

INSTITUTO DE INFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

CECÍLIA RAFAEL JOSÉ TIVIR

**Disseminação de conteúdo educacional através de sua catalogação automática em  
Repositório Educacional**

Porto Alegre

2017

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
INSTITUTO DE INFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

CECÍLIA RAFAEL JOSÉ TIVIR

**Disseminação de conteúdo educacional através de sua catalogação automática em  
Repositório Educacional**

Dissertação apresentada como requisito  
parcial para a obtenção do grau de Mestre  
em Ciência da Computação.

Orientadora: Profa. Dra. Rosa Maria Vicari

Co-orientador: MSc Marcos Freitas Nunes

Porto Alegre

2017

## CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Tivir, Cecília Rafael José

Disseminação de conteúdo educacional através de sua catalogação automática em Repositório Educacional/ Cecília Rafael José Tivir. – 2017.

65 f.:il.

Orientadora: Rosa Maria Vicari. Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação. Porto Alegre, BR – RS, 2017.

1. Objetos de Aprendizagem. 2. Ambientes Virtuais de Aprendizagem. 3. Interoperabilidade. 4. Informática na Educação. I. Vicari, Rosa Maria. II. Nunes, Marcos Freitas. III. Disseminação de conteúdo educacional através de sua catalogação automática em Repositório Educacional.

“Quando deixamos nossa luz própria brilhar,  
inconscientemente damos às outras pessoas  
permissão para fazer o mesmo”.

Nelson Mandela

## **Agradecimentos**

Este trabalho considero fruto de uma cooperação e apoio de diversas pessoas que direta ou indiretamente fizeram com que o mesmo se materializa. E é com muita honra e prazer que tenho de expor a minha imensa gratidão, em especial a minha querida orientadora Profa. Rosa Maria Vicari, pela sua paciência, disponibilidade, dedicação, e carinho.

À Cognitiva Brasil, um especial agradecimento ao Marcos meu coordenador e ao Luiz.

Ao Centro de Informática da Universidade Pedagógica de Moçambique, um especial agradecimento ao Prof. Felix Singo, por me proporcionar a oportunidade de aperfeiçoamento e busca de conhecimento.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico em cooperação com Ministério de Ciências e Tecnologias de Moçambique (CNPq/MCT-Moz 2014) pelo financiamento da bolsa.

À Universidade Federal Rio Grande do Sul (UFRGS), através do Programa de Pós-Graduação em Computação do Instituto de Informática.

Aos colegas e amigos do grupo de pesquisa GIA, à Fabiane, pelo encorajamento, compreensão, disponibilidade, muito obrigada pelo apoio constante nas grandes e pequenas coisas.

À família Flores Penteado pelo acolhimento, carinho e disponibilidade, eternamente agradecida.

À Comunidade de estudantes moçambicanos em Porto Alegre, pelos momentos de reflexão, debates e confraternização, em especial a Arlete, a Cacilda, a Cristina, ao Cuco a Glória e a Sónia.

Aos meus pais, por terem apostado em mim sempre e por me proporcionar as maiores heranças que poderiam ter me dado, a educação, cultura e força de prosseguir sempre.

## RESUMO

Atualmente, existem vários recursos para catalogar e disponibilizar objetos de aprendizagem em vários padrões de metadados. No entanto, encontramos um problema relacionado ao armazenamento e recuperação de objetos de aprendizagem. Talvez porque os professores/autores não tenham tempo para armazenar seus objetos ou mesmo porque o processo de armazenamento de metadados em um repositório é muito trabalhoso. Como resposta ao problema encontrado é apresentada uma possibilidade de minimizar o mesmo e, pretende-se com a pesquisa facilitar o processo de catalogação e disponibilização dos objetos de aprendizagem através da interoperabilidade da plataforma virtual de aprendizagem Moodle com um repositório de objetos de aprendizagem Cognix. Este trabalho tem como objetivo propor uma ferramenta que permite a catalogação automática de objetos de aprendizagem do Moodle no repositório Cognix. Para tal, foram descritas quatro etapas metodológicas para o desenvolvimento do projeto, bem como para a implementação da ferramenta. Espera-se com esta ferramenta possibilitar a reutilização e o acesso aos objetos de aprendizagem em outros ambientes/sites, garantindo assim a descrição uniforme e completa dos metadados.

**Palavras-chave:** Objetos de Aprendizagem. Ambientes Virtuais de Aprendizagem. Interoperabilidade. Informática na Educação.

## **ABSTRACT**

Nowadays, there are several resources for cataloging and making available learning objects in various metadata standards. However, we found a problem related to the existence of the storage and retrieval of learning objects. Perhaps because the teachers/authors do not have time to storage their objects or even because the process of storage metadata in a repository is very laborious. As one way to this problem, it is here presented a possibility of minimizing it and with this research we aim to facilitate the process to storage and making available the learning objects through the interoperability of the virtual learning platform Moodle with the repository of Cognix. This work aims to propose a tool that allows the automatic storage of Moodle learning objects in the Cognix repository. For such, four methodological steps were described for the development of the project as well as for the implementation of the tool. This tool is expected rant reutilization and access to learning objects for their users, thus ensuring the uniform and complete description of the metadata.

**Keywords:** Learning Objects. Learning Environments. Interoperability. Computers in Education.

## LISTA DE FIGURAS

Figura 2.1 - Estrutura hierárquica arquitetura do padrão OBAA. ....	20
Figura 2.2 - Softwares usados para a criação de repositórios mais usados .....	23
Figura 2.4 - Níveis de interoperabilidade de acordo com IoT .....	30
Figura 3.1 - Modelo de integração Proposto .....	35
Figura 3.2 - Telas de pesquisa e resultados .....	35
Figura 3.3 - Esquema de comunicação proposta por Díaz et al (2013).....	37
Figura 4.1 - Interface <i>web</i> do Repositório Cognix .....	41
Figura 4.2 - Arquitetura do Repositório Cognix.....	43
Figura 4.3 - Passos de catalogação de OAs no repositório Cognix.....	44
Figura 5.1 - Arquitetura proposta .....	46
Figura 5.2 - Tela do Plugin.....	49
Figura 5.3 - Fluxograma geral de depósito.....	54
Figura 5.4 - Diagrama de depósito .....	55
Figura 5.5 - Tela de submissão.....	56



## LISTA DE TABELAS

Tabela 1.1 - Etapas da pesquisa.....	15
Tabela 2.1 - Descrição resumida das categorias do padrão LOM.....	19
Tabela 2.2 - Metadados do padrão DC .....	21
Tabela 3.1 - Análise comparativa dos trabalhos relacionados.....	38
Tabela 5.1 - Estrutura de diretórios do <i>plugin</i> .....	47
Tabela 5.2 - Descrição de predicados do <i>Statement</i> .....	52
Tabela 5.3 - Parâmetros SWORD.....	56
Tabela 5.4 - Mapeamento do metadados do padrão DC para o OBAA .....	58
Tabela 5.5 - Verbos e Códigos de resposta HTTP .....	59

## LISTA DE ABREVIATURAS

API	<i>Application Program Interface</i>
AVA	Ambiente Virtual de Aprendizagem
CESTA	Coletânea de Entidades de Suporte ao uso de Tecnologias na Aprendizagem
CMS	<i>Content Management System</i>
CRUD	<i>Create, Retrieve (or Read), Update e Delete</i>
DC	<i>Dublin Core</i>
EDIFACT	<i>Electronic Data Interchange for Administration, Commerce and Transport</i>
FEB	Federação de Repositórios Educa Brasil
IEEE	<i>Institute of Electrical and Electronic Engineers</i>
HTTP	<i>HyperText Transfer Protocol</i>
IoT	<i>Internet of Things</i>
IP	<i>Internet Protocol</i>
LMS	<i>Learning Management System</i>
IRI	<i>Internationalized Resource Identifier</i>
LOM	<i>Learning Object Metadata</i>
LORs	<i>Learning Object Repositories</i>
METS	<i>Metadata Encoding and Transmission Standard</i>
MOODLE	<i>Modular Object-Oriented Dynamic Learning Environment</i>
OAs	Objetos de Aprendizagem
OAI	<i>Open Archives Initiative</i>
OAI-PMH	<i>Open Archives Initiative Protocol for Metadata Harvesting</i>
OBAA	Objetos de Aprendizagem Baseados em Agentes
PDF	<i>Portable Document Format</i>
ROA	Repositório de Objeto de Aprendizagem
SCORM	<i>Sharable Content Object Reference Model</i>
SOA	<i>Service-Oriented Architecture</i>
SRU/W	<i>Search/Retrieve via URL/Web Service</i>
SWORD	<i>Simple Web-service Offering Repository Deposit</i>
UFRGS	Universidade Federal do Rio Grande Sul
XML	<i>Extensible Markup Language</i>

## SUMÁRIO

1 INTRODUÇÃO.....	12
1.1 Objetivos.....	14
1.1.1 Objetivo geral .....	14
1.1.2 Objetivos específicos.....	14
1.2 Roteiro da Pesquisa .....	14
1.3 Contribuições.....	16
1.4 Estrutura do Trabalho .....	16
2 FUNDAMENTAÇÃO TEÓRICA .....	18
2.1 Objeto de Aprendizagem .....	18
2.2 Padrões de Metadados para OAs .....	19
2.4 Recursos tecnológicos para a criação de repositórios digitais.....	22
2.5 Ambiente Virtual de Aprendizagem.....	26
2.6 Integração de AVA e ROA.....	28
2.6.1 Interoperabilidade .....	28
2.6.2 Catalogação Semiautomática.....	30
2.6.3 Extração de Metadados.....	31
3 TRABALHOS RELACIONADOS .....	32
3.1 Integração de repositório e AVA.....	32
3.3 Integração de Ambientes Virtual de Aprendizagem com Repositório Digital.....	34
3.4 Integração de LMS com repositório digital: Caso de estudo.....	36
3.5 Depósito de objetos de aprendizagem em repositórios a partir da integração com ambientes virtuais de aprendizagem .....	38
4 REPOSITÓRIO COGNIX.....	40
4.1 Contextualização .....	40
4.2 Arquitetura do Repositório Cognix .....	42
4.3 Processo de catalogação .....	43
5 FERRAMENTA PARA FACILITAR A DISSEMINAÇÃO DE CONTEÚDO EDUCACIONAL .....	45
5.1 Arquitetura da Ferramenta.....	45
5.2 Desenvolvimento .....	46
5.2.1 Implementação do protocolo SWORD no Moodle .....	46
5.2.2 Implementação do protocolo SWORD no ROA Cognix.....	50
5.3 Implementação e validação dos Resultados .....	53
5.4 Teste .....	60
6 CONSIDERAÇÕES E TRABALHOS FUTUROS .....	61
REFERÊNCIAS .....	63

## 1 INTRODUÇÃO

Nas últimas décadas, as tecnologias educacionais vêm se tornando uma das áreas de maior enfoque no contexto do aprendizado suportado por mídias educativas e plataformas virtuais de ensino. Assim sendo, este trabalho parte da seguinte questão de pesquisa: **projetar e desenvolver uma ferramenta para facilitar a disseminação de conteúdo educacional através de sua catalogação automática em repositório de objetos de aprendizagem.**

Nesse sentido, parte-se do pressuposto de que a falta de metadados dos objetos de aprendizagem nos repositórios é considerado um dos desafios a ser ultrapassado. Ainda, por exemplo, se considera a existência de um grande volume de objetos de aprendizagem desenvolvidos e que por sua vez não são reutilizados. Neste contexto, escolheu-se como o objeto de estudo Moodle um ambiente virtual de aprendizagem *open source* de grande utilidade pedagógica e atualmente tem sido utilizado em muitas instituições de ensino.

Um dos focos patentes na pesquisa é o reaproveitamento dos metadados já descritos no Moodle quando o professor faz o *upload* da aula (formato de recurso), o que poderá se tornar um dos ganhos para a disseminação dos conteúdos em repositórios, fazendo com que o professor não despenda mais tempo na especificação detalhada ao catalogar os objetos de aprendizagem direto em repositórios, e que por muitas vezes o mesmo acaba desistindo.

Nesta ordem de ideias, a catalogação dos conteúdos educacionais poderá ser feita automaticamente a partir do ambiente Moodle direto ao repositório de objetos de aprendizagem, com auxílio de protocolo de comunicação e, que neste caso, será utilizado o SWORD<sup>1</sup> que se mostrou mais adequado, de acordo com várias pesquisas realizadas, como Tarouco et al (2013) e Díaz et al (2013). Diante deste contexto, o estudo tem como especial finalidade integrar o Moodle e o Cognix<sup>2</sup>, repositório de objeto de aprendizagem que por si só, já possui um mecanismo que facilite a descrição dos metadados para recursos educacionais como o plano de aula. Neste trabalho focamos no protocolo SWORD na versão 2, pois foi o selecionado para ser utilizado nesta implementação, pelo fato deste permitir uma interoperabilidade entre sistemas de publicação de recursos digitais, (LEWIS et al, 2009). Para além de ser uma especificação bastante usada por repositórios digitais para depósito de recursos a partir de outros sistemas e em diferentes formatos, (STEVENSON, 2009). Lewis, Castro e Jones (2012) apresentam nove senários de aplicação deste protocolo, todos voltados ao depósito de objetos em repositórios,

---

<sup>1</sup> SWORD - *Simple Web-service Offering Repository Deposit*, protocolo de comunicação usado em repositórios digitais para que os mesmos aceitem o depósito de conteúdos digitais provenientes de outros sistemas.

<sup>2</sup> <http://cognix-repo.inf.ufrgs.br/repositorio/documents>

dos quais tem-se: (1) editores para repositórios; (2) sistemas de informação de pesquisa para repositórios; (3) computadores pessoais para repositórios; (4) repositórios para repositórios; (5) interfaces de depósito para usuários especializados para repositórios; (6) Sistemas de gerenciamento de conferências para repositórios; (7) equipamentos de laboratório para repositórios; (8) depósito em massa para repositórios; (9) autoria colaborativa.

O repositório Cognix foi escolhido para esta implementação por se tratar de uma tecnologia desenvolvida no grupo de pesquisa e que também é o primeiro repositório que utiliza como padrão de metadados OBAA (VICARI et al, 2009), que também foi desenvolvido por este grupo de pesquisa. Além disso, o repositório Cognix providência facilidades para a extração automática de metadados a partir do upload do arquivo de conteúdo do objeto. A esta facilidade vai ao encontro dos objetivos do presente trabalho.

Assim, pretende-se adequar uma nova forma de se catalogar os objetos de aprendizagem baseando-se em uma ferramenta *open source* e que diminua não só o esforço do professor no momento da catalogação, mas também a descrição de metadados que outrora era feita manualmente.

## **1.1 Objetivos**

Para que a pesquisa se foque no que diz respeito a catalogação e conseqüentemente a disseminação de conteúdo educacional, se espera o alcance dos seguintes objetivos:

### **1.1.1 Objetivo geral**

Como objetivo geral do estudo, propomos uma ferramenta que permite a catalogação automática de objetos de aprendizagem do ambiente virtual de aprendizagem Moodle no repositório Cognix.

### **1.1.2 Objetivos específicos**

- E1 - Estudar as formas de integração de Moodle com os repositórios de objetos de aprendizagem;
- E2 - Implementar no Moodle a ferramenta de comunicação com repositórios;
- E3 - Implementar no repositório Cognix o protocolo de comunicação com o Moodle;
- E4 - Desenvolver um módulo para preenchimento automático de metadados do objeto de aprendizagem;
- E5 - Testar a ferramenta desenvolvida para a integração do repositório Cognix com o Moodle;
- E6 - Publicar os resultados tanto na forma de dissertação de mestrado como na forma de artigos científicos.

## **1.2 Roteiro da Pesquisa**

De modo a estruturar o trajeto da pesquisa, a Tabela 1.1 apresenta um roteiro de quatro etapas de modo a classificar as atividades do desenvolvimento da ferramenta de catalogação de objetos de aprendizagem, identificadas por A, B, C e D, seguidas de seus objetivos (refere-se aos objetivos específicos do trabalho) e atividades realizadas.

Tabela 1.1- Etapas da pesquisa

<i>Etapa</i>	<i>Objetivos</i>	<i>Atividades</i>
A – Teórica	Estudar as formas de integração de Moodle com os repositórios de objetos de aprendizagem.	Revisão bibliográfica em torno de ambientes virtuais de aprendizagem, metadados, repositórios de objetos de aprendizagem e interoperabilidade. Revisão bibliográfica sobre integração de ambientes virtuais de aprendizagem com repositórios. Revisão sobre o estado da arte.
B – Análise e Desenvolvimento	Desenvolver um módulo para preenchimento automático de metadados dos objetos de aprendizagem.	Desenho da arquitetura, análise e desenvolvimento da ferramenta.
C – Implementação	Implementar no Moodle a ferramenta de comunicação com repositórios.	Implementação do protótipo.
	Implementar no repositório Cognix o protocolo de comunicação com o Moodle.	
D – Teste e Resultados	Testar a ferramenta desenvolvida para a integração do repositório Cognix com o Moodle.	Teste do protótipo e avaliação dos resultados.
	Publicar os resultados tanto na forma de dissertação de mestrado como na forma de artigos científicos.	

