2
Esquemas conceituais em lógica de descrição

Lógica de descrição (*Description Logic* – LD) [4] é o nome dado para uma família de formalismos de representação de conhecimento. Para modelar um domínio de aplicação em LD, primeiramente define-se conceitos relevantes ao domínio - suas terminologias - e então utiliza-se esses conceitos para especificar propriedades de objetos e indivíduos que ocorrem no domínio - a descrição do mundo.

Este capítulo fornece uma introdução à lógica de descrição como uma linguagem formal para representação de conhecimento. A Seção 2.1 apresenta formalmente uma família de linguagens atributivas, a noção de esquemas conceituais elementares e a definição do ambiente de mediação. A Seção 2.4 define formalmente o ambiente de mediação, ilustrando esta noção com exemplos.

2.1. Uma família de linguagens atributivas

Para este trabalho, foi adotada uma família de linguagens atributivas, conhecida como $\mathcal{ALCUI}+$ [39]. Uma linguagem $\mathcal{L}$ na família $\mathcal{ALCUI}+$ é caracterizada por um alfabeto $\mathcal{A}$, consistindo de um conjunto de *conceitos atômicos*, um conjunto de *papéis atômicos*, os *conceitos universal* (*ou supremo*) e *vazio* (ou *ínfimo*) denotados por $\top$ e $\bot$, respectivamente, e os *papéis universal* (ou *supremo*) e *vazio* (ou *ínfimo*), também respectivamente denotados por $\top$ e $\bot$, e um conjunto de *constantes*.

O conjunto de *descrições de papel* de $\mathcal{L}$ é indutivamente definido como:

- Um papel atômico e os papéis universal e vazio são descrições de papel
- Se p e q são descrições de papel, então as seguintes expressões são descrições de papel

  p^- é a *inversa* de p

  $p \circ q$ é a *composição* de p e q

$p \sqcup q$	é a *união* de p e q

O conjunto de descrições de conceito de $\mathcal{L}$ é indutivamente definido como:

- Um conceito atômico e os conceitos universal e vazio são descrições de conceito
- Se $a_1,\ldots,a_n$ são constantes, então $\{a_1,\ldots,a_n\}$ é uma descrição de conceito, chamada de *descrição de conjunto*
- Se e e f são descrições de conceito e p é uma descrição de papel, e n um inteiro não negativo, então as seguintes expressões são descrições de conceito:

$\neg e$	negação
$e \sqcap f$	interseção
$e \sqcup f$	união
$\exists p$	quantificação existencial restrita
$\exists p.e$	quantificação existencial irrestrita
$\forall p.e$	valor restrito
$(\leq n\ p)$	restrição de máximo
$(\geq n\ p)$	restrição de mínimo

- Uma *interpretação* $\boldsymbol{s}$ para os símbolos do alfabeto $\mathcal{A}$ consiste de um conjunto não vazio $\Delta^{\boldsymbol{s}}$, o *domínio* de $\boldsymbol{s}$, cujos elementos são chamados de *indivíduos*, e uma *função de interpretação*, também denotada por $\boldsymbol{s}$, onde:
- $\boldsymbol{s}(\top) = \Delta^{\boldsymbol{s}}$, quando $\top$ denota o conceito universal
- $\boldsymbol{s}(\bot) = \emptyset$, quando $\bot$ denota o conceito vazio
- $\boldsymbol{s}(A) \subseteq \Delta^{\boldsymbol{s}}$, para cada conceito atômico A de $\mathcal{L}$
- $\boldsymbol{s}(\top) = \Delta^{\boldsymbol{s}} \times \Delta^{\boldsymbol{s}}$, quando $\top$ denota o papel universal
- $\boldsymbol{s}(\bot) = \emptyset$, quando $\bot$ denota o papel vazio
- $\boldsymbol{s}(P) \subseteq \Delta^{\boldsymbol{s}} \times \Delta^{\boldsymbol{s}}$, para cada papel atômico P de $\mathcal{L}$
- $\boldsymbol{s}(a) \in \Delta^{\boldsymbol{s}}$, para cada constante a de $\mathcal{L}$, tal que constantes distintas denotam indivíduos distintos (*suposição de unicidade*)

A função $\boldsymbol{s}$ é estendida para descrições de papel e conceito de $\mathcal{L}$ como a seguir:

