

Uma ferramenta para fornecer apoio a catalogação de metadados de objetos de aprendizagem - LINNAEUS

Ederson Luiz Silveira¹, Matheus Campezzato Galão¹, João Carlos Gluz¹

¹PIPCA – Programa Interdisciplinar de Pós-Graduação em Computação Aplicada –
Universidade do Vale dos Sinos (UNISINOS)
Av. Unisinos, 950 – 93.022-000 – Cristo Rei – São Leopoldo – RS – Brasil

esilveiral@gmail.com, matheusz.cg@gmail.com, jcgluz@unisinos.br

Abstract. *This work presents Linnaeus, a tool that supports the cataloging process of learning objects, working with the OBAA metadata standard. The Linnaeus uses an innovative technological solution to support this process of cataloging, integrating the technologies of intelligent agents and educational ontologies in its project. The project aims to provide an intelligent and proactive support, helping users without technical knowledge about learning objects or metadata standards to correctly catalog their objects. This article presents the architecture of Linnaeus and system's ontologies, showing the main features of the prototype tool and the initial results of its use.*

Resumo. *Este trabalho apresenta a ferramenta Linnaeus que apoia o processo de catalogação de objetos de aprendizagem, considerando o padrão de metadados OBAA. O Linnaeus usa uma solução tecnológica inovadora para suportar este processo de catalogação, integrando as tecnologias de agentes inteligentes e de ontologias educacionais em seu projeto. O projeto se propõe a fornecer um apoio inteligente e pró-ativo, ajudando usuários sem conhecimentos técnicos sobre metadados ou padrões de objetos de aprendizagem a catalogar de forma correta seus objetos. O artigo apresenta a arquitetura do Linnaeus e as ontologias aplicadas no sistema, mostra as principais características do protótipo da ferramenta e os resultados iniciais de sua utilização.*

1. Introdução

Objetos de Aprendizagem (OA) são apresentados como qualquer entidade, que pode ser utilizada, reutilizada ou referenciada durante o aprendizado apoiado por computador (IEEE-LTSC, 2002). Para possibilitar a reutilização, descoberta e facilitar a interoperabilidade entre os objetos foram criados padrões de metadados, tais como: IEEE LOM (IEEE-LTSC, 2002), Dublin Core (Kunze e Baker, 2007), SCORM (ADL, 2001) e OBAA (Bez et al., 2010).

Devido a necessidade contínua de atingir mais aplicações e domínios, os padrões de metadados vêm se tornando cada vez mais complexos e extensos. Consequentemente, o tempo e esforço gastos para o preenchimento dos metadados pelos projetistas de OAs têm aumentado. O preenchimento dos metadados é uma atividade

importante, formando a base dos processos de catalogação, indexação e localização dos objetos. O preenchimento manual destas informações, levando em consideração o complexo cenário de padrões de metadados, tende a gerar divergências e erros nas informações, devido a possíveis diferenças de interpretação quanto ao significado do metadado.

Neste contexto, a criação de ferramentas que auxiliem no processo de autoria de OAs, incluindo o processo de catalogação, torna-se um ponto fundamental na utilização de qualquer padrão de OA. Diversas ferramentas de autoria foram desenvolvidas com o intuito de facilitar o desenvolvimento de OAs nos diversos padrões existentes. Dentre os exemplos encontrados pode-se citar: *Aprenderis.cl*, *Atenex*, *CourseLab*, *eXe Learning*, *FreeLOms*, *Lectora*, *LOMPad*, *Xerte* e *DSPACE*. Várias dessas ferramentas, como *Aprenderis.cl*, *Atenex*, *CourseLab*, *FreeLoms*, *LOMPad* e *Lectora* são basicamente editores de conteúdo, não tratando das questões de catalogação do objeto. Assim, *eXe Learning* (BARBONE e RIFON, 2009) e *Xerte* (BALL e TENNEY, 2008), que combinam facilidades de autoria de conteúdo com suporte à catalogação, e *DSPACE* (SMITH et al., 2003), que é exclusivamente direcionada ao processo de catalogação, são as ferramentas mais relacionadas ao presente trabalho. Todas essas ferramentas disponibilizam uma interface gráfica para o preenchimento dos metadados. Apesar destas ferramentas oferecerem um mecanismo de entrada de informação organizado com vocabulário enumerado e explicações sobre a semântica do metadado, as informações ainda tem que ser manualmente preenchidas pelo autor do OA. Faltam mecanismos que auxiliem o preenchimento dos metadados de forma inteligente, ou seja, sem que o autor ou projetista do OA tenha que informar todos os dados explicitamente, este é um tópico que pode ser melhorado.