Fonte: O autor (2015)

Na primeira etapa (etapa A - Teórica) como atividades tem-se: (a) revisão bibliográfica sobre ambientes virtuais de aprendizagem, repositórios, padrões de metadados, objetos de aprendizagem, e meios de integração em sistemas; (b) levantamento e estudo de pesquisas relacionadas com a disseminação de conteúdo educacionais em sistemas/ambientes virtuais de aprendizagem. A etapa B, referente à análise e desenvolvimento, tem como atividades desenho da arquitetura, análise e desenvolvimento da ferramenta recorrendo aos seguintes recursos tecnológicos (a) Moodle, (b) repositório Cognix, (c) protocolo de comunicação SWORD. A terceira etapa (etapa C – implementação) tem como enfoque a implementação da ferramenta nas plataformas Moodle, DSpace e Cognix. E por fim, a etapa D, teste e resultados com enfoque nas atividades de teste do protótipo, avaliação dos resultados e publicação dos resultados.

### 1.3 Contribuições

Com o aumento considerável de repositórios de objetos de aprendizagem, que permite o acesso à grande quantidade de dados em diferentes áreas do saber, favorecendo deste modo, tanto os alunos como o professor, com este trabalho pretendemos facilitar o partilha e acesso aos conteúdos educacionais. A presente pesquisa enquadra-se no projeto de pesquisa ROA-MOBRAS/2013 (Repositório de Objetos de Aprendizagem Moçambique-Brasil), uma cooperação entre a Universidade Pedagógica de Moçambique (UPM) e Universidade Federal do Rio Grande do Sul (UFRGS) financiado pela Capes.

Como contribuição, a pesquisa auxiliará a área de Tecnologias Educacionais na reutilização de conteúdo educacional através do ambiente virtual de aprendizagem Moodle. Particularmente em Moçambique, no contexto do programa de ensino à distância que é um regime de ensino adotado por várias universidades, pode-se vincular o uso da ferramenta em ambientes virtuais de aprendizagem de modo a apoiar nas atividades dos professores e alunos bem como os gestores. Do ponto de vista do desenvolvimento de soluções tecnológicas para sistemas educacionais, a contribuição desta pesquisa e desenvolvimento encontra-se na interoperabilidade entre sistemas. Ou seja, na busca da interoperabilidade entre um repositório de OAs e um Ambiente Virtual de Aprendizagem. Esta interoperabilidade facilita o trânsito de conteúdo educacional entre estes dois *softwares*, seja para seu armazenamento ou para a sua utilização. É também contribuição deste trabalho o desenvolvimento de um modelo para o enriquecimento dos metadados armazenados no repositório.

### 1.4 Estrutura do Trabalho

A dissertação segue a seguinte organização: no capítulo 1 faz-se um relato introdutório do trabalho apresentando a justificativa e motivação que levaram ao estudo e investigação, objetivos, problema de pesquisa, metodologia e etapas da pesquisa e, por fim, os trabalhos relacionados são apresentados. No capítulo 2 são tratadas as definições de conceitos que facilitam o acompanhamento e a percepção do trabalho em causa. No capítulo 3 são apresentados os trabalhos relacionados. No capítulo 4 faz-se uma breve apresentação em torno do repositório de objeto de aprendizagem Cognix. O capítulo 5 trata-se do desenvolvimento, implementação e validação da ferramenta no Moodle e no repositório Cognix. Por fim, no capítulo 6, segue-se apresentando as considerações finais contendo reflexões referentes às limitações encontradas, recomendações em relação aos resultados da pesquisa, além disso,



sugestões à possível continuidade de estudo e pesquisa em outros ambientes de ensino e/ou repositórios digitais

## 2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados estudos que darão suporte à pesquisa e desenvolvimento da ferramenta de integração entre ambiente virtual de aprendizagem e o repositório de objetos de aprendizagem de modo a facilitar a catalogação automática dos objetos de aprendizagem, bem como a disseminação de conteúdo educacional. Neste contexto, discutimos os conceitos de objetos de aprendizagem, padrões e características, recursos tecnológicos para a criação de repositórios digitais, ambiente virtual de aprendizagem, integração de ambiente virtual de aprendizagem e repositório de objetos de aprendizagem, interoperabilidade, catalogação semiautomática e extração de metadados.

### 2.1 Objeto de Aprendizagem

O avanço da tecnologia educacional possibilitou o desenvolvimento e criação de material pedagógico que é possível armazenar para posterior reutilização em diferentes contextos de ensino. Tais materiais são designados também como Objetos de Aprendizagem (OAs) entidades digitais utilizadas em diferentes contextos de aprendizagem. Nessa perspectiva, diversas definições são encontradas para o conceito de “objeto”, outros ainda ressaltam aspectos da “aprendizagem”, como Wiley (2001), considera que um objeto de aprendizagem é qualquer recurso digital que possa ser reutilizado no suporte ao ensino em múltiplos contextos educativos. Esta definição (WILEY, 2001) será a utilizada neste trabalho. No entanto apresentamos a seguir outras definições, que também são utilizadas pela comunidade da área. Para IEEE (2002), um Objeto de Aprendizagem é definido como qualquer entidade, digital ou não, que possa ser usada, reutilizada e referenciada durante o processo de aprendizagem ou treinamento. OAs podem ser fotos ou imagens digitais, animações, simulações e pequenas aplicações, *sites* ou mesmo aplicações. Uma das principais propriedades de um objeto de aprendizagem é sua reusabilidade. Tal característica pode ser alcançada através da modularidade que descreve o grau de separação e conseqüente recombinação dos componentes do OA, interoperabilidade capacidade de operar em plataformas heterogêneas e recuperação que é capacidade de poder ser encontrado em função da descrição de suas propriedades e funcionalidades.

## 2.2 Padrões de Metadados para OAs

Metadados ou “dados sobre dados” permitem que um determinado recurso educacional seja identificado e descrito, dando a conhecer a proveniência e o autor do mesmo, a que grupo é direcionado, sua aplicação, é ou não interativo e outras características ou informações relevantes que possam auxiliar o seu entendimento e utilização. De acordo com Vicari et al, 2010, um dos principais objetivos do metadados é facilitar a busca, avaliação, recuperação e utilização/reutilização dos recursos digitais.

Metadados descrevem ainda, as características de objeto (dado) de modo a serem catalogado num determinado repositório e recuperados através de sistemas de busca. Para que os OAs sejam armazenados em repositórios e reutilizados em diversos ambientes de ensino, é necessária a descrição de seu conteúdo de forma padronizada, o que permitirá o intercâmbio de informação (VICARI et al. 2009). Existem hoje, especificações/padrões de metadados aceitos mundialmente para OAs dos quais se destacam:

**LOM - Learning Object Metadata** da *IEEE Learning Technology Standards Committee* é uma das primeiras especificações de metadados educacionais a surgir, garante a descrição das principais características do OA (IEEE, 2002). Tais características foram agrupadas nas seguintes categorias: *general, life cycle, meta-metadata, technical, rights, relation, annotation* e *classification*. Em cada categoria há uma sequência de metadados com seus elementos e valores. O padrão LOM foi desenvolvido exclusivamente para especificar OAs podendo ser digitais ou não. O LOM inclui pouco mais de 70 elementos divididos em 9 categorias distintos de metadados, a Tabela 2.1 faz referências às categorias e respetiva descrição do padrão LOM.

Tabela 2.1 - Descrição resumida das categorias do padrão LOM

<i>No</i>	<i>Categoria</i>	<i>Descrição</i>
1	General	Informações gerais sobre o objeto de aprendizagem.
2	Lifecycle	Agrupa informações sobre o histórico e o estado atual do OA.
3	Meta-metadata	Informações sobre os metadados do OA.
4	Technical	Agrupa informações sobre requisitos e características técnicas do OA.
5	Educational	Agrupa informações sobre características educacionais e pedagógicas do OA.
6	Rights	Informações sobre propriedade intelectual
7	Relation	Relacionamentos entre este OA e outros OAs
8	Annotation	Anotações e comentários sobre uso educacional;
9	Classification	Posicionamento do OA frente a sistemas de classificação.

Fonte: O autor (2016)

**OBAA** - padrão de metadados brasileiro para objetos de aprendizagem, é uma extensão do padrão de metadados LOM (VICARI et al, 2010). OBAA é abreviação de Objetos de Aprendizagem Baseados em Agentes, foi desenvolvido por pesquisadores da UFRGS em parceria com outras universidades, cujo intuito é atender os requisitos de portabilidade e que também facilite o uso de recursos digitais no contexto educacional brasileiro. O grande diferencial em relação aos outros padrões de metadados é o fato deste estar orientado a interoperabilidade dos OAs em diferentes plataformas tais como: Web, dispositivos móveis, televisão digital e demais plataformas (VICARI et al, 2009).

A especificação do padrão OBAA tem como base o padrão de metadados LOM, ou seja, é formado por todas categorias do LOM e foram complementadas as categorias técnica e educacional, também acrescentou-se duas novas categorias acessibilidade e segmentação. A Figura 2.1 mostra a estrutura hierárquica arquitetura do padrão LOM.

Figura 2.1 - Estrutura hierárquica arquitetura do padrão OBAA.



**SCORM** - *Sharable Content Object Reference Model* da *Advanced Distributed Learning* (ADL) é um conjunto de especificações e padrões para objetos de aprendizagem baseado na *web* usados normalmente e sistemas *e-Learning*. O SCORM permite encapsular conteúdos (objetos de aprendizagem) de uma unidade de aprendizagem em um único local que são definidos em uma determinada sequência e contendo dados (TAROUCO et al, 2014).

**Dublin Core** - é uma especificação de metadados mantida pela *Dublin Core Metadata Initiative* (DCMI, 2015), apresenta um vocabulário composto por 15 elementos incluindo os mais usados pelos professores e *designers* instrucionais para especificar recursos digitais podendo repetir caso haja necessidade. No contexto educacional é muito usado, pois a sua simplicidade e adaptabilidade facilita a descrição de qualquer recurso digital. A Tabela 2.2, mostra os elementos descritores do padrão Dublin Core (DC) e respetiva definição.

Tabela 2.2 - Metadados do padrão DC

<i>No.</i>	<i>Elemento</i>	<i>Definição</i>
1	Title	Nome dado ao recurso
2	Subject	Tópico do conteúdo do recurso (exemplo: palavra-chave, código de classificação)
3	Description	Descrição do recurso (exemplo: resumo, sumário, notas)
4	Language	Idioma do recurso
5	Source	Referência a um recurso a partir do qual o recurso descrito foi derivado (exemplo: documento origem da digitalização)
6	Relation	Referência a recursos relacionados (exemplo: coleção a que faz parte, livro a partir do qual o filme foi adaptado)
7	Coverage	Tópico espacial (nome de lugar ou coordenadas geográficas), temporal (período, data) ou jurisdição do recurso (abrangência de uma lei)
8	Creator	Entidade (pessoa, organização ou serviço) primariamente responsável pela confecção do recurso (exemplo: autor)
9	Contributor	Entidade (pessoa, organização ou serviço) responsável por ter realizado contribuições ao recurso (exemplo: ilustrador, tradutor, orientador, diretor).
10	Publisher	Entidade responsável por tornar o recurso disponível
11	Rights	Informações sobre os direitos sobre o recurso
12	Date	Data associada a um evento do ciclo de vida do recurso
13	Type	Natureza ou gênero do recurso (exemplo: texto, imagem, tese, dissertação, livro ...)
14	Format	Formato de um arquivo, meio físico ou dimensão de um recurso (exemplo: html, os, doc, mp3, wave, jpeg, mpeg, duração de 30 min, tamanho 300MB, ...)
15	Identifier	Uma referência não ambígua do recurso em um dado contexto (exemplo: URI, ISSN, ISBN, ...)

Fonte: Adaptado, <http://www.ufrgs.br/snote/wiki/doc.php?id=190> (2016)

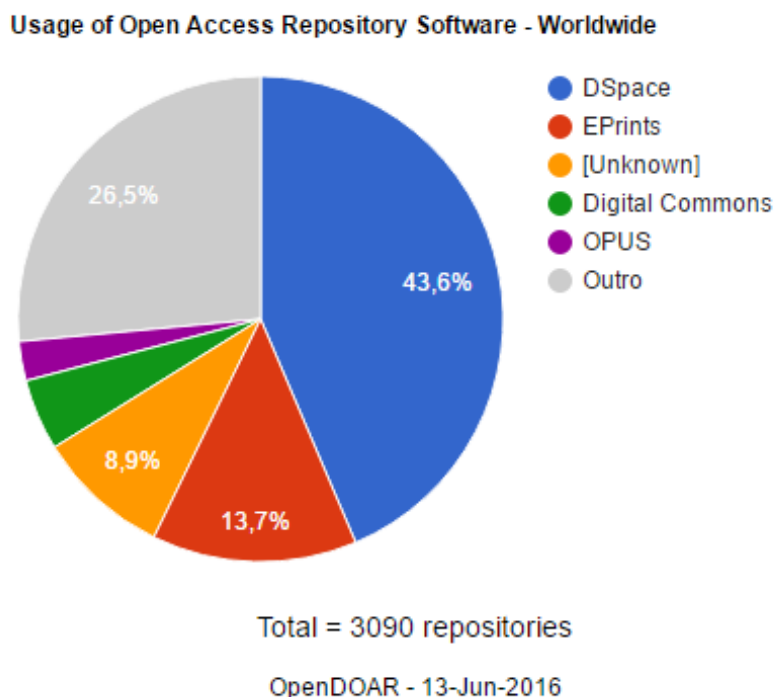
## 2.4 Recursos tecnológicos para a criação de repositórios digitais

Repositórios digitais são ferramentas que permitem a coleta, o armazenamento e compartilhamento de dados digitais institucionais ou de outras áreas de interesse, de modo que tais dados possam ser disponibilizados on-line.

Os repositórios digitais podem ainda, serem considerados bancos de dados online que reúnem de maneira organizada a produção científica de uma instituição ou área temática, armazenam arquivos de diversos formatos. Ainda, resultam em uma série de benefícios tanto para os pesquisadores quanto às instituições ou sociedades científicas, proporcionam maior visibilidade aos resultados de pesquisas e possibilitam a preservação da memória científica de sua instituição (IBICT, 2015).

No que se refere ao ensino, é possível criar um ambiente onde serão armazenados objetos de aprendizagem, nomeados Repositório de Objetos de Aprendizagem (ROAs). OS ROAs também designado por LORs (do inglês *Learning Object Repositories*) são bibliotecas digitais que contém material educacional e com o propósito de garantir o seu compartilhamento e reuso em outros ambientes educacionais (OCHOA e DUVAL, 2009). Os repositórios de objetos de aprendizagem garantem as funcionalidades típicas de um repositório digital (armazenamento, pesquisa e recuperação de OAs como de metadados) aos usuários finais (SAMPSON e ZERVAS, 2013). Tarouco et al (2014) afirmam que repositórios de objetos de aprendizagem são espaços que permitem o armazenamento, pesquisa e a reutilização de objetos de aprendizagem. Atualmente, existem vários recursos tecnológicos (*softwares*) de código aberto como também comerciais que o possibilitam a criação e gestão dos repositórios digitais. A plataforma DSpace, até hoje, é a mais usada mundialmente para a criação de repositórios digitais, conforme a Figura 2.2 detém mais de 43% do número total de softwares usados mundialmente para a criação de repositórios digitais, de acordo com dados disponíveis no OpenDOAR (2016).

Figura 2.2 - Softwares usados para a criação de repositórios mais usados



Fonte: Adaptado, OpenDOAR (2016)

**DSpace**<sup>3</sup> desenvolvido pelo *Hewlett Packard* (HP) em coordenação com *Massachusetts Institute of Technology* (MIT), atualmente o mais utilizado para a criação de repositórios institucionais e multidisciplinares para bibliotecas digitais como também de objetos de aprendizagem. De acordo com Smith (2003), o DSpace fornece meios de gerenciamento de materiais de pesquisa e publicações em um repositório profissionalmente mantido que garanta maior visibilidade e acessibilidade.

O DSpace surgiu com a necessidade de resolver o problema que as universidades tinham em coletar, preservar e disponibilizar os resultados de suas pesquisas e publicações acadêmicas. Como *software* livre e sem fins lucrativos tem como intuito fornecer meios de gerenciamento de recursos digitais (publicações acadêmicas) de modo a atender as necessidades de qualquer organização.

