

# Um Experimento com a Ontologia IMS LD na Construção de Modelos Conceituais para E-learning

Francisco Hélio de Oliveira<sup>1</sup>, Lais Nascimento Salvador<sup>2</sup>, Renato Lima Novais<sup>3</sup>

<sup>1</sup>Instituto Federal Baiano - Campus Itapetinga  
Itapetinga – BA – Brasil

<sup>2</sup>Departamento de Ciência da Computação – Universidade Federal da Bahia  
Salvador – BA – Brasil.

<sup>3</sup>Instituto Federal da Bahia - Campus Salvador  
Salvador – BA – Brasil

asihelio@gmail.com, laisns@dcc.ufba.br, renato@ifba.edu.br

**Abstract.** *Difficulties in the early stages of software development can negatively affect conceptual modeling. Errors at this stage (modeling) give rise to conceptual models that do not correctly represent the interests of users. In turn, domain ontologies can be used to minimize such errors. The purpose of this study is to show that the use of the IMS LD ontology can help in the construction of a more correct conceptual model for e-learning. For this, an experimental study was carried out in which the traditional approach of constructing the conceptual model was confronted with the approach that uses this ontology. The results were evaluated through statistical analysis. The main result of this study demonstrates, with statistical significance, that the use of the IMS LD ontology helps in the conceptual modeling.*

**Resumo.** *Dificuldades nas fases iniciais de desenvolvimento de software podem afetar de forma negativa a modelagem conceitual. Erros nesta fase (modelagem) dão origem a modelos conceituais que não representam corretamente os interesses dos usuários. Por sua vez, ontologias de domínio podem ser usadas para minimizar esses erros. O objetivo deste estudo é mostrar que o uso da ontologia IMS LD pode ajudar na construção de modelo conceitual mais correto para e-learning. Para tal, foi realizado um estudo experimental em que a abordagem tradicional de construção do modelo conceitual foi confrontada com a abordagem que usa essa ontologia. Os resultados foram avaliados através de análises estatísticas. O principal resultado deste estudo demonstra, com significância estatística, que o uso da ontologia IMS LD ajuda na modelagem conceitual.*

## 1. Introdução

Modelos de desenvolvimento de software são compostos de várias e diferentes etapas. Elas podem compreender desde a análise de requisitos até a manutenção do software. Como exemplos de modelos de desenvolvimento de software temos: Modelo Cascata [Royce et al. 1970], Modelo Espiral [Boehm 1988], entre outros. Independente do modelo de desenvolvimento adotado, a partir da elicitação de requisitos, o modelo conceitual do sistema de informação poderá ser construído.

A modelagem conceitual de um software é uma atividade não trivial, que se relaciona com vários desafios, como: dificuldades na elicitação de requisitos; problemas de comunicação entre o analista e o usuário; uso de documentos de requisitos que nem sempre expressam claramente as funcionalidades do sistema. Esses desafios se agravam quando o trabalho está relacionado com domínios mais complexos como bioinformática onde o acesso ao especialista pode ser mais complicado e o vocabulário usado pode ter pouco ou nenhum uso por parte do analista. Na área de educação essa questão também se verifica, através de projetos de software que devem atender padrões para ambientes de aprendizagem como o IMS Learning Design (IMS LD) [IMS 2003] e o Shareable Content Object Reference Model (SCORM) [Wisher et al. 2004] – especificações para a disponibilização de conteúdos e serviços de e-learning. Para minimizar esse problema, ontologias podem ser utilizadas por ajudar na compreensão do domínio. Uma ontologia é uma especificação explícita e formal de uma conceitualização compartilhada [Studer et al. 1998]. Segundo [Pinto et al. 2011], as ontologias estão assumindo um papel importante como elemento de origem na criação de modelos de domínio de software. De acordo com [Vasilecas et al. 2015], o uso de ontologia pode reduzir a complexidade da modelagem conceitual de sistemas de informação. Além disso, [Hesse 2005] afirma que Ontology-based Software Engineering (OBSE) pode reduzir custos na área de desenvolvimento de software. Observamos assim, por todas essas evidências teóricas, que o uso de ontologia pode ajudar na modelagem conceitual de sistema de informação.

