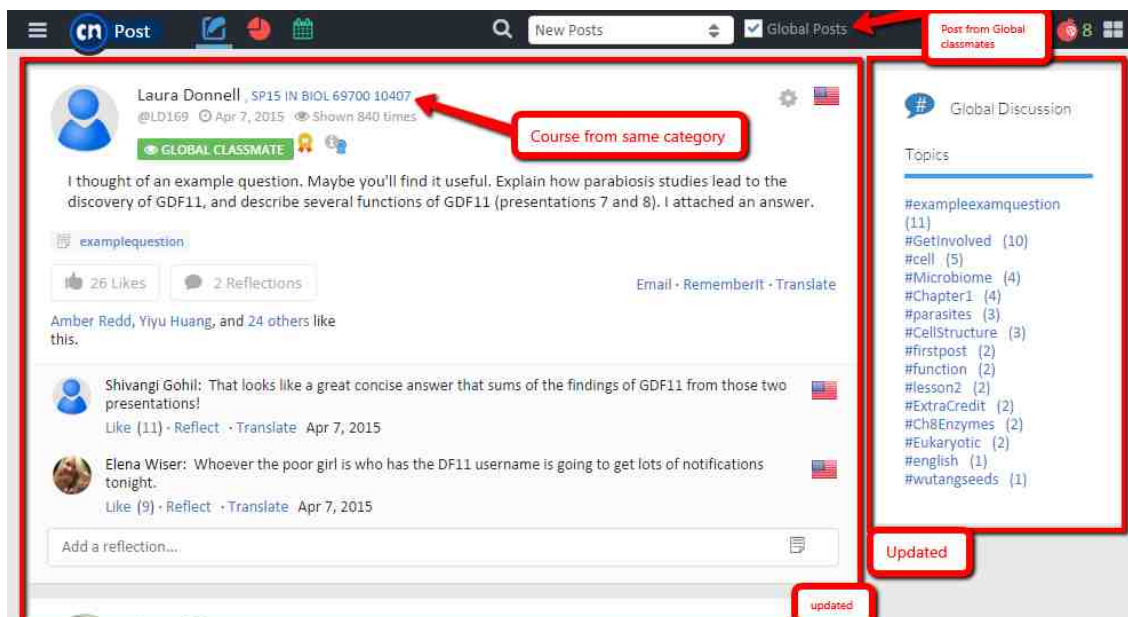
Fig. 4.14.: Posts from Global Classmates

According to Figure 4.15, the member shown on the far right of the scale has accumulated the most Anar seeds in the Course or Conexus. Thus, besides student's rank, increases competitiveness between students through Anar seeds by showing the reward system as a race.

In addition, the color of Anar bar: green for good status and red for bad status, psychologically alerts the students to work harder to participate in the course more than before.
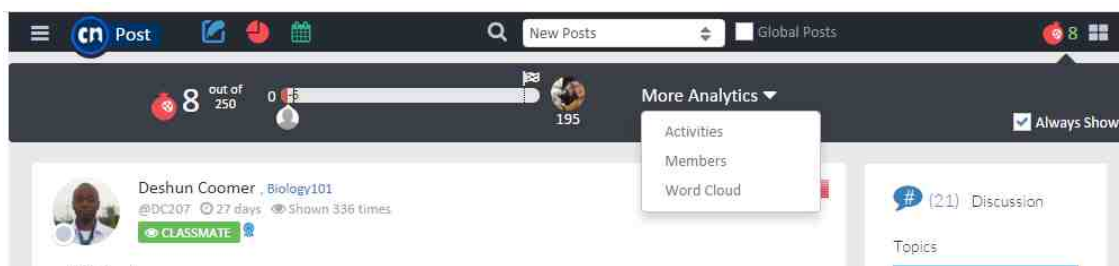
Fig. 4.15.: Anar Bar and Analytics

4.3.3.7 Engagement Analytics

CN Post brings Engagement Analytics from CN to offer useful analytics to the users. These analytics are offered in three different categories including: comparable activity records among the course, individual activity records and semantic analysis of course content.

The activity report in the course shows meaningful graphs to convey the activity of each user in terms of logins into the course, participation (like, reflection, post publication) and earning Anar seeds. In Figure 4.16 activities analytics are shown:
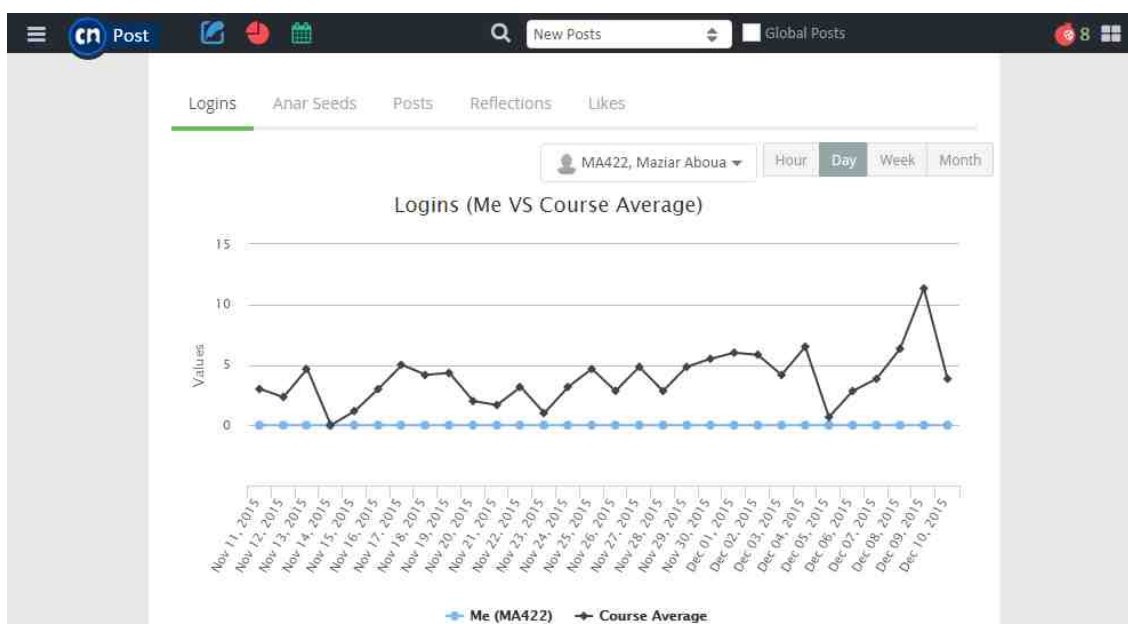
Fig. 4.16.: Course Analytics

Individual analytics are shown in the course roster and in a few columns in (Figure 4.17).