DSpace foi desenvolvido usando a linguagem Java e suporta ainda ferramentas *open source* tais como: PostgreSQL<sup>4</sup>, Tomcat, Apache Solr<sup>5</sup> (*software* de busca), dentre outros.

<sup>3</sup> <http://www.dspace.org/>

<sup>4</sup> <https://www.postgresql.org/>

<sup>5</sup> Solr é uma plataforma de busca corporativa de código aberto que permite a pesquisa e navegação filtrada (facetada) de todos os objetos (<http://www.dspace.org/techspecs>). O Solr tem como base projeto Apache Lucene.

O processo de submissão de objetos no DSpace é descrito usando uma versão qualificada do vocabulário de metadados Dublin Core, no qual três campos são obrigatórios: título, idioma e data de submissão.

**Cognix** é um repositório de objetos de aprendizagem que tem como objetivo principal facilitar a catalogação de materiais educacionais no padrão de metadados OBAA, desenvolvido pela Cognitiva Brasil<sup>6</sup>, uma empresa vocacionada e especializada em tecnologias educacionais. O Cognix surgiu a partir de uma demanda de instituições educacionais brasileiras de ter um repositório de fácil instalação e que facilitasse a catalogação de materiais educacionais no padrão OBAA, visto que os outros existentes no mercado tinham uma complexidade de instalação bem alta e não disponibilizavam a alteração do padrão de metadados OBAA. O Cognix possui um módulo para facilitar a catalogação, onde é possível definir um perfil de metadados, como por exemplo “plano de aula” e ao inserir um objeto com esse perfil o sistema gera vários metadados automaticamente. Esse repositório é utilizado em diversas universidades brasileiras inclusive pela secretaria de ensino básico do Ministério da Educação<sup>7</sup>.

O Cognix possui algumas características importantes, como por exemplo, foi desenvolvido em Java, possui a implementação do protocolo OAI-PMH (*Open Archives Initiative Protocol for Metadata Harvesting*) para que seja possível a coleta por outro repositório ou federação de repositórios e possui uma empresa brasileira que presta suporte para o sistema. Além de utilizar o padrão OBAA que é uma extensão do LOM, permitindo assim a compatibilidade com outros padrões como Dublin Core, do LOM ou todos do OBAA.

**EPrints** desenvolvido pela *University of Southampton*, tendo se mostrado apropriado para a construção de repositórios institucionais, *software* de código aberto adaptável às necessidades de qualquer instituição de ensino e pesquisa. Segundo Gutteridge (2002), EPrints<sup>8</sup> é um *software* livre no qual pode-se criar um depósito *web* de objetos educacionais. Esse depósito *online* de "coisas", que possuem metadados e arquivos de conteúdo que podem ser carregados para *web*. Estas "coisas" são geralmente material acadêmico, como trabalhos de pesquisa e teses. Obviamente é uma ferramenta voltada para a Internet das Coisas.

EPrints também realiza a coleta dos dados pelo protocolo OAI-PMH, permitindo ainda, o depósito de produções de pesquisa podendo estas serem *pre-print* ou *post-print* e é baseado no padrão de metadados o Dublin Core.

---

<sup>6</sup> <http://cognitivabrasil.com.br/>

<sup>7</sup> <http://cognix-repo.inf.ufrgs.br/repositorio/documents>

<sup>8</sup> <http://software.eprints.org>



**Fedora** - *Flexible Extensible Digital Object and Repository Architecture* é uma plataforma de código aberto, desenvolvido pela *University of Virginia* em cooperação com *Cornell University*, oferece uma arquitetura projetada para servir como base para a implementação de repositórios digitais para uma grande variedade de aplicações, podendo ser bibliotecas digitais, repositórios institucionais e outros sistemas gerenciadores de informação, (PAYETTE E LAGOZE, 1998). Tem como padrão de metadados Dublin Core.

**i-Tor**<sup>9</sup> (*Tools and technologies for Open Repositories*) foi desenvolvido pelo ITA (*Innovative Technology Applied*), uma seção do Instituto para Serviços de Informação Científica da Holanda. Este repositório permite que seus usuários cataloguem vários tipos de informação, independentemente do local onde os dados são armazenados ou o formato em que são apresentados.

**MyCoRe**<sup>10</sup> desenvolvido pela Universidade de Essen oferece ferramenta de suporte para bibliotecas digitais ou repositórios de conteúdo digital (*Content Repositories, CoRe*). O software foi projetado para ser configurado e adaptado às necessidades de cada instituição (de onde provém o *My*), sem que haja esforços de programação. O MyCoRe suporta padrão de metadados Dublin Core e pode-se importar e exportar qualquer outro tipo de metadado como XML juntamente com os arquivos associados.

**OPUS**<sup>11</sup> (*Online Publications of the University of Stuttgart*) desenvolvido em 1998 pela *University Library* e o *Computing Center of the University of Stuttgart*, tendo como principal foco oferecer um sistema onde estudantes, professores e universidade em geral pudessem gerenciar suas publicações eletrônicas, incluindo artigos dissertações e teses. O OPUS utiliza o padrão de metadados Dublin Core.

**Archimède** desenvolvido pela *Laval University Library*, Canadá. Modelado para permitir o depósito de *pre-prints* (primeira versão de um documento publicado) e *post-prints* (versão final de um documento após revisões do autor) de universidades e pesquisadores, inspirado no modelo DSpace o repositório Archimède emprega a concepção de comunidades de pesquisa auto reguladas que podem ser responsáveis pela criação e gestão dos conteúdos, (BENJELLOUN, 2005). Armazena documentos em diversos formatos de arquivo como: HTML, PDF, MS Word, MS PowerPoint, MS Excel, Open Office Suíte, TXT, RTF e outros. O sistema está em conformidade com os padrões e princípios do modelo dos arquivos abertos (OAI-PMH) e o conjunto de elementos metadados baseado no Dublin Core.

---

<sup>9</sup> file:///C:/Users/Rafael/Downloads/GL6\_Dijk.ppt

<sup>10</sup> <http://www.mycore.de/>

<sup>11</sup> <http://www.ub.uni-stuttgart.de/forschen-publizieren/opus/>

**CDSware**<sup>12</sup>(*Document Server Software*) desenvolvido pela CERN (*European Organization for Nuclear Research*). O CERN utiliza o CDSware para gerenciar cerca de 350 coleções de dados, englobando cerca de 550.000 registros bibliográficos e 220.000 documentos completos, incluindo *pre-prints*, jornais, artigos, livros e fotografias, (SITAS, 2006). O CDSware suporta padrões de metadados no formato Dublin Core e o formato MARC 21<sup>13</sup> (*Machine Readable Cataloging* padrão de estrutura de metadados descritivos para o compartilhamento de dados bibliográficos).

De forma geral, observou-se que determinados softwares para a criação de repositórios já apresentam e utilizam mecanismos de interoperabilidade e integração com outros sistemas. O DSpace é o mais complexo de todos e apresenta em sua estrutura diferentes módulos inclusive o SWORD já padronizado nas suas versões v1 e v2.

## 2.5 Ambiente Virtual de Aprendizagem

O ambiente virtual de aprendizagem (AVA) também designado por *Learning Management System* (LMS) integra recursos de aprendizagem com o objetivo de construir um conjunto de instruções, orientações, atividades e interações que formam um curso (TAROUCO et al, 2014). Os ambientes de virtuais de aprendizagem tendem a evoluir bastante tornando-se sistemas robustos. A Figura 2.3 mostra exemplos de LMS, dos quais o Moodle, Rooda e Blackboard são descritos a seguir.

Figura 2.3 – Exemplo de AVAs



Fonte: O autor (2015)

<sup>12</sup> <https://cds.cern.ch/record/725772/files/cer-002439316.pdf>

<sup>13</sup> <https://www.loc.gov/marc/faq.html>

Atualmente, existem várias LMS disponíveis de distribuição livre ou comerciável dos quais se destacam o Moodle atualmente mais utilizado pelas comunidades acadêmicas, o ROODA e Blackboard comercialmente mais utilizada.

**Moodle** - *Modular Object-Oriented Dynamic Learning Environment* plataforma virtual de aprendizagem robusto de código aberto desenvolvido sob princípios pedagógicos (DOUGIAMAS, 2003). O Moodle além de permitir um ambiente virtual de aprendizagem oferece vários instrumentos (*plugin*) para a criação de cursos e conteúdos educacionais. Possui ainda suporte a padrões de metadados e interoperatividade agregando aplicações externas a ele.

Atualmente, é um dos ambientes mais populares entre as comunidades acadêmicas, dentre várias, pode-se citar:

- Composição flexível das atividades dos cursos, tais como fóruns, *wikis*, questionários, pesquisas de opinião, tarefas, chats.
- Extensibilidade, por meio de *plugins* com novos temas, novos módulos de atividades e pacotes de idiomas, entre outras opções.
- Suporte aos padrões SCORM, Dublin Core e IMS *Content Package*, permitindo o compartilhamento de recursos com outros ambientes.
- Documentação extensa, tanto para administradores quanto para desenvolvedores, e uma comunidade ativa, como se pode perceber pelo número de usuários registrados.

**ROODA** - Rede Cooperativa de Aprendizagem é um ambiente virtual de educação à distância, desenvolvida pelo Núcleo de Tecnologias Digitais Aplicadas à Educação (NUTED) da Universidade Federal do Rio Grande do Sul, (BEHAR et al, 2001). A ROODA foi desenvolvida de acordo com a filosofia de software livre e incorpora ferramentas como fórum e chat também usadas em outras plataformas.

**Blackboard**<sup>14</sup> - ambiente virtual de aprendizagem baseado na plataforma web, além de facilitar o acesso *online* dos alunos, oferecendo uma flexibilidade e complexidade que garantem um gerenciamento completo dos cursos. De acordo com Bradford et al (2007), Blackboard tornou-se o LMS comercial dominante.

---

<sup>14</sup> <http://www.blackboard.com/about-us/index.aspx>

## 2.6 Integração de AVA e ROA

Com surgimento de plataformas de ensino mais inteligentes, surgem também várias discussões em torno do reuso dos recursos educacionais. A integração entre sistemas é uma das formas na qual pode-se viabilizar o reuso dos OAs como também a diminuição de etapas de catalogação. A partir da versão 2.0 o Moodle já disponibiliza *plugins* específicos para que se possa integrar repositórios digitais como Picasa, Amazon S3, youtube, Facebook, DSpace, dentre outros. Tal integração faz com que tanto o Moodle quanto ROA adotem mecanismo de comunicação (protocolos de comunicação) entre si, dos quais se destacam: ATOM<sup>15</sup>, SRU/W<sup>16</sup> (*Search/Retrieve via URL/Web service*), SWORD<sup>17</sup> (*Simple Web-service Offering Repository Deposit*), OAI-PMH<sup>18</sup> e RSS<sup>19</sup> (*Real Simple Syndicate*). Assim sendo, é meta do estudo adequar um mecanismo de comunicação, especificamente o protocolo SWORD para integração do Moodle e o repositório Cognix.

### 2.6.1 Interoperabilidade

A interoperabilidade pode ser entendida como sendo uma capacidade que pressupõe a comunicação de múltiplos sistemas, garantindo assim o compartilhamento de informação e serviços. De acordo com o glossário<sup>20</sup> da IEEE, a interoperabilidade é uma habilidade que um sistema possui, de forma a interagir com outros sistemas sem que haja um esforço por parte dos clientes, e isso só é possível quando implementado padrões de comunicação. Em ambientes de aprendizagem permitem que recursos sejam recomendados, recuperados, compartilhados e disponibilizados com ajuda de uma padronização de metadados e de comunicação. Protocolos com o SWORD, SRU/SRW (evolução simplificada do protocolo Z39.50) e OAI-MH são exemplos de regras e padrões adotados pelos repositórios.

No contexto da Internet das Coisas (IoT), que é uma proposta que visa conectar todos os dispositivos entre si e também permitir a troca de informações desses dispositivos pela web, mudando a nossa interação com o ambiente. Hoje em dia, já podemos encontrar refrigeradores,

---

<sup>15</sup> <http://www.atomenabled.org/>

<sup>16</sup> <http://www.loc.gov/standards/sru/>

<sup>17</sup> <http://SWORDapp.org>

<sup>18</sup> <http://www.openarchives.org/pmh/>

<sup>19</sup> <http://www.rssboard.org/rss-specification>

<sup>20</sup> Disponível: [https://www.ieee.org/education\\_careers/education/standards/standards\\_glossary.html](https://www.ieee.org/education_careers/education/standards/standards_glossary.html)

carros, lâmpadas, entre outros dispositivos, que podem ter acesso a internet e usar seus dados para fornecer ao usuário facilidades.

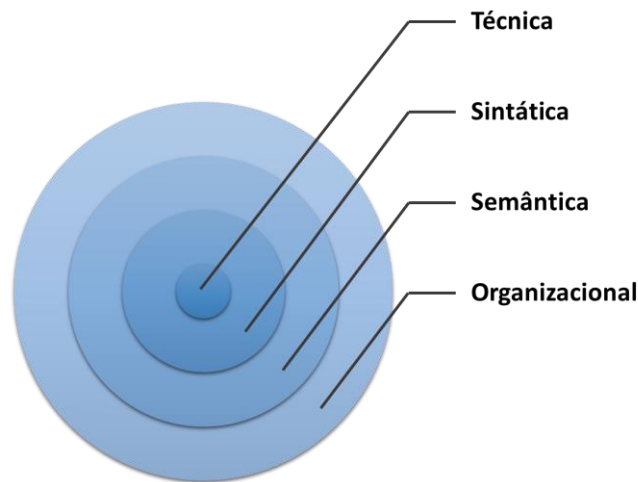
Apesar de ser simples a definição, compreender o que é realmente IoT pode ser bastante difícil (ATZORI; IERA; MORABITO, 2010). Isso porque existem três perspectivas que devem ser analisadas. Na perspectiva das “Coisas” as tecnologias visam identificar de forma única, acessar e manipular um objeto, um exemplo disso é o RFID, que foi o precursor do conceito de IoT. A perspectiva de “Internet” tem como foco fundamentar IoT sobre o protocolo IP, afim de tornar os dispositivos endereçáveis a qualquer outro. A visão “Semântica” tem como principais desafios a representação dos dispositivos e informações do sistema. A convergência dessas três perspectivas leva a Internet das Coisas.

A arquitetura que mais vem ganhando espaço em IoT é o SOA (*Service Oriented Architecture*). Em (KRAFZIG; BANKE; SLAMA, 2004) é proposta uma convergência das três perspectivas que resulta no conceito de Internet das Coisas

As componentes da SOA são: *Application frontend* que é a forma como os usuários conseguem interagir com as outras aplicações SOA; *Service* que representa serviços que são compostos por *contrat* (como o serviço é chamado), *Implementation* (a implementação do serviço) e *interface* (interface de acesso ao serviço). O *service repository* é utilizado pelo usuário ou aplicação para obter maiores informações sobre o serviço que será utilizado. O *Service Bus* é responsável pela integração dos elementos da arquitetura SOA.

Atualmente a pesquisa e a proposição de padrões para a interoperabilidade vêm alcançando grande destaque, principalmente no contexto da Internet das coisas (IoT). Um exemplo é o protocolo EDIFACT (*Electronic Data Interchange for Administration, Commerce and Transport*) da IoT (KUBICEK e CIMANDER, 2009). Ressalta-se que o foco deste trabalho não está ligado ao protocolo IoT, mas por ser um tópico da atualidade o mesmo traz o conceito da interoperabilidade dividido em níveis de acordo com a Veer e Wiles (2008) conforme mostra a Figura 2.4.

Figura 2.4 - Níveis de interoperabilidade de acordo com IoT



Fonte: Adaptado, Veer e Wiles (2008)

De acordo com a Figura 2.4, o nível de interoperabilidade técnica é responsável pela segurança da transferência de dados, a interoperabilidade semântica é responsável pelo processamento e interpretação dos dados recebido, de modo que haja uma compreensão por parte de todos envolvidos na comunicação, a interoperabilidade sintática é responsável pelo tratamento dos dados recebidos e por fim o nível de interoperabilidade organizacional responsável pela ligação automática dos processos em diferentes sistemas quer seja nos modelos arquiteturais ou modelos de padronização de processos como SOA.

No contexto deste trabalho, o conceito de interoperabilidade é definido como sendo a capacidade de uso de um mecanismo interpretável (protocolo) comum para a comunicação entre dois ambientes.