Em vista disso, este trabalho apresenta a ferramenta *Linnaeus* que apoia o processo de catalogação de OAs relacionado ao preenchimento dos metadados. A ferramenta auxilia as atividades de criação e edição dos metadados de OAs para domínios educacionais. Para isto o *Linnaeus* utiliza mecanismos para preenchimento automático dos metadados, utilizando técnicas de inferência aplicadas sobre ontologias de conteúdos educacionais e a ontologia OBAA (Gluz e Vicari, 2011).

O sistema *Linnaeus* opera somente na camada de metadados. O protótipo deste sistema será integrado à infraestrutura MILOS (Gluz e Vicari, 2010; Gluz et al., 2012), fornecendo parte do serviço de autoria de OA a ser disponibilizado pela infraestrutura. Assim o *Linnaeus* efetua um intercâmbio de informações com os demais subsistemas da infraestrutura.

A organização deste artigo é a seguinte: a Seção 2 a arquitetura do sistema e dos seus componentes, na Seção 3 é apresentada um uso de caso do sistema juntamente com um passo-a-passo e na seção 4 são apresentas as considerações finais.

2. Arquitetura do Sistema *Linnaeus*

O principal cenário de aplicação da ferramenta *Linnaeus* (Silveira e Gluz, 2012) é fornecer apoio a usuários sem conhecimentos técnicos sobre metadados ou padrões de objetos de aprendizagem e as atividades de criação e edição dos metadados de Objetos de Aprendizagem para domínios educacionais. Neste contexto, os principais requisitos

levados em conta no projeto da interface do sistema são a disponibilização de uma interface de edição *Web* para a troca de informações com o usuário, em conjunto com o desenvolvimento de mecanismos de preenchimento e validação do conteúdo dos metadados baseado em ontologias de domínios educacionais. Os mecanismos de preenchimento e validação de metadados permitem que a interface com o usuário esteja mais próxima do domínio educacional do OA.

2.1 Infraestrutura MILOS

A arquitetura de software definida para o *Linnaeus* (Silveira e Gluz, 2012) deve estar alinhada e integrada com a infraestrutura de agentes MILOS (Gluz e Vicari, 2010; Gluz et al. 2012), devendo ser considerada um dos componentes da própria MILOS. A arquitetura da MILOS é dividida em três grandes níveis de abstração (ver Figura 1): a) Nível das Ontologias: responsável pela especificação dos conhecimentos que serão compartilhados entre os agentes da infraestrutura; (b) Nível de Agentes: responsável pela implementação do suporte aos requisitos de adaptabilidade, interoperabilidade e acessibilidade previstos no padrão de metadados OBAA (Bez et al., 2010); (c) Nível das Facilidades de Interface: responsável pela comunicação dos agentes da MILOS com servidores *Web*, ambientes virtuais, repositórios de OA, bancos de dados, serviços de diretórios e demais tipos de aplicações educacionais.

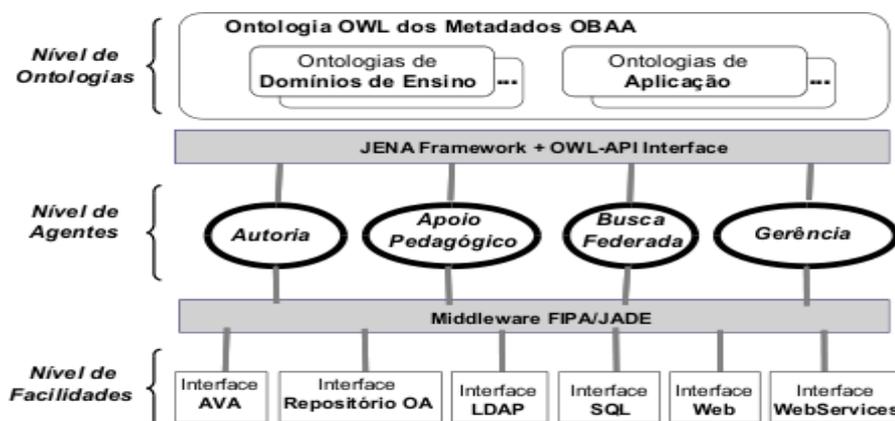


Figura 1 – Organização geral da MILOS. Fonte (Gluz e Vicari, 2010).

O *Linnaeus* faz parte do Sistema de Autoria da MILOS, cujo principal objetivo é prover assistência para os processos de autoria de OA, incluindo autoria de metadados e conteúdos (ver Figura 2).

Os agentes da camada de inteligência do sistema de preenchimento de metadados de OAs incorporam as principais facilidades do sistema de catalogação. Estas facilidades são acessadas diretamente por meio de uma interface *Web*. Por trás da operação dos agentes de preenchimento existe um conjunto de agentes em forma de *Wizards* de edição que são responsáveis pelo apoio a atividades de catalogação específica.