Neste estudo, estamos considerando apenas os sistemas de e-learning. Atualmente a Internet tem desempenhado um papel significativo relacionado ao uso de tais sistemas na disseminação de conhecimento. Portanto, torna-se relevante apoiar seu desenvolvimento de forma consistente. Sendo assim, o objetivo deste estudo é mostrar dados experimentais sobre o uso da ontologia de domínio IMS LD como ferramenta de apoio na construção de diagrama de classe (perspectiva conceitual) na área de e-learning. Neste trabalho, são apresentados os resultados de um experimento com a ontologia IMS LD e sua replicação, com análise e comparação dos dados. Uma das motivações para realização da replicação do experimento foi a diferença entre as médias de corretude do primeiro experimento entre os grupos de controle (GC) e experimental (GE). No primeiro experimento [Oliveira et al. 2014], mesmo apesar de uma diferença de 44,1% a favor do GE, não houve significância estatística. Alguns pontos que prejudicaram o primeiro experimento foram detectados e as devidas correções foram feitas na replicação, a saber: i) a taxonomia da ontologia foi traduzida; ii) a realização das tarefas foi separada, com tempo de início e fim para cada uma delas; iii) os grupos foram separados durante a realização do experimento; iv) ao término de cada uma das tarefas era reforçado, resumidamente, a forma de uso da ontologia. Nesta replicação, os resultados mostraram nova vantagem para o GE, porém com significância estatística.

O artigo está organizado da seguinte forma: a Seção 2 apresenta fundamentação teórica, breve discussão sobre ontologias e modelos conceituais, os dados dos experimentos realizados e sua análise. A Seção 3 apresenta os trabalhos correlatos. A Seção 4 conclui o artigo apresentando limitações e trabalhos futuros.

## **2. Ontologia de Domínio e Modelo Conceitual**

Neste trabalho, modelo conceitual é utilizado como descrito em [Guizzardi 2005] que o define como uma abstração da realidade de acordo com uma determinada

conceitualização. Conceitualização, por sua vez, é definida como o conjunto de conceitos utilizados para articular abstrações do estado de coisas em um determinado domínio. Ontologias de domínio e modelos conceituais (diagrama de classe de análise UML) são formalismos que apresentam pontos em comum. Através do exame desses pontos serão apresentados e analisados dados acerca do uso da ontologia de domínio como ferramenta de apoio à modelagem conceitual de sistemas de informação de e-learning.

## **2.1. Fundamentação Teórica**

Podemos chamar a descrição de um modelo conceitual de esquema conceitual que basicamente descreve o conhecimento geral apresentado em um modelo conceitual. Para tentar capturar todos os requisitos de um software, de acordo com a necessidade do usuário, é essencial a criação de um modelo conceitual. A importância do esquema conceitual é mostrada em [Olivé 2007] onde, segundo o autor, a identificação do conhecimento requerido por um sistema de informação é uma atividade necessária. Além disso, os sistemas de informação não podem ser concebidos ou programados sem a obtenção prévia do conhecimento que precisa ser modelado. Assim, é necessário definir seu esquema conceitual. Em [Conesa and Olivé 2004] é mostrada a relação próxima entre ontologia e modelo conceitual quando a própria ontologia é usada como schema conceitual. É possível que um dos artefatos mais importantes na construção de software seja o modelo conceitual. Nos softwares desenvolvidos para ambientes de aprendizagem isso não é diferente, fato mostrado por [Franciosi et al. 2002]. Dada sua importância, é compreensível a busca por ferramentas para ajudar na modelagem. A ontologia de domínio é uma candidata a assumir esse papel. Neste estudo, será utilizada a ontologia IMS LD criada em OWL [W3C 2004] para o domínio de e-learning como ferramenta de apoio à modelagem.

De acordo com [Guizzardi et al. 2008], há uma aceitação de uso de ontologias na modelagem conceitual, engenharia de software, inteligência artificial e web semântica observando as peculiaridades de uso. A abordagem declarativa baseada na lógica, usada pela ontologia, permite a descrição de um domínio de conhecimento, sem fazer uma conexão com a implementação de um sistema. Assim, o conhecimento modelado é independente da plataforma de software e hardware. O uso de ontologias na modelagem conceitual é promissora, uma vez que ontologias buscam fornecer uma conceituação explícita sobre seus elementos, ajudando as pessoas a compreenderem melhor a área de conhecimento [Gava and de Menezes 2003]. Ontologias de domínio escritas em OWL e modelos de software (diagrama de classes), especificados em UML [OMG 2011], possuem características semelhantes. O padrão ODM (Ontology Definition Metamodel) [OMG 2009], faz uma correlação entre esses dois universos.

## **2.2. Ontologia IMS LD**

Uma ontologia representa um domínio por meio de seus conceitos (classes), propriedades, axiomas, hierarquia de conceitos (taxonomia de conceitos) e hierarquia de relações (taxonomia de relações). A ontologia IMS LD é baseada na especificação IMS Learning Design [IMS 2003] e apresenta um formalismo orientado a classes e relações. A especificação IMS Learning Design utiliza-se da descrição do processo educacional em unidades de aprendizagem com a ideia de um fluxo de aprendizagem que se desenvolve a partir da realização de atividades. Essas atividades são apoiadas pelo uso de materiais digitais e ferramentas de interação, definidas a partir de objetivos de aprendizagem em um

determinado planejamento didático. A IMS LD demonstra sua importância ao descrever um ambiente de aprendizagem tratando o contexto pedagógico ao modelar o processo de ensino aprendizagem em e-learning. Nesse contexto, a ontologia IMS LD foi desenvolvida com o propósito de obter uma descrição formal do padrão IMS-LD a fim de facilitar o desenvolvimento de aplicações que implementem a especificação associada, além de compartilhar este conhecimento consensual usando uma linguagem formal.