### 2.6.2 Catalogação Semiautomática

A catalogação é um processo de descrição de características e aspetos de um determinado recurso cuja finalidade é atender as necessidades dos usuários, tais como acessibilidade, identificação e localização, permitindo, assim o acesso remoto. Diante desta necessidade, catalogar OAs adequadamente melhora seus resultados de busca e recuperação.

Em ROAs, a catalogação dos OAs pode ser feita recorrendo aos processos manual e/ou automático. Na catalogação manual o usuário informa as características do objeto preenchendo o formulário e posteriormente o *upload* do objeto. Todavia, o processo de catalogação manual

é suscetível a erros de digitação ou a não informação (preenchimento) de certos metadados referente ao objeto.

A catalogação automática possui uma filosofia diferente no seu processo e descrição do objeto, permitindo assim, a obtenção de bons resultados de busca e diminuição dos passos usados na catalogação manual. No repositório Cognix a catalogação dos objetos é semiautomática e com inferência uma boa parte dos metadados facilitando assim, o processo de submissão de OAs que é apresentado no subcapítulo 4.2.

### 2.6.3 Extração de Metadados

A extração de metadados é uma tarefa complexa devido à diversidade dos recursos, no entanto a extração melhora a qualidade dos metadados e facilita o processo de submissão e busca de recursos em repositórios diminuindo os passos (desnecessários) que o usuário normalmente os seguem.

Com catalogação automatizada em repositório digitais e inclusive a extração de metadados provenientes do Moodle, de forma a ajudar o usuário no processo de depósito de OAs, diminuindo assim os passos de descrição e melhorando a quantidade e qualidade dos metadados Casali et al (2014) estabelecem os seguintes benefícios:

- A possibilidade da reutilização dos objetos de aprendizagem em outros ambientes/sites;
- A garantia de uma descrição uniforme e completa dos metadados que é uma das dificuldades verificadas em muitos ROAs;
- A contribuição para a minimização da falta de metadados nos ROAs.

A melhora da qualidade dos metadados e diminuição de passos no processo de catalogação dos OAs, quer em repositórios como através de sistemas intermediários requer uma padronização unificada, visto que existe uma grande diversidade de repositórios como também de recursos.

### 3 TRABALHOS RELACIONADOS

Com o avanço tecnológico e conseqüentemente a evolução de novas formas de ensino baseados em tecnologia surgem os AVAs, que são *software* que visam garantir a interação entre o professor e aluno virtualmente. A necessidade de se reaproveitar e conservar os OAs produzidos pelos professores de modo que os mesmos sejam usados em AVAs é um dos desafios emergentes, dos quais surgiram vários trabalhos relacionados com a integração dos ROAs em AVAs, em que foram propostos e desenvolvidos modelos (métodos) de integração dos mesmos. Assim, o Capítulo 3 vem responder o primeiro objetivo específico (E1) que é estudar as diferentes formas de integração de AVAs e ROAs. São apresentados modelos de integração que mostraram-se eficazes e por fim uma tabela (Tabela 3.1) comparativa dos mesmos.

#### 3.1 Integração de repositório e AVA

Leal e Queirós (2010) trazem um estudo sobre a integração do repositório de *CrimsonHex* e o Moodle 2.0. A ideia central do trabalho é testar a interoperabilidade do repositório. O *CrimsonHex* é um repositório educacional que permite armazenar OAs referente a problemas de programação para uso pedagógico no ensino secundário e superior.

De acordo com Leal e Queirós (2009) os usuários do repositório devem conhecer a sua arquitetura, nomeadamente, os seus componentes e as suas funções, baseado na especificação IMS *Digital Repositories Interoperability*. Para além da especificação IMS (2002) o *CrimsonHex* suporta ainda o padrão de metadados DC.

Para integrar o *CrimsonHex* ao Moodle Leal e Queirós (2010) fazem referência a seguintes *interfaces*: *Repository API* (busca e recuperação de objetos nos repositórios externos) e *Portfolio API* (exportação de objetos do Moodle para repositórios externos).

*Repository API*- esta *interface* é composta por duas partes: *Administrators* – usada pelos administradores do Moodle para configurar repositórios e *File picker* usada pelos professores para interagir como os repositórios. No entanto, para a criação da integração usando a *interface* do repositório:

- Criou-se uma pasta para o plugin (*moodle/repositorio/crimsonHex*);
- Colocou-se na pasta do *plugin repository.class.php* (sub-classificação da classe padrão do API padrão e substituindo seus métodos) e *icon.png* - ícone exibido no *file picker*;



- Adicionou-se o arquivo de idiomas *repository\_crimsonHex.php* a pasta *moodle/repositório/lang/en\_utf8/*.

O *file repository.class.php* é responsável pela comunicação entre Moodle e todos tipos de ROAs, neste caso, serviu para comunicação com *crimsonHex*, mas existem outros que usam como Merlot, YouTube, Flickr, DSpace, dentre outros.

*Portfolio API - plugin* que permite que o autor de conteúdo educacional faça *upload crimsonHex* e criar um novo conteúdo usando metadados requeridos. Para tal:

- Criou-se a pasta para o *plugin (moodle/portfolio/type/crimsonHex)*;
- Colocou-se na pasta do *plugin lib.php* (sub-classificação da classe padrão do API padrão e substituindo seus métodos);
- Adicionou-se o arquivo de idiomas *repository\_crimsonHex.php* na pasta *moodle/portfolio/type/crimsonHex/lang/en\_utf8*.

O arquivo *lib.php* é responsável por garantir a comunicação entre o Moodle e repositórios externos do mesmo tipo que *crimsonHex*, dos quais destacam-se o Mahara, GoogleDocs e o Box.net.

### 3.2 MrCute

Acrônimo de *Moodle Repository Create Upload Tag Embed* é um *plugin* que permite integrar mecanismos de pesquisa de objetos bem como de catalogação de OAs no Moodle, desenvolvido por JISC<sup>21</sup> projeto da *Worcester College of Technology and Learning Objectivity*.

De acordo com Brady (2009) o MrCute permite que os objetos de aprendizagem carregados para qualquer curso do AVA possam ser compartilhados entre diversos usuários e usado em diversos ambientes (cursos). Segundo Tarouco et al (2014) o módulo MrCute “amplia as funções do Moodle ao permitir que conteúdos educacionais sejam armazenados na estrutura interna do AVA e, exatamente, à estrutura de um curso específico, possibilitando que outros cursos instalados no mesmo servidor tenham acesso a tal conteúdo. Outra vantagem do módulo é a possibilidade de agregar conteúdo de um repositório externo, no caso, o repositório JORUM<sup>22</sup>.

O MrCute permite incorporar funcionalidades de um CMS ao Moodle, na qual é possível realizar recuperação de conteúdo como também pesquisa local usando palavras-chave,

---

<sup>21</sup> Joint Information Systems Committee (<http://www.jisc.ac.uk/>)

<sup>22</sup> <http://www.jorum.ac.uk/>

descrição, título, autor e categoria. No que se refere ao funcionamento do módulo MrCute após a sua instalação, que neste caso específico, faz-se referência a versão 2 do MrCute também designado por MrCute 2, e é observado duas opções:

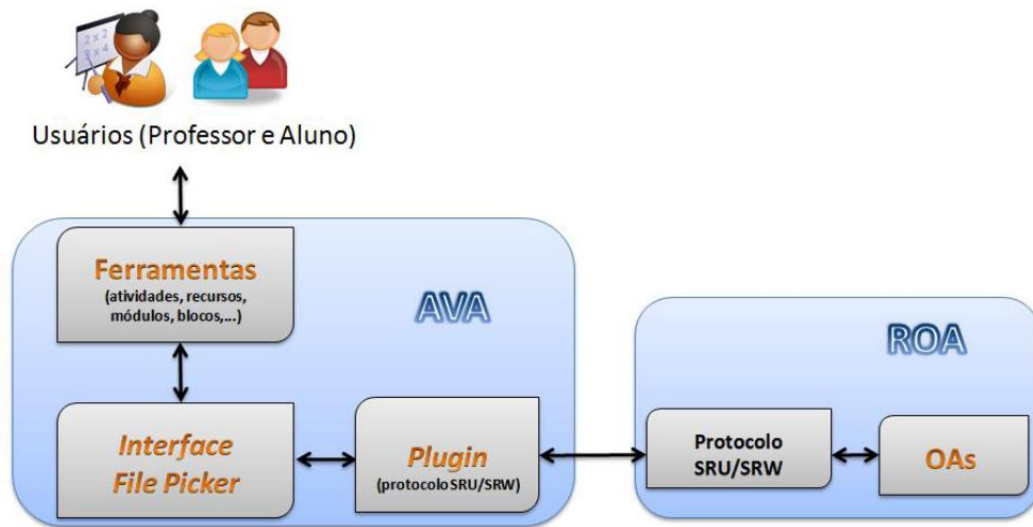
- *Find ready made materials* – permite ao usuário fazer o uso dos objetos já existentes no repositório local, NLN (*National Learning Network*) bem como no repositório externo JORUM. Quando o usuário pesquisa um determinado objeto, preenche os campos e informa as opções de pesquisa, e como resultados da pesquisa poderá ter, tanto dados encontrados no repositório local (Moodle) como do repositório JORUM.
- *Upload learning materials* – permite ao usuário fazer o *upload* das matérias para o repositório local. Existem duas possibilidades de se carregar os OAs: *Upload a file or create a package* (possibilita a criação de pacote no formato IMS, o usuário seleciona os arquivos do objeto e são empacotados em um *zip* gerando um XML com informações do objeto) e o *Upload existing IMS zip package* (o usuário seleciona um arquivo do seu computador local já confeccionado no padrão IMS e zipado e atribui o título e a descrição nos campos).

No entanto, não foi possível, até o momento, encontrar trabalhos recentes que dão ênfase à evolução do *MrCute 2*, doutro modo o estudo viabilizou outros trabalhos de integração de ROA a AVA como a pesquisa de Rodrigues (2013).

### **3.3 Integração de Ambientes Virtual de Aprendizagem com Repositório Digital**

Rodrigues (2013), propõe um modelo de integração de um AVA com um ROA para fins de busca e recuperação de OA, através do padrão SRU/W (*Search/Retrieve via URL/Web service*) que permite a execução de consultas complexas, fazendo com que o cliente possa solicitar ao servidor apenas os metadados dos objetos de aprendizagem que apresentem determinadas características. A Figura 3.1 apresenta o modelo de integração proposto na pesquisa.

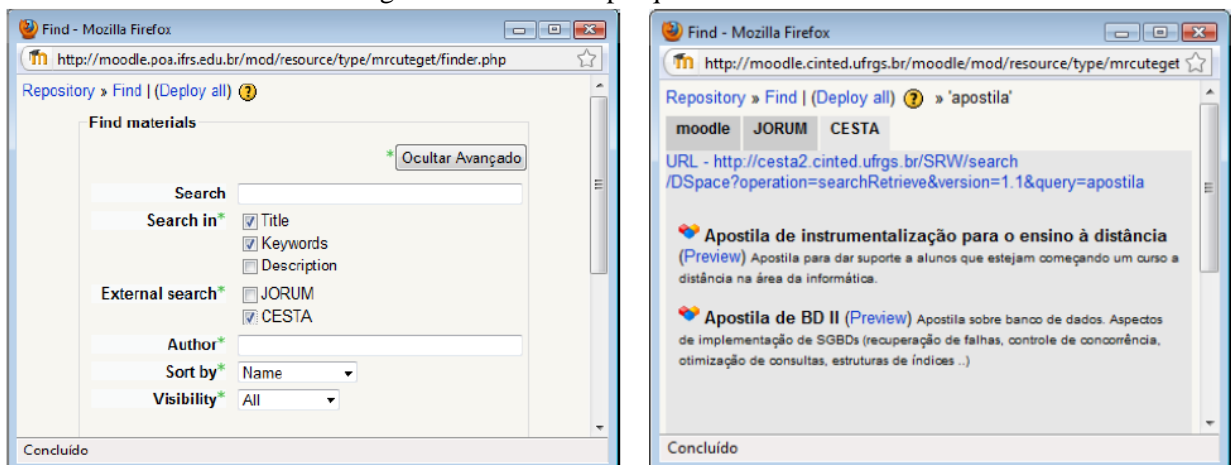
Figura 3.1 - Modelo de integração Proposto



Fonte: Rodrigues (2013, p. 77)

O modelo proposto para a integração do AVA e do Moodle foi feito por meio de uma *API Repository* padrão. Recorreu-se a versão 2.0 do Moodle e o repositório CESTA <sup>23</sup> (Coletânea de Entidades de Suporte ao uso de tecnologias na Aprendizagem), na qual usou-se como protocolo de comunicação o *SRU/W*. Para que as consultas fossem realizadas a partir do ambiente Moodle modificou-se o código fonte do *MrCute* (já visto anteriormente). Como pode ser visto na Figura 3.2.

Figura 3.2 - Telas de pesquisa e resultados



Fonte: Tarouco et al (2013)

<sup>23</sup> <http://cesta.cinted.ufrgs.br>

A Figura 3.2 mostra duas telas que fazem referência ao processo de busca de recursos educacionais com o auxílio do plugin MrCute já modificado e adaptado ao repositório CESTA, na primeira tela o usuário inicia a consulta informando o que pretende consultar, onde poderá encontrar tal termo (no título, nas palavras-chave ou na descrição), informa também se a consulta pode ser feita fora do ambiente Moodle (nos repositórios JORUM ou CESTA), o nome do autor e também pode fazer uma ordenação pelo nome por exemplo, isso caso haja necessidade. A segunda tela apresenta o resultado da busca como termo de pesquisa que, no exemplo, foi a palavra ‘apostila’. A informação fornecida ao usuário é proveniente do repositório Cesta, e nos mostra o título e uma pequena descrição do OA pesquisado.

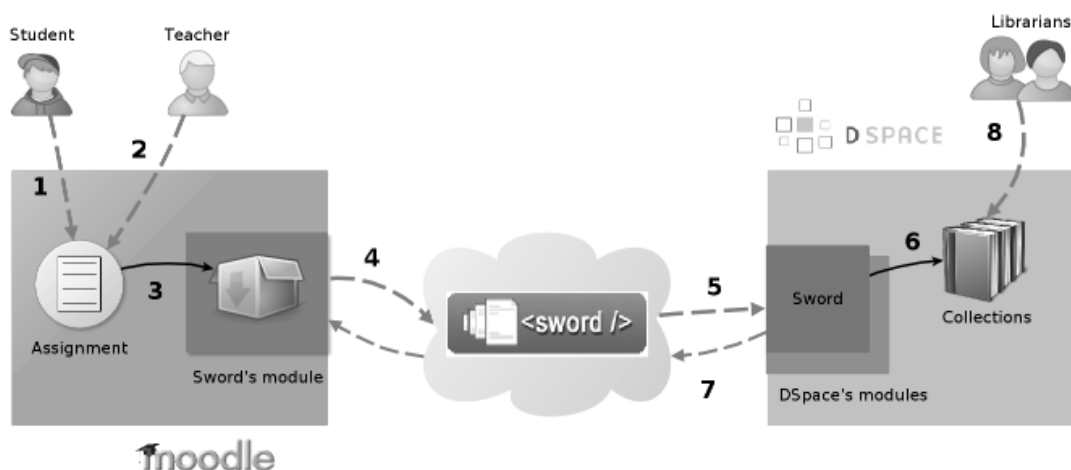
Como conclusão e prova de conceito os autores ressaltaram que a realização da integração apoiou os professores na construção de um plano de ensino e aprendizagem na medida em que permitiu e facilitou a pesquisa e a seleção de materiais educacionais digitais adequados à disciplina, como também ao acesso e reutilização das matérias.

### **3.4 Integração de LMS com repositório digital: Caso de estudo**

Díaz et al (2013), trazem um modelo de integração do Moodle e repositório DSpace que consiste em estabelecer a conexão entre os mesmos, pois garantem que ambos (Moodle e DSpace) possuem suas próprias tecnologias de acordo com suas necessidades e que a integração não é imediatamente direta.

Inicialmente estabeleceu-se o canal de comunicação com o auxílio do protocolo SWORD que foi incorporada na interface do módulo *Assignment* no Moodle e DSpace que a partir da sua versão 1.8 já possui SWORD *server*. A Figura 3.3 apresenta o esquema de comunicação proposta pelos autores.

Figura 3.3 - Esquema de comunicação proposta por Díaz et al (2013)



Fonte: Díaz et al (2013, p. 213)

- 1 – O aluno realiza o envio de uma determinada tarefa.
- 2 – O professor após avaliação, seleciona a tarefa e envia os mesmos ao repositório.
- 3 – O módulo pega cada uma das tarefas e empacota com seus metadados correspondentes.
- 4 – O pacote é enviado pelo SWORD para a coleção especificada pelo professor
- 5 – O módulo SWORD do DSpace desempacota e recupera os metadados DC e arquivos associados.
- 6 – É criado um novo recurso no não coleção especificada do repositório.
- 7 – O módulo SWORD do DSpace envia uma resposta XML para o Moodle informando o *status* da operação.
- 8 – Os usuários catalogam objetos.