The last section of Analytics is not related directly to student engagement but does represent a meaningful word cloud for users based on posted content in the course. The graph is generated based on the frequency of words used in the course (in the course posts, polls and events) and generates a word cloud based on that collected information. In the word cloud below, larger sized words show the words'

Fig. 4.17.: Course Roster Analytics for Individual Members

usage frequency and allow them to be considered as the main topics of that course. A sample result of this tool for a Biology 101 sample course can be shown in the Figure 4.18:

Fig. 4.18.: Word Cloud Analytics for a Biology 101 Course

4.3.3.8 Left side bar Navigation

There is a side bar on left hand side of CN Post used to give users quick access to different parts of CN Post such as settings and quick links. All of the options are listed and quickly described as follow (Figure 4.19):

1. Profile edit: After clicking on this area, a new browser tab open and users are able to review their profile page.

2. Anar Seeds Settings (only for instructors): In this section instructors can modify Anar seeds related parameters such as number of Anar seeds for each type of participation and its relation to the course grade.

3. Go to Full CN: After clicking on this area, the course page will open in a new browser tab.

4. CN Quick Start Tutorial: This link takes users to the CourseNetworking YouTube Channel where they can view the CN Quick Start Tutorial video.

5. Change Password: By clicking this, user can change his password.

6. Email Notification: This page allows users to modify their email notification preferences.

7. RememberIt List: A list of bookmarked posts.

8. Roster: A roster table that provides different information and analytics for course members.

9. Badges: A list of badges and their recipients.

4.3.3.9 Right hand side Notification bar

There is a notification bar on the right hand side of CN Post, which appears after clicking on the notification icon on the top right of the CN Post. This notification
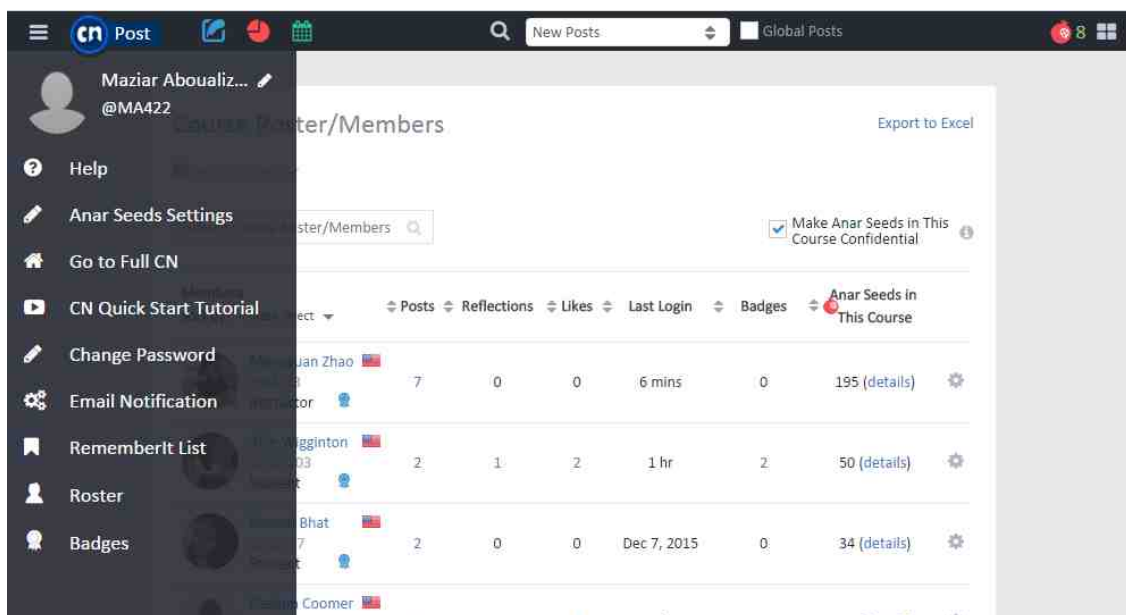
Fig. 4.19.: CN Post Left Hand Side Navigation Bar

center shows all types of notifications, emails and follower's notifications to the user. If a user clicks on any of the notifications, a new browser tab will open and allow the user to see more details in the full version of CN, this is illustrated in Figure 4.20.

4.4 Development Process

In this section, the development process of CN Post as an external tool offers a subset of services from CourseNetworking including system design, challenges in development process, technical details, certification and a quick installation guide for LTI compliant learning management systems.

4.4.1 System Design

CN Post has two major sub-systems, which are the LTI license manager and the main LTI interface which makes CN Post available to consumers.
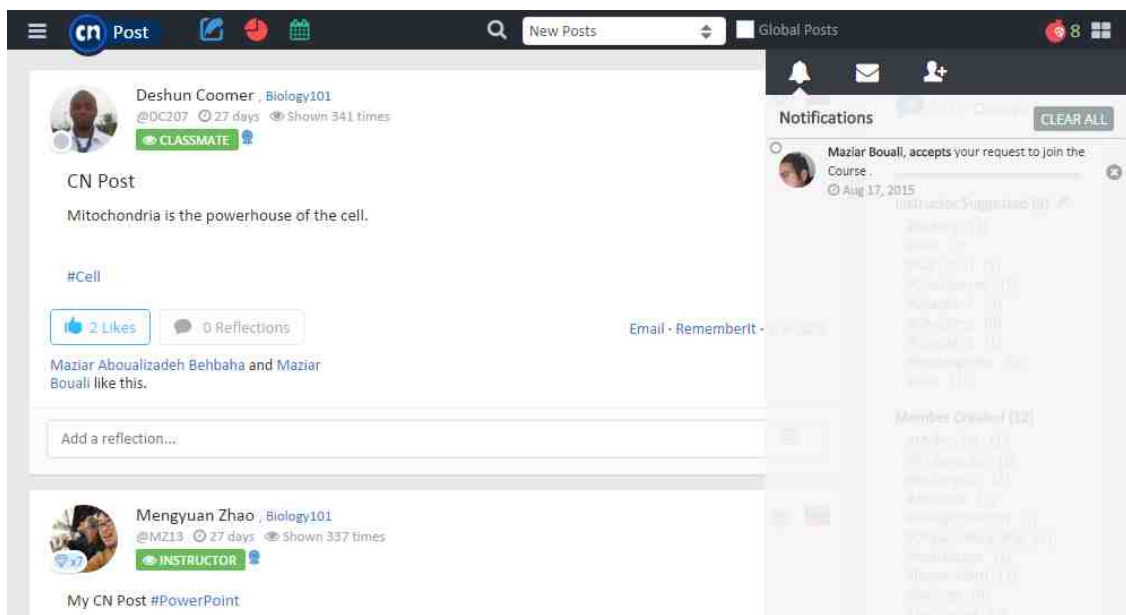
Fig. 4.20.: CN Post Notification bar