### 2.3. Processo de Modelagem Conceitual com Ontologia de Domínio

Nossa proposta mostra uma alternativa para ajudar na atividade de modelagem conceitual, assumindo-se que haja uma ontologia de domínio para a área (domínio) do sistema a ser desenvolvido. Nós propomos basicamente duas etapas de trabalho: (i) o analista obtém os requisitos do software de modo tradicional (sem ontologia) com entrevista, análise de documentos e com o auxílio de outros aplicativos tendo como resultado dessa etapa a produção do documento de requisitos; (ii) o analista aplica o processo de modelagem proposto a seguir com o documento de requisitos e a ontologia de domínio para a construção do modelo conceitual (diagrama de classe). Estas duas etapas são ilustradas na Figura 1. Neste estudo, não foi levado em consideração nenhum modelo de desenvolvimento de

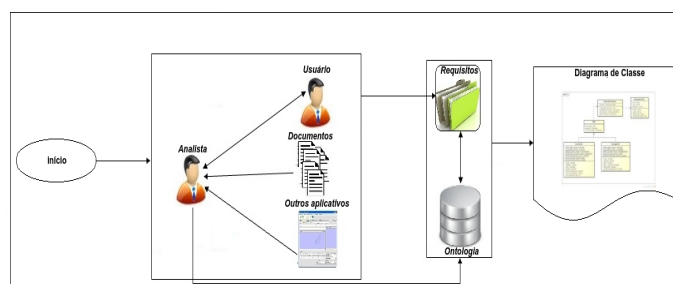


Figura 1. Modelagem conceitual com ontologia

software em particular. O foco foi na atividade (etapa) de modelagem conceitual. Nessa etapa, os analistas fazem um estudo detalhado dos dados levantados e constroem modelos a fim de representar o software a ser desenvolvido. A nossa proposta é utilizar a ontologia de domínio e o documento de requisitos para ajudar a construir o diagrama. Passos:

a) definição das classes: (i) seleciona-se os conceitos (classes) do mundo real que são relevantes para o software no documento de requisitos; (ii) verifica-se a existência da classe selecionada na ontologia. É necessário que a ontologia esteja atualizada, caso não exista a classe na ontologia ela será descartada; (iii) após selecionar as classes, cria-se um diagrama de classe preliminar apenas com classes;

b) definição das relações de herança: (i) de posse das classes, verifica-se a relação de subclasse existente entre elas na ontologia (taxonomia); (ii) caso exista, marca-se essa relação com herança; (iii) após analisar todas as classes, atualiza-se o diagrama de classe (classe e herança);

c) seleção das propriedades : verifica-se na Manchester<sup>1</sup> OWL quais propriedades

<sup>1</sup>De forma geral, vários axiomas de domínio/escopo devem ser interpretados como a interseção das classes identificadas na sequência. Em OWL uma sequência de elementos sem operador explícito representa uma conjunção/interseção. Por outro lado, múltiplas classes no domínio/escopo podem ser especificadas pela a união de classes através do construtor owl:unionOf [Horridge and Patel-Schneider 2008].

de objeto possuem apenas uma classe no domínio (domain) e apenas uma no escopo (range);

d) definição das relações de composição : (i) para cada propriedade encontrada verifica-se no documento de requisitos se existe uma relação de composição; (ii) caso exista, marca-se essa relação com composição; (iii) após analisar todas as propriedades (passo 2.3), atualiza-se o diagrama de classe (classe, herança e composição);

e) definição das relações de agregação : (i) para as propriedades restantes do item anterior verifica-se no documento de requisitos se existe uma relação de agregação; (ii) caso exista, marca-se esta relação com agregação; (iii) após analisar todas as propriedades (passo 2.3), atualiza-se o diagrama de classe (classe, herança, composição e agregação);

f) definição das relações de associação: (i) para as propriedades restantes, marca-se a relação como associação; (ii) atualiza-se o diagrama de classe com classe, herança, composição, agregação e associação;

g) finalização da análise e da construção: (i) é feita uma última análise utilizando o documento de requisitos para finalizar a construção do diagrama de classe; (ii) finaliza-se a construção do diagrama com a identificação das classes e das relações entre elas.