Os autores afirmam que, a implementação da comunicação entre o ambiente Moodle e o repositório DSpace permite reduzir a sobrecarga no armazenamento de mesmos recursos nas duas plataformas, bem como incentivar os professores que gerem seus cursos publiquem mais projetos desenvolvidos pelos seus alunos em repositórios externos sem que estejam familiarizados com a interface de catalogação do repositório DSpace.

### 3.5 Depósito de objetos de aprendizagem em repositórios a partir da integração com ambientes virtuais de aprendizagem

Tarouco et al (2013) apresentam o *plugin SWORD\_upload*<sup>24</sup> que serve para fazer o *upload* de OAs no repositório DSpace e indexação dos mesmo no Moodle. Para a comunicação entre os dois sistemas usou-se como protocolo de comunicação o SWORD, visto que facilita o processo de depósito de OAs e pelo fato de preencher os principais requisitos do ambiente de integração Moodle e repositório.

Assim, para que seja feito o depósito do OA, o professor inicialmente autentica-se no repositório no qual possui autorização de publicação a partir do AVA (usando o API), em seguida descreve o objeto preenchendo os campos correspondentes aos metadados, e por fim escolhe o arquivo ou o *link* que corresponde ao objeto de aprendizagem a ser publicado. Ao passar essas informações para o plugin, o objeto é criado no repositório e também é adicionado ao ambiente virtual. A adição ao Ambiente Virtual de Aprendizagem pode ocorrer através de uma referência ao repositório ou através da duplicação do arquivo do próprio AVA, (TAROUCO et al, 2013). Os autores consideram ainda que como se desenvolveu um módulo para o Moodle, é possível desenvolver módulos semelhantes para outros Ambientes Virtuais de Aprendizagem.

Fez-se um quadro resumo dos trabalhos relacionado, no qual apresenta-se os sistemas usados para integração (sejam eles AVAs ou ROAs), o mecanismo de comunicação entre ambos para a interação, e a finalidade da integração dos sistemas que se classificou com depósito ou busca de OAs, tal resumo pode ser visto na Tabela 3.1 que se segue.

Tabela 3.1 - Análise comparativa dos trabalhos relacionados

<i>Modelo de Integração</i>	<i>Sistemas</i>	<i>Mecanismo de comunicação</i>	<i>Depósito de OAs</i>	<i>Busca de OAs</i>
MrCute (2009)	Moodle e JORUM	-	Sim. O depósito pode ser feito já com o pacote IMS criado ou pode-se criar o pacote na altura do <i>upload</i> .	Sim. A busca de objetos de aprendizagem é feita
Leal e Queirós (2010)	Moodle e <i>CrimsonHex</i>	<i>Repository API</i>	Sim	Sim
Rodrigues (2013)	Moodle e DSpace	SRU/W	Não	Sim

<sup>24</sup> [https://github.com/marceloschmitt/SWORD\\_upload](https://github.com/marceloschmitt/SWORD_upload) - disponível para *download*

Díaz et al (2013)	Moodle e DSpace	SWORD	Sim	Não
Tarouco et al (2013)	Moodle e DSpace	SWORD	Sim. Depósito a partir do Moodle e por sua vez o <i>link</i> do OA é visível no ambiente virtual.	Não

Fonte: O autor (2015)

De acordo com a Tabela 3.1 acima, pode-se observar as características que se pretendem adequar à ferramenta de catalogação de objetos de aprendizagem no repositório Cognix. É visto também, que algumas iniciativas mostram algo em comum que é o uso do Moodle. Dentre os quatro trabalhos analisados escolheu-se a iniciativa de Díaz et al (2013) e de Tarouco et al (2013), pelo fato destas apresentarem uma estratégia de integração dos *softwares* interessante, o uso do protocolo de comunicação SWORD contribui de forma a garantir uma facilidade na comunicação entre cliente (o ambiente Moodle) e servidor (repositório DSpace) garantindo assim a interoperabilidade entre os sistemas, para além do que se deseja com a ferramenta cujo propósito é a catalogação dos OAs em um repositório.

## **4 REPOSITÓRIO COGNIX**

Repositório digital é mais do que um portal ou uma forma de acesso ao conteúdo digital. O que torna um repositório digital muito mais do que um portal é a capacidade de descobrir um objeto de aprendizagem e reutilizá-lo. O propósito de um repositório digital não é simplesmente de armazenamento seguro e distribuição, mas o compartilhamento e reutilização do conteúdo digital. Esse Capítulo tem como finalidade apresentar de forma clara e concisa, aspectos genéricos em torno do repositório Cognix.

### **4.1 Contextualização**

Com o surgimento da Universidade Aberta do Brasil vem crescendo o incentivo à criação e ao compartilhamento de conteúdo educacionais nacionais, sejam eles vídeos, textos, mapas, infográficos, etc. Com toda a heterogeneidade de materiais, aliadas à grande quantidade de dispositivos existentes torna-se necessário o armazenamento e a disseminação deste conhecimento. Certamente, esta necessidade é universal e cada dia demanda por novas soluções devido, principalmente, ao crescimento da oferta de conteúdo educacional na WEB.

No sentido de se catalogar, armazenar e distribuir tais conteúdos educacionais surgem os repositórios educacionais, que objetivam o armazenamento, a preservação e a disseminação da produção intelectual (Costa e Leite, 2006), abrangem um conjunto de serviços que incluem: a gestão, compartilhamento, preservação, organização, reutilização e acesso de seus conteúdos (Lynch 2003 e Duncan 2003).

O repositório mais utilizado hoje é o DSpace que também mantém os direitos do Fedora, já discutidos no Capítulo 2. Porém o DSpace, apesar de robusto, é extremamente complicado de se instalar, configurar, customizar e manter. Como pretende atingir um público muito variado, apresenta decisões de projeto muito genéricas, desde o conjunto de metadados até a variedade de interfaces, necessitando grandes customizações para satisfazer o público da área educacional. Nesse contexto, muitas instituições (principalmente no Brasil) não dispõem, nem de recursos humanos tampouco de infraestrutura para tal. Um exemplo é a Diretoria de Ensino Básico do MEC, onde existem milhares de instituições filiadas, criando OAs, porém sem a capacidade de implementar um repositório local que armazena e torne disponível ao compartilhamento tais recursos.

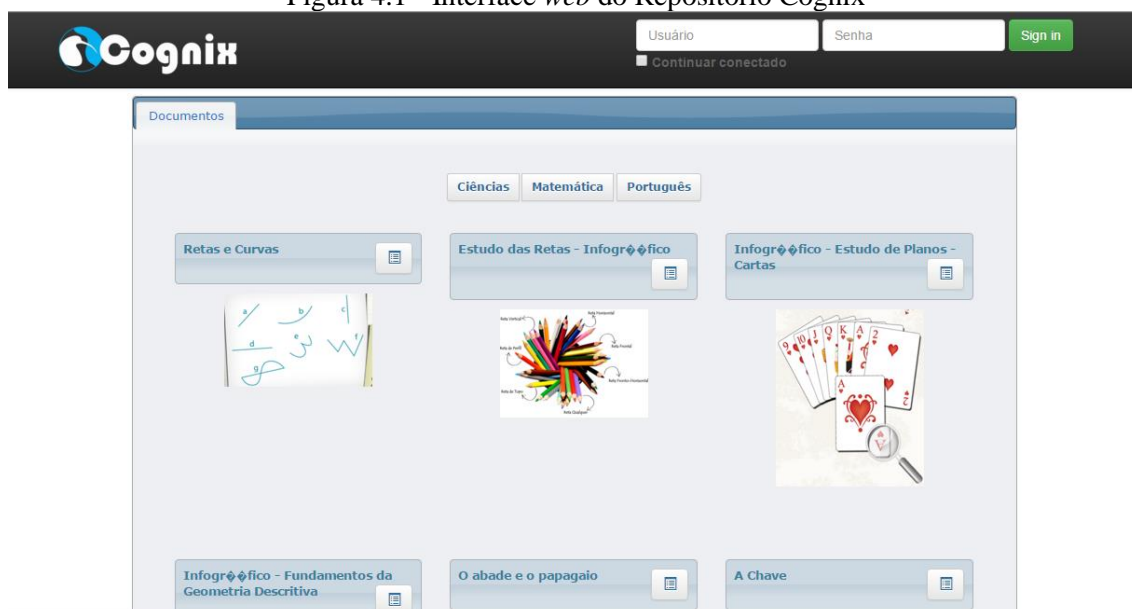
Além disso, o trabalho exigido do autor, que no geral é um professor, para se cadastrar um OA é extremamente grande, e por vezes, repetitivo. Muitos dos dados que necessitam ser



catalogados, no repositório, poderiam ser inferidos do próprio arquivo (OA), como, por exemplo, seu tamanho ou formato. O Cognix é um novo repositório, focado em propiciar uma satisfação de uso para o usuário, seja ele um administrador, um professor ou um consumidor dos recursos educacionais. Diante desta contextualização são citadas as seguintes características

- Sistema robusto e de fácil instalação e de pequena complexidade em relação aos outros softwares;
- Portabilidade para diversas plataformas (sistemas operacionais);
- Possui controle de versionamentos no padrão Git, uma ferramenta utilizada no desenvolvimento distribuído de controle de versões, na qual é possível criar *forks* (bifurcações) de um código fonte, modificá-lo e criar então: ou uma nova versão diferente em um novo ciclo de vida, ou fazer uma requisição de atualização (*pull-request*). Muito desse conceito é aplicável ao ciclo de vida do OA, que tem por característica principal a sua reusabilidade. A inclusão desse tipo de controle de versionamento, no repositório Cognix, auxilia muito também no controle do versionamento dos OAs.
- Interface (pode ser vista na Figura 4.1) para a utilização dos objetos direta e com portabilidade para dispositivos móveis, sejam eles *tablets* ou celulares.

Figura 4.1 - Interface *web* do Repositório Cognix



Fonte: O autor (2016)

A Figura 4.1 mostra como os OAs são apresentados ao usuário através da interface de acesso ao Cognix. Ou seja, diferentemente de outros repositórios onde o usuário tem acesso

apenas ao link para o conteúdo, no Cognix os OAs são apresentados contendo uma ‘capa’ ilustrada e o conteúdo de cada OA também é apresentado no formato de um livro contendo título, título das divisões, se possui ou não atividades interativas, etc. Além disso são apresentadas opções de navegação no conteúdo, opções de abrir ou baixar um determinado OA, bem como opções de acessibilidade (aumentar o tamanho das letras e figuras e escutar o conteúdo).

Por outro lado, a interface para o consumidor dos conteúdos do repositório, também precisa ser facilitada. A interface do DSpace é complicada para os usuários desse tipo de conteúdo (professores e alunos, dentre outros). O Cognix reduziu o número de cliques que o usuário precisa realizar, para que ele seja capaz de chegar ao OA desejado, da maneira mais fácil possível. A interface atual, do DSpace, chega a exibir os metadados para o usuário, coisa que normalmente não é o que se deseja.

Outra limitação existente hoje é a restrição a padrões de metadados sem hierarquia, o problema não reside simplesmente nesse fato, mas sim, que os padrões para objetos educacionais mais avançados hoje existentes, usam este recurso (como o IMS , o OBAA e o LOM), e como os repositórios não acompanham estas evoluções nos padrões de metadados, isso impossibilita catalogações que podem facilitar muito para, por exemplo, portadores de necessidades especiais, ou ainda para realizar buscas de conteúdo de vídeo.

## **4.2 Arquitetura do Repositório Cognix**

A arquitetura do repositório Cognix é apresentada na Figura 4.2. Trata-se de uma arquitetura composta por três componentes: a) A camada de apresentação que é responsável pela exibição da informação. Possui componentes para a comunicação das outras camadas como de outras aplicações, a qual é constituída basicamente, pela interface do sistema, o protocolo OAI-PMH e protocolo SWORD, que é contribuição deste trabalho. b) A camada de negócio é responsável por implementar a lógica de negócio da aplicação, controle do sistema, e administração da informação. Nesta camada é feito o gerenciamento de usuários, objetos e metadados, permissões e controle de acesso dos usuários, fluxo de depósito e demais ferramentas de gerenciamento. c) A camada de dados responsável pela persistência, acesso e armazenamento físico dos dados do repositório e é composto basicamente pelo sistema de gerenciamento de banco de dados PostgreSQL

Figura 4.2 - Arquitetura do Repositório Cognix



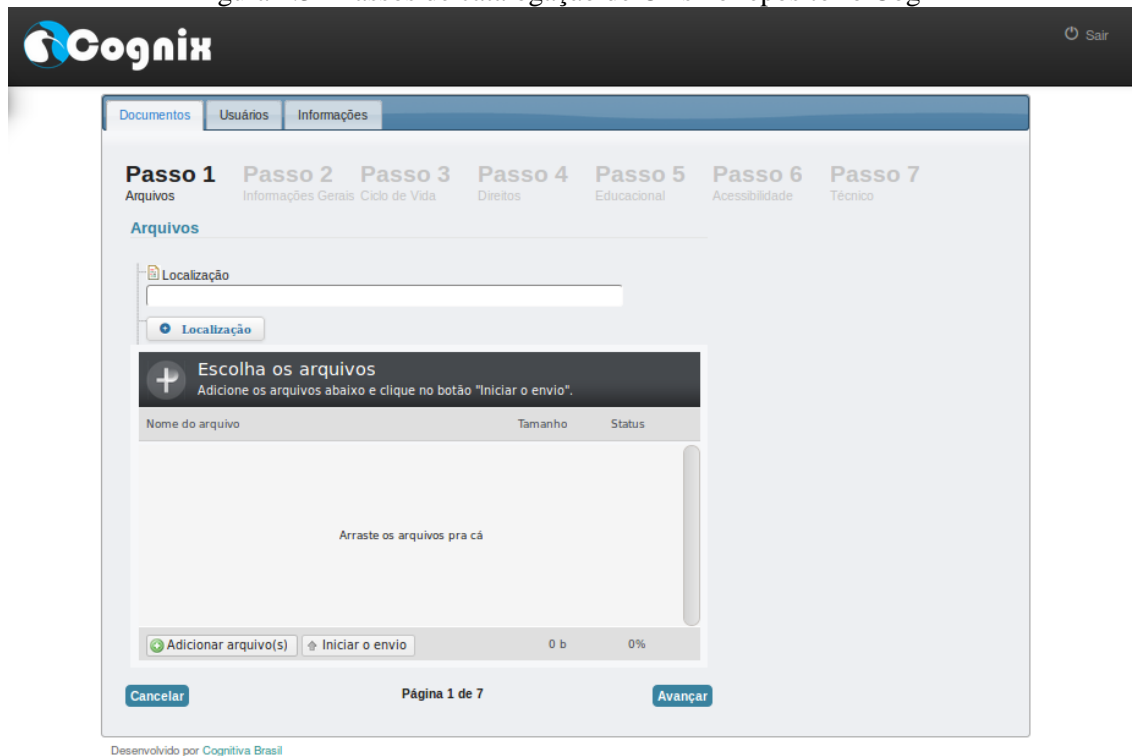
Fonte: O autor (2016)

### 4.3 Processo de catalogação

Um dos pontos de maior resistência na adesão aos repositórios existentes, por parte da comunidade que produz conteúdo educacional, está no difícil trabalho para a catalogação dos conteúdos. O repositório Cognix propõe uma mudança na forma de como o processo de catalogação é feito nos repositórios existentes atualmente. Inicialmente, o usuário faz o upload do arquivo a ser catalogado, e com isso, boa parte dos metadados são inferidos automaticamente pelo serviço de catalogação do repositório (Figura 4.3). Outro fator que facilita muito a catalogação é a criação de perfis de objetos, por exemplo, a meta é catalogar 50 planos de aula, de matemática, para o ensino fundamental, o público alvo para esse conteúdo são os professores, o formato dos OAs é texto, o tema é ensino de matemática, etc... Estas informações permitem a criação de um perfil comum a todos 50 OAs e, com isso, restará apenas um pequeno

conjunto de metadados para o usuário preencher, que não puderam ser inferidos e nem pertençam ao perfil previamente delineado.

Figura 4.3 - Passos de catalogação de OAs no repositório Cognix



Fonte: O autor (2016)

A Figura 4.3 mostra o processo de catalogação dos OAs, que é composta por 7 passos, o primeiro é o carregamento do OA, na qual são inferidos as seguintes informações: título (o nome do arquivo carregado), o idioma, o identificador, o tamanho, a extensão, dentre outros elementos. Os restantes seis (6) passos fazem referência aos elementos das categorias do padrão OBAA, que o usuário informa de acordo com o tipo de OA.