4.4.1.1 Overview

CN Post and LTI license manager are two subsystems in the CourseNetworking web application that work together to provide services to LTI consumers. CN Post uses a few public and private APIs from CourseNetworking so it is highly service oriented (RESTful) and independent to CN's backend underlying technologies.

The License manager provides an interface for system admin and manages LTI license information from the database. CN Post will offer a service oriented interface, based on LTI 1.0, which is a well-known standard for web services in learning systems' domain, to the service consumer. CN will access the database through the RESTful APIs and consider the licensing info when it is creating course and users or performing any other tasks. This system design allows CN to have a completely separate and independent backend system from CN Post. LTI consumers should feel safe about the technical changes in CN Post knowing both sides are LTI 1.0 compliant. An overview of system blocks is shown in Figure 4.21.
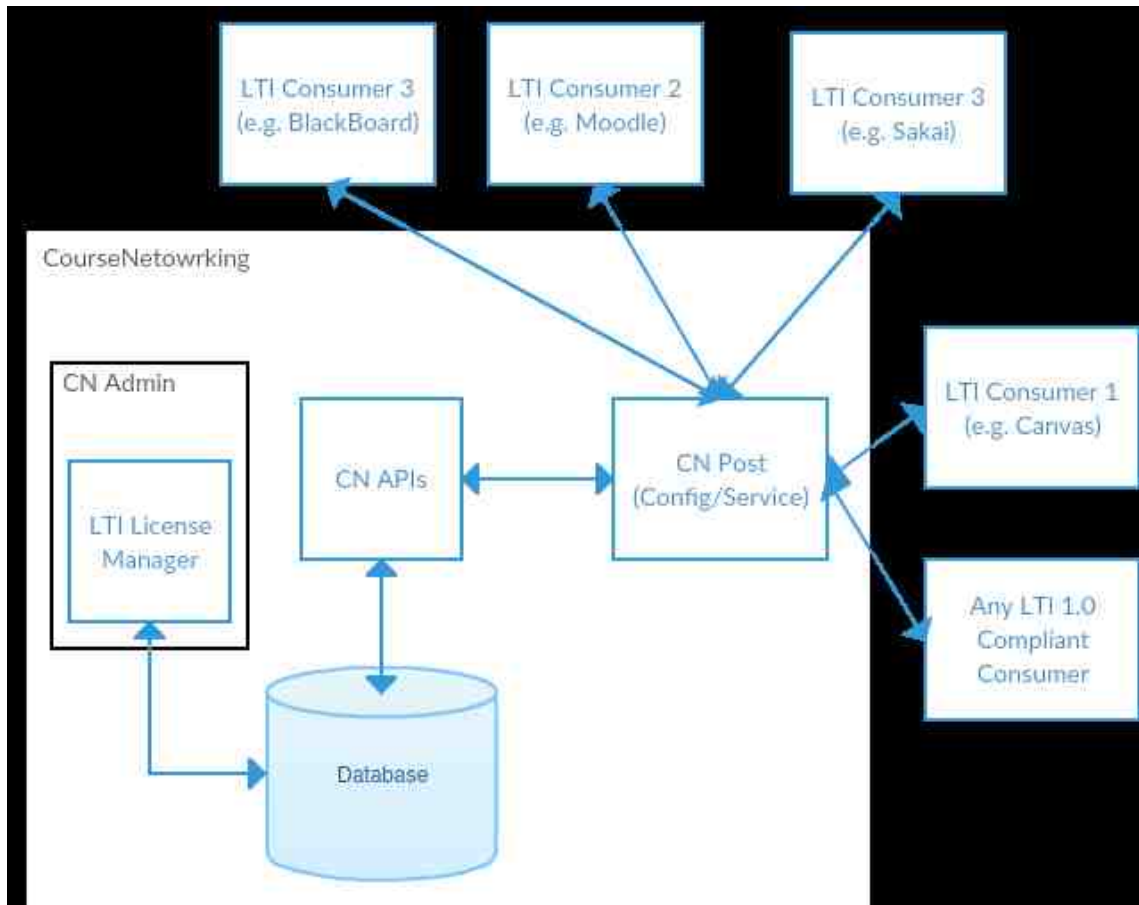
Fig. 4.21.: CN Post System Architecture Overview

4.4.1.2 LTI License Manager

LTI License manager is a subsection of CourseNetworking Administrator panels (CN Admin) and allow the system administrator to create a licensing package for LTI consumers. Each record in this portion of maintained information regarding a certain LTI consumer can be considered institution information. We have defined a new institution as Global CourseNetworking Classroom, which is a worldwide license that allows instructors to have CN Post license regardless of their institution. Obviously they all created their courses under the CN Global Classroom institution (conceptually) and they have institutional administration permissions. The LTI License manager maintains the following information:

- LTI Consumer Key (will be consider as unique identifier for institution in CN's end).

- LTI Secret Key.

- Institution's Name.

- Account type (Limited/Unlimited).

  - Number of allowed users to be created.

  - Number of allowed courses to be created.

- Contact Person information (name/email/phone).

- Validity period for license.

- Country in which institution is located.

- URL of institution's website.

- Unique Identifier of Connected CN Channel.

- Notes (Content and visibility settings) in agreement page.

- Term of Services (Content and visibility settings) in agreement page.

- Additional information from institution (Content and visibility settings) in agreement page.

CN Post will use 1-legged OAuth to authenticate the consumers and make sure they are authorized.