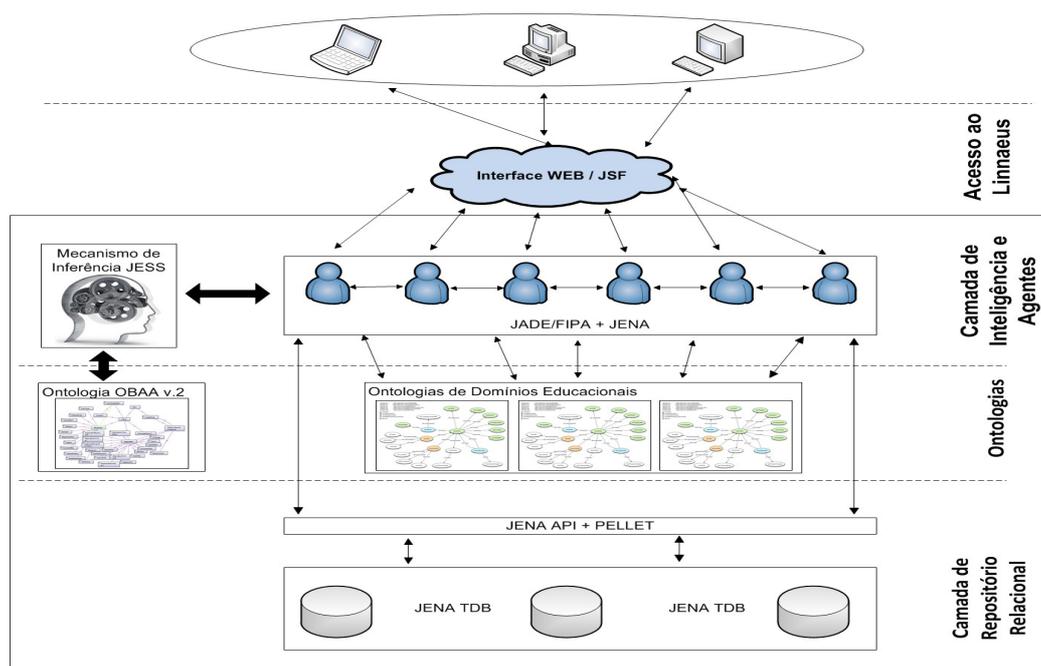


Figura 2 – Organização do sistema Linnaeus. Fonte (autor, 2013).

2.2 Agentes do Sistema *Linnaeus*

A principal função do *Linnaeus* dentro da MILOS é disponibilizar um conjunto de *wizards* para o apoio ao preenchimento de metadados compatível com os metadados para objetos OBAA. Porém, tendo em vista o estágio inicial de desenvolvimento da infraestrutura MILOS, o *Linnaeus* também se tornou um protótipo inicial para um sistema de apoio inteligente e pró-ativo, ajudando usuários sem conhecimentos técnicos sobre metadados ou padrões de objetos de aprendizagem a catalogar de forma correta seus objetos. Assim, essa ferramenta foi concebida desde o início nas três camadas da MILOS: ontologias, agentes e interface com o usuário. As operações e tarefas do *Linnaeus* são implementadas por meio de agente de *software* (Wooldridge, 2002; Weiss, 1999), usando o conceito de agentes *wizards* inteligentes. A criação do catálogo de metadados do sistema se dará de forma automática sem o conhecimento do usuário, o catálogo seguirá como uma de suas premissas o domínio do OA, sua aplicabilidade, plataforma de operação, para a criação do catálogo.

A modelagem do comportamento dos agentes deste sistema é baseada em *workflow* de regras de negócio BPM (*Business Process Management*) com o auxílio da ferramenta *WADE* (Caire et al, 2008), que é um subsistema da plataforma JADE. Seguindo esta ideia, serão apresentados o comportamento de dois importantes agentes do sistema.