## **2.4. Experimentos**

Para demonstrar o uso de ontologia de domínio como ferramenta de apoio à modelagem conceitual de e-learning foi planejado e executado a replicação de um experimento controlado mostrado em [Oliveira et al. 2014]. A replicação utilizou o template Goal Question Metric GQM [Caldiera and Rombach 1994], conforme Tabela 1. É importante salientar que o GQM não foi utilizado como proposto (objetivo, questões e métricas), mas apenas para definição do Goal. Antes da realização do experimento foi estruturado e realizado um estudo piloto. Esse estudo foi aplicado a 20 estudantes de nível técnico (10 no GE e 10 no grupo de controle (GC)) e assim foi testado o “design” do experimento.

### **2.4.1. Resumo do Primeiro Experimento**

No primeiro experimento, o nível de confiança foi de 95% ( $\alpha=0,05$ ). Os resultados mostraram que a abordagem com ontologia teve uma média de acertos maior, para a corretude, que a abordagem sem ontologia culminando em uma vantagem de 44,1%. Entretanto, o t-test apresentou resultado p-value igual a 0,229 que é maior que 0,05. Isso indica que não houve significância estatística. Por outro lado, a diferença entre as médias de acerto mostrou que o uso da ontologia de domínio ajudou na construção de diagrama de classe mais correto. O tempo, que também era uma variável a ser medida, não apresentou vantagem em termos de eficiência a favor do GE. Os tempos gastos pelo GE e GC foram, respectivamente, 50,92 e 48,33 minutos. Este fato também nos motivou a fazer a replicação deste experimento.

### **2.4.2. Replicação do Primeiro Experimento**

A identificação dos objetivos deste estudo foi feita segundo o template Goal Question Metric (GQM): o objetivo do estudo foi avaliar o uso da ontologia IMS LD na modela-

**Tabela 1. Característica do estudo**

Objeto de estudo	artefato:ontologia IMS LD
Propósito	Caracterizar o impacto da adoção de ontologias na modelagem conceitual
Foco de qualidade	Corretude da modelagem e tempo de desenvolvimento do diagrama de classe
Ponto de vista	Desenvolvedor, Analista de sistema
Contexto	Análise de desenvolvimento de software e de alunos de graduação e pós-graduação além de profissionais

gem de sistemas, com o propósito de caracterizar o impacto da adoção de ontologias na modelagem conceitual, tendo como foco de qualidade a corretude da modelagem e tempo de desenvolvimento do diagrama de classe, do ponto de vista dos pesquisadores deste estudo, no contexto da análise de desenvolvimento de software por alunos e profissionais da área de computação.

Considerando a abordagem baseada em ontologias para construção de sistemas de informação, planeja-se caracterizar sua diferença com relação a abordagem não baseada em ontologia, com respeito a: corretude e eficiência. Para a corretude, foram avaliadas e contadas classes, associações, agregações, heranças e composições pertencentes ao diagrama de classe do modelo de referência (oráculo) descrito na especificação do domínio IMS Learning Design Information Model [IMS 2003]. As duas abordagens tiveram seus tempos contados, dessa forma, a eficiência pôde ser medida. O tempo total foi calculado, em minutos, considerando tempo de início e de fim de cada tarefa. O modelo criado pelos participantes foi confrontado com o modelo de referência para determinar sua corretude. A replicação do experimento foi conduzida em ambiente acadêmico. Os indivíduos, 12 estudantes, foram selecionados entre alunos do curso de Sistema de Informação da Faculdade de Tecnologia e Ciências (FTC) e Sistema de Informação do Instituto Federal de Educação, Ciência e Tecnologia da Bahia (IFBA) além de 1 técnico em informática (profissional). A alocação dos indivíduos ao GE e ao GC foi de acordo com sua experiência. Para participantes com a mesma experiência, a alocação foi feita aleatoriamente aos grupos através de sorteio. Apenas 1 dos participantes tinha experiência de 6 meses com UML e não conhecia ontologia, o restante não tinha experiência nem com desenvolvimento UML e nem com ontologia. Não houve nenhuma recompensa para participação no experimento, sendo que a motivação foi, basicamente, pelo aprendizado adquirido. Todos participantes foram convidados e, os que aceitaram, assinaram um termo de consentimento com cláusula explícita sobre a confidencialidade dos dados. Todos receberam um número de acordo com a assinatura na lista de presença e desse modo a confidencialidade foi assegurada. Os dois grupos receberam materiais diferentes. O GC recebeu apenas um texto (documento de requisitos) em português onde estava descrito todos os conceitos (classes) e relações (associação, herança, agregação e composição) pertinentes a parte do domínio do problema. Enquanto GE recebeu, além do texto, a ontologia IMS LD (taxonomia), onde constavam as classes do domínio. Além disso, esse último grupo recebeu também uma tabela na codificação Manchester de OWL com a descrição das propriedades (objectProperty) da ontologia. Os participantes desse grupo seguiram o processo descrito na subseção 2.3.

