

## 4 Tratamento de Heterogeneidade dos Metadados

Além da própria arquitetura de integração é importante considerar o tratamento da heterogeneidade dos metadados. Atualmente, há uma variedade de conjuntos de metadados relacionados a *e-learning*, incluindo padrões diferentes. Essa heterogeneidade dificulta a interoperabilidade de recursos e, conseqüentemente, o compartilhamento de LOs. Esta dissertação propõe uma arquitetura capaz de operar entre diferentes conjuntos de metadados para descrição de LOs e suas diversas implementações (modelos, meta-esquemas e esquemas). Ao longo deste capítulo, é apresentada uma técnica de mapeamento entre meta-esquemas/padrões de metadados visando à interoperabilidade entre os diferentes repositórios de LOs.

### 4.1. Meta-esquemas e Padrões de Metadados para LOs

Existem diversos padrões para representação de metadados educacionais, dentre os quais são utilizados no protótipo deste trabalho os seguintes: IEEE-LOM, Dublin Core e Ariadne. Esses padrões e outros meta-esquemas provêm representações distintas para os metadados de um LO. LORIS permite que as fontes de dados sejam descritas por qualquer meta-esquema de metadados para LOs. Tal fato, o obriga a fazer um tratamento da heterogeneidade entre esses meta-esquemas/padrões de metadados, visando a integração das fontes.

Para solucionar o problema da heterogeneidade é necessário, primeiramente, entender os conceitos associados a cada um dos elementos representados nos meta-esquemas/padrões de metadados e, a partir disso, estabelecer as possíveis relações de equivalência entre os mesmos.

A descrição (metadados) de um LO reúne os elementos necessários para sua organização e facilitar sua recuperação. Esses elementos podem ser analisados sob duas visões: *estrutural* e *semântica*. A visão estrutural representa a informação que descreve a organização e estrutura dos dados (dado sobre o dado),

por exemplo, informações sobre o formato, os tipos de dados usados, etc. A visão semântica fornece as informações sobre o significado (conceito) dos dados disponíveis e seus relacionamentos semânticos [72].

Muitas vezes um conceito pode estar associado a mais de um elemento de metadados como é o caso do conceito “autor” que no padrão IEEE LOM é representado pela combinação de dois elementos: *lifecycle.contribute.entity* e *lifecycle.contribute.role*. O conceito “autor” é especificado no padrão como o elemento *lifecycle.contribute.entity*, quando *lifecycle.contribute.role* tiver valor igual a “*author*”. Este mesmo conceito é especificado pelo elemento *general.authors*, no padrão Ariadne.

Conforme exemplificado anteriormente, um meta-esquema/padrão de metadados pode exprimir um mesmo conceito de forma estruturalmente distinta de um outro meta-esquema/padrão. A partir de uma análise minuciosa dos diferentes elementos representados pelos padrões de metadados propostos para descrição de LOs, pôde-se verificar a necessidade de um "entendimento semântico" entre os conceitos expressos pelos mesmos. A estratégia utilizada em LORIS para obter esse entendimento faz uso do formalismo de ontologias para descrever os conceitos necessários às descrições de LOs e encontrar as possíveis equivalências entre os meta-esquemas/padrões de metadados (mapeamentos).

#### **4.2. Mapeamento entre Modelos**

A construção de um mapeamento entre diferentes meta-esquemas de descritores de LOs constitui uma tarefa árdua e propensa a erros, uma vez que esses metadados têm sido freqüentemente desenvolvidos de uma forma independente.

Como mencionado anteriormente, os elementos que compõem os meta-esquemas de metadados podem ser analisados estruturalmente e semanticamente, sendo assim, o mapeamento entre conjuntos de metadados distintos pode ser dividido em *mapeamento estrutural* e *mapeamento semântico*, respectivamente. As subseções seguintes explicam os detalhes envolvidos em cada mapeamento.

### 4.2.1. Mapeamento Semântico

No processo de mapeamento semântico são observadas as informações referentes ao significado (conceito) dos dados educacionais, assim como seus relacionamentos. Na visão semântica pretende-se encontrar correspondências entre os conceitos expressos pelos meta-esquemas de metadados.

Ao longo dos anos, diversos modelos semânticos têm sido propostos com o intuito de representar os diferentes domínios de conhecimento. Algumas propostas são Taxonomias, Tesouros, Mapas de Tópicos (*Topic Maps*), Modelo Entidade Relacionamento (ER), Redes Semânticas e Ontologias.

As taxonomias representam uma classificação ou categorização de forma hierárquica. Numa taxonomia, as entidades são representadas através de relações de generalização e especialização, o que leva a construção de modelos de semântica relativamente pobre [73].

Um tesouro constitui uma estrutura de controle de vocabulário, na qual os termos ou conceitos podem estar relacionados por meio de relações de hierarquia, associação e equivalência. [74].

Mapas de Tópicos (*Topic Maps*) constituem um paradigma que permite a organização, manutenção e navegabilidade de informação, criando para isso, um mapa virtual de informação, sem que seja necessário modificar a informação original [75].

Uma ontologia consiste de uma especificação formal e explícita da conceitualização compartilhada de um determinado domínio de conhecimento [76]. A especificação compreende a descrição de classes de conceitos, propriedades, relações, funções, restrições, axiomas que são explicitamente definidos.

Com o advento das pesquisas na área da Web Semântica, o estudo de ontologias passou a ser um dos principais focos de interesse. O desenvolvimento de especificações e linguagens de consulta tornou viável a implementação de ontologias como estratégia para prover interoperabilidade semântica.