Este capítulo teve como finalidade fazer uma abordagem em torno do repositório Cognix, descrevendo suas principais características e o processo de catalogação de OA.

Com a integração entre o AVA MOODLE e o ROA Cognix busca-se permitir que o módulo do Cognix responsável pela catalogação de OAs possa inferir um maior número de metadados, que estão presentes no conteúdo educacional proveniente do MOODLE.

## 5 FERRAMENTA PARA FACILITAR A DISSEMINAÇÃO DE CONTEÚDO EDUCACIONAL

A integração de sistemas sugere uma interoperatividade da qual é acrescida a eficiência na busca, localização e uso/reuso dos OAs de modo a facilitar a disseminação dos recursos. Este capítulo vem responder os objetivos específicos E1 e E2, conseqüentemente apresenta aspectos inerentes ao desenvolvimento da ferramenta de integração do ROAs e AVA Moodle tais como a arquitetura proposta e codificação, bem como a implementação e teste e validação dos resultados. Como, a parte de prática desta pesquisa, trata fundamentalmente da implementação e instalação do protocolo de comunicação SWORD, tanto no Moodle como no repositório Cognix. Este capítulo vai apresentar rotinas e linhas de código extraídas do código completo do Moodle e do Cognix. Consideramos este procedimento importante, pois trata-se de uma tarefa difícil de ser executada. Sendo assim pretendemos contribuir com futuros pesquisadores que venham a enfrentar este mesmo tipo de problema, em outras situações de integração de *software*.

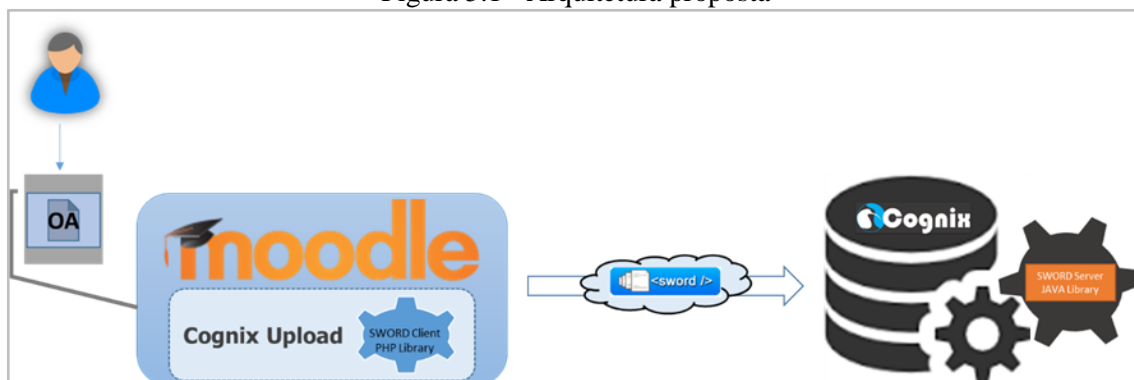
### 5.1 Arquitetura da Ferramenta

O estudo propõe uma arquitetura genérica e suas entidades composta por dois sistemas, o AVA Moodle que é uma aplicação *web* que funciona como serviço cliente e o ROA Cognix, como o provedor de serviço, ou seja, o nosso servidor. A Figura 5.1 apresenta o cenário geral da arquitetura. De salientar que em ambos sistemas foram incorporados dois módulos baseados no protocolo SWORD. No Moodle o módulo que conta com serviços cliente composto pela biblioteca cliente do SWORD na linguagem de programação PHP e o módulo recursos do próprio Moodle que foi modificado e adequado segundo às necessidades e características do repositório Cognix. No repositório Cognix implementou-se SWORD server na linguagem Java inicialmente modificada. É de referencial que ambas bibliotecas do protocolo SWORD encontram-se disponíveis<sup>25</sup> livremente para a sua reutilização e implementação.

---

<sup>25</sup> <https://github.com/swordapp>

Figura 5.1 - Arquitetura proposta



Fonte: O autor (2016)

A Figura 5.1 apresenta o cenário geral da arquitetura que é caracterizada pela tecnologia cliente-servidor. É apresentada também, um elemento importante para a comunicação entre os dois sistemas que é o protocolo de comunicação SWORD na versão v2 que propõe uma interface interoperável fazendo com que o depósito em repositórios seja facilitado. Embora o Moodle já possuía APIs<sup>26</sup> para comunicação com repositórios digitais, neste caso, optou-se pelo uso do SWORD e foi necessário acrescentar funcionalidades que serão apresentadas a seguir.

## 5.2 Desenvolvimento

Conforme o diagrama arquitetural (representado pela Figura 5.1), seguiu-se ao desenvolvimento da ferramenta de catalogação de OAs com a implementação do protocolo SWORD que possibilita o depósito remoto de conteúdos e seus respectivos metadados em repositório sem que haja a necessidade do uso da *interface* de catalogação do próprio repositório.

Durante o desenvolvimento da ferramenta, estabeleceu um roteiro metodológico caracterizado por duas seções:

- I. Implementação do protocolo SWORD no Moodle
- II. Implementação do protocolo SWORD no ROA Cognix

### 5.2.1 Implementação do protocolo SWORD no Moodle

Após análise da versão v2.9 do Moodle, utilizada para o desenvolvimento se escolheu o módulo recurso que já vem por padrão instalado, cuja finalidade é inserir recursos

<sup>26</sup> [https://docs.moodle.org/dev/Web\\_service\\_API\\_functions](https://docs.moodle.org/dev/Web_service_API_functions)

educacionais que serão usados num determinado curso. A partir deste módulo é publicado no ambiente um número considerável de OAs e o mesmo suporta vários tipos de arquivos como HTML, Flash, imagens, PDF, Word, vídeos, dentre outros, permite também publicar OAs usando pacotes SCROM, IMS *Content Packaging* e outros.

Realizou-se também alterações/adequações de métodos e acréscimo da biblioteca SWORD cliente versão 2 na linguagem PHP, isto pelo fato de sua compatibilidade com o Moodle. Abaixo segue-se a Tabela 5.1 que mostra a estrutura de diretórios do *plugin*.

Tabela 5.1 - Estrutura de diretórios do *plugin*

<i>Diretório / Arquivo</i>	<i>Descrição</i>
db/	Diretório responsável pelos pacotes de banco de dados
lang/	Diretório responsável pelos pacotes de idiomas.
sword2/	Diretório que contém as classes do protocolo SWORD cliente, que foram adicionadas ao módulo. De salientar que, usou-se as classes SWORDAPPClient que serve para estabelecer a comunicação com o servidor, e PackagerMetsSwap permite criar um arquivo XML com os metadados do OA inclusive o pacote a ser enviado através do protocolo SWORD.
index.php	Lista todas as instâncias existentes do <i>plugin</i>
mod_form.php	Formulário que contém elementos necessários para adicionar ou atualizar instâncias
lib.php	Arquivo responsável pela lógica de processamento, responsável pela ligação entre o plugin e core do sistema, pelo menos três funções devem ser implementadas (exemplo: <code>cognix_add_instance()</code> , <code>cognix_update_instance()</code> , e <code>cognix_delete_instance()</code> )
locallib.php	Responsável pelas classes e funções internas do plugin, inclusive a classe <code>cognix_sword</code> que possui funções para o envio de recursos do Moodle ao servidor
setting.php	Configuração global do plugin
version.php	Arquivo responsável pelo versionamento do plugin
view.php	Responsável pela visualização de uma determinada instância.

Fonte: O autor (2016)

#### Funções do lib.php

- `cognix_add_instance($data, $mform)` - adicionar uma nova instância com os dados vindo do `mod_form.php`, adicionou-se duas linhas de código que chamam as funções de criação (`sword_create_package`) e envio (`sword_deposit`) do pacote contendo os metadados e o objeto carregado.

```

94 | // create package and send it to repository
95 | $p = sword_create_package($data, $mform);
96 | sword_deposit($p);

```

- `cognix_update_instance($data, $mform)` - atualiza uma instância no banco de dados após o recebimento de dados vindos do `mod_form.php` como um objeto.
- `cognix_delete_instance($data, $mform)` - recebe o id da instância do Cognix de forma a usar para exclusão dos registros relacionados no banco de dados.

No `localib.php` adicionou-se as funções `sword_metadata($p, $data)`, `sword_mets($p)` responsável pela criação de um arquivo XML contendo os metadados do OA e a `sword_create_package($data, $mform)` cria um pacote composto pelo arquivo XML e o OA, que será enviado para o repositório. Estas funções foram criadas neste trabalho com o objetivo de integração entre Moodle e o Cognix.

```

485 | function sword_mets($data){
486 |     $p = new PackagerMetsSwap($data['rootin'], $data['dirin'], $data['rootout'], $data['fileout']);
487 |     sword_metadata($p, $data);
488 |     $p->create();
489 | }
490 |
491 | function sword_metadata($p, $data){
492 |     $p->setTitle($data['title']);
493 |     $p->addCreator($data['author']);
494 |     $p->setAbstract($data['abstract']);
495 |     $p->setLanguage($data['language']);
496 |     $p->addRights($data['license']);
497 |
498 |     foreach ($data['subject'] as $subject) {
499 |         $p->addSubject($subject);
500 |     }
501 |     // $p->addFile($data['filename'], $data['mimetype']);
502 |     //foreach($data['files'] as $file){
503 |
504 |         $p->addFile($data['filename'], $data['mimetype']);
505 |     //}
506 | }

```

O trecho do código a seguir é referente a função `sword_deposit($p)` que é responsável pelo envio do pacote criado contendo o XML e o conteúdo do OA. Criou-se uma instância da classe `SWORDAPPClient` para estabelecer a comunicação e depósito do OA no repositório, onde inicialmente é verificada a autenticação e a autorização do usuário de modo que seja feito o depósito.



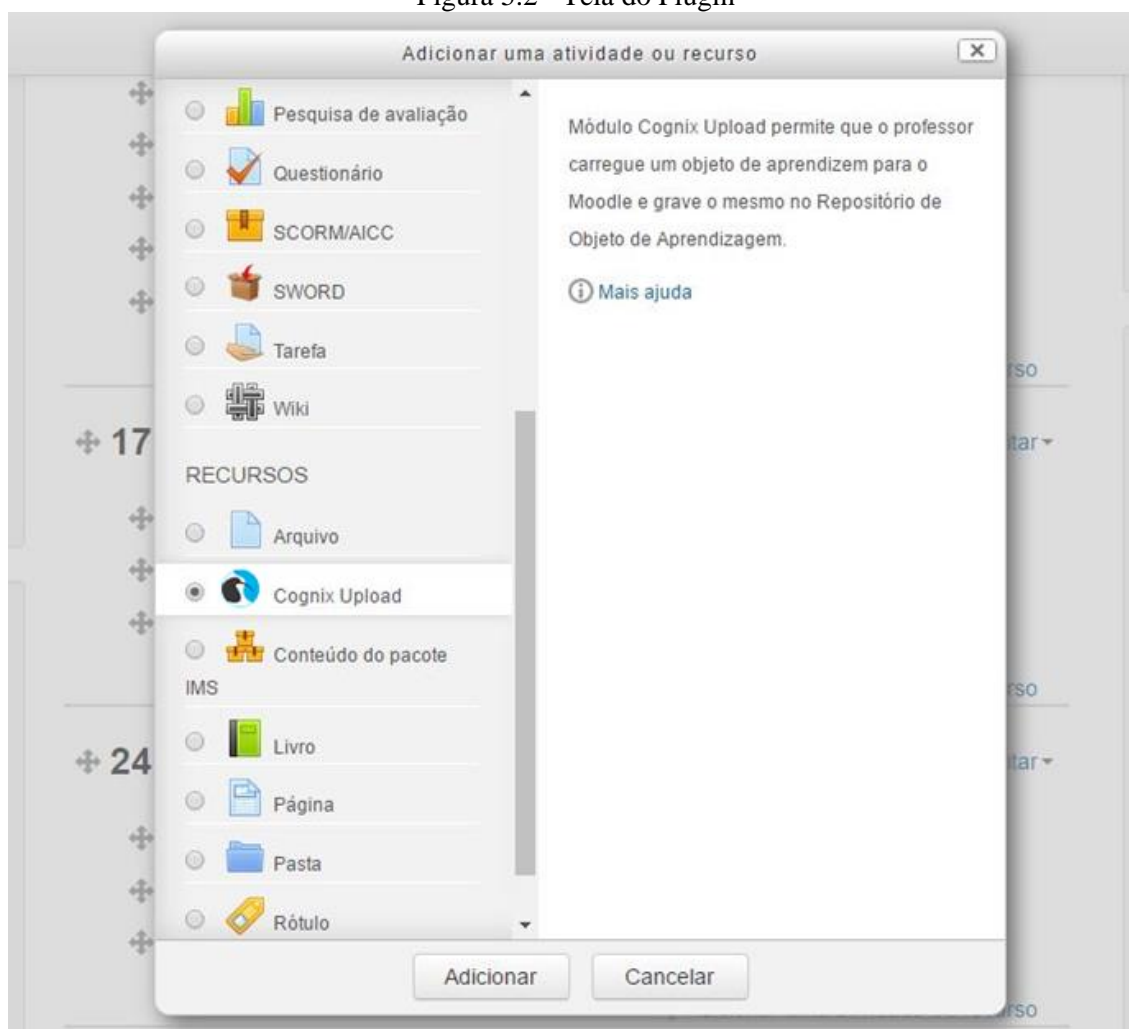
```

524     $client = new SWORDAPPClient();
525     $response = $client->deposit($colurl, $sworduser, $swordpw, '', $p,
526                             $CFG->dirroot . '/mod/cognix/temp/mets_swap_package.zip',
527                             $pf, 'application/zip');
528     if(($response->sac_status >= 200) || ($response->sac_status < 300)) {
529         copy($CFG->dirroot . '/mod/cognix/temp/files/mets.xml', $CFG->dirroot . '/mod/cognix/temp/files/mets-bak.xml');
530         @unlink($CFG->dirroot . '/mod/cognix/temp/files/mets.xml');
531         @unlink($CFG->dirroot . '/mod/cognix/temp/files/.'.$filename);
532         @unlink($CFG->dirroot . '/mod/cognix/temp/mets_swap_package.zip');
533         return array($response->sac_links[0]);
534     } else {
535         throw new moodle_exception('upload_error','cognix');
536     }
537 }

```

A Figura 5.2, mostra a tela (visão do usuário) para o início da submissão de OA a partir do módulo Cognix Upload, onde o professor ao carregar um OA no ambiente Moodle e armazena também no repositório Cognix, já com uma boa parte da descrição dos objetos.

Figura 5.2 - Tela do Plugin



Fonte: O autor (2016)

### 5.2.2 Implementação do protocolo SWORD no ROA Cognix

O repositório Cognix diferentemente do DSpace não possui por padrão os serviços do protocolo SWORD server, de modo a facilitar a comunicação com o Moodle, teve-se de criar/implementar tais serviços a partir da biblioteca na linguagem Java<sup>27</sup>, que corresponde a versão v2 do SWORD com funções CRUD (*Create, Retrieve or Read, Update e Delete*). Esta é uma das contribuições deste trabalho de mestrado.

Na versão v2.2 do repositório Cognix adicionou-se a biblioteca Java do SWORD, visto que o repositório Cognix foi desenvolvido utilizando a Java. Assim, implementou-se as seguintes classes conforme a descrição abaixo:

- ✓ *CollectionDepositManager* – possui métodos necessários para a criar um novo recurso no servidor, podendo ser criado de três formas: *Binary File Deposit*, *Multipart Deposit* e *Atom Entry*.

*Binary File Deposit*: cria-se um recurso postando o conteúdo binário como o corpo de uma solicitação HTTP para IRI da Coleção, com os cabeçalhos *Content-Type*, *Content-Disposition* e *Packaging*.

*Multipart Deposit*: de modo a garantir que todos os clientes e servidores SWORD possam trocar uma gama completa de conteúdo e metadados de arquivos, a utilização do *atom multipart* permite combinar um pacote (possivelmente um simples *zip*) com um conjunto de termos de metadados DC incorporado em uma entrada ATOM, não sendo necessário que servidor SWORD suporte formatos de empacotamento, mas este perfil recomenda que o servidor possa aceitar um arquivo *zip*, com a *media part* de uma requisição (*request*) *atom multipart*. O cliente cria um recurso postando uma mensagem mime *multipart* na IRI da coleção com duas partes: Uma entrada ATOM contendo os metadados para o depósito (conhecida como a *entry part*) e o conteúdo binário (conhecida como a *media part*). Um exemplo de depósito de um pacote *zip* no DSpace (example.zip contendo um arquivo METS com os metadados e 3 PDF). Neste exemplo, o arquivo example.zip é depositando em uma coleção com o identificador teste/2, como podemos verificar no código apresentado a seguir.

```
curl -i http://127.0.1.1:8088/swordv2/collection/teste/2 --data-binary
"@example.zip" -H "Content-Disposition: filename=example.zip" -H
"Content-Type: application/zip" -H "Packaging:
http://purl.org/net/sword/package/METSDSpaceSIP" --user admin@mail.com
```

<sup>27</sup> Disponível em: <https://github.com/swordapp/JavaServer2.0>

A seguir apresentamos as funções do protocolo SWORD implementadas para permitir a recuperação de informações do repositório.