Classes presentes na ontologia: act, academic, activity, structured-activity, environment, evaluation, learn-activity, object-activity, objective-learning, learner, method, play, prerequisite, professional, role, role-part, service, staff e support-activity. Entre as classes existiam as seguintes relações: 6 associações, 6 heranças, 8 agregações, 3 composições. O experimento continha 5 tarefas, a saber: T1 - encontrar as classes corretamente como descritas no oráculo; T2 - encontrar todas as relações de herança corretamente como descritas no oráculo; T3 - encontrar todas as relações de associação corretamente como descritas no oráculo; T4 - encontrar todas as relações de composição corretamente como descritas no oráculo e T5 - encontrar todas as relações de agregação corretamente como descritas no oráculo.

Para a replicação do experimento duas hipóteses foram levantadas: H1) a abordagem baseada em ontologia produz artefato mais correto que a abordagem sem ontologia; H2) a abordagem baseada em ontologia diminui o tempo gasto na construção do modelo conceitual. Desejava-se medir o quanto o diagrama de classe obtido a partir do texto ou do texto e ontologia estava correto. As hipóteses nulas foram as negações de cada uma delas. As variáveis independentes foram: texto em linguagem natural e ontologia. As variáveis dependentes foram: corretude (número de classes e relações identificadas corretamente) e eficiência (medida através do tempo). Para determinar a corretude foi feito o seguinte: (i) contagem do número de elementos encontrados corretamente (EC) para cada tarefa; (ii) contagem do número de elementos encontrados incorretamente (EI) para cada tarefa; (iii) e, por fim, subtração do EC por EI. O número encontrado indicava o grau de corretude alcançado pelo participante em uma tarefa. A corretude do diagrama foi dada pela soma das corretudes das tarefas. Essa métrica foi definida de acordo com [Novais et al. 2012]. O tempo (minuto) foi contado levando-se em consideração cada uma das tarefas. Cada participante teve uma hora de início (HI) e fim (HF) e assim o tempo foi calculado subtraindo-se HF da HI. Portanto, o tempo total gasto pelo participante foi calculado somando-se o tempo de cada tarefa. Para fins de treinamento, uma aula de 50 minutos foi ministrada no horário reservado a uma disciplina na instituição onde o experimento foi executado, cobrindo fundamentos de UML e de ontologia. Além de mostrar o uso da ontologia como ferramenta de apoio à modelagem.

Foi iniciada a replicação do experimento que tinha duração máxima prevista para 100 minutos. Procedimentos: os participantes assinaram a lista de presença; ii) foram distribuídos os artefatos aos participantes de acordo com o grupo, além da folha de resposta e do Termo de Consentimento; iii) o termo de consentimento foi lido, assinado e recolhido; iv) foi feita uma leitura dos artefatos; v) foi proposto como objetivo: obtenção do diagrama de classe que representava o modelo conceitual do Learning Design com classes e relações; vi) cada participante executou as tarefas T1, T2, T3, T4 e T5 com o tempo sendo cronometrado e ao final da T5 o diagrama de classe ficou esboçado na folha de resposta, onde constava o identificador do participante; vii) após a entrega da folha de resposta o participante preencheu o questionário de feedback.

## **2.5. Resultados e Análise**

O planejamento da replicação foi alterado por pequenas mudanças: i) a taxonomia da ontologia foi traduzida; ii) as tarefas foram separada com tempo de início e fim para cada uma delas; iii) os grupos foram separados durante a realização do experimento; iv) ao término de cada tarefa era reforçado, resumidamente, a forma de uso da ontologia.

Corretude							
Grupo	Média	min	max	Desvpad	SW p-value	Levene p-value	t.test
GE	13,830	7	24	5,98	0,68	0,690	0,003
GC	0,850	-5	10	6,517	0,0867		

**Figura 2. Resultado da replicação para variável corretude**

Tempo				
Grupo	Média	df	min	max
GE	55,200	35,62%	43	74
GC	40,700		18	68

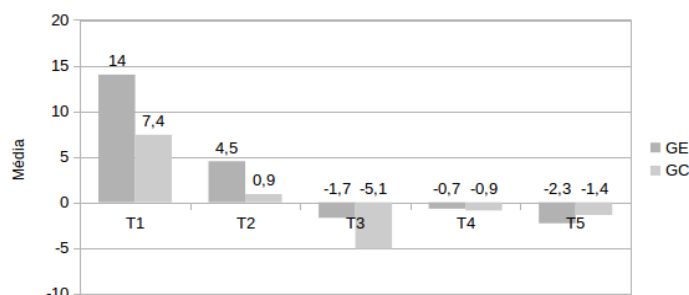
**Figura 3. Resultado da replicação para variável tempo**