Este trabalho utiliza ontologias para representação semântica dos padrões de metadados educacionais, devido, principalmente, ao alto poder de expressividade; à possibilidade de utilizá-las como veículo de formalização dos

conceitos, além das razões que viabilizam sua implementação, citadas no parágrafo anterior.

Desta forma, o mapeamento semântico obtido em LORIS traduz-se como um mapeamento inter-ontologias, o qual será discutido em detalhes na seção 4.3.

#### **4.2.2. Mapeamento Estrutural**

No mapeamento estrutural faz-se necessário definir regras de mapeamento ou cruzamento entre os diferentes esquemas de metadados, originando os modelos de correspondências. Dá-se o nome de mapeamento de esquema ao processo de associação de elementos de um esquema com elementos de outro esquema.

Segundo [77], o mapeamento estrutural entre padrões de metadados pode ser classificado em: mapeamento de elementos e de mapeamento de valores.

##### **4.2.2.1. Mapeamento de Elementos**

Nesta visão de mapeamento, os elementos do modelo de metadados origem são traduzidos para os elementos equivalentes no modelo destino. Esse processo pode apresentar-se das seguintes formas:

- **um-para-um (1-1)**: um mapeamento 1-1 acontece quando um elemento de um modelo de metadados tem a mesma característica de um elemento de outro modelo de metadados. Assim, este elemento é mapeado diretamente do modelo origem para um elemento correspondente no modelo destino.
- **um-para-muitos (1-N)**: um mapeamento 1-N acontece quando um elemento do modelo de metadados origem pode ser mapeado para mais de um elemento do modelo de metadados destino e não há qualquer perda de informação, preservando assim, a semântica do metadado. É importante destacar que questões como multiplicidade e restrições de tipos de dados devem ser cuidadosamente tratadas durante o processo de mapeamento.
- **muitos-para-um (N-1)**: mapeamento de mais de um elemento do modelo de metadados origem para um elemento do modelo de metadados destino. Vale lembrar que no mapeamento N-1, os elementos do modelo origem podem, expressar um nível semântico maior que o nível suportado pelo

elemento do modelo destino. Dessa forma, pode-se ocasionar perda de informação.

- **Mapeamento Hierárquico:** Um tipo especial de mapeamento de elementos de dados acontece quando os elementos possuem um relacionamento hierárquico ou um relacionamento pai-filho. Esses elementos devem ser mapeados de forma que preservem a semântica de sua hierarquia.

#### 4.2.2.2.

#### **Mapeamento de Valores de Elementos**

Este tipo de mapeamento ocorre quando um elemento do modelo origem possui um valor pertencente a uma lista fixa (pré-determinada), a qual não encontra correspondência direta com a lista de valores do elemento do padrão destino. O mapeamento de valores também ocorre na formas 1-1, 1-N e N-1, conforme apresentado anteriormente. É possível também que os valores encontrados na lista apresentem um tipo de dado diferente da lista encontrada no modelo de metadados correspondente.

#### 4.3.

#### **Mapeamento no LORIS**

Inicialmente, o processo realizado para obter-se o mapeamento entre os repositórios de metadados que compõem uma comunidade de *e-learning* pode ser dividido em duas etapas. A primeira etapa consiste de um estudo minucioso das especificações de cada conjunto de metadados, gerando o que denominamos de representação estrutural, esse tipo de representação corresponde a um esquema mais próximo da especificação, sem maiores abstrações. A segunda fase corresponde à representação semântica dos conceitos extraídos da especificação de cada padrão, reunindo abstrações descomprometidas com detalhes de esquema (estrutura).

A representação estrutural é descrita pela linguagem de descrição de vocabulário *RDF Schema* (RDFS) [78], uma linguagem utilizada para escrever vocabulários a serem usados em instâncias RDF [79]. A representação semântica é descrita pela linguagem de ontologia OWL *Web Ontology Language* [80]. A

linguagem OWL emerge como um padrão recomendado pelo W3C para a descrição de ontologias. A OWL visa prover uma linguagem voltada para a descrição de classes e relacionamentos que são inerentes a documentos *web* e aplicações. Esta linguagem pode também ser usada para formalizar um domínio através da definição de classes e suas propriedades; definir indivíduos e afirmar propriedades sobre eles, além de prover raciocínios lógicos sobre estas classes e indivíduos de acordo com o grau permitido pela semântica formal da OWL.

As representações estrutural e semântica dos meta-esquemas/padrões são detalhadas nas subseções 4.3.1 e 4.3.2, respectivamente. Ao longo do texto, o termo “ontologia” é tratado como representação semântica, enquanto o termo “esquema” refere-se à representação estrutural. Faz-se uma ressalva ao termo ontologia, o qual pode ser atribuído, num escopo mais amplo, ao próprio esquema de definição. Neste caso, considera-se o esquema como uma ontologia com menos semântica.

#### **4.3.1. Representação Estrutural**

Cada meta-esquema/padrão de metadados possui uma especificação, na qual são estabelecidos quais atributos/tipos o compõem, além dos conjuntos fixos de valores para alguns atributos. Os detalhes descritos na especificação de cada meta-esquema/padrão, assim como sua estrutura organizacional são representados num “esquema” correspondente, compondo a representação estrutural.

A figura 4.1 apresenta uma visão da representação estrutural dos padrões Ariadne, LOM e DC. Essa visão é apenas um subconjunto de cada padrão, considerando os atributos “autor” e “requisito”. Esses atributos serão utilizados nos exemplos apresentados nas próximas seções.