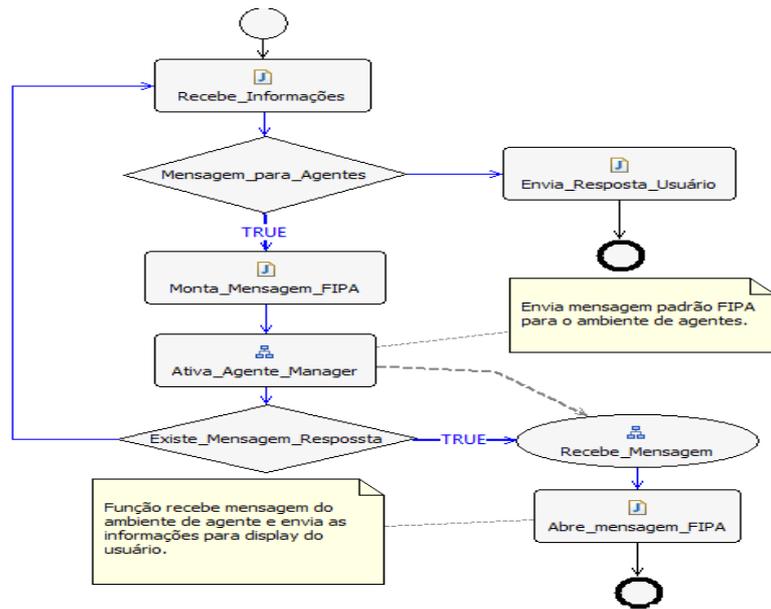


Figura 3 – Comportamento do agente responsável pela comunicação com o usuário.
Fonte (autor, 2013).

O agente *Gateway* (figura 3) é o responsável por toda interação com o usuário. A cada solicitação ou resposta do projetista, este agente acionará o agente *Manager* que por sua vez acionará os demais agentes quando necessário, para que as tarefas solicitadas sejam atendidas. A troca de mensagens entre o agente *Gateway* e o *Manager* ocorre de forma que as informações fornecidas pelo usuário são convertidas para mensagens no formato FIPA, por uma classe extra adicionada ao agente.

O agente *Manager* (figura 4) é o organizador das informações sobre definição dos metadados, validação de valores preenchidos manualmente e também pela solicitação de informações ao usuário. Este agente é o responsável por disparar o agente que realizará a inferência dos valores dos metadados a partir das ontologias definidas em conjunto com a ontologia OBAA (Gluz e Vicari, 2011). Além dos agentes responsáveis pelo controle e organização do mecanismo de inferência, este agente também coordena os agentes responsáveis pelo acesso a base de triplas RDF, onde são armazenados os novos metadados gerados pelo mecanismo de inferência .

A geração dos metadados de saída é responsabilidade do agente *Manager*. Este agente também é responsável pela validação das informações fornecidas pelo projetista, no momento em que o mecanismo de inferência não consiga resolver a análise que esteja executando. As mensagens trocadas entre os agentes são, principalmente, os próprios conjuntos de metadados gerados no processo de catalogação.

Este agente é responsável pela execução correta dos demais agentes do sistema. Basicamente o agente *Manager* centraliza as informações e as distribui de forma correta aos demais, sem que haja conflitos de requisição e sobrecarga de tarefas a serem executadas.

O agente *Manager* possui suporte a mensagens provenientes de outros sistemas multiagentes, em caso do recebimento de mensagens deste tipo o sistema possui uma chamada para um agente que fara a comunicação com outros sistemas para eventuais

consultas a base de metadados ou até mesmo a alteração de informações contidas nesta base.