Os alunos da FTC e do IFBA realizaram o experimento no laboratório de informática da FTC e o técnico utilizou o laboratório de informática do Instituto Federal de Educação, Ciência e Tecnologia Baiano. Utilizando a técnica de interquartil, não foram detectados outliers em nenhuma das amostras. As amostras de corretude para os grupos se evidenciaram, respectivamente, normais e com variância igual, como demonstram os testes de Shapiro-Wilk para o GE (0,68), GC (0,0867) e Levene (0,690), mostrados na Figura 2. Em correspondência com os dados foi aplicado o t-test para a hipótese H1 sendo o p-value(0,003) menor alpha(0,05) o que indica a não rejeição da hipótese H1. Assim, podemos afirmar com 95% de confiança que a ontologia ajuda na criação de diagrama de classe mais correto. A melhora do resultado pode ser explicada por basicamente dois fatores: (i) garantia de uso da ontologia pelo GE em cada uma das tarefas, através da divisão e acompanhamento das tarefas; (ii) o documento de requisitos pode ter se mostrado simples para os participantes do primeiro experimento e assim ter preterido o uso da ontologia. Enquanto pode ter se mostrado complexo para os do segundo e, por conseguinte, incentivado o uso da ontologia. Em ambos os casos o nível de experiência pode ter influenciado. A hipótese H2 não se confirmou, pois a média de tempo gasto pelo GE foi maior que o GC como mostra a Figura 3 e por esse motivo a estatística não foi calculada. Observa-se nos resultados do primeiro experimento e da replicação, que os resultados são convergentes com relação à corretude. A Seção 2.4.1 mostra que o primeiro experimento não obteve relevância estatística, porque o p-value (t test) foi maior que 0,05, mas apresentou média de acertos maior para o GE. A subseção 2.4.2 confirma o resultado da subseção 2.4.1 e, além disso, mostra estatisticamente a significância (p-value menor que 0,05) do resultado do experimento replicado. O resultado da média de tempo também converge em ambos experimentos mostrando que o GE gastou mais tempo.

O experimento realizado tem como objetivo final verificar se a ontologia de domínio ajuda a modelagem conceitual através da produção de diagrama de classe mais correto. Pode haver o argumento de que o uso da ontologia já indicaria uma melhora nos resultados, uma vez que o GE tem mais recursos. Entretanto, isso precisa ser mostrado experimentalmente, justificando assim o desenvolvimento deste trabalho. Os dados do experimento são especialmente importantes por mostrar a influência do uso da ontologia sobre a modelagem de sistemas de informação de e-learning. A Figura 4 apresenta a média da corretude da replicação do experimento por grupo. Essa média é a soma das corretudes de cada tarefa realizada por cada participante, dividida pelo número de participantes. Dessa forma, conseguimos mostrar quantos elementos (classe, associação,



herança, agregação ou composição) foram identificados corretamente pelo grupo. Como exemplo podemos citar o elemento classe da tarefa T1, onde o GE obteve em média 14 elementos identificados corretamente enquanto o GC obteve 7,4. Os dados mostram que



**Figura 4. Média das tarefas por grupo**

a média da corretude para as tarefas T1, T2, T3 e T4 são maiores para o GE. Evidenciado um maior número de elementos UML (classe e relações) identificados corretamente. Podemos observar que as tarefas T3, T4 apresentaram resultados negativos e mesmo assim o GE obteve vantagem por errar menos. Apenas a tarefa T5 apresenta média maior para o GC. Sendo assim, este estudo comprova que a ontologia IMS LD ajuda na modelagem conceitual de projetos educacionais mais corretos relacionados aos padrões de e-learning. Além disso, para analista com pouco ou nenhum conhecimento do domínio, a ontologia ajuda na compreensão desse domínio. Esses dois fatores talvez possam contribuir para construção de modelos conceituais de e-learning.

### 2.5.1. Validade

Os possíveis riscos à validade deste experimento foram analisados e tratados de forma a serem minimizados. Validade interna: a experiência dos participantes foi levada em consideração, pois poderia influenciar a corretude no desenvolvimento do diagrama de classe. Esse problema foi minimizado colocando os participantes nos grupos de acordo com sua experiência que foi previamente obtida através de formulário. Participantes com a mesma experiência foram alocados a grupos diferentes. Validade externa: os participantes foram selecionados a partir de alunos de graduação de uma Universidade e um Instituto. Além de um profissional técnico em programação. Essa amostra pode não ser representativa.

## 3. Trabalhos Relacionados

Modelos baseados em UML e ontologias OWL se constituem em abordagens com diferentes pontos fortes e fracos. Isso os tornam adequados para a especificação de aspectos diferentes de software. Dessa forma, o autor propõe uma utilização integrada de ambas as abordagens (UML e OWL) de modelagem. Nesse estudo, o autor apresenta um framework com uma abordagem que permite a modelagem UML com expressividade semântica da OWL DL [Parreiras and Staab 2010].