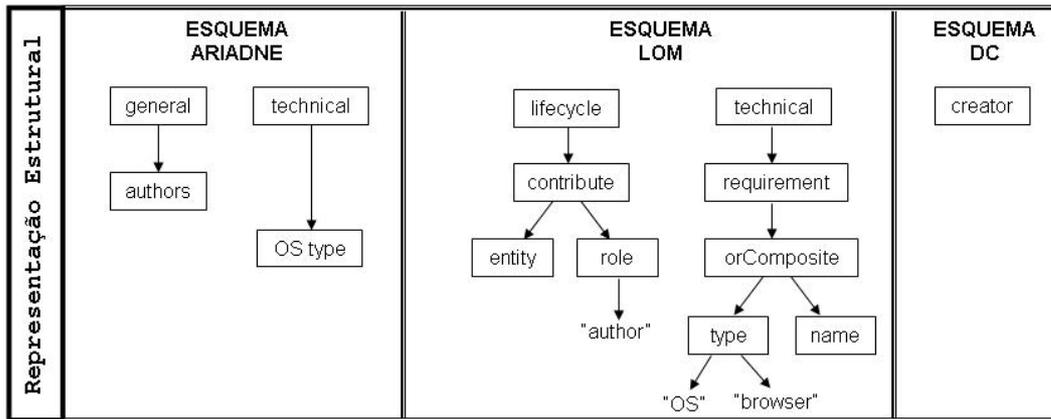


Figura 4.1: Representação estrutural dos padrões

Na figura 4.1, os atributos correspondem aos retângulos e as setas indicam o relacionamento de composição entre eles. Cada atributo possui uma série de considerações, tais como: cardinalidade, tipo de dados, ordenação, etc. Esses aspectos foram desprezados por questões de simplicidade e focamos na forma como os atributos estão estruturados. Percebe-se que os meta-esquemas/padrões utilizam diferentes estruturas para representar um mesmo conceito (autor, por exemplo).

Em LORIS é gerado um esquema em RDFS para cada meta-esquema/padrão de metadados para LOs utilizado. Cada “esquema” criado serve como referência para as fontes que armazenam LOs representados no meta-esquema/padrão em questão. Os tradutores (*wrappers*) dessas fontes são responsáveis pelo processamento do esquema.

Tomando como base a representação estrutural dos meta-esquemas/padrões, foram construídas suas respectivas visões semânticas, a partir da abstração dos conceitos envolvidos. A seção seguinte trata desse assunto.

#### 4.3.2. Representação Semântica

Sendo definida a representação estrutural dos meta-esquemas/padrões de metadados, o passo seguinte consiste na elaboração da representação semântica dos mesmos. A representação semântica expressa os conceitos presentes em cada meta-esquema/padrão de metadados.

O modelo semântico utilizado para prover a representação semântica dos padrões corresponde à “ontologia”. Neste caso, deve ser criada uma ontologia

para representar cada meta-esquema/padrão com o intuito de se obter uma descrição semântica do mesmo. A construção de cada ontologia é obtida a partir de um estudo minucioso das especificações de cada meta-esquema/padrão de metadados, objetivando-se fazer um levantamento dos conceitos existentes em cada um deles.

A partir das ontologias geradas é construída uma “ontologia comum” capaz de refletir uma visão generalizada dos conceitos presentes nas demais ontologias. A ontologia comum expressa o modelo global da arquitetura proposta, sobre o qual as consultas dos usuários são submetidas.

A figura 4.2 ilustra a representação semântica de padrões de metadados para LOs, juntamente com sua representação estrutural, considerando os conceitos “autor” e “requisito”. A especificação do padrão é espelhada em sua representação estrutural. O conceito “autor” refere-se aos criadores do material didático em questão. Considerando-o como exemplo, há uma representação estrutural distinta do mesmo em cada padrão. Na especificação do padrão Ariadne, o conceito “autor” equivale ao atributo *authors*, o qual corresponde a uma lista de autores do LO (no formato *vcard*), que faz parte da categoria *general*. Na especificação do padrão IEEE LOM, *lifecycle* corresponde à categoria que descreve a história do LO ao longo de sua evolução. O atributo *contribute* representa as entidades que afetaram o LO, ao longo do tempo. Cada *contribute* compreende os atributos *role* (tipo de contribuição) e *entity* (identificação da entidade). Desta forma, o conceito “autor” corresponde ao atributo *entity*, desde que *role* tenha valor igual a “*author*”. Na especificação DC, o conceito “autor” corresponde apenas ao atributo *creator*.