- $s(p^{-}) = s(p)^{-}$, a inversa de $s(p)$
- $s(p \circ q) = s(p) \circ s(q)$, a composição de $s(p)$ com $s(q)$
- $s(p \sqcup q) = s(p) \cup s(q)$, a união de $s(p)$ com $s(q)$
- $s(\{a_1,...,a_n\}) = \{s(a_1),..., s(a_n)\}$, o conjunto que consiste dos indivíduos $s(a_1),..., s(a_n)$, uma descrição de conjunto
- $s(\neg e) = \Delta^s - s(e)$, o complemento de $s(e)$ com referência a Δ^s
- $s(e \sqcap f) = s(e) \cap s(f)$, a interseção de $s(e)$ e $s(f)$
- $s(e \sqcup f) = s(e) \cup s(f)$, a união de $s(e)$ e $s(f)$
- $s(\exists p) = \{I \in \Delta^s \ / \ (\exists J \in \Delta^s)\ ((I,J) \in s(p))\}$, o conjunto de indivíduos que $s(p)$ relaciona a algum indivíduo
- $s(\exists p.e) = \{I \in \Delta^s \ / \ (\exists J \in \Delta^s)((I,J) \in s(p) \wedge J \in s(e))\}$, o conjunto de indivíduos que $s(p)$ relaciona a algum indivíduo em $s(e)$
- $s(\forall p.e) = \{I \in \Delta^s \ / \ (\forall J \in \Delta^s)\ ((I,J) \in s(p) \Rightarrow J \in s(e))\}$, o conjunto de indivíduos I tal que, se $s(p)$ relaciona I a um indivíduo J, então J está em $s(e)$
- $s(\geq n\ p)=\{I \in \Delta^s \ / \ |\{J \in \Delta^s \ / \ (I,J) \in s(p)\}| \geq n\}$, o conjunto de indivíduos que $s(p)$ relaciona a pelo menos n indivíduos distintos
- $s(\leq n\ p)=\{I \in \Delta^s \ / \ |\{J \in \Delta^s \ / \ (I,J) \in s(p)\}| \leq n\}$, o conjunto de indivíduos que $s(p)$ relaciona a no máximo n indivíduos distintos

Uma *fórmula* de $\mathcal{L}$ é uma expressão da forma $u \sqsubseteq v$, chamada de *inclusão*, ou da forma $u \equiv v$, chamada de *equivalência*, onde u e v são ambas descrições de conceito ou são ambas descrições de papel de $\mathcal{L}$. Uma *definição* é uma equivalência da forma $T \equiv u$, onde T é um conceito atômico e u é uma descrição de conceito, ou T é um papel atômico e u é uma descrição de papel. Uma *disjunção* é uma inclusão da forma $u \sqsubseteq \neg v$, que será abreviada como $u \mid v$.

Uma interpretação s para $\mathcal{L}$ *satisfaz* $u \sqsubseteq v$ sse $s(u) \subseteq s(v)$, s *satisfaz* $u \mid v$ sse $s(u) \cap s(v)=\emptyset$, ou seja, $s(u)$ e $s(v)$ são conjuntos disjuntos. Da mesma forma, uma interpretação s para $\mathcal{L}$ *satisfaz* $u \equiv v$ sse $s(u) = s(v)$.

A seguinte notação também será usada para indicar satisfação e implicação lógica, onde σ é uma fórmula e Σ e Γ são conjuntos de fórmulas:

- $s \models \sigma$ indica que uma interpretação s satisfaz a fórmula σ

- $\boldsymbol{s} \models \Sigma$ indica que uma interpretação $\boldsymbol{s}$ satisfaz todas as fórmulas de Σ; neste caso $\boldsymbol{s}$ é um *modelo* de Σ
- $\Sigma \models \sigma$ indica que qualquer modelo de Σ satisfaz σ, ou seja, para qualquer interpretação $\boldsymbol{s}$, se $\boldsymbol{s} \models \Sigma$, então $\boldsymbol{s} \models \sigma$. Neste caso, diz-se que Σ *implica logicamente* σ
- $\Sigma \models \Gamma$ indica que qualquer modelo de Σ também é um modelo de Γ, ou seja, para qualquer interpretação $\boldsymbol{s}$, se $\boldsymbol{s} \models \Sigma$, então $\boldsymbol{s} \models \Gamma$. Neste caso, diz-se que Σ *implica logicamente* Γ
- $Th(\Sigma)$ denota a *teoria induzida* por Σ, que é o menor conjunto de fórmulas que contém Σ e é fechado sob a implicação lógica
- Σ e Γ são *tautologicamente equivalentes* sse $Th(\Sigma) = Th(\Gamma)$

Nas Seções 2.2 e 2.4 serão usadas descrições de conceito e papel sob um alfabeto $\mathcal{A}$ que é definido como sendo a união de alfabetos $\mathcal{A}_1,...,\mathcal{A}_n$ cujos únicos símbolos em comum são $\top$ e $\bot$. Embora possam ser uniões de alfabeto, a sintaxe usada para descrições de papel e conceito será a mesma. Uma interpretação $\boldsymbol{s}$ para $\mathcal{A}$ é construída a partir de interpretações $\boldsymbol{s}_1,...,\boldsymbol{s}_n$ para $\mathcal{A}_1,...,\mathcal{A}_n$ como a seguir:

- o domínio de $\boldsymbol{s}$ é a união dos domínios de $\boldsymbol{s}_1,...,\boldsymbol{s}_n$
- $\boldsymbol{s}(\boldsymbol{x}) = \boldsymbol{s}_i(\boldsymbol{x})$ *sse* $\boldsymbol{x}$ é um símbolo qualquer, exceto $\top$ e $\bot$, de $\mathcal{A}_i$ (que é único por suposição)
- $\boldsymbol{s}(\top)$ e $\boldsymbol{s}(\bot)$ são definidos normalmente

Assume-se também que os domínios das interpretações são disjuntos:

- *Suposição da disjunção dos domínios* – qualquer par de interpretações para $\mathcal{A}_i$ e $\mathcal{A}_j$ possui domínios disjuntos, para cada $i, j \in [1,n]$, e $i \neq j$.