Segundo [Tavares et al. 2009], é possível melhorar um diagrama de classe existente através de um procedimento, o PrOntoCon, que visa realizar análise ontológica do

diagrama de classe. Dessa forma, ajuda o analista nos estágios iniciais de desenvolvimento de software. O PrOntoCon traz a vantagem do analista não precisar estar familiarizado com conceitos filosóficos sofisticados que fundamentam o procedimento. Em [Pinto et al. 2011] é discutido o uso de ontologia como ferramenta de apoio à modelagem. Eles apresentam a ferramenta MDAOnto. Esta ferramenta recebe como entrada uma ontologia. O usuário pode interagir para remover conceitos sem relevância para o sistema a ser desenvolvido. Em seguida essa ontologia passa por uma transformação OWL2UML obtendo-se um diagrama de classe. Em [El-Ghalayini et al. 2006] os autores mostram a transformação de uma ontologia de domínio OWL em um diagrama de classe UML através do algoritmo Transformation-Engine (TE). Na prática o algoritmo TE é aplicado à ontologia e dessa forma os conceitos e regras relevantes dessa ontologia são selecionados gerando um modelo conceitual. O artigo analisa o modelo gerado de duas perspectivas quantitativa e qualitativa analisando a qualidade através da precisão do modelo.

Pode ser visto que existem estudos relacionando UML e ontologias no sentido de otimizar, melhorar a criação de modelos conceituais nas fases iniciais de desenvolvimento de software. Basicamente o que propomos é utilizar uma ontologia de domínio como ferramenta de apoio à modelagem conceitual. Todos os estudos citados usam ontologia para melhorar o desenvolvimento de software, entretanto, nenhum deles discute a relação entre ontologia e desenvolvimento (modelo conceitual). Este estudo se diferencia dos demais por apresentar, discutir e analisar dados experimentais sobre a corretude de um diagrama de classe quando se usa ontologia de domínio como apoio.

#### **4. Conclusões**

Com o aumento da complexidade dos softwares educacionais, da diversidade de tecnologias adotadas e do grande número de pessoas envolvidas, tornou-se inadequado projetar um programa educacional sem utilizar-se de um processo bem definido para orientar o seu desenvolvimento [Marczak et al. 2003]. Nesta linha, o presente estudo mostra o uso da ontologia IMS-LD como apoio no processo de desenvolvimento de software para a área educacional. Este trabalho faz uma análise e discute a viabilidade da ontologia de domínio para a modelagem de sistemas de informação, tendo em vista que o modelo conceitual é um dos artefatos mais importantes para efetiva implementação de software.

Com base nos estudos experimentais apresentados, podemos afirmar que o uso de ontologias de domínio podem ajudar na modelagem conceitual de sistemas de informação, mais especificamente sistemas para e-learning, pois: (i) a identificação de classe de interesse do sistema pode ficar mais clara; (ii) as relações de herança estão descritas de forma explícitas na taxonomia da ontologia facilitando sua identificação; (iii) as relações de associação, agregação e composição podem ser mapeadas através da propriedade de objeto da ontologia. As duas últimas – agregação e composição - podem apresentar maior dificuldade de serem identificadas, pois não estão explicitamente descritas em OWL, mas observou-se que o uso combinado da ontologia com o documento de requisitos tornou mais fácil para o analista verificar qual é o relacionamento (associação, agregação ou composição) em questão. O experimento avaliado em [Oliveira et al. 2014] mostrou indícios de que o uso de ontologia pode ajudar na criação de diagramas de classe (modelo conceitual). O presente trabalho, mostrou que os resultados do primeiro experimento e da sua replicação são convergentes e, adicionalmente, demonstrou que a replicação apre-

sentou relevância estatística através de teste de hipótese. Este trabalho também contribui na busca por uma abordagem experimental na área de Ontology-based Software Engineering, pois: (i) provê dados experimentais que podem indicar o uso da ontologia como uma maneira interessante de construir ambientes virtuais de aprendizagem; (ii) mostra que a ontologia IMS-LD constitui-se em uma ferramenta auxiliar na engenharia de software educacional; (iii) indica também que a mesma ontologia pode ser um artefato de aprendizado do domínio para analistas com pouca experiência na área de e-learning.

Como fatores limitantes, podemos citar: i) a ausência de outros elementos que compõem um diagrama UML, por exemplo, a relação de dependência; ii) o uso de apenas uma ontologia: a IMS LD; iii) somente a avaliação do diagrama de classe de análise UML como modelo conceitual; iv) não tratamento de outros elementos da ontologia como os axiomas. Como trabalhos futuros, pretende-se realizar experimentos para sistemas de informação de outra natureza como os sistemas de informação gerencial (sig), e assim elucidar algumas questões de pesquisa relacionadas às limitações deste trabalho.