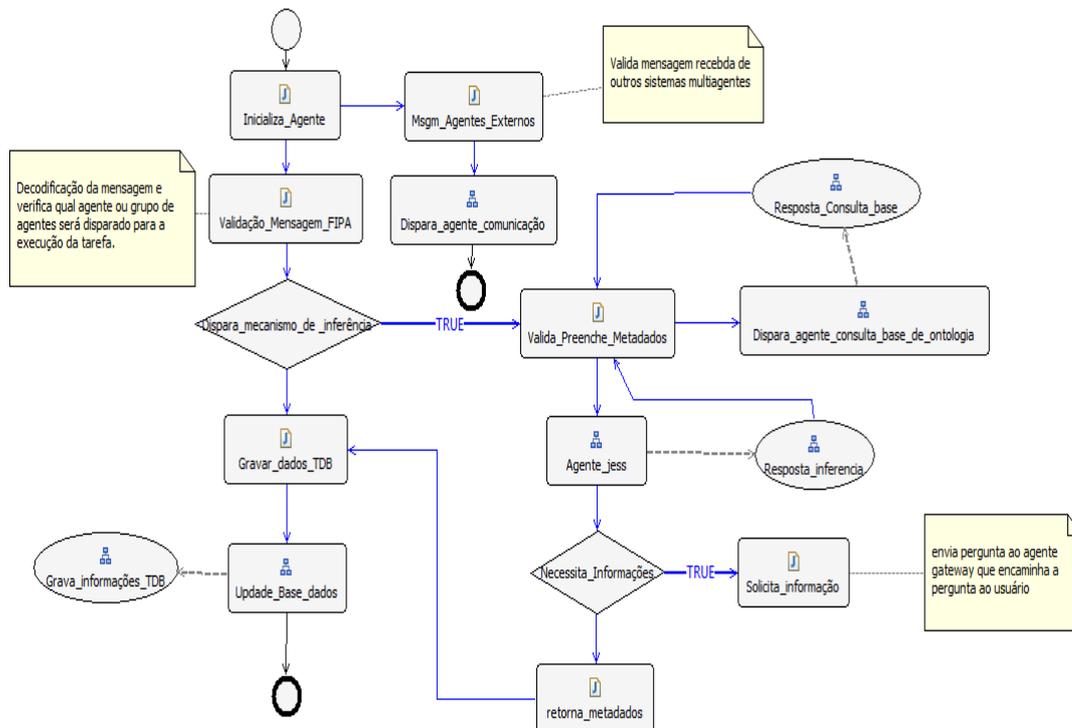


Figura 4 – Comportamento do agente responsável pelo gerenciamento e controle dos demais agentes. Fonte (autor, 2013).

3. Interface Adaptativa do Linnaeus

O protótipo do sistema *Linnaeus* foi projetado para ser utilizado por desenvolvedores de OA. O *Linnaeus* oferece distintos de interface de usuário, que se adaptam aos conhecimentos do desenvolvedor em relação aos aspectos técnicos dos OA. São considerados três níveis de conhecimentos: desenvolvedores leigos, sem conhecimentos técnicos sobre OA ou metadados, desenvolvedores com conhecimentos intermediários sobre estes temas e desenvolvedores que se consideram especialistas (*experts*) nestes conhecimentos. Seguindo esta caracterização, as interfaces de usuário do *Linnaeus* são especificadas em três casos de uso distintos (ver Figura 5), sendo o primeiro para usuários classificados como iniciantes devido ao nível de conhecimento técnico sobre OA, o segundo em nível intermediário de conhecimento técnico e o terceiro em nível *expert* com grande conhecimento técnico em OA.

A seguir são detalhados as principais características do caso de uso de usuários leigos, opção que leva à interface de usuário do *Linnaeus* que oferece maior suporte ao processo de catalogação. Será apresentado um processo passo-a-passo de catalogação de um OA utilizando o *Linnaeus*, mostrando a interação do sistema com o usuário através de *pop-ups web* que solicitam informações ao usuário sempre que o mecanismo de inferência necessitar de informações adicionais para a catalogação. As perguntas são geradas pelo agente que coordena o mecanismo de inferência, estas perguntas são geradas de uma base de perguntas pré-selecionadas para que o usuário possa respondê-las de forma imediata sem a necessidade de buscar informações e ajuda para

tal questionamento. O sistema *Linnaeus* está definido com dois tipos de perguntas que serão aplicadas ao desenvolvedor, a primeira pergunta é do tipo *EditBox*, para entrada de dados do tipo texto. O segundo tipo de pergunta é do tipo *RadioBox* para respostas do tipo booleanas do tipo VERDADEIRO/FALSO ou SIM/NÃO.

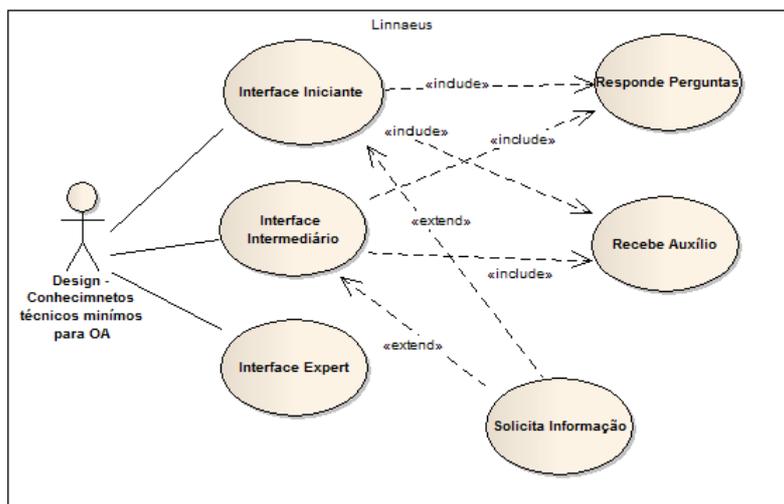


Figura 5 – Diagrama de Casos de Uso do Linnaeus. Fonte [autor, 2013].

Ao acessar o sistema o usuário encontra a área onde deve fornecer informações de identificação (figura 6) e grau de conhecimento técnico com relação a objetos de aprendizagem. Juntamente com o idioma nativo do OA que será catalogado. No momento em que o botão de *login* é executado os agentes do sistema juntamente com o mecanismo de inferência são inicializados e o processo de catalogação de metadados fica em modo de espera para realizar a consulta na base de ontologias para o domínio definido pelo desenvolvedor de OA.