2.2. Esquemas elementares

Neste trabalho serão usados *esquemas conceituais elementares* (*extralite schemas*) que correspondem parcialmente a OWL Lite [7]. Os esquemas elementares suportam *classes* e *propriedades,* e admitem *restrições de domínio* e *imagem, restrições de subconjunto e exclusão*, *restrições de minCardinalidade e maxCardinalidade*, todas com o significado usual.

Formalmente, um *esquema elementar* é um par $S = (\mathcal{A}, \Sigma)$ tal que:

- $\mathcal{A}$ é um alfabeto, chamado de *vocabulário* de S, cujos conceitos e papéis atômicos são chamados de *classes* e *propriedades* de S, respectivamente
- Σ é um conjunto de fórmulas, chamado de *restrições* de S, que necessariamente possuem uma das formas a seguir, onde C e D são classes e P é uma propriedade de S:
  - *Restrição de domínio*

    $\exists P \sqsubseteq D$, a propriedade P possui domínio D
  - *Restrição de imagem*

    $\exists P^{-} \sqsubseteq C$, a propriedade P possui imagem C
  - *Restrição de minCardinalidade*

    $C \sqsubseteq (\geq k\ P)$ ou $C \sqsubseteq (\geq k\ P^{-})$, a propriedade P, ou sua inversa P^{-}, mapeia cada indivíduo em C para no mínimo k indivíduos distintos
  - *Restrição de maxCardinalidade*

    $C \sqsubseteq (\leq k\ P)$ ou $C \sqsubseteq (\leq k\ P^{-})$, a propriedade P, ou sua inversa P^{-}, mapeia cada indivíduo em C para no máximo k indivíduos distintos
  - *Restrição de subconjunto*

    $C \sqsubseteq D$, a classe C é uma subclasse da classe D
  - *Restrição de disjunção*
    - $C \mid D$, a classe C é disjunta da classe D
  - Também são admitidas restrições de uma das formas a seguir
    - $C \sqsubseteq \bot$, a classe C é sempre vazia
    - $\exists P \sqsubseteq \bot$ ou $\exists P^{-} \sqsubseteq \bot$, a propriedade P é sempre vazia, ou seja, P possui domínio vazio ou possui imagem vazia.

Diz-se que uma interpretação $\boldsymbol{s}$ para $\mathcal{A}$ é um *estado consistente* de S sse $\boldsymbol{s}$ satisfaz todas as restrições em Σ.

Uma observação importante é que, apesar das restrições serem fórmulas muito limitadas, elas capturam semântica suficiente para modelar diagramas UML simples. Também é possível, através destas restrições, abranger algumas das restrições de propriedade e classes de axiomas de OWL [7].

Ao longo do texto, serão usados os termos *classe*, *propriedade* e *vocabulário* como sinônimos dos termos *conceito atômico*, *papel atômico* e *alfabeto,* respectivamente.

As restrições de domínio e imagem recebem, de agora em diante, o nome coletivo de *restrições de propriedade*. Bem como as restrições de minCardinalidade e maxCardinalidade serão chamadas de *restrições de cardinalidade*, e as restrições de disjunção e de subconjunto serão as *restrições de classe.* Note que é permitido mais de uma restrição de domínio ou imagem por propriedade, e mais de uma restrição de minCardinalidade ou maxCardinalidade para a mesma classe e propriedade.

Além disso, a representação $C \sqsubseteq (=k\ P)$ será usada como abreviação do par de restrições da forma $C \sqsubseteq (\geq k\ P)$ e $C \sqsubseteq (\leq k\ P)$. Para referência posterior, observa-se que uma restrição de domínio $\exists P \sqsubseteq C$ pode ser reescrita formalmente como $(\geq 1\ P) \sqsubseteq C$. Da mesma forma, a restrição de imagem $\exists P^- \sqsubseteq C$ pode ser reescrita como $(\geq 1\ P^-) \sqsubseteq C$, posto que $\exists P \equiv (\geq 1\ P)$ e $\exists P^- \equiv (\geq 1\ P^-)$.

As restrições de subconjunto capturam hierarquia de classes e não necessitam de discussão adicional.

Para justificar a formalização das restrições de domínio, tome $\mathbf{s}$ uma interpretação de $\mathcal{L}$ com domínio $\Delta^{\mathbf{s}}$. Então, $\mathbf{s}$ satisfaz uma restrição de domínio $\exists P \sqsubseteq D$ sse, para qualquer $a \in \Delta^{\mathbf{s}}$, se $a \in \mathbf{s}(\exists P)$ então $a \in \mathbf{s}(D)$. Mas $a \in \mathbf{s}(\exists P)$ sse existe $b \in \Delta^{\mathbf{s}}$ tal que $(a,b) \in \mathbf{s}(P)$, o que significa que a está no domínio de $\mathbf{s}(P)$. Em outras palavras, se a está no domínio de $\mathbf{s}(P)$, então $a \in \mathbf{s}(D)$.