## Referências

- Boehm, B. W. (1988). A spiral model of software development and enhancement. *Computer*, 21(5):61–72.
- Caldiera, V. R. B.-G. and Rombach, H. D. (1994). Goal question metric paradigm. *Encyclopedia of Software Engineering*, 1:528–532.
- Conesa, J. and Olivé, A. (2004). Pruning ontologies in the development of conceptual schemas of information systems. In *International Conference on Conceptual Modeling*, pages 122–135. Springer.
- El-Ghalayini, H., Odeh, M., McClatchey, R., and Arnold, D. (2006). Deriving conceptual data models from domain ontologies for bioinformatics. In *Information and Communication Technologies, 2006. ICTTA'06. 2nd*, volume 2, pages 3562–3567. IEEE.
- Franciosi, B. R., de Medeiros, M. F., Vargas, R. F., Pernigotti, J. M., Marques, J. C., and Colla, A. L. (2002). Experiência de modelagem de ambientes de aprendizagem da pucrs virtual. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 1, pages 178–183.
- Gava, T. B. S. and de Menezes, C. S. (2003). Uma ontologia de domínio para a aprendizagem cooperativa. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 1, pages 336–345.
- Guizzardi, G. (2005). *Ontological foundations for structural conceptual models*. PhD thesis, University of Twente.
- Guizzardi, G., Falbo, R., and Guizzardi, R. S. S. (2008). A importância de ontologias de fundamentação para a engenharia de ontologias de domínio: o caso do domínio de processos de software. *Revista IEEE América Latina*, 6(3):244–251.
- Hesse, W. (2005). Ontologies in the software engineering process. In *EAI*, pages 3–16.
- Horrige, M. and Patel-Schneider, P. F. (2008). Manchester syntax for owl 1.1. *OWL: Experiences and Directions, Washington, DC*.

- IMS (2003). Ims learning design information model - version 1.0 final specification. [https://www.imsglobal.org/learningdesign/ldv1p0/imsld\\_infov1p0.html](https://www.imsglobal.org/learningdesign/ldv1p0/imsld_infov1p0.html).
- Marczak, S., Giraffa, L., Almeida, G., and Blois, M. (2003). Modelando um ambiente de aprendizagem na web: a importância de formalização do processo de desenvolvimento. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 1, pages 535–544.
- Novais, R., Nunes, C., Lima, C., Cirilo, E., Dantas, F., Garcia, A., and Mendonça, M. (2012). On the proactive and interactive visualization for feature evolution comprehension: An industrial investigation. In *Proceedings of the 34th International Conference on Software Engineering*, pages 1044–1053. IEEE Press.
- Oliveira, F. H., do Nascimento Salvador, L., and Novais, R. (2014). Uma análise do uso da ontologia ims ld na construção de modelos conceituais para e-learning. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 25, page 1213.
- Olivé, A. (2007). *Conceptual Modeling of Information Systems*. Springer Verlag NY, 15th edition edition.
- OMG (2009). Ontology definition metamodel - version 1.0. <http://www.omg.org/spec/ODM/1.0>.
- OMG (2011). Omg unified modeling language (omg uml), infrastructure. <http://www.omg.org/spec/UML/2.4.1/Infrastructure/>.
- Parreiras, F. S. and Staab, S. (2010). Using ontologies with uml class-based modeling: The twouse approach. *Data & Knowledge Engineering*, 69(11):1194–1207.
- Pinto, G. P., Salvador, L. N., and Amorim, R. J. R. (2011). A model driven solution based on ontology in the implementation of units of learning authoring tools. *IEEE Technology and Engineering Education (ITEE)*, 6(3):31–49.
- Royce, W. W. et al. (1970). Managing the development of large software systems. In *proceedings of IEEE WESCON*, volume 26, pages 1–9. Los Angeles.
- Studer, R., Benjamins, V. R., and Fensel, D. (1998). Knowledge engineering: principles and methods. *Data & knowledge engineering*, 25(1-2):161–197.
- Tavares, D. B., de Paiva Oliveira, A., Braga, J. L., and Lisboa Filho, J. (2009). Analysis procedure for validation of domain class diagrams based on ontological analysis. In *International Conference on Conceptual Modeling*, pages 159–168. Springer.
- Vasilecas, O., Kalibatiene, D., and Guizzardi, G. (2015). Towards a formal method for the transformation of ontology axioms to application domain rules. *Information Technology and Control*, 38(4).
- W3C (2004). Web ontology language reference. <https://www.w3.org/TR/2004/REC-owl-ref-20040210/>.
- Wisher, R. A., Fletcher, J. D., Barrett, J. W., Danylova, O., Garza, P., Mihalka, B., Synytsya, K., Voychenko, O., Ivashchenko, A., Christman, W., et al. (2004). Advanced distributed learning. *Information & Security*, 14.