Atom *Entry*: cria um contêiner dentro do servidor SWORD e opcionalmente é agregado os metadados sem adicionar nenhum conteúdo binário. Isso é feito usando o método POST ATOM *entry* à IRI da coleção. No DSpace por exemplo, inicialmente é criado o item contendo somente os metadados, podendo fazer o POST dos restantes arquivos posteriormente.

- ✓ `CollectionListManager` - possui métodos necessários para listar o conteúdo de uma determinada coleção.
- ✓ `ContainerManager` – possui métodos para recuperar e atualizar um objeto no servidor.
- ✓ `MediaResourceManager` – possui métodos para recuperar e atualizar um conteúdo dentro do container no servidor.
- ✓ `ServiceDocumentManager` – possui métodos para recuperar documentos do serviço do servidor a partir do método `getServiceDocument` que retorna um documento XML com as informações sobre a versão do protocolo SWORD, o tamanho máximo de *upload*, o título da coleção, a URL de depósito, o conjunto de formatos de pacotes aceitos, dentre outras informações.

```

55      @Override
56      public ServiceDocument getServiceDocument(String string, AuthCredentials ac, SwordConfiguration sc)
57      throws SwordError, SwordServerException, SwordAuthException {
58          Resource resource = new ClassPathResource("/config.properties");
59          Properties props = null;
60          try {
61              props = PropertiesLoaderUtils.loadProperties(resource);
62          } catch (IOException io) {
63              throw new SwordServerException("Erro ao carregar o arquivo properties", io);
64          }
65
66          ServiceDocument document = new ServiceDocument();
67
68          document.setVersion(props.getProperty("sword.generator.version"));
69          document.setMaxUploadSize(Integer.parseInt(props.getProperty("sword.max.uploaded.file.size", "-1")));
70          SwordWorkspace workspace = new SwordWorkspace();
71          workspace.setTitle(props.getProperty("Repositorio.name"));
72
73          SwordCollection collection = new SwordCollection();
74          collection.setTitle(props.getProperty("sword.collection.name"));
75          collection.setHref(props.getProperty("sword.base.url") + "/collection");
76          collection.addAcceptPackaging(UriRegistry.PACKAGE_SIMPLE_ZIP);
77          collection.setMediation(false);
78          workspace.addCollection(collection);
79
80          document.addWorkspace(workspace);
81
82          return document;
83      }
84  }

```

- ✓ StatementManagerImpl - possui métodos para recuperar a descrição de objetos no servidor SWORD, tendo em conta duas características: a estrutura e estado objeto no servidor. A Tabela 5.2, mostra os predicados das características e sua descrição.

Tabela 5.2 - Descrição de predicados do *Statement*

<i>Predicado</i>	<i>Descrição</i>
sword:originalDeposit	Fornecer IRI do recurso no servidor
Sword:state	Fornecer IRI que representa o estado do recurso no servidor
sword:packaging	Fornecer o formato do pacote do recurso
sword:stateDescription	Fornecer uma descrição legível do estado em que o item se encontra no servidor
sword:depositedBy	Fornecer o nome de usuário que fez o depósito

Fonte: O autor (2016)

- ✓ Classe SwordConfigurationImpl – fornece as configurações básicas do servidor SWORD, conforme mostra o trecho do código a seguir, algumas informações como: a versão do serviço SWORD usado pelo servidor, o e-mail do administrador, o tipo de autenticação, o tamanho limite de carregamento, etc...

```

81  @Override
82  public String generatorVersion() {
83      return props.getProperty("sword.generator.version");
84  }
85
86  @Override
87  public String administratorEmail() {
88      return props.getProperty("administrator.email");
89  }
90
91  @Override
92  public String getAuthType() {
93      return props.getProperty("sword.auth.method");
94  }
95
96  @Override
97  public boolean storeAndCheckBinary() {
98      return true;
99  }
100
101  @Override
102  public String getTempDirectory() {
103      return System.getProperty("java.io.tmpdir");
104  }
105
106  @Override
107  public int getMaxUploadSize() {
108      return Integer.parseInt(props.getProperty("sword.max.uploaded.file.size", "-1"));
109  }

```

### 5.3 Implementação e validação dos Resultados

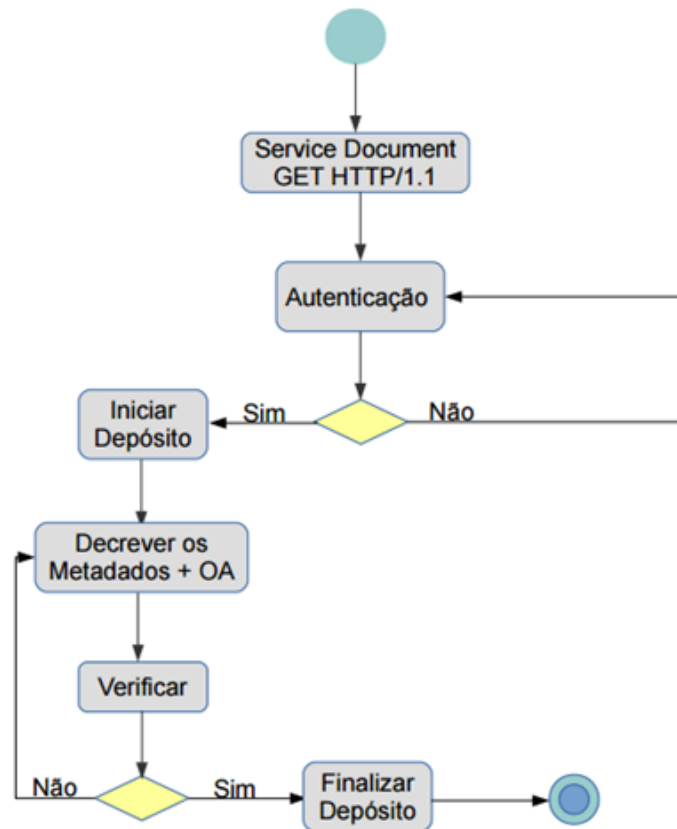
No processo da implementação da ferramenta, conforme previsto no roteiro da pesquisa como sendo uma atividade da etapa C, tem como início a descrição do funcionamento do protocolo SWORD v2 que é caracterizado por duas fases:

- Recuperação do *Service Document* - que representa a descrição da interface SWORD server (que neste caso a do repositório Cognix) e fornece informação sobre a capacidade do provedor SWORD e a lista de coleção disponível. O repositório Cognix possui uma coleção para depósito à qual podem ser armazenados OAs através do protocolo SWORD através do localizador: <http://endereço/repositorio/sword/servicedocument>.

```
<service xmlns="http://www.w3.org/2007/app"
xmlns:atom="http://www.w3.org/2005/Atom">
<workspace>
  <atom:title type="text">Repositório da Cognitiva Brasil</atom:title>
  <collection
href="http://127.0.1.1:8088/repositorio/sword/collection">
  <atom:title type="text">Coleção SWORD</atom:title>
  <mediation
xmlns="http://purl.org/net/sword/terms/">false</mediation>
  <acceptPackaging xmlns="http://purl.org/net/sword/terms/">
    http://purl.org/net/sword/package/SimpleZip
  </acceptPackaging></collection>
</workspace>
<generator
xmlns="http://www.w3.org/2005/Atom"
uri="http://www.swordapp.org/" version="2.0">
  developers@cognitivabrasil.com.br
</generator><version
xmlns="http://purl.org/net/sword/terms/">2.0</version>
</service>
```

- Depósito do recurso – neste caso faz-se uso da informação recebida pelo *Service Document* contendo a descrição completa do serviço, tais como a coleção para o depósito, o tipo de pacote a ser recebido, dentre outros, prepara-se e inicia-se o depósito do recurso no servidor. A Figura 5.3, ilustrada por um fluxograma mostra o processo genérico da recuperação do serviço e do depósito.

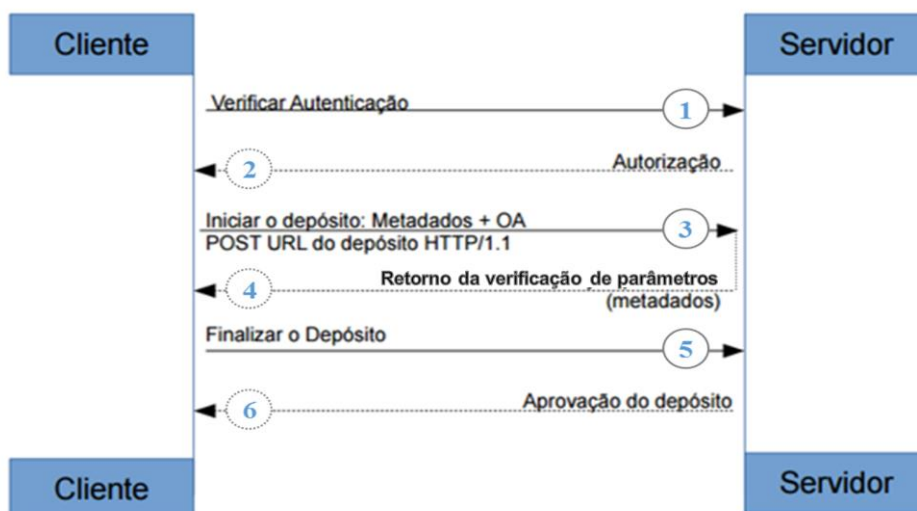
Figura 5.3 - Fluxograma geral de depósito



Fonte: O autor (2016)

O processo inicia com a requisição GET do *service document* por parte do cliente, logo em seguida é verificada a autenticação e dá-se início ao depósito, que é caracterizado pelo método POST na URL específica de depósito. No entanto, é enviado o pacote que outrora foi verificado na interface do SWORD cliente se possuía os metadados obrigatórios e o OAs. Embora os metadados já vem do Moodle num padrão deferente do OBAA a interface do SWORD servidor faz uma conversão, ou seja um mapeamento para o padrão OBAA para que o OAs seja armazenado no repositório Cognix.

Figura 5.4 - Diagrama de depósito



Fonte: O autor (2016)

Após o estabelecimento da comunicação entre o cliente (Moodle) e o servidor SWORD (repositório Cognix), inicia-se o fluxo de depósito representado pelo diagrama abaixo (Figura 5.4). Dando seguimento à descrição do fluxo de depósito:

1. O usuário autentica-se (*login* do repositório Cognix), um trecho do código que verifica a autenticação pode ser visto logo a seguir.

```

139  |
140  |  /**
141  |   * checks the AuthCredentials of the user
142  |   *
143  |   * @param auth
144  |   * @return
145  |   * @throws SwordAuthException when user can not authenticate
146  |   */
147  |  private boolean checkAuthCredentials(AuthCredentials auth) throws SwordAuthException {
148  |      log.debug("Dados do sword para autenticação. Usuário: " + auth.getUsername() + " senha: " + auth.getPassword());
149  |
150  |      User user = userService.authenticate(auth.getUsername(), auth.getPassword());
151  |      return (user != null && user.hasPermission(User.CREATE_DOC));
  
```

2. Quando a autenticação for sucedida o usuário é autorizado e passa para a etapa 3.
3. O usuário informa os metadados (segundo a lógica do *plugin* sword-upload apresentado no subcapítulo 3.5, todos elementos são obrigatórios exceto o autor, licença uri, e conteúdo (arquivo). A descrição completa pode ser vista na Tabela 5.3. No nosso caso, somente o nome, palavras-chave e o conteúdo são parâmetros obrigatórios.

Conforme mostra a Figura 5.5, a tela de submissão de um novo recurso do plugin, que é feito no Moodle, é possível ainda, acrescentar informações opcionais como: o resumo, idioma e tipo de conteúdo do OA a ser armazenado.

Figura 5.5 - Tela de submissão

Fonte: Autor

4. Retorno, caso os elementos obrigatório não forem informados.
5. É feito o empacotamento e enviado ao repositório.
6. O servidor cria um novo recurso no repositório e é finalizado o depósito (retornando o código de aceitação HTTP 201).

De uma forma resumida, a Tabela 5.3, apresenta o processo de submissão por parte do servidor SWORD, obedecendo uma sequência composta por três passos após autorização do depósito, é detalhado os parâmetros, regras HTTP bem como a respectiva descrição.

Tabela 5.3 - Parâmetros SWORD

<i>Parâmetros SWORD</i>	<i>Descrição</i>	
<i>Nova transferência</i>		
Requerido pelo HTTP 1.1 (POST)	Host	<i>Hostname</i> do repositório
	Content-Type	Tipo de arquivo: application/atom+xml;type=entry ...
	Content-Length	Tamanho
	Content-MD5	Checksum MD5 codificado para o conteúdo
	Content-Disposition	Com o nome do arquivo
	Packaging	Fornecendo a IRI do formato de pacote usado.



Requerido pelo Repositório	Authorization	Contém a chave da API e o usuário que logou
Requerido pelo SWORD (v2)	In-Progress	Assume True (False ou True)
	On-Behalf-Of (Opcional)	
	Slug (Opcional)	Identificador sugerido pelo conteúdo depositado
Requerido no Body (Atom Entry Document)	atom:entry	Adicionar a marcação de metadados externa (Dublin Core ou outro tipo, o OBAA por exemplo)
	Id	Identificador
	Author	Nome do autor
	Summary	Resumo
	Updated	Data de submissão
Exemplo	<p>POST Col-IRI HTTP/1.1  Host: example.org  Authorization: Basic ZGFmZnk6c2VjZXJldA==  Content-Length: [content length]  Content-Type: application/atom+xml;type=entry  In-Progress: true  On-Behalf-Of: jbloggs  Slug: [suggested identifier]</p> <pre>&lt;?xml version="1.0"?&gt; &lt;entry xmlns="http://www.w3.org/2005/Atom"   xmlns:dcterms="http://purl.org/dc/terms/"&gt;   &lt;title&gt;Title&lt;/title&gt;   &lt;id&gt;urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a&lt;/id&gt;   &lt;updated&gt;2005-10-07T17:17:08Z&lt;/updated&gt;   &lt;author&gt;&lt;name&gt;Contributor&lt;/name&gt;&lt;/author&gt;   &lt;summary type="text"&gt;The abstract&lt;/summary&gt;   &lt;!-- some embedded metadata --&gt;   &lt;dcterms:abstract&gt;The abstract&lt;/dcterms:abstract&gt;   ...   &lt;dcterms:contributor&gt;Contributor&lt;/dcterms:contributor&gt;   &lt;dcterms:description&gt;Description&lt;/dcterms:description&gt;   &lt;dcterms:hasPart&gt;Has Part&lt;/dcterms:hasPart&gt;   &lt;dcterms:hasVersion&gt;Has Version&lt;/dcterms:hasVersion&gt;   &lt;dcterms:identifier&gt;Identifier&lt;/dcterms:identifier&gt;   &lt;dcterms:title&gt;Title&lt;/dcterms:title&gt;   &lt;dcterms:type&gt;Type&lt;/dcterms:type&gt; &lt;/entry&gt;</pre>	
Adicionar arquivos à transferência		
HTTP POST body EM-IRI (Atom Edit Media IRI)	atom:entry	Deve especificar o formato de metadados (normalmente o DC)
	atom:link@rel="edit"	Media <i>Entry</i> IRI ( <i>Edit</i> -IRI)
	atom:link@rel="edit-media"	Media <i>Resource</i> IRI
	atom:link@rel="http://purl.org/net/sword/terms/add"	SWORD Edit IRI (Pode ser o mesmo que o Edit-IRI)

	sword:packaging	Formatos que o <i>Media Resource</i> pode recuperar
	sword:treatment	-
	sword:verboseDescription	-
	atom:link	-
	atom:link@rel="alternate"	-
Finalização da transferência		
HTTP POST body SE-IRI (SWORD Edit IRI)	POST SE-IRI HTTP/1.1 Host: example.org Content-Type: application/atom+xml;type=entry Authorization: Basic ZGFmZnk6c2VjZXJldA== Content-Length: [content length] In-Progress: true On-Behalf-Of: jbloggs [entry document]	

Fonte: Autor (2017)

Do Moodle é possível extrair-se um conjunto de metadados que posteriormente são empacotados e enviados juntos com o OA carregado no ambiente e enviados ao repositório. No ato da submissão de uma determinada aula o professor informa alguns metadados, tais como: o título, resumo, palavras-chave, descrição, licença, idioma, dentre outros.