Da mesma forma, $\mathbf{s}$ satisfaz uma restrição de imagem $\exists P^- \sqsubseteq C$ sse, para qualquer $b \in \mathbf{s}(\exists P^-)$ então $b \in \mathbf{s}(C)$. Mas $b \in \mathbf{s}(\exists P^-)$ sse, existe $a \in \Delta^{\mathbf{s}}$ tal que $(a,b) \in \mathbf{s}(P)$, o que significa que b está na imagem de $\mathbf{s}(P)$. Logo, pode-se dizer que, se b está na imagem de $\mathbf{s}(P)$, então $b \in \mathbf{s}(C)$.

É importante frisar que a formalização aqui apresentada não faz distinção entre *object* e *datatype properties*, na terminologia OWL. A distinção ficará visível apenas nos exemplos, onde a imagem de uma *object property* será uma classe definida no esquema, enquanto que a imagem de uma *datatype property* será um XML Schema type, ou seja, um conjunto de *datatype values* ou *literals*. Uma vez que a noção de domínio de uma interpretação não separa indivíduos que denotam elementos de classe (*class elements*) de indivíduos que correspondem a valores de tipo de dados (*datatype values*), o desenvolvimento formal não tem como capturar esta distinção.

Exemplo 2-1 – Esquemas *Amazon* e *eBay*

A Figura 2 e a Figura 3 contém esquemas para fragmentos de bancos de dados da *Amazon* e *eBay*, com uma notação informal. Foram usados como prefixos de *namespace* "`a:`" e "`e:`" para se referir aos vocabulários dos esquemas da *Amazon* e *eBay* respectivamente.

```
a:Product
  a:title          range string
  a:price          range decimal
  a:currency       range string
a:Book
  a:isbn           range string
  a:author         range string
  a:pub            range a:Publ
```

```
a:Publ
  a:name           range string
  a:city           range string

a:Book is-a a:Product
a:Music is-a a:Product

a:Book disjoint-from a:Music
```

Figura 2 – Definição informal do esquema da *Amazon*.

```
e:Seller
  e:name           range string
e:Offer
  e:qty            range integer
  e:price          range double
  e:currency       range string
  e:seller         range e:Seller
  e:product        range e:Product
```

```
e:Product
  e:type           range string
  e:ean            range integer
  e:title          range string
  e:author         range string
  e:edition        range integer
  e:year           range integer
  e:place          range string
```

Figura 3 – Definição informal do esquema do *eBay*.

Pode-se observar a partir da Figura 2, que `a:title` é definido como uma propriedade com domínio `a:Product` e imagem `string` (um *XML Schema datatype*), `a:Book` é declarado como uma subclasse de `a:Product`, e `a:pub` é definido como uma propriedade com domínio `a:Book` e imagem `a:Publ`. A Figura 3 segue o mesmo raciocínio.

R. de Propriedade	*R. de Cardinalidade*	*R. de Classe*
∃ a:title ⊑ a:Product ∃ a:title^{-} ⊑ string ... ∃ a:pub ⊑ a:Book ∃ a:pub^{-} ⊑ a:Publ ... ∃ a:city ⊑ a:Publ ∃ a:city^{-} ⊑ string ...	a:Product ⊑ (≤ 1 a:title) a:Product ⊑ (≤ 1 a:price) a:Product ⊑ (≤ 1 a:currency) a:Book ⊑ (≤ 1 a:isbn) a:Book ⊑ (≥ 2 a:pub) a:Publ ⊑ (≤ 1 a:name) a:Publ ⊑ (≥ 3 a:city)	a:Book ⊑ a:Product a:Music ⊑ a:Product a:Book | a:Music

Figura 4 – Definição formal das restrições do esquema da *Amazon*

A Figura 4 e a Figura 5 formalizam as restrições do esquema da seguinte forma: a primeira coluna mostra as restrições de domínio e imagem; a segunda apresenta as restrições de cardinalidade; e a terceira coluna contém as restrições de classe.

R. de Propriedade	*R. de Cardinalidade*	*R. de Classe*
∃ e:name ⊑ e:Seller ∃ e:name^{-} ⊑ string ... ∃ e:seller ⊑ e:Offer ∃ e:seller^{-} ⊑ e:Seller ∃ e:product ⊑ e:Offer ∃ e:product^{-} ⊑ e:Product ...	e:Seller ⊑ (≤ 1 e:name) e:Offer ⊑ (≤ 1 e:qty) e:Offer ⊑ (≤ 1 e:price) ... e:Product ⊑ (≤ 1 e:type) e:Product ⊑ (≤ 1 e:ean) e:Product ⊑ (≤ 1 e:title) ... e:Product ⊑ (≥ 1 e:place)	Não há

Figura 5 – Definição formal de algumas das restrições do esquema do *eBay*

Em relação a Figura 4 pode-se verificar, na segunda coluna, que: (1) todas as propriedades possuem maxCardinalidade igual a 1, exceto `a:author`, `a:pub` e `a:city`; (2) `a:author` possui maxCardinalidade ilimitada, devido ao fato que um livro pode ter mais de um autor; (3) `a:pub` tem maxCardinalidade igual a 2; e (4) `a:city` possui minCardinalidade igual a 3. A terceira coluna mostra que `a:Book` e `a:Music` são sub-classes de `a:Product` e, além disso, são classes disjuntas.