4.4.1.3   CN Post LTI Interface and Logic

CN Post provides an interface for LTI consumers such as Canvas or Blackboard and adds the CN into their learning management systems. After adding the CN Post

as an external tool in the consumer system, users can then click on the button, which will send them a HTTP post request to the CN Post; this is based on the consumer system's policy for placing hyperlinks to CN Post in the course, it would most likely appear in the navigation sidebar. The sample Post request that was sent by Moodle and Canvas is available in the appendix. On the receiver side of the operation, the request will be parsed and handled. CN Post checks all of the information for validity and in the case of invalidity or missing information, it will show a proper message.

If the user was not the registered in CN before (based on user's academic email) an agreement page will appear to inform the user about terms of services and agreement for CN. Additionally, the user can link their existing account in CN to their institutional or academic account. If the user chose to click on the Account Create button, then an account for that user will be created and subsequently a corresponding course to the course in the consumer side will be created. The next step would be enrollment of the user into the created course and then logging them. They would then be redirected to the CN Post. The logic behind this procedure and the data flow is visualized in Figure 4.22.

Figure 4.23 visualizes the interaction between LTI 1.0 compliant consumer, CN Post and CourseNetworking web application. This is considered a sequential process.

4.4.2 Challenges

CN Post was a challenging project in itself, specifically because of the project's original idea; an idea that is not yet published and will most likely face challenges in the future. These challenges can be sorted into the two main categories: Conceptual and Technical challenges.

Conceptual challenges are challenges related directly to the domain of the project. For example, in this case study we experienced a few challenges regarding linking existing CN accounts with students' CN accounts.
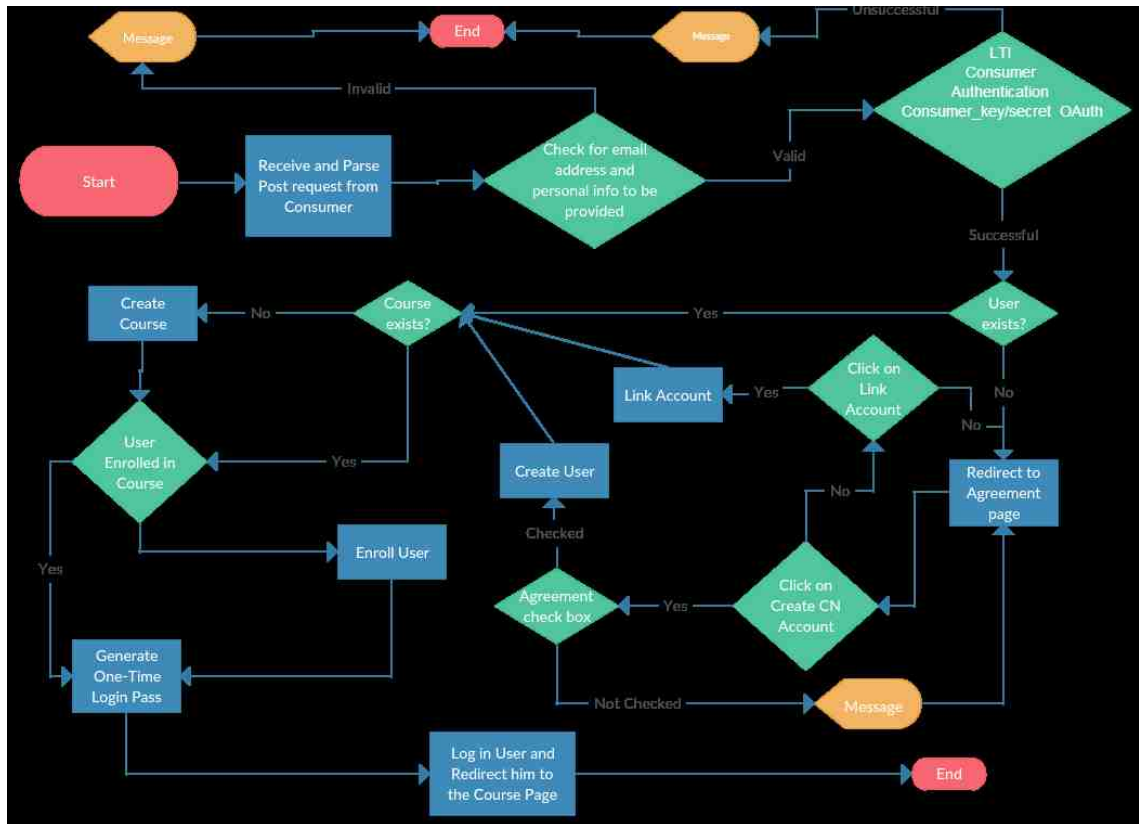
Fig. 4.22.: CN Post LTI Interface Logic

Another conceptual challenge was the start date and the end date of the course, which would vary from university to university. The proposed solution for this problem was to consider the data of the course creation as the first day of course and +5 months as the end date. Obviously, these dates are default values and can be modified later by the instructor. Start dates are used by CN Post to start grading based on Anar seeds, and the end dates are used to hide the ended courses from ongoing the course list in the full CN.