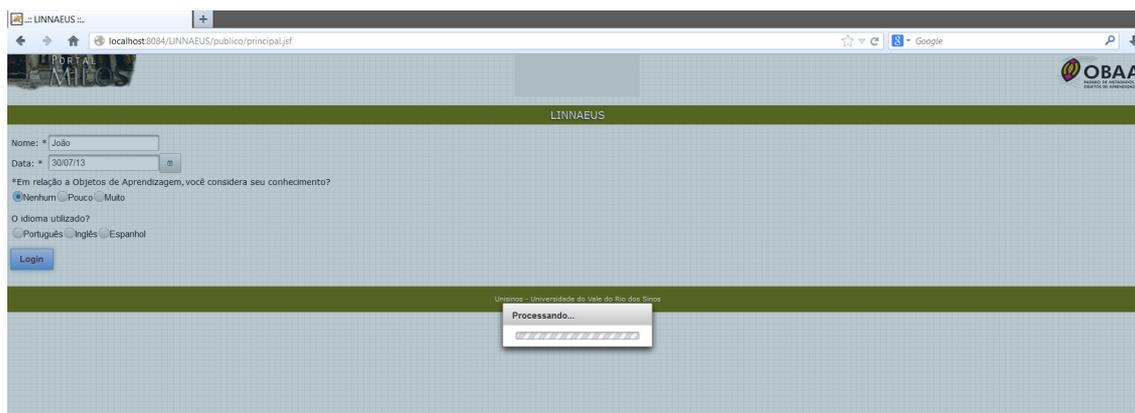


Figura 6 – Tela de informações iniciais fornecidas pelo desenvolvedor. Fonte [autor, 2013].

O próximo passo executado pelo desenvolvedor no *Linnaeus* é responder a solicitação do domínio do OA que será catalogado. Através de um entrada de dados do tipo *EditBox* (figura 7) com esta informação o sistema executa a pesquisa na base ontologias de domínios educacionais e transmite estas informações ao mecanismo de inferência para a continuidade do processo de preenchimento de metadados. Quando o

mecanismo de inferência não conseguir resolver qual informação será atribuída a determinado metadado e este tiver o tipo texto o mecanismo de inferência envia a solicitação de que precisa de informações adicionais ao seu agente de gerenciamento e este encaminha a solicitação para que o desenvolvedor de OA possa responde-la e encaminha o conteúdo desta resposta novamente para o mecanismo de inferência e o processo segue até o termino do preenchimento dos metadados.

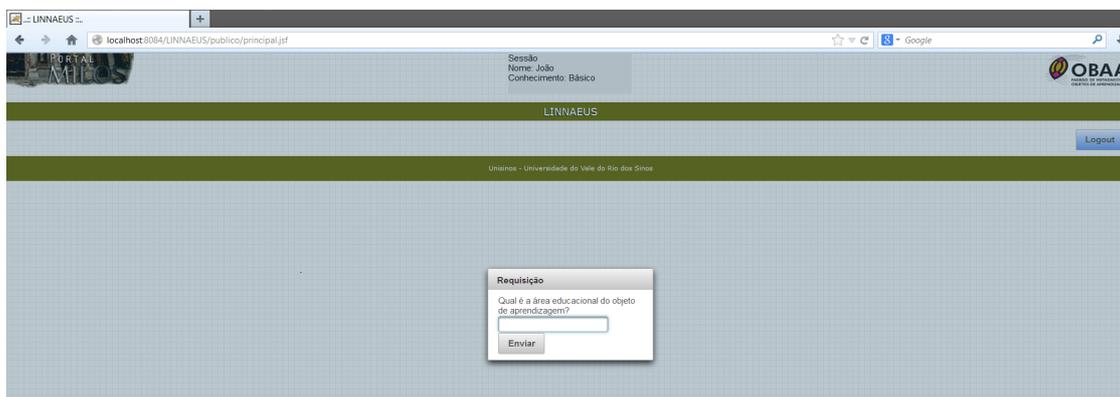


Figura 7 – Tela de entrada de dados do tipo texto. Fonte [autor, 2013].

Caso ocorra a necessidade de informações que utilizem respostas booleanas o sistema encaminha ao desenvolvedor de OA um questionamento do tipo *RadioBox* conforme mostrado na figura 8.