2.3.
Esquemas ultra elementares

Os *esquemas ultra elementares* são um dialeto dos *esquemas elementares* da Seção 2.2. Eles suportam classes e propriedades e admitem restrições de domínio, imagem e subconjunto da forma como é apresentado na Seção 2.2. No entanto este dialeto não captura expressões com restrições de cardinalidade. Ele foi adotado por questões de complexidade dos algoritmos. Porém, mesmo não admitindo restrições de cardinalidade, modelam as restrições mais comumente usadas em projetos de bancos de dados, como será visto no Capítulo 3.

Formalmente, um *esquema ultra elementar* é um par $S = (\mathcal{A}, \Sigma)$ tal que:

- $\mathcal{A}$ é um alfabeto, chamado de *vocabulário* de S, cujos conceitos e papéis atômicos são chamados de *classes* e *propriedades* de S, respectivamente
- Σ é um conjunto de fórmulas, chamado de *restrições* de S, que necessariamente possui uma das formas a seguir, onde C e D são classes e P é uma propriedade de S:

  - *Restrição de domínio* $\exists P \sqsubseteq D$
  - *Restrição de imagem* $\exists P^{-} \sqsubseteq C$
  - *Restrição de subconjunto* $C \sqsubseteq D$

2.4. Ambiente de mediação

Um *ambiente de* mediação contém um *esquema mediado* M, um *mapeamento de mediação* γ e, para cada $k = 1,...,n$, um *esquema exportado* E_k, um *esquema importado* I_k e um *mapeamento local* γ_k.

Os esquemas importados foram introduzidos com o intuito de dividir as definições do mapeamento em dois estágios: a definição dos mapeamentos locais e a definição do mapeamento de mediação. Além disso, o mapeamento de mediação foi restrito para que ele defina os conceitos de M como uniões de conceitos a partir de esquemas importados. De fato, no contexto da *Web,* fontes de dados são independentes. Portanto, o mapeamento de mediação deve ser especificado de tal forma que seja possível automatizar as tarefas de adicionar e remover fontes de dados do ambiente de mediação. Esta restrição sobre o mapeamento de mediação representará um papel importante no que está sendo proposto para modelar as restrições do esquema mediado.

Os esquemas importado, exportado e o esquema mediado são esquemas elementares, conforme definido na Seção 2.2. Formalmente, o esquema mediado será então um par $M=(\mathcal{A}, \Sigma)$, onde $\mathcal{A}$ é um vocabulário (i.e. alfabeto) e Σ é um conjunto de restrições em $\mathcal{A}$.

No que se segue, assume-se que o vocabulário do esquema mediado M consiste das classes $C_1,...,C_u$ e das propriedades $P_1,...,P_v$.

Assume-se ainda que o esquema importado está restrito como segue

- para $k=1,...,n$, o vocabulário de I_k é o mesmo vocabulário de M, no sentido de que eles possuem as mesmas classes e propriedades, mas com *namespace prefixes* distintos.

Nas definições que vem a seguir, não foram adotados *namespace prefixes*, como no exemplo. Em seu lugar, uma notação mais abstrata foi usada para distinguir a ocorrência de um símbolo no vocabulário de M da ocorrência do mesmo símbolo no vocabulário de I_k. Para cada classe C_i (ou propriedade P_j) no vocabulário de M, a ocorrência de C_i (ou P_j) no vocabulário de I_k será denotada por C_i^k (ou P_j^k). Diz-se ainda que C_i^k (ou P_j^k) *alinha com* C_i (ou P_j).

O mapeamento de mediação γ define as classes e propriedades de M como uniões de classes e propriedades do esquema importado. Mais precisamente, o mapeamento de mediação está restrito como segue:

- para cada $i=1,...,u$, o mapeamento γ contém uma definição da forma $C_i \equiv e_i^1 \sqcup ... \sqcup e_i^n$ onde e_i^k é a classe C_i^k de I_k que alinha com C_i, para cada $k=1,...,n$;
- para cada $j=1,...,v$, o mapeamento γ contém uma definição da forma $P_j \equiv p_j^1 \sqcup ... \sqcup p_j^n$ onde p_j^k é a propriedade P_j^k de I_k que alinha com P_j, para cada $k=1,...,n$.

É possível ver que a definição de C_i (ou P_j) possui exatamente uma descrição de conceito (ou descrição de papel) para cada esquema importado, mesmo que o esquema importado não tenha classe (ou propriedade) que alinhe C_i (ou P_j), caso em que é usado o símbolo $\perp$.