A technical challenge also transpired in the course start and end dates, creating another challenge. As mentioned before, an LTI has the capability of adding custom parameters into a Post request, which comes from the consumer side. The problem was that not all of the consumers have the capability of pulling up the course creation date from their systems. For instance, Canvas and Moodle have that capability, but
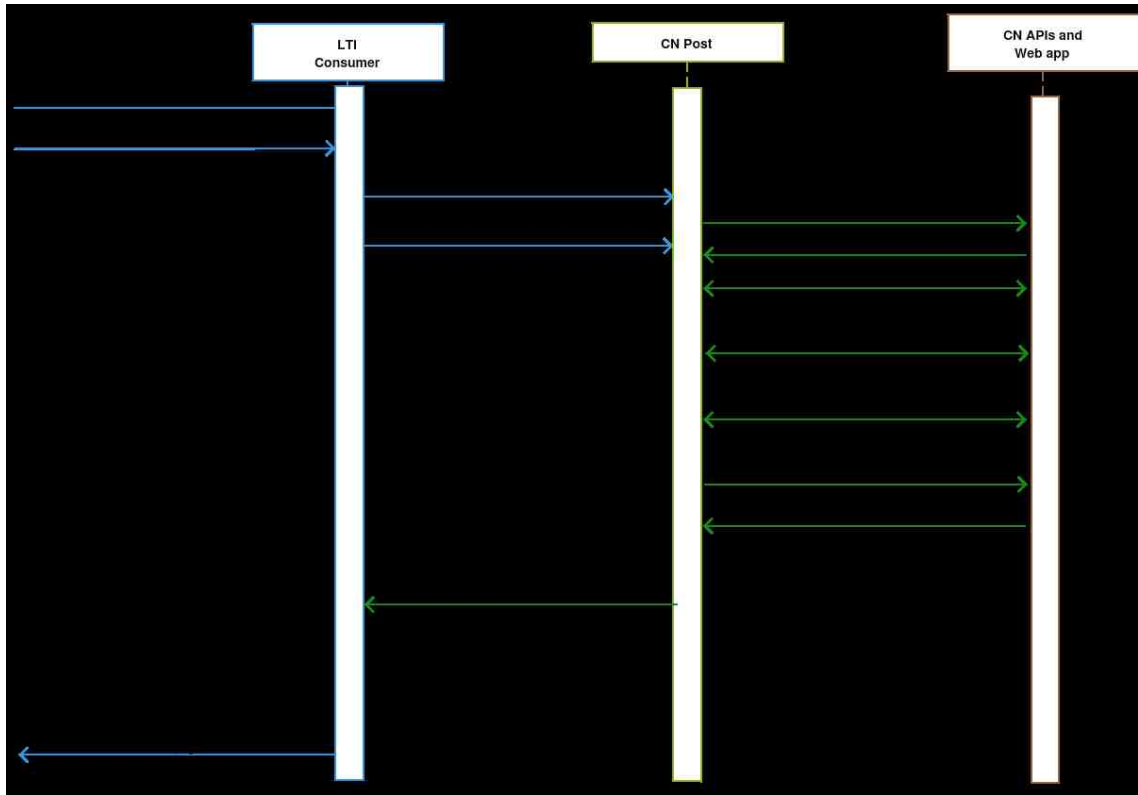
Fig. 4.23.: Interaction Between Consumer, CN Post and CN

Sakai does not nor does Blackboard. There were not any free documents to describe any potential custom parameters for this action. Therefore, we could not rely on this value and had to implement a default value.

The last, and most important technical challenge we learned was the OAuth, which is a method of verification for the consumer key and secret between LTI consumer and LTI provider without the transmission of sensitive info such as a password. In this method, the consumer calculates a hash value based on request content, time and secret. This post request would be sent to the provider (CN Post) and on receiver's side, the provider would recalculate the hash value and compare the result. This process seems pretty straightforward until some data, other than ASCII, comes into play. We just followed Moodle's method of OAuth from their GitHub repository in order to resolve this technical challenge.

4.4.3   Development Details

In this section all of the remaining details about development of CN Post are covered.

4.4.3.1   CN Post as a module in CourseNetworking

CN Post has two different parts that handle the requests from consumers. The first part is an option one, which is a configuration file that makes it easier for consumers to add the CN Post to their system. This XML configuration file is accessible from: https://www.thecn.com/program/lti/config [1] and support Canvas extensions, in addition to LTI Basic Launch. This configuration URL also supports Canvas extensions which make CN Post available in a course's navigation in the Canvas.

Another end that handles requests based on the consumer's input parameters, is the target URL of LTI course, which is accessible from:

https://www.thecn.com/program/lti/course [1].

This URL will not work in a browser without LTI's basic parameters.

We have only the two handlers to handle requests from LTI consumers. When a request from consumer received on CN's end, the CN Post will analyze that request and then, based on its logic, do a few API calls. This analysis uses native PHP functions. Based on the rules that has been implemented in the CN Post LTI interface, data in LTI basic launch POST request will be analyzed and validated. CN Post will call a few API in order to find out the course and the user are valid (or create them if they are not available in the database) and go for the browser validation. During the browser validation first our interface recognize the type and version of web browser then based on that show the proper response. This logic is simple and only categorize the web browsers to two different categories: Internet Explorer and Safari in one group and other web browsers in the other one. If the browser was Internet Explorer or Safari we have to open the CN Post in a new browser tab instead of an iframe.

---

[1]Last Date Accessed: April 1, 2016

4.4.3.2 Request Handler Algorithm

After CN Post has received a request from the consumer, the entire procedure follows these actions:

1. Check the request format.

   - Using $_POST array to extract information.

2. Authentication with LTI License Manager and make sure consumer is valid and authorized.

   - Read data from the collection corresponding to LTI License Manager.
   - Recalculate the signature value generated by consumer based on 1-legged OAuth approach.
   - Verify received value by value calculated from database received.

3. Extract information from request and check them for any invalid or missing essential info.

4. Detect if user already exists and if not create user.

   - Use CN RESTful API: https://www.thecn.com/api/user[1].

5. Detect if course already exists and if not create course.

   - Use CN RESTful API: https://www.thecn.com/api/course[1].

6. Detect if user already enrolled and if not enroll user in course.

   - Use CN RESTful API: https://www.thecn.com/api/course[1].

Because CourseNetworking is a commercial product, some details concerning source code and data models behind the scenes cannot be released. However, we can release courses and users, which are created through the LTI processes associated with a consumer license record in the LTI license manager.

---

[1]Last Date Accessed: April 1, 2016

4.4.3.3 Responsive Design

After CN Post receives and handles the LTI Basic Launch request from the consumer and makes the necessary changes in the CN database, it is then ready to automatically log the user into CN and redirect them into the CN Post course page. The data object behind this course is exactly same as that of the course on the full CN, so any data modification will reflect both sides.