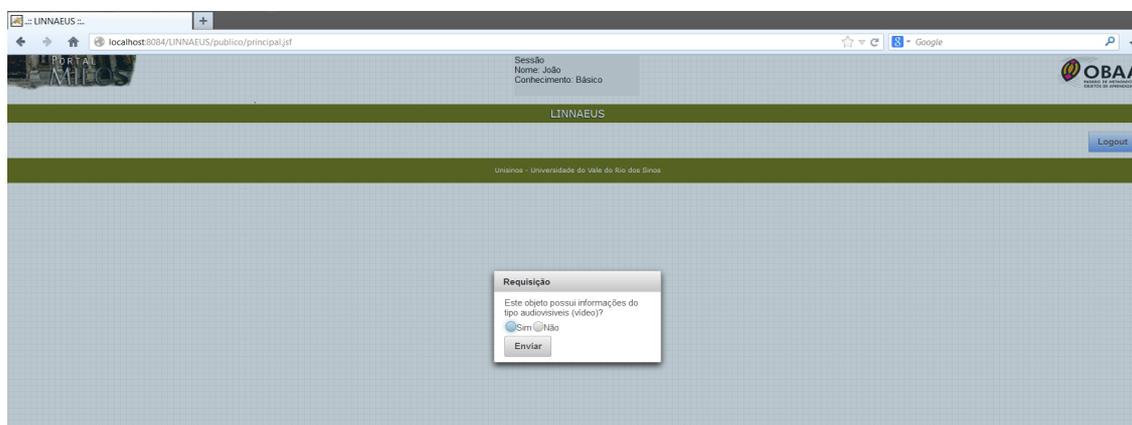


Figura 8 – Tela de entrada de dados do tipo *RadioBox*. Fonte (autor, 2013).

Se o *Linnaeus* não encontrar nenhum problema ou dificuldade para o preenchimento dos metadados, e o processo tiver percorrido todos os itens definido pelo padrão de metadados OBAA o sistema finaliza todas as tarefas e gera os metadados para serem gravados na base de dados de triplas RDF.

O próximo passo a ser executado pelo desenvolvedor de OA é a finalização do processo. O sistema apresenta ao usuário os metadados gerados (figura 9) que podem ser alterados por edição manual sem o auxílio do sistema de agentes ou o desenvolvedor poderá finalizar o processo aceitando os metadados gerados pelo sistema através do botão de conclusão da tarefa. Ao executar o botão de conclusão o sistema dispara a

gravação dos metadados na base de dados de triplas RDF e uma nova sessão é iniciada para a criação de um novo catalogo.

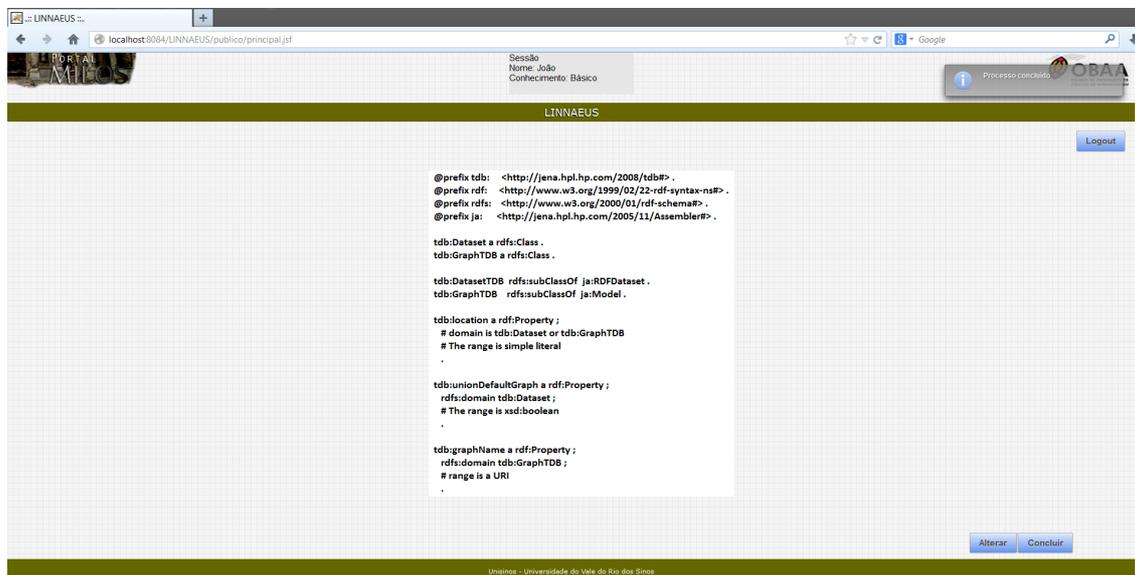


Figura 9 – Tela de apresentação dos metadados gerados pelo sistema no formato de triplas RDF. Fonte [autor, 2013].