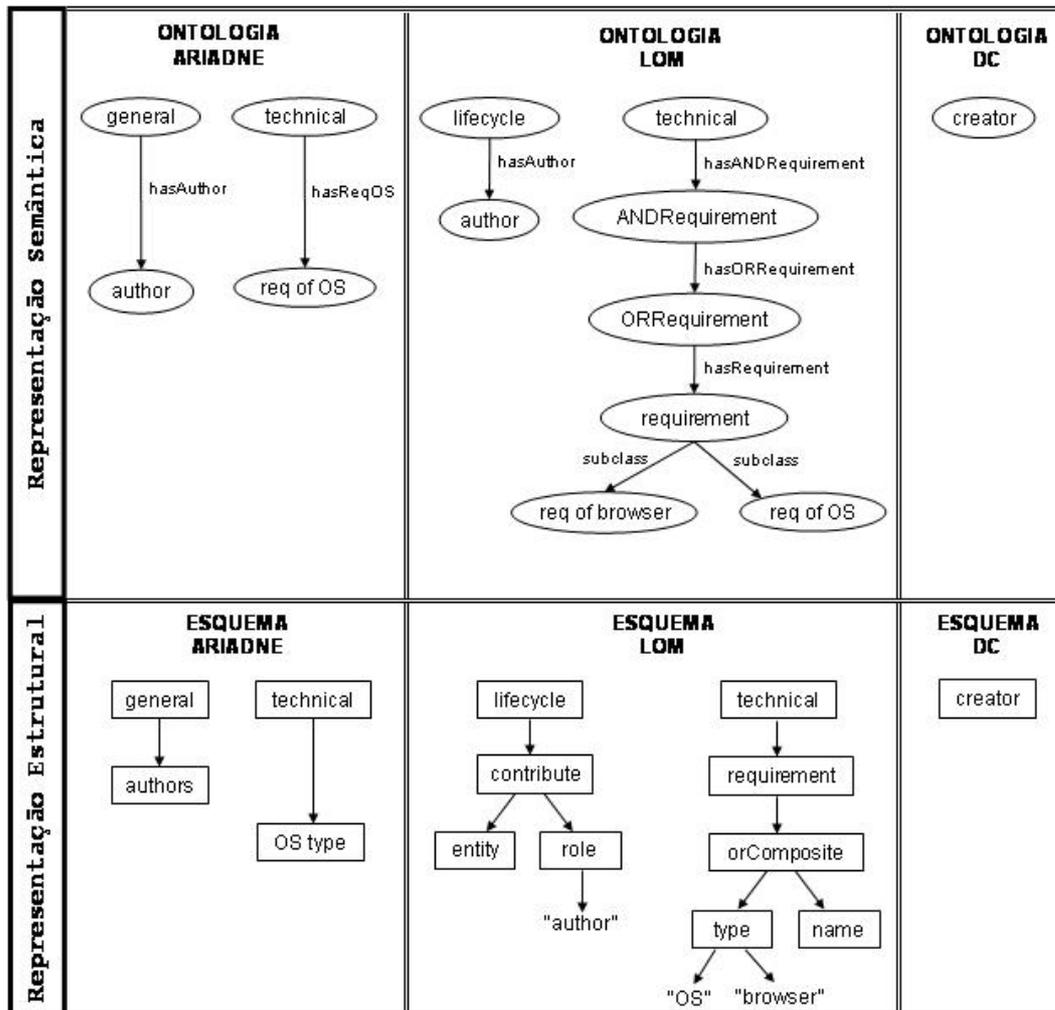


Figura 4.2: Representação semântica e estrutural dos padrões

O conceito “requisito” refere-se às necessidades técnicas necessárias para o uso do LO. Na especificação do padrão Ariadne, este conceito equivale ao atributo *OS type*, que corresponde ao tipo de sistema operacional em que o LO pode ser executado e faz parte da categoria *technical*. Na especificação do padrão IEEE LOM, os requisitos do LO estão organizados na categoria *technical*, através do atributo *requirement* (pode ter até 40 ocorrências), onde cada ocorrência de *requirement* pode abrigar até 40 ocorrências do atributo *orComposite*. O atributo *orComposite* abriga os atributos *type* e *name*, onde *type* indica qual o tipo de requisito (“OS” ou “browser”) em questão. O atributo *name* representa o nome da tecnologia necessária (por exemplo “Windows”, “Unix”, etc.).

Em DC, não há uma representação para este atributo, uma vez que considera um conjunto bem restrito de elementos.

A representação semântica dos meta-esquemas/padrões de metadados corresponde a uma abstração dos conceitos presentes na especificação e descritos anteriormente para os conceitos “autor” e “requisito”. Assim, a partir da especificação de cada padrão, foi criada uma ontologia correspondente, a subseção 4.3.2.1 discute em detalhes os conceitos representados por essas ontologias.

#### **4.3.2.1. Ontologias dos Padrões**

A figura 4.2 apresenta as ontologias dos padrões Ariadne, IEEE LOM e DC. As elipses representam as classes e as setas representam as propriedades. Na construção das ontologias, surgem algumas considerações interessantes. Na ontologia LOM, o conceito “autor” passa a ser representado como uma classe, evidenciando o conceito que na especificação apresenta-se como combinação dos atributos *role* e *entity*. Ainda na ontologia LOM, os requisitos técnicos de um LO são representados a partir da classe *requirement*, a qual é especializada nas subclasses “requisito de navegador” (*req of browser*) que representa as necessidades em relação ao navegador *Web* e “requisito de sistema operacional” (*req of OS*), o qual representa as necessidades técnicas em relação ao sistema operacional.

As ontologias de cada padrão reúnem todos os conceitos presentes nas especificações dos mesmos. A figura 4.3 apresenta uma visão completa das ontologias de cada padrão, desenvolvidas na ferramenta Protégé [80]. As classes em azul, representam os conceitos “autor” e “requisito” já apresentados anteriormente.