LTI consumers' users usually prefer to open the LTI 1.0 tools in an iframe and that means the tools has a variety of screen sizes. In addition to this, CN Post has to handle the screen sizes like they would for mobile view and touch screens. This is because, for example, Canvas has a very convenient mobile app that is supports LTI tools on mobile devices. Therefore, the CN Post has to be equipped with a responsive design. Figure 4.24 shows the responsive design of CN Post more clearly:
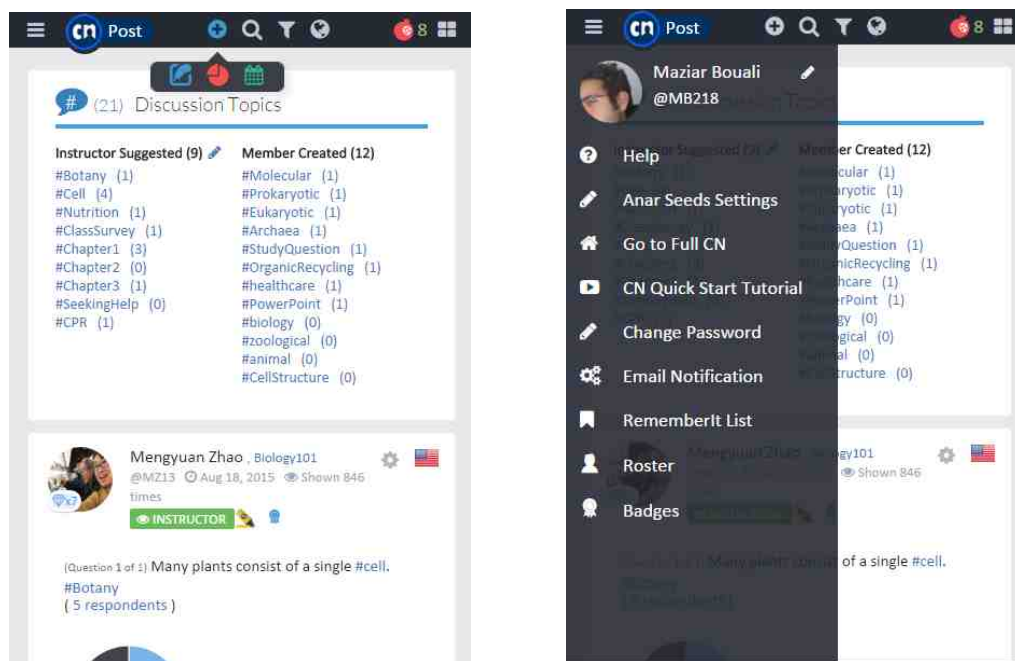
(a) Normal View   (b) Left Context Menu

Fig. 4.24.: Responsive Design of CN Post

Table 4.1: Certification Result - Tool Registration

Test	Required for	Status
1: Tool registration		
1.1: Perform a rejected registration request	2	
1.2: Perform an accepted registration request	2	

4.4.4 IMS Certification and Compatibility

CN Post is an IMS LTI tool, which is certified through the IMS conformance utilities and received the certification from the IMS Global Learning Consortium. The certification page is available in IMS Global at:

https://www.imsglobal.org/compliance/coursenetworking-cn-post-v20150814[1]

In Tables 4.1, 4.2, 4.3, 4.4, results regarding the tests on CN Post using IMS Global Certification utilities is demonstrated.

4.4.5 Installation

A trial version of the CN Post is available for testing, exclusively for instructors and the related configuration parameters can be as follows:

- **Consumer Key:** 55e6369fa21f46a70f8b457b

- **Secret Key:** a21d97e242c30e54988238b0

- **Configuration URL:** https://www.thecn.com/program/lti/config[1]

- **Target URL:** https://www.thecn.com/program/lti/course[1]

---

[1]Last Date Accessed: April 1, 2016

Table 4.2: Certification Result - Types of Launch Request

Test	Required for	Status
2: Types of Launch Request		
2.1: Launch as an instructor	1.0, 1.1, 1.2, 2.0	Pass
2.2: Launch as a learner	1.0, 1.1, 1.2, 2.0	Pass
2.3: Launch as an instructor from a different resource link	1.0, 1.1, 1.2, 2.0	Pass
2.4: Launch as an instructor with max privacy	1.0, 1.1, 1.2, 2.0	Pass
2.5: Launch as a learner with email but no name	1.0, 1.1, 1.2, 2.0	Pass
2.6: Launch as a learner with given/family but no full name	1.0, 1.1, 1.2, 2.0	Pass
2.7: Launch as a learner with a full name but no given or family names	1.0, 1.1, 1.2, 2.0	Pass
2.8: Launch as an instructor using a fully-qualified URN	1.0, 1.1, 1.2, 2.0	Pass
2.9: Launch as an instructor with a list of roles	1.0, 1.1, 1.2, 2.0	Pass
2.10: Launch as an instructor using a full urn role in a list	1.0, 1.1, 1.2, 2.0	Pass
2.11: Launch as a learner using a full urn role in a list	1.0, 1.1, 1.2, 2.0	Pass

Table 4.3: Certification Result - Levels of User Information

Test	Required for	Status
3: Levels of User Information		
3.1: Launch as an instructor	1.0, 1.1, 1.2, 2.0	Pass
3.2: Launch as an instructor with no personal info	1.0, 1.1, 1.2, 2.0	Pass
3.3: Launch as an instructor with no context info	1.0, 1.1, 1.2, 2.0	Pass
3.4: Launch as Instructor with no context_id	1.0, 1.1, 1.2, 2.0	Pass
3.5: Launch with no roles	1.0, 1.1, 1.2, 2.0	Pass
3.6: Launch with no roles and no context	1.0, 1.1, 1.2, 2.0	Pass
3.7: Launch with only resource_link_id	1.0, 1.1, 1.2, 2.0	Pass

Table 4.4: Certification Result - Invalid Launch Request

Test	Required for	Status
4: Invalid Launch Request		
4.1: No resource_link_id provided	1.0, 1.1, 1.2, 2.0	Pass
4.2: No resource_link_id or return URL provided	1.0, 1.1, 1.2, 2.0	Pass
4.3: Invalid OAuth signature	1.0, 1.1, 1.2, 2.0	Pass
4.4: Invalid LTI version	1.0, 1.1, 1.2, 2.0	Pass
4.5: Missing LTI version	1.0, 1.1, 1.2, 2.0	Pass
4.6: Invalid LTI message type	1.0, 1.1, 1.2, 2.0	Pass
4.7: Missing LTI message type	1.0, 1.1, 1.2, 2.0	Pass

A quick guide in order to add this tool to an existing learning management system or more generally, to a LTI 1.0 compliant consumer can be found at IMS Global official website: https://www.imsglobal.org/setting-lti-tools-specific-tool-consumers[1].

---

[1]Last Date Accessed: April 1, 2016

5. CONCLUSION AND FUTURE WORK

5.1 Conclusions

In order to facilitate the integration and interoperability between different software applications, it is necessary to follow a standard regardless of the domain or industry. However, standard methods, which have been specialized to a certain domain, will work much better on certain domains and this helps save time and money. Therefore, in reviewing existing technologies, the best approach in enhancement of conventional learning management systems to new seamless integration capability was found. The problem defined by this research was at the first step, the need to equip CourseNetworking, a social educational learning management system, with interoperability and more integration features. In order to come up with a reliable solution, the first step was studying feasibilities. Subsequently, based on the findings of the first step, studying, an evaluation and categorization of the existing approaches helped narrow down the area of study.