4. Considerações Finais

Este trabalho apresentou a evolução da ferramenta *Linnaeus*, uma implementação da arquitetura multiagente proposta para apoiar o processo de catalogação de objetos de aprendizagem. A expectativa do sistema LINNAEUS é reduzir de forma significativa o volume de informações que o projetista tem que catalogar. Atualmente o sistema Linnaeus está em processo de validação de suas principais funcionalidade e de aprimoramento do mecanismo de inferência.

O sistema LINNAEUS terá sua aplicabilidade testada por usuários finais no próximo mês (agosto), estando disponível para acessos *online* e preenchimento da avaliação de seu desempenho visto por este usuário. Espera-se com os resultados deste trabalho validar a aplicabilidade da proposta de catalogação para objetos de aprendizagem compatíveis com padrão de metadados OBAA.

Espera-se que o sistema *Linnaeus* seja uma contribuição para um modelo inicial para a arquitetura do subsistema de Autoria da Infraestrutura MILOS (Gluz e Vicari, 2010; Gluz et al., 2012), mostrando a viabilidade do sistema de forma prática, através do desenvolvimento de um protótipo funcional para este subsistema.

5. Agradecimentos

Os autores agradecem ao MCT, FINEP, FUNTTEL, CNPq e a CAPES por financiarem esta pesquisa.

Referências

- ADL. Advanced Distributed Learning Initiative. *Sharable Content Object Reference Model (SCORM) Version 1.3: The SCORM Overview*. Alexandria: ADLnet, 2001. Disponível em: <<http://www.adlnet.org>>. Acesso em: out/2012.
- BARBONE, V.G., RIFON, L.A. From SCORM to Common Cartridge: A step forward, 2009. *Computers & Education*, 2010.88-102.
- BALL, S., TENNEY, J. Xerte – A User-Friendly Tool for Creating Accessible Learning Objects, 2008. *Lecture Notes in Computer Science* 5105, pp. 291 a 294.
- BEZ, M., VICARI, R. M., SILVA, J., RIBEIRO, A., GLUZ, J. C., PASSERINO, L. , SANTOS, E., PRIMO, T., ROSSI, L., BEHAR, P., FILHO, R. Proposta Brasileira de Metadados para Objetos de Aprendizagem Baseados em Agentes (OBAA). *RENOTE*, v. 8, p. 1-10, 2010
- CAIRE, G., GOTTA, D., BANZI, M. WADE: A software platform to develop mission critical applications exploiting agents and workflows. *Procs. of AAMAS'08*. Portugal: 29-36.
- GLUZ, J.C. e VICARI, R. MILOS: Infraestrutura de Agentes para Suporte a Objetos de Aprendizagem OBAA. *Anais do SBIE 2010*, João Pessoa, 2010.
- GLUZ, J. C.; VICARI, R. Uma Ontologia OWL para Metadados IEEE-LOM, Dublin-Core e OBAA. *Anais do SBIE 2011*, Aracaju, 2011. v. 1. p. 204-213.
- GLUZ, J.C., VICARI, R e PASSERINO, L. An Agent-based Infrastructure for the Support of Learning Objects Life-Cycle. *Procs. of ITS 2012*, Chania, Creta, 2012. *Lecture Notes in Computer Science*. New York: Springer, 2012. v. 7315. p. 691-693.
- IEEE Learning Technology Standards Committee (LTSC). *Standard for Learning Object Metadata, IEEE Standard 1484.12.1*. Nova York, Institute of Electrical and Electronics Engineers, 2002.
- JENNINGS, N.; SYCARA, K.; WOOLDRIDGE, M. A Roadmap of Agent Research and Development. *Journal Autonomous Agents and Multi-Agent Systems*, Holanda, 1998.
- KUNZE, J.; BAKER, T. *The Dublin Core Metadata Element Set: RFC 5013*. California: IETF, 2007.
- SILVEIRA, E. L., GLUZ, J., Sistema LINNAEUS: apoio inteligente para a catalogação e edição de metadados de objetos de aprendizagem. *Anais do SBIE 2012*, Rio de Janeiro, 2012.
- SMITH, M., BASS, M., MCCLELLAN, G., TANSLEY, R., BARTON, M., BRANSCHOFKY, M., STUVE, D., WALKER, J. H. DSpace An Open Source Dynamic Digital Repository, *D-LIB Magazine.*, 2003.
- WOOLDRIDGE, M. *An Introduction to MultiAgent Systems*. 2002. John Wiley & Sons Ltd, paperback.
- WEISS G., 1999. Learning in Multiagent Systems. In G.Weiss (Ed), *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, p. 559-298. The MIT Press, Cambridge, MA.