Para cada $k=1,...,n$, o mapeamento local γ_k define as classes e propriedades de I_k em termos do vocabulário do esquema exportado E_k. Então, o mapeamento local γ_k é limitado como a seguir:

- para cada classe C_i^k de I_k, o mapeamento local γ_k contém uma definição da forma $C_i^k \equiv \rho_i^k$ onde ρ_i^k é uma descrição de conceito sobre o vocabulário de E_k

- para cada propriedade P_j^k de I_k, o mapeamento local γ_k contém uma definição da forma $P_j^k \equiv \pi_j^k$ onde π_j^k é uma descrição de papel sobre o vocabulário de E_k

O símbolo $\bar{\gamma}_k$ representa a *função induzida por* γ_k, definida como a função de estados de E_k em estados de I_k tal que, para cada estado $\boldsymbol{s}$ de E_k, $\bar{\gamma}_k(\boldsymbol{s}) = \boldsymbol{r}$ sse

- $\boldsymbol{r}(C_i^k) = \boldsymbol{s}(\rho_i^k)$, se $C_i^k \equiv \rho_i^k$ define a classe C_i^k em γ_k
- $\boldsymbol{r}(P_j^k) = \boldsymbol{s}(\pi_j^k)$, se $P_j^k \equiv \pi_j^k$ define a propriedade P_j^k em γ_k

De forma semelhante, foi introduzido o símbolo $\bar{\gamma}$ para representar a *função induzida* pelo mapeamento de mediação γ e os mapeamentos locais $\gamma_1,...,\gamma_n$ como sendo o mapeamento dos estados de $E_1,...,E_n$ em estados de M tal que, para estados $s_1,...,s_n$ de $E_1,...,E_n$, $\bar{\gamma}(\boldsymbol{s}_1,...,\boldsymbol{s}_n) = \boldsymbol{r}$ sse, para $i=1,...,u$ e $j=1,...,v$, vale

- $\boldsymbol{r}(C_i) = \boldsymbol{s}_1(e_i^1) \cup ... \cup \boldsymbol{s}_n(e_i^n)$, se $C_i \equiv e_i^1 \sqcup ... \sqcup e_i^n$ define C_i em γ
- $\boldsymbol{r}(P_j) = \boldsymbol{s}_1(p_j^1) \cup ... \cup \boldsymbol{s}_n(p_j^n)$, se $P_j \equiv p_j^1 \sqcup ... \sqcup p_j^n$ define P_j em γ

Exemplo 2-2 – Ambiente Mediado *Sales*

Para ilustrar os conceitos introduzidos até aqui, considere um ambiente mediado, onde *Sales* é o esquema mediado, cujo vocabulário é mostrado na Figura 6. Para este vocabulário será usado o prefixo "`s:`".

Classes:	Propriedades:

`s:Product`
`s:Book`
`s:Music`
	`s:title`
	`s:city`

Figura 6 – Vocabulário do esquema mediado *Sales*

O vocabulário do esquema importado da *Amazon* é o mesmo vocabulário do esquema *Sales*, exceto que o prefixo de *namespace* adotado é "`ai`", como pode ser visto na Figura 7.

Classes:	Propriedades:

`ai:Product`
`ai:Book`
`ai:Music`
	`ai:title`
	`ai:city`

Figura 7 – Vocabulário do esquema importado da *Amazon*

O conjunto de restrições do esquema importado é construído a partir do conjunto de restrições do esquema exportado da Figura 4. Tais restrições podem ser vistas na Figura 8. O processo de construção não será detalhado nesta seção; o objetivo aqui é apenas mostrar como, após as traduções, as restrições do esquema importado são apresentadas ao esquema mediado. Note, porém, que algumas das restrições do esquema importado não são traduções de restrições do esquema exportado, mas sim conseqüência das definições do mapeamento local da *Amazon*, mostrado na Figura 9.

R. de Propriedade	***R. de Cardinalidade***	***R. de Classe***
$\exists$ ai:title $\sqsubseteq$ ai:Product $\exists$ ai:title$^-$ $\sqsubseteq$ string $\exists$ ai:city $\sqsubseteq$ ai:Book $\exists$ ai:city$^-$ $\sqsubseteq$ string	ai:Product $\sqsubseteq$ ($\leq$ 1 ai:title) ai:Book $\sqsubseteq$ ($\geq$ 3 ai:city)	ai:Book $\sqsubseteq$ ai:Product ai:Music $\sqsubseteq$ ai:Product ai:Book | ai:Music

Figura 8 – Restrições do esquema importado da *Amazon*

ai:Product $\equiv$ a:Product ai:Music $\equiv$ a:Music ai:Book $\equiv$ a:Book	ai:title $\equiv$ a:title ai:city $\equiv$ a:pub $\circ$ a:city

Figura 9 – Mapeamento local do esquema da *Amazon*

Algumas observações importantes podem ser feitas. Considere as definições `ai:city` $\equiv$ `a:pub` $\circ$ `a:city` e `ai:Book` $\equiv$ `a:Book` mostradas na Figura 9. Primeiro, observe que o domínio de `ai:city` é `ai:Book` e a imagem é `string`. Segundo, note que `ai:city` tem minCardinalidade 3 com respeito ao domínio `ai:Book` já que, de acordo a Figura 4, `a:pub` tem minCardinalidade 2 com respeito a `a:Book`, `a:city` possui minCardinalidade 3 com respeito a `a:Publ`, e `a:Publ` é tanto a imagem de `a:pub` quanto o domínio de `a:city`.