Adapting a learning management system to service oriented architecture can be done in a number of ways. One of them is by using the RESTful API, which would be a good approach to port the features from a web application to the third party applications such as mobile apps or other applications. Standard framework is always an option when facilitating later integration and interoperability in learning. In the area of learning systems, frameworks can be categorized into two different types:

1. Abstract Frameworks, mainly focus on the creation of specification and best practices (e.g. IEEE LTSA, OKI OSID and IMS Global interoperability initiatives).

2. Concrete Frameworks, which extend abstract frameworks go a few steps forward in implementation (e.g. OpenUSS, SIF, E-Framework).

Another key component in a learning environment is content, stated technically, learning objects. Therefore, compliancy with learning object standards (or at least to most popular standard) is another step. Learning objects are self-contained, reusable, hierarchical data packages with meta data, offered by content providers and used by providers. The most popular standards for learning objects are SCORM, which was kind of the first mover in this area and IMS Common Cartridge, which learned lessons from SCORM. Common Cartridge is offering many different advantages over SCORM such as more flexibility and better compatibility with other IMS standards like LTI and LIS.

After studying the different learning interoperability and SOA initiatives IMS Global was selected as the most popular creator of standards, we then moved forward with its set of standards. IMS Global offers different standards for requirements in learning systems. The most popular one is the LTI, which helps providers (provide the tool) and consumers (avail the tool in LMS) interact with each other. Community App Sharing Architecture (CASA) is a method defined by IMS to create the collection of LTI applications (LTI app store).

The most popular implementation of CASA was done by Instructure company (creator of Canvas), which is known as Eduappcenter. CASA does not maintain any data regarding the relation between consumers and providers. Recently, an open-source framework named TSUGI, released by Dr. Charlese Severance for developing LTI applications and tools, can maintain the information regarding consuming LTI tool. The new proposed architecture adds the TSUGI to CASA, which enables CASA to maintain data associated with consumer-provider interaction. TSUGI as LTI external tool container provides extra backups for service and data for consumer, therefore it would be helpful in two cases; First, in the case of failure of provider, it would be helpful. Second, by using TSUGI, each institution or even instructor could maintain all of the information on his or her private server.

The procedure involving making the learning management system able to interoperate with other systems is started by making the system LTI (as the dominant standard) compliant. This enhancement was the goal behind the development of CN Post, as a LTI compliant tool, providing social functionalities of CourseNetworking in other learning management systems. CN Post helps educators to take a step beyond the traditional learning management systems by providing an interactive environment for students and instructors to interact, collaborate, socialize and share their knowledge.

5.2   Future Work

Based on the current state of this research and its results it would be necessary to define new studies to execute and assess the proposed ideas. Additionally, this case study is a product based on a certain learning management system and can be equipped with new features. Therefore, adding new features to CN Post can form other groups of future works. Additional details on future works are categorized and discussed as follows:

- A comprehensive study on existing standards and technologies in learning environments would be very helpful, but after a couple of years the organizations or even learning system manufacturers would improve their standards and probably release new versions. Keeping track of these changes and contributions in the development of new standards could be a valuable research topic.

- The proposed refinements to CASA could be executed and assessed. These refinements could be done in one of the open source implementations of CASA or considered in a new implementation of CASA. The most popular implementation of CASA is in Ruby.

- Another big refinement in CASA is addition to TSUGI as the native LTI application container. At this moment TSUGI is developing in PHP and JAVA, which can both be ported to other programming languages.

- Improving the LTI compliancy of CourseNetworking would be another high priority future work. A general guideline would be useful to show procedures in making a learning management system LTI compliant. This guideline could define a best practice for detection of entities in a learning management system and enhance them so they are LTI compliant.

- Improving IMS compliancy of CourseNetworking by making it compliant with IMS standards such as Common Cartridge, LIS and OneRoster, Caliper, CASA and self-configurable LTI tools.

- Providing more external tools in order to avail more features from CourseNetworking in the consumer learning management systems.

- Additionally supplemented new features to CN Post. The most important feature added could a be more meaningful order. One could filter the posts (personalized) especially for global posts in CN Post. Other improvements in CN Post could be to push notifications from CN Post into the host learning management system. Actually, this feature would need refinement in LTI standards to support propagating a notification to host.

Other improvements on CN Post could be performance analysis and refinements on its system design, which would improve performance of CourseNetworking as well.

LIST OF REFERENCES

LIST OF REFERENCES

[1] J. P. Leal and R. Queirós, "Elearning Frameworks : a Survey," *Inted2*, no. August, 2010.

[2] M. Á. C. González, F. J. G. Peñalvo, M. J. C. Guerrero, and M. A. Forment, "Adapting LMS architecture to the SOA: An architectural approach," *Proceedings of the 2009 4th International Conference on Internet and Web Applications and Services, ICIW 2009*, pp. 322–327, 2009.

[3] M. T. Mahmoudi, M. Parsanasab, M. K. Haghighat, M. A. Behbahani, and F. Taghiyareh, "Automatic Creation of Semantic Schema for Accurate Retrieving of Education-Supportive Documents," *6th IEEE International Conference on E-Learning in Industrial Electronics (ICELIE)*, 2012.

[4] V. Gonzalez-Barbone and L. Anido-Rifon, "From SCORM to Common Cartridge: A step forward," *Computers & Education*, vol. 54, no. 1, 2010.

[5] P. R. Humanante-ramos, F. J. García-peñalvo, and M. Á. Conde-gonzález, "Personal Learning Environments and Online Classrooms: An Experience With University Students," vol. 10, no. 1, pp. 26–32, 2015.

[6] P. R. H. Ramos, M. Á. Conde, and F. J. G. Peñalvo, "Differences and similarities in use and acceptance of PLEs between universities in Ecuador and Spain," *Proceedings of the 2014 Workshop on Interaction Design in Educational Environments - IDEE '14*, pp. 70–76, 2014.

[7] IMS Global Learning Consortium, "Learning Tools Interoperability," Last Date Accessed: April 1, 2016. Internet Link: http://www.imsglobal.org/activity/learning-tools-interoperability.