Tabela 5.4 - Mapeamento do metadados do padrão DC para o OBAA

<i>Parâmetros</i>	<i>DSpace</i>	<i>Cognix</i>
Autor ( <i>login</i> do moodle)	dc.contributor.creator	obaa.general.author
Título	dc.title	obaa.general.title
Resumo	dc.description.abstract	obaa.general.abstract
Descrição	dc.description	obaa.general.description
Tipo (extensão)	dc.type	obaa.technical.format
Assunto (palavras-chave)	dc.subject	obaa.general.keyword
Idioma	dc.language	obaa.general.language
Licença	dc.rights	obaa.rights.description

Fonte: Autor (2017)

Os parâmetros estabelecidos no repositório Cognix seguem a especificação OBAA, assim há necessidade de os mapear do moodle para o repositório Cognix (Tabela 5.4), um trecho do código que apresenta o método do mapeamento dos metadados DC para OBAA pode ser visto logo a seguir, o mesmo se encontra na classe `CollectionDepositManager`.

```

169 private OBAA fromDublinCore(Map<String, List<String>> dc) {
170     OBAA obaa = new OBAA();
171     General g = new General();
172     for (String s : safe(dc.get("dc.description"))) {
173         g.addDescription(s);
174     }
175     for (String s : safe(dc.get("dc.subject"))) {
176         g.addKeyword(s);
177     }
178     for (String s : safe(dc.get("dc.title"))) {
179         g.addTitle(s);
180     }
181     for (String s : safe(dc.get("dc.language"))) {
182         g.addLanguage(s);
183     }
184     for (String s : safe(dc.get("dc.identifier"))) {
185         g.addIdentifier(new Identifier("", s));
186     }
187     obaa.setGeneral(g);
188     Technical t = new Technical();
189     for (String s : safe(dc.get("dc.type"))) {
190         t.addFormat(s);
191     }
192     obaa.setTechnical(t);
193     LifeCycle lc = new LifeCycle();
194     Contribute c = new Contribute();
195     for (String s : safe(dc.get("dc.contributor.creator"))){
196         c.addEntity(s);
197         c.setRole(Role.CREATOR);
198         lc.addContribute(c);
199     }

```

A seguir é apresentada a Tabela 5.5 com os verbos usados e possíveis códigos de resposta HTTP no processo de depósito dos objetos.

Tabela 5.5 - Verbos e Códigos de resposta HTTP

Métodos/Verbos	GET	Requisitar um recurso
	POST	Criar novo recurso
	PUT	Atualizar um recurso existente
	DELETE	Deletar recurso
Códigos de Resposta	200	OK
	201	CREATED
	400	BAD REQUEST
	404	NOT FOUND
	415	UNSUPPORTED MEDIA TYPE
	501	NOT IMPLEMENTED

Fonte: O autor (2016)

## 5.4 Teste

Inicialmente realizou-se testes locais no laboratório e em computadores pessoais em que foi validada a interface do usuário. Posteriormente o teste de comunicação cliente servidor, e por fim realizou-se o teste de envio dos OAs, juntamente com os metadados. É importante lembrar que os teste de depósito ainda não se mostram totalmente eficientes.

Para verificar-se a comunicação entre o cliente e o servidor foram feitos testes a partir do navegador e via cURL (ferramenta de linha de comando utilizada para transferência de dados) através da requisição simples estabelecida pelo método GET do *service document*, de modo a se obter informações da interface do servidor. Este teste se mostrou satisfatório, visto que a comunicação entre o cliente e o servidor foi bem sucedida.

Quanto aos testes de depósito, apenas observou-se do lado do cliente Moodle a captura dos metadados e dos OAs faltando do lado do servidor, ou seja, o recebimento e armazenamento do pacote. Os testes foram feitos pela interface do Moodle, com a submissão de OAs no formato PPT, WORD, PDF e áudio.

Por fim, este capítulo apresentou a solução tecnológica que buscamos desenvolver e validar nesta dissertação. É importante ainda, a efetivação de testes mais completos integrando cada situação que encontramos nos diversos OAs utilizados para estes testes preliminares. Embora seja necessário realizar-se mais testes e integração de grupo de usuários para que esta solução possa ser utilizada efetivamente pela comunidade interessada.

## 6 CONSIDERAÇÕES E TRABALHOS FUTUROS

Este capítulo tem como objetivo, apresentar as considerações finais sobre a ferramenta desenvolvida para a catalogação e disseminação de conteúdo educacional. A solução aqui proposta foca a interoperabilidade entre ambientes educacionais. São apresentadas a conclusão e os trabalhos futuros com sugestões a serem desenvolvidas visando a melhoria do trabalho aqui apresentado.

O estudo e pesquisa apresentado se propôs a contribuir para a busca de uma solução para o problema da interoperabilidade entre sistemas educacionais, em particular entre um repositório de conteúdo educacional (o repositório Cognix) e um ambiente virtual de aprendizagem (o Moodle). A solução que buscamos foi o desenvolvimento de uma ferramenta baseada no protocolo SWORD para a catalogação automática de OAs provenientes do Moodle no repositório Cognix de modo a ajudar na disseminação (recuperação e uso) de conteúdo educacional.

O estudo e levantamento das formas de integração de AVA com os repositórios de objetos de aprendizagem, auxiliou bastante, visto que, para que a catalogação automática de OAs seja feita, recorreu-se ao ambiente de aprendizagem Moodle e o repositório Cognix que é utilizado pela Federação de Repositórios Educa Brasil (FEB<sup>28</sup>), entretanto, até o presente momento o Cognix não possui ainda nenhuma conexão com LMS. O ambiente Moodle foi escolhido por ser amplamente utilizado, de código aberto e bem documentado para desenvolvimento de *plugins*. A escolha do repositório Cognix deveu-se ao fato deste ser um repositório novo e inteligente, pois possui algoritmos para a obtenção automática de um certo número de metadados além de armazenar os objetos organizados por conteúdo, tem como padrão de metadados OBAA, que é objeto de pesquisa do grupo ao qual esse trabalho está inserido e por possuir contato com os desenvolvedores deste projeto. Além disso é de código aberto.

O desenvolvimento deste trabalho criou maior facilidade ao desenvolvedor de OAs no processo de catalogação automática do OA em um repositório a partir do ambiente do Moodle tende a aumentar a quantidade de objetos disponíveis, pelo fato de já ser um ambiente conhecido pelos professores e reduzindo assim o trabalho necessário para disponibilizar seu OA para a comunidade.

A implementação da interface de comunicação, através do uso do protocolo de comunicação SWORD tanto no Moodle como no Cognix, trouxe um ganho no que se refere ao

---

<sup>28</sup> <http://www.portalobaa.org/feb-federacao-de-repositorios-obaa>

melhoramento da qualidade e quantidade de metadados que são catalogados, dos recursos, visto que, alguns elementos, tais como: o título, autor, extensão, dentre outros já existem no Moodle e, ainda há a extração de metadados realizada automaticamente pelo repositório Cognix, do próprio arquivo contendo o OA. No entanto, embora seja um dos objetivos deste trabalho a implementação do SWORD no repositório Cognix este objetivo não foi alcançado no período de duração deste trabalho.

Como trabalhos futuros, se pretende melhorar a ferramenta de modo que se possa aumentar a quantidade de metadados que pode ser obtido automaticamente a partir do Moodle, acrescentando assim mais parâmetros descritivos no armazenamento dos OAs. Também, acrescentar mais funcionalidades ao módulo de interoperabilidade, a partir das restantes operações CRUD que já se encontram na biblioteca SWORD faltando assim a implementação dos mesmos. Visto que somente é criado o OA no repositório sem que haja as opções de atualização, recuperação ou remoção a partir do moodle. Pretende-se ainda melhorar os testes da ferramenta de modo a garantir melhor confiabilidade. Também se faz necessária a realização de uma quantidade maior de testes desta implementação. Ainda ressaltar que os metadados que existem em outros padrões como o LOM, SCROM e DC não são objetos de estudo deste trabalho.

## REFERÊNCIAS

ATZORI, L.; IERA, A.; MORABITO, G. The internet of things: A survey. *Comput. Netw.*, v. 54, 2010.

BEHAR, P. A; et al. ROODA - Rede cOOperativa De Aprendizagem – Uma plataforma de suporte para aprendizagem à distância. **Revista Informática na Educação Teoria Prática**. Porto Alegre: v.4, n.2, p. 87-96, 2001.

BENJELLOUN, Rida. Archimède: a Canadian solution for institutional repository. *Library Hi Tech*, v. 23, n.4, pp. 481-489, 2005.

BRADY, Helen. The MrCute Repository: The Next Phase, **Ariadne**, 2009. Disponível em: <<http://www.ariadne.ac.uk/issue58/brady>>. Acesso em: 02 de mai. de 2015.

BRADFORD, P.; PORCIELLO, M.; BALKON, N.; BACKUS, D. The Blackboard learning System: The be all and end all in educational instruction? **Journal of Educational Technology Systems**, 2007.

CASALI, A.; DECO, C.; BENDER, C.; FONTANARROSA, S. Extracción Automática de Metadatos de Objetos Digitales Educativos. **In Proceedings Novena Conferencia Latinoamericana de Objetos y Tecnologías de Aprendizaje LACLO 2014**, Colombia, p. 23-29, out. 2014.

COSTA, Sely Maria de Souza; LEITE, Fernando César Lima. Repositórios institucionais: potencial para maximizar o acesso e o impacto da pesquisa em universidades, 2006. **In 1ª Conferência Ibero Americana de Publicações Eletrônicas no Contexto da Comunicação Científica, Brasília, 2006**, p. 1-10. Universidade de Brasília.

DCMI. Dublin Core Metadata Initiative. Disponível em: <<http://dublincore.org/>>. Acesso em: 26 out. 2015.

DÍAZ, F.; SCHIAVONI, A.; AMADEO, A.; CHARNELLI, M. Integrating a learning management system with a student assignments digital repository. A case study. **In Proceedings of the IADIS Internationale-Learning 2013**, pp. 207-215, jul. 2013

DOUGIAMAS, M.; TAYLOR, P. Moodle: Using Learning Communities to Create an Open Source Course Management System. In D. Lassner & C. McNaught (Eds.), **Proceedings of EdMedia: World Conference on Educational Media and Technology 2003** (pp. 171-178). Association for the Advancement of Computing in Education (AACE). 2003.

DUNCAN, Charles. Digital Repositories: e-Learning for Everyone. Presented at eLearnInternational, Edinburgh 9-12 February 2003.

GUTTERIDGE, Christopher. GNU EPrints 2 Overview, 2002. Disponível em: <<http://eprints.soton.ac.uk/256840/2/eprintoverview.pdf>>. Acesso em: 30 abr. 2015

IBICT. Repositórios Digitais. Disponível em: <<http://www.ibict.br/informacao-para-ciencia-tecnologia-e-inovacao%20/repositorios-digitais>>Acesso em: 26 out. 2015.

IEEE. Standard for Learning Object Metadata. Learning Technology Standards Committee of the IEEE. 2002.

IMS. IMS Global Learning Consortium. 2002.

KRAFZIG, D.; BANKE, K.; SLAMA, D. Enterprise SOA: Service-Oriented Architecture Best Practices (The Coad Series). Upper Saddle River, NJ, USA: Prentice Hall PTR, 2004. ISBN 0131465759.

KUBICEK, H.; CIMANDER, R. Three dimensions of organizational interoperability, European Journal of ePractice, No. 6, January 2009, epractice.eu.

LEAL, J. P.; QUEIRÓS, R. CrimsonHex: Um Repositório de Objectos de Aprendizagem, 2009. Disponível em: <[http://recipp.ipp.pt/bitstream/10400.22/4528/1/COM\\_RicardoQueiros\\_2009\\_1.pdf](http://recipp.ipp.pt/bitstream/10400.22/4528/1/COM_RicardoQueiros_2009_1.pdf)>. Acesso em: 26 abr. 2015.

LEAL, J. P.; QUEIRÓS, R. Integration of repositories in Learning Management Systems, 2010. Disponível em: <[http://www.dcc.fc.up.pt/~rqueiros/pubs/XATA\\_2010\\_3.pdf](http://www.dcc.fc.up.pt/~rqueiros/pubs/XATA_2010_3.pdf)>. Acesso em: 29 abr. 2015.

LEWIS, S.; CASTRO, P.; JONES, R. SWORD: Facilitating Deposit Scenarios. **D-Lib Magazine**, vol 18, no. 1-2. 2013. Disponível em: <<http://www.dlib.org/dlib/january12/lewis/01lewis.html>>. Acesso em: 29 abr. 2015

LEWIS, S.; et al. If SWORD is the answer, what is the question?: Use of the Simple Web-service Offering Repository Deposit protocol. Program: electronic library and information systems, Vol 43, Issue 4, pp. 407-418. 2009. Disponível em: <<http://hdl.handle.net/2292/5315>>. Acesso em: 29 abr. 2015.

LYNCH, Clifford. A. Institutional repositories: essential infrastructure for scholarship in the Digital Age [Énlinea]. (2003). ARL: Bimonthlyreport, n.226. 2003

OCHOA, X.; DUVAL, E. Quantitative analysis of learning object repositories. **IEEE Transactions on Learning Technologies**, v.2, n.3, p. 226–238, Jul-Set. 2009.

OPENDOAR (2016). Directory of open access repositories. <<http://www.opendoar.org/onechart.php?cID=&ctID=&rtID=&clID=&lID=&potID=&rSoftwareName=&search=&groupby=r.rSoftwareName&orderby=Tally%20DESC&charttype=pie&width=600&height=300&caption=Usage%20of%20Open%20Access%20Repository%20Software%20-%20Worldwide>>. Acesso em: 13 de jun. 2016.

PAYETTE, Sandra; LAGOZE, Carl. Flexible and Extensible Digital Object and Repository Architecture (FEDORA). **Proceedings of the Second European Conference on Research and Advanced Technology for Digital Libraries**, p. 41-59, September 21-23. 1998. Disponível em: <<http://www.cs.cornell.edu/payette/papers/ECDL98/FEDORA.html>>. Acesso 30 abr. 2015.



RODRIGUES, Alessandra Pereira. **Integração de ambiente virtual de aprendizagem com repositório digital**. 2013. 187 f. Tese (Doutorado em Informática na Educação) – CINTED, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2013.

SAMPSON, D.; ZERVAS, P. Learning object repositories as knowledge management systems. **Knowledge Management & E-Learning: An International Journal (KM&EL)**, v.5, n.2, 2013.

STEVENSON, A. (2009). SWORD2 project final report. Disponível em: <<http://www.webarchive.org.uk/wayback/archive/20140615083222/http://www.jisc.ac.uk/media/documents/programmes/reppres/sword2finalreport.pdf>>. Acesso em: 30 abr. 2015.

SITAS, Anestis. CDSware (CERN Document Server Software), **Library Hi Tech**, v. 24 ISS: 3, p. 420 – 429, 2006

SMITH, MacKenzie; et al. DSpace: An Open Source Dynamic Digital Repository. 2003. Disponível em: <<http://www.dlib.org/dlib/january03/smith/01smith.html>>. Acesso em: 30 abr. 2015.

TAROUCO, L. M.; et al. **Objetos de Aprendizagem: Teoria e Prática**. Evangraf. Porto Alegre. 2014.

TAROUCO, L. M. R.; RODRIGUES, A. P; SCHMITT, M A.; VIDEIRA, J. A. Depósito de objetos de aprendizagem em repositórios a partir da integração com ambientes virtuais de aprendizagem. **RENOTE**, v.11, n3, 2013.

VEER, H. van der; WILES, A. Achieving technical interoperability - the ETSI approach. **ETSI White Paper**, n3. 3ed, 2008.

VICARI, R. M.; et al. Padrão para Metadados de Objetos de Aprendizagem Multiplataforma. Porto Alegre: UFRGS, 2009.

VICARI, R. M.; et al. Proposta de Padrão de Objetos de Aprendizagem Baseados em Agentes (OBAA). In: **LATIN-AMERICAN CONFERENCE ON LEARNING OBJECTS (LACLO 2010)**, S. Paulo. Anais... São Paulo, 2010.

WILEY, D. Connecting Learning Objects to Instructional Design Theory: A Definition, A Metaphor, and A Taxonomy. 2001. In **Wiley, D. The Instructional Use of Learning Objects: Online Version**. Disponível em: <<http://reusability.org/read/chapters/wiley.doc>>. Acesso em: set. 2015.