Intuitivamente, cada livro possui pelo menos duas editoras e cada uma delas está localizada em pelo menos três cidades. No entanto, este conjunto mínimo de três cidades de uma editora não é necessariamente distinto do conjunto de três cidades de uma outra editora, ou das demais editoras definidas para cada livro. Logo, o que se pode garantir é que cada livro é associado a pelo menos 3 editoras.

Para o esquema *eBay*, o processo acontece da mesma forma. O vocabulário do esquema importado do *eBay*, mostrado na Figura 10, é o mesmo que pode ser visto na Figura 6, exceto que o prefixo de *namespace* adotado é "`ei:`". Na Figura 11 podem ser vistas as restrições do esquema importado *eBay* que foram construídas a partir do mapeamento local da Figura 12 e das restrições do esquema externo do *eBay* da Figura 5

Classes:	Propriedades:
ei:Product ei:Book ei:Music	ei:title ei:city

Figura 10 – Vocabulário do esquema importado eBay

R. de Propriedade	*R. de Cardinalidade*	*R. de Classe*
$\exists$ ei:title $\sqsubseteq$ ei:Product $\exists$ ei:title^{-} $\sqsubseteq$ string $\exists$ ei:city $\sqsubseteq$ ei:Product $\exists$ ei:city^{-} $\sqsubseteq$ string	ei:Product $\sqsubseteq$ ($\leq$ 1 ei:title) ei:Product $\sqsubseteq$ ($\geq$ 1 ei:city)	ei:Book $\sqsubseteq$ ei:Product ei:Music $\sqsubseteq$ ei:Product ei:Book | ei:Music

Figura 11 – Restrições do esquema importado *eBay*

ei:Product $\equiv$ e:Product ei:Music $\equiv$ e:Product $\sqcap$ $\exists$e:type.{'music'} ei:Book $\equiv$ e:Product $\sqcap$ $\exists$e:type.{'book'}	ei:title $\equiv$ e:title ei:city $\equiv$ e:place

Figura 12 – Mapeamento local do esquema *eBay*

Em particular, é possível observar, na Figura 12, que ei:Music e ei:Book são definidas como restrições de e:Product (dado um conceito atômico A, a *restrição* de A é uma interseção da forma $A \sqcap e$). Como conseqüência, tem-se duas restrições de subconjunto e uma restrição de disjunção, como pode-se ver na terceira coluna da Figura 11, muito embora o esquema original *eBay* não possua tais restrições (veja Figura 5). É bom lembrar que as restrições de disjunção requerem que constantes distintas denotem indivíduos distintos.

Por fim, as restrições do esquema mediado *Sales* são mostradas na Figura 13 e o mapeamento de mediação pode ser visto na Figura 14. Note que este mapeamento define as classes e propriedades de *Sales* como uniões de classes e propriedades do esquema importado, conforme exigido.

R. de Propriedade	*R. de Cardinalidade*	*R. de Classe*
$\exists$ s:title $\sqsubseteq$ s:Product $\exists$ s:title^{-} $\sqsubseteq$ string $\exists$ s:city $\sqsubseteq$ s:Product $\exists$ s:city^{-} $\sqsubseteq$ string	s:Product $\sqsubseteq$ ($\leq$ 1 s:title) s:Book $\sqsubseteq$ ($\geq$ 1 s:city)	s:Book $\sqsubseteq$ s:Product s:Music $\sqsubseteq$ s:Product s:Book | s:Music

Figura 13 – Restrições do esquema mediado *Sales*

s:Product $\equiv$ ai:Product $\sqcup$ ei:Product s:Music $\equiv$ ai:Music $\sqcup$ ei:Music s:Book $\equiv$ ai:Book $\sqcup$ ei:Book	s:title $\equiv$ ai:title $\sqcup$ ei:title s:city $\equiv$ ai:city $\sqcup$ ei:city

Figura 14 – Mapeamento de mediação do esquema *Sales*

Na Figura 15 foram agrupados todos os esquemas que dizem respeito à inclusão do esquema da *Amazon* ao esquema mediado *Sales*. Tal resumo repete as figuras que já foram inseridas ao longo do texto, mas serve para uma melhor compreensão do ambiente de mediação.