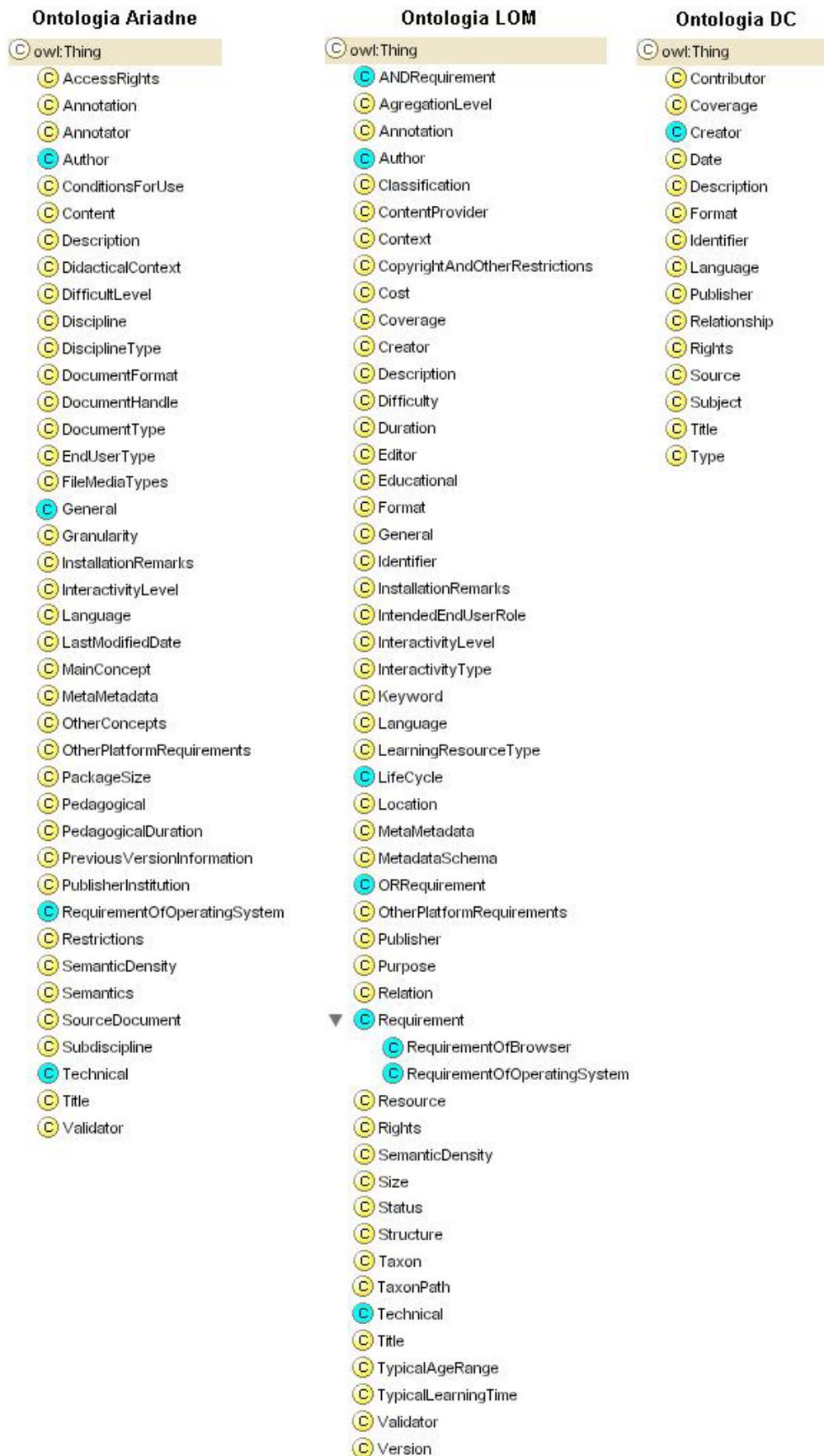


Figura 4.3: Visão completa das ontologias Ariadne, LOM e DC

#### 4.3.2.2. Ontologia Comum

Como mencionado anteriormente, a partir das ontologias criadas é construída uma ontologia comum, que corresponde a uma visão geral dos conceitos representados pelas demais ontologias.

A ontologia comum (ou global) para o exemplo considerado anteriormente é apresentada na figura 4.4, juntamente com as ontologias de cada padrão e suas correspondências. Nesta figura, as elipses representam as classes, as linhas contínuas representam as propriedades e as linhas tracejadas representam as correspondências.

A ontologia comum, ilustrada na Figura 4.4, considera como exemplos os conceitos “autor” e “requisito”. Na ontologia, a classe *author* está relacionada (por meio da propriedade *hasAuthor*) à classe conceito *lifecycle*, a qual representa a história do LO (entidades que afetaram o LO durante sua evolução). Essa representação é idêntica à ontologia LOM e tal representação atende aos demais padrões, conforme indicam as correspondências na figura (linha tracejada).

Como dito anteriormente, o conceito “requisito” refere-se às necessidades técnicas necessárias para o uso do LO. Sua representação na ontologia comum, também advém da ontologia LOM, sendo a classe *requirement* especializada em duas subclasses *req of browser* (requisito de browser), que representa as necessidades em relação ao navegador *Web*, e *req of OS* (requisito de sistema operacional), a qual representa as necessidades técnicas em relação ao sistema operacional. Essa forma de representação atende aos demais padrões, conforme mostra a correspondência na figura. Observe a inexistência desse conceito no padrão DC.

A construção da ontologia comum para os padrões Ariadne, IEEE LOM e DC teve como base a ontologia do padrão IEEE LOM, pois este padrão emerge como forte candidato a padrão de fato, além de ser bastante abrangente, incorporando quase todos os conceitos representados pelos demais padrões de metadados para LOs.