[8] G. Henrick, "Moodle as the Central Hub of Learning with Tools Plugged in - Learning Tool Interoperability," *1st Moodle Research Conference*, pp. 208–211, 2012.

[9] M.-W. online, "Metadata," Last Date Accessed: April 1, 2016. Internet Link: http://www.merriam-webster.com/dictionary/metadata.

[10] J. B. De, Miltiadis D Lytras Patricia Ordonez and HermannMaurer, Pablos AdrianZiderman AlanRoulstone and Imber, *Knowledge Management, Information Systems, E-Learning, and Sustainability Research - Third World Summit on the Knowledge Society, WSKS 2010, Proceedings*, vol. 111 CCIS. 2010.

[11] M. A. Conde, F. J. García-Peñalvo, J. Piguillem, M. J. Casany, and M. Alier, "Interoperability in eLearning Contexts. Interaction between LMS and PLE," *1st Symposium on Languages, Applications and Technologies (SLATE 2012)*, pp. 205–223, 2012.

[12] E. M. Al-mukhaini, W. S. Al- Qayoudhi, and A. H. Al- Badi, "Adoption Of Social Networking In Education : A Study Of The Use Of Social Networks By Higher Education Students In Oman," *Journal of International Education Research*, vol. 10, no. 2, pp. 143–155, 2014.

[13] M. A. Conde, F. J. Garci, M. J. Casany, and M. Alier, "Applying Web Services to Define Open Learning Environments," *Database and Expert Systems Applications (DEXA), 2010 Workshop on DOI - 10.1109/DEXA.2010.36*, pp. 79–83, 2010.

[14] M. Á. Conde, F. García-Peñalvo, M. Alier, E. Mayol, and C. Fernández-Llamas, "Implementation and design of a service-based framework to integrate personal and institutional learning environments," *Science of Computer Programming*, vol. 88, pp. 41–53, 2014.

[15] N. Galanis, M. Alier, M. José, E. Mayol, and C. Severance, "TSUGI : A Framework for Building PHP-Based Learning Tools," in *TEEM '14*, (Salamanca), pp. 409–413, ACM, 2014.

[16] S. Kurz, M. Podwyszynski, and A. Schwab, "A Dynamically extensible, service-based infrastructure for mobile applications," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5232 LNCS, pp. 155–164, 2008.

[17] H. L. Grob, F. Bensberg, and D. Blasius, "Open University Support System (OpenUSS)," Last Date Accessed: April 1, 2016. Internet Link: http://openuss.sourceforge.net/openuss.

[18] "School Interoperability Framework," Last Date Accessed: April 1, 2016. Internet Link: http://www.nsip.edu.au/systems-interoperability-framework.

[19] "SIF Implementation Specification," Last Date Accessed: April 1, 2016. Internet Link: https://www.sifassociation.org/Specification/Pages/default.aspx.

[20] S. Wilson, K. Blinco, and D. Rehak, "An e-Learning Framework - A Summary," *Joint Information Systems Committee*, no. July 2004, 2004.

[21] W. Barritt, Chuck ; Lewis, Deborah ; Wieseler, "Cisco Systems Reusable Information Object Strategy," *RIO Strategy v3.0*, pp. 23–43, 1999.

[22] O. Santos, *Educational Recommender Systems and Technologies: Practices and Challenges: Practices and Challenges.* Premier reference source, Information Science Reference, 2011.

[23] "What Are Learning Objects?," Last Date Accessed: April 1, 2016. Internet Link: http://www4.uwm.edu/cie/learning_objects.cfm?gid=56.

[24] Advanced Distributed Learning Initiative, "SCORM overview," Last Date Accessed: April 1, 2016. Internet Link: http://www.adlnet.gov/Technologies/scorm/.

[25] M. J. Casany, M. Alier, and E. Mayol, "Application of mobile and software as a service (SAAS) in learning processes," *Proceedings of the First International Conference on Technological Ecosystem for Enhancing Multiculturality - TEEM '13*, pp. 207–208, 2013.

[26] C. Severance, T. Hanss, and J. Hardin, "Ims learning tools interoperability: Enabling a mash-up approach to teaching and learning tools," *Technology, Instruction, Cognition and Learning*, vol. 7, no. 3-4, pp. 245–262, 2010.

[27] J. P. Leal and R. Queirós, "Using the Learning Tools Interoperability Framework for LMS Integration in Service Oriented Architectures," *Techeducation11*, 2011.

[28] J. Fontenla, R. Pérez, and M. Caeiro, "Using IMS Basic LTI to Integrate Games in LMSs," in *2011 IEEE Global Engineering Education Conference*, pp. 299–306, 2011.

[29] R. Queirós, "Json on mobile: is there an efficient parser?," *3rd Symposium on Languages, Applications and Technologies*, pp. 93–100, 2014.

[30] S. Vickers, "Learning Tools Interoperability® Content-Item Message," Last Date Accessed: April 1, 2016. Internet Link: https://www.imsglobal.org/specs/lticiv1p0.

[31] IMS Global Learning Consortium and R. Rocchio, "Community App Store Architecture Approach to Interoperable Educational Apps and Content." 2013.

[32] IMS Global Learning Consortium, "Learning Measurement for Analytics Whitepaper," *Consortium, Ims Global Learning*, pp. 1–12, 2013.

[33] IMS Global Learning Consortium, "Common Cartridge," Last Date Accessed: April 1, 2016. Internet Link: http://www.imsglobal.org/activity/common-cartridge.

[34] IMS Global Learning Consortium, "OneRoster and Learning Information Services," Last Date Accessed: April 1, 2016. Internet Link: http://www.imsglobal.org/activity/onerosterlis.

[35] O. Sitthisak, L. Gilbert, and D. Albert, "Learning in Moodle using Competence-Based Knowledge Space Theory and IMS QTI," *Computer Science and Engineering Conference (ICSEC), 2013 International*, no. Icsec, pp. 53–57, 2013.

[36] IMS Global Learning Consortium, "Accessible Portable Item Protocol (APIP)," Last Date Accessed: April 1, 2016. Internet Link: http://www.imsglobal.org/apip/index.html.

[37] A. Jafari and Amy L. Baylor, "From Course Management to Course Networking," vol. 1, no. 1, pp. 1–8, 2012.