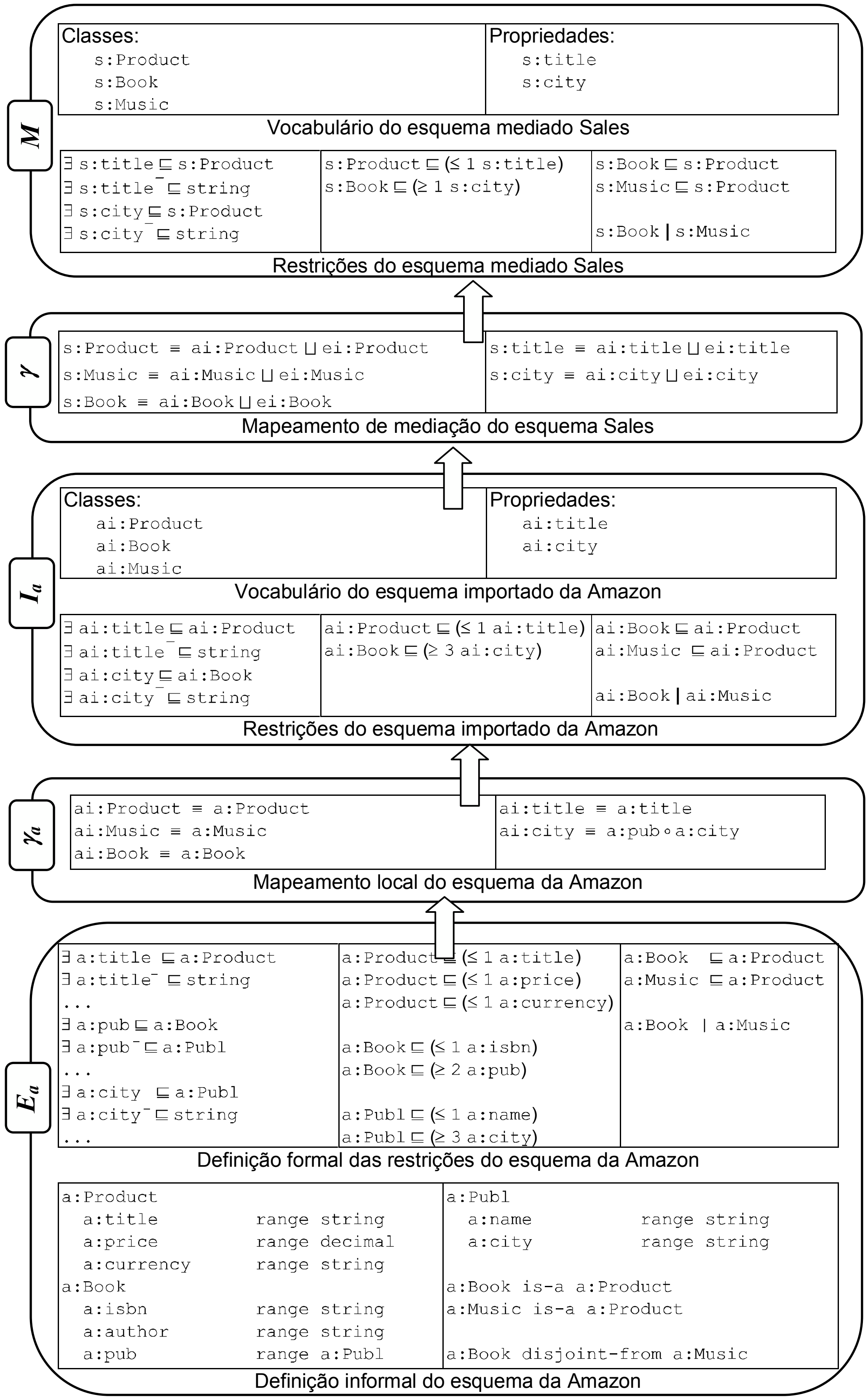

Figura 15 – Ambiente de mediação *Sales* com o esquema da *Amazon*

2.5. Conclusões do capítulo

Neste capítulo foi primeiramente apresentada uma visão geral de Lógica de Descrição. Então foi introduzido o formalismo para os esquemas elementares e ultra elementares que serão usados nos próximos capítulos. Por fim, o ambiente de mediação considerado nesta tese foi definido.

Em particular, introduziu-se a noção de esquema importado no intuito de dividir as definições dos mapeamentos em dois estágios: a definição dos mapeamentos locais e a definição do mapeamento de mediação. Além disso, restringiu-se o mapeamento de mediação para que defina os conceitos de M como uniões de conceitos a partir dos esquemas importados. Esta restrição reflete a idéia de que, no contexto da *Web,* fontes de dados são independentes. Portanto, o mapeamento de mediação deve ser especificado de forma que seja possível automatizar as tarefas de adicionar e remover as fontes de dados do ambiente de mediação, sem propagar mudanças além do próprio mapeamento de mediação. Esta limitação representa um papel fundamental no que está sendo proposto para modelar as restrições do esquema de mediação.

Cada aplicativo que faça uso do ambiente de mediação deve conhecer a semântica comum dos dados retornados pelas fontes de dados, de modo que possa ter informações suficientes para processar corretamente os resultados das consultas. Tal conhecimento, de fato, é a motivação para adicionar restrições ao esquema mediado. É afirmado no Capítulo 4 que as restrições do esquema mediado podem ser modeladas como sendo tautologicamente equivalentes ao ínfimo do conjunto de restrições dos esquemas importados envolvidos. Desta forma, os dados retornados por consultas definidas sobre o esquema mediado sempre irão satisfazer as restrições. De fato, os dados retornados por cada fonte de dados irão satisfazer as restrições de cada esquema importado (que estão corretos, por construção) e, conseqüentemente, as restrições do esquema mediado (que são modelados como o ínfimo das restrições do esquema importado).