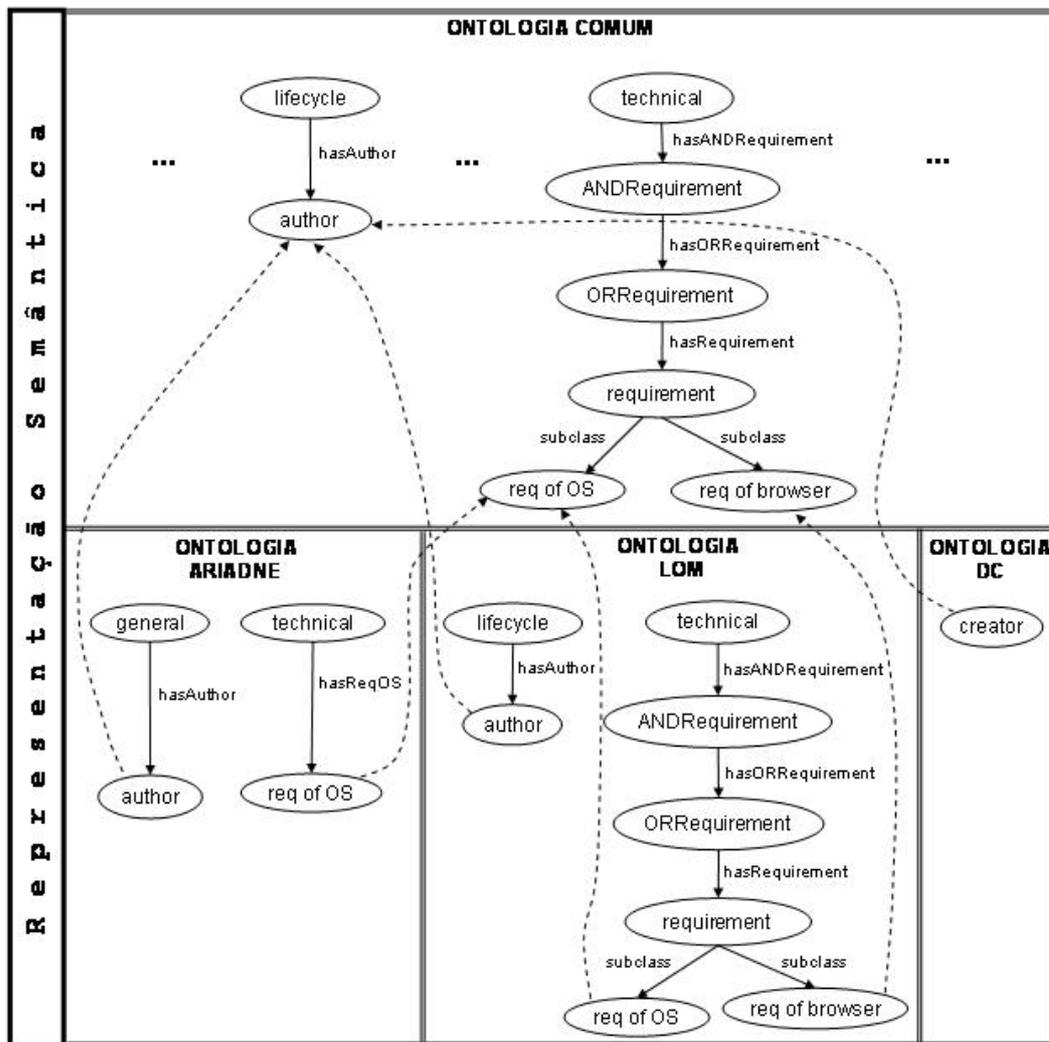


Figura 4.4: Visão semântica de *autor* e *requisito* na ontologia comum

A figura 4.5 apresenta uma visão completa dos conceitos presentes na ontologia comum. A representação da ontologia comum em OWL encontra-se no Anexo C.

### Ontologia Comum

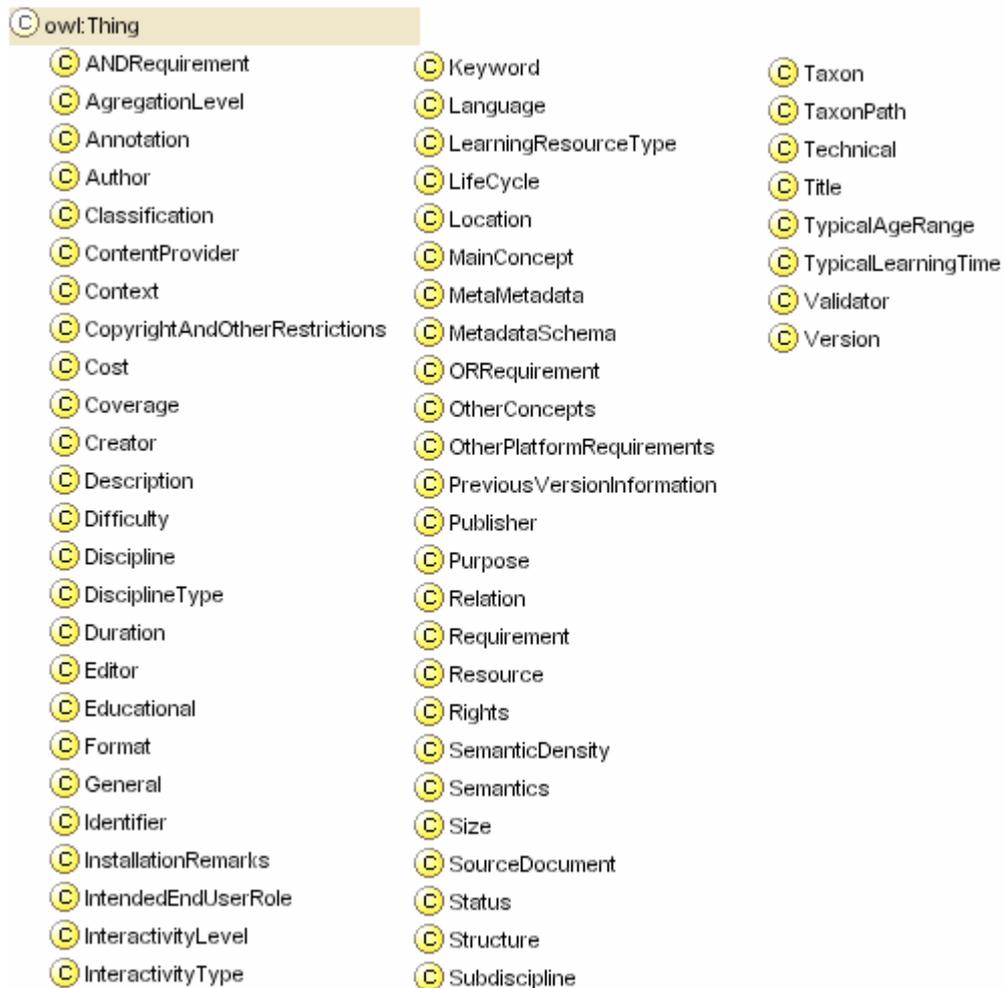


Figura 4.5: Visão completa da ontologia comum

#### 4.3.3. Processo de Mapeamento

Nesta seção discutimos o processo de mapeamento passando pelos níveis semântico e estrutural. A figura 4.6 ilustra esse processo, sendo o nível semântico representado pela ontologia comum descrita em OWL e o nível estrutural representado pelos esquemas descritos em RDFS. Nesta figura é representada a correspondência entre a ontologia comum e os esquemas dos padrões (conforme indicado pelas linhas tracejadas da figura).

A linguagem OWL que descreve a ontologia comum possui propriedades especiais para a definição de equivalências entre as classes. Essas propriedades definem que uma classe ou propriedade particular de uma ontologia seja dita equivalente a uma determinada classe ou propriedade de outra ontologia.

Em OWL, a propriedade *owl:sameAs* indica que duas ou mais classes expressam o mesmo significado. Outra propriedade útil no mapeamento interontologias é a *owl:equivalentProperty*, a qual implica no fato de duas propriedades terem a mesma extensão de propriedade, ou seja, mesmo *domain* e *range* [81].

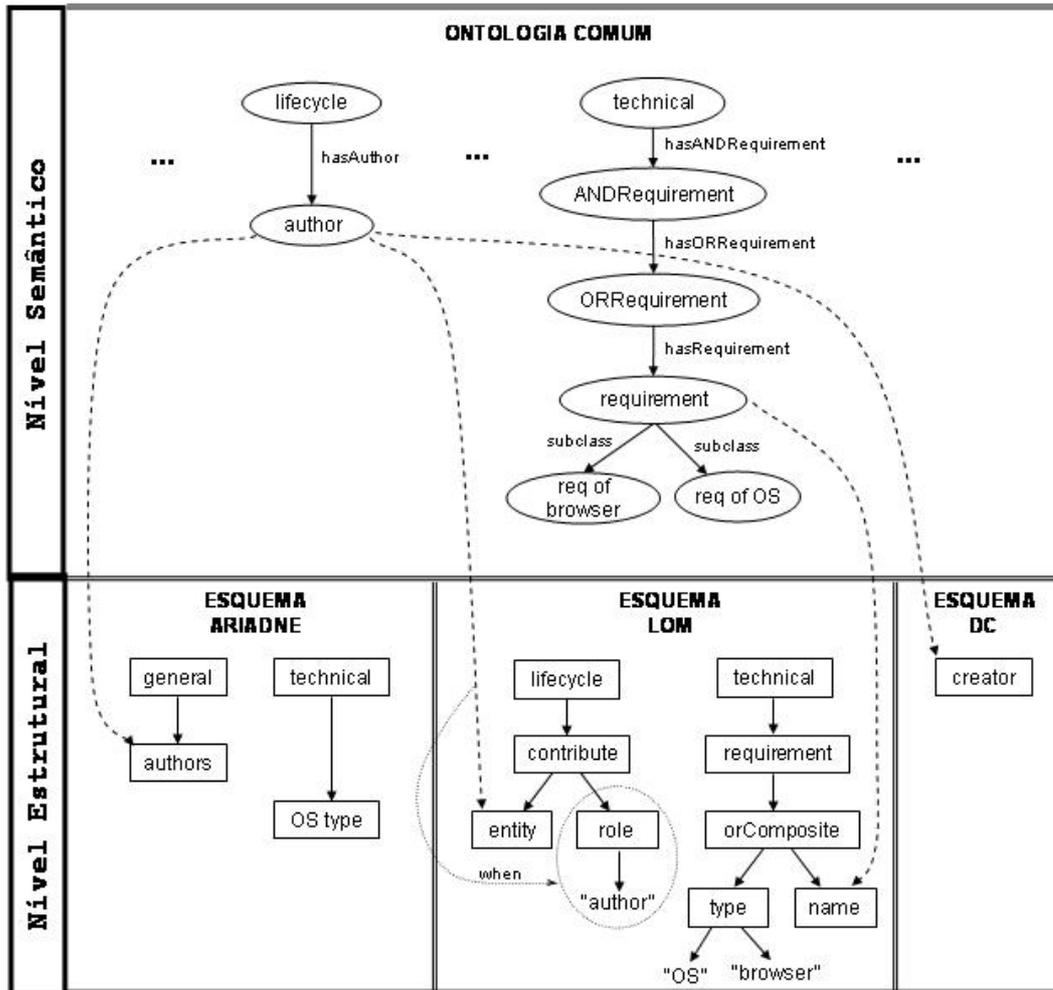


Figura 4.6: Exemplo de Mapeamento em LORIS

Para o melhor entendimento do processo de mapeamento utilizado em LORIS, consideremos a seguinte consulta:

**Consulta:** *Encontre os LOs de autoria de "Astrogildo".*

Esta consulta é submetida ao modelo global (ontologia comum) e encontra correspondência a cada padrão de metadados. De acordo com a consulta acima, é inquirido a respeito do conceito autor (*author*). A figura 4.6 ilustra o mapeamento

partindo do nível semântico (ontologia comum) e chegando até o nível estrutural (especificação dos meta-esquemas/padrões). Considerando-se o padrão Ariadne, na figura 4.6 é indicada a correspondência entre a classe *author* (ontologia comum) e a classe *authors* pertencente à classe *general* (esquema Ariadne). Para o padrão IEEE LOM, a correspondência ocorre entre a classe *author* (ontologia comum) e a classe *entity* que pertence à classe *contribute*, a qual faz parte da classe *lifecycle* (esquema LOM). Contudo, essa correspondência somente é válida quando a classe *role* tiver valor "author". Quanto ao padrão DC, a figura indica correspondência entre a classe *author* (ontologia comum) e a classe *creator* (esquema DC).

Ainda considerando o exemplo exposto no parágrafo anterior, a figura 4.7 apresenta a implementação da correspondência do conceito "author" (ontologia comum) e os demais esquemas (*dc*, *ariad* e *lom*).

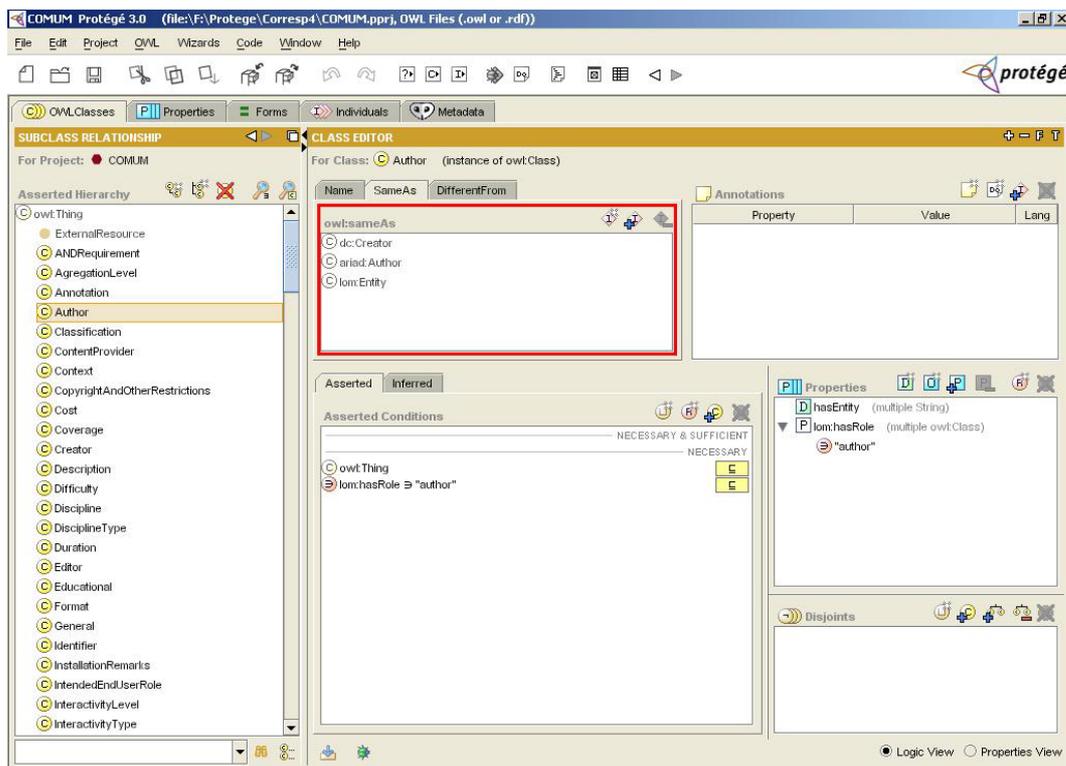


Figura 4.7: Implementação do mapeamento do conceito *author*

A seguir, é apresentado a figura 4.8 que contém a representação OWL da correspondência do conceito *author* por meio da propriedade *owl:sameAs*.

```
<owl:Class rdf:ID="Author">
  <rdfs:subClassOf
    rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
  <owl:sameAs rdf:resource="http://www.tecbd.inf.puc-
    rio.br/LORIS/schemas/dc.rdfs#Creator"/>
  <owl:sameAs rdf:resource="http://www.tecbd.inf.puc-
    rio.br/LORIS/schemas/ariadne.rdfs#Authors"/>
  <owl:sameAs rdf:resource="http://www.tecbd.inf.puc-
    rio.br/LORIS/schemas/lom.rdfs#Entity"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="http://www.tecbd.inf.puc-
        rio.br/LORIS/schemas/lom.rdfs#hasRole"/>
      <owl:hasValue
        rdf:datatype="http://www.w3.org/2001/XMLSchema#string">a
        uthor</owl:hasValue>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

Figura 4.8: Representação em OWL do mapeamento do *author*

#### 4.4. Considerações Finais

Neste capítulo foi abordado o tratamento da heterogeneidade dos padrões de metadados utilizados na descrição de LOs. Foi descrito o estudo realizado neste trabalho acerca do mapeamento entre modelos, sendo dividido em mapeamento semântico e mapeamento estrutural. Para cada padrão foi construída uma ontologia e, a partir das ontologias geradas, foi definida uma ontologia comum. O processo de mapeamento passa pelos níveis semântico e estrutural.