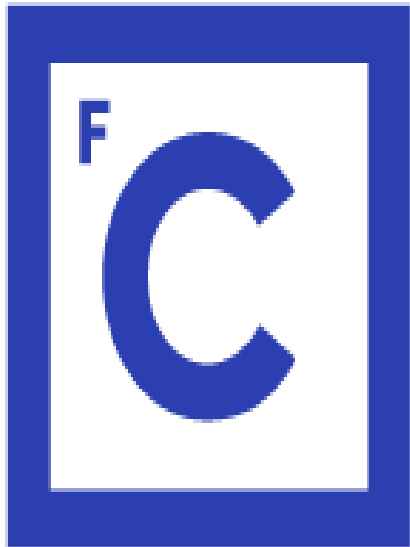


Trabalho de Engenharia do Conhecimento



**Ciências
ULisboa**

Grupo nº30:

André Monteiro nº51718

José Salgueiro nº50004

Miguel Octávio nº48321

2 Especificação do Conhecimento a Representar

Justificação das afirmações dadas

Ponto nº1:

Para o ponto número 1 criamos apenas uma classe Publicação onde esta teria os dois grandes tipos, as publicações periódicas e não periódicas.

Ponto nº2:

Para este ponto adicionamos os jornais e as revistas á publicação periódica.

Ponto nº3:

Neste ponto fizemos uma classe Periodização onde esta teria de conter as instâncias criadas: diária, semanal, quinzenal, mensal, bimestral, trimestral, semestral ou anual.

Depois disto tivemos de por na publicação periódica que cada uma tem exatamente um período de periodicidade. ([temPeriodicidade exactly 1 Periodização](#))

Ponto nº4:

Foi criada uma classe contribuição onde cada uma contribuição tem no min 1 Autor Contribuição (Separamos os autores de contribuição dos de publicação para ficar mais legível e para que não houvesse confusões entre eles). Em seguida tivemos de dizer que em cada publicação periódica existe sempre a um conjunto de contribuições.

Ponto nº5:

O Individual Publico é um Jornal. Adicionei as informações do jornal (O diretor, as suas edições, e a sua periodicidade) com object properties.

O Individual edição1, adicionamos as suas data properties (número da edição e data de edição) e object porperties (Ligação ao jornal Publico, a contribuição da edição (contribuiçãoEd1)).

O Individual contribuiçãoEd1, é a contribuição referente a edição1 adicionamos todas as object properties (o AutorContribuição Cai Run,o tipo de (artigo de opinião), qual edição faz parte(neste caso edição 1)) e adicionei as data properties (o titulo da contribuição).

Ponto nº6:

Cada publicação tem no min 1 autor por isso foi adicionado o [AutorPublicação min 1 autor](#) na Classe Publicação

Em relação as contribuições os livros podem ou não ter por isso adicionamos no livro quando ele tem contribuições, mas não especificamos em nenhum equivalent.

Ponto nº7:

Criação da classe Biblioteca Central, esta biblioteca vai ter os seus exemplares que pode requisitar.

Adicionado o Individual LivroOdisseia com Object Properties (As edições dele o Autor desta Publicação Homero (individual) e qual o tipo deste livro, neste caso é poema épico(individual)), foi também adicionado as Data Properties (tem Linguagem grego antigo).

Adicionado o Individual EdiçãoOdisseia que é uma edição traduzida do LivroOdisseia. Esta edição vai ter como objectProperties(traduzido por Frederico Lourenço(individual), tem Editora Quetzal Editores (individual) e vai ter 2 exemplares (individuals). Tem como data properties (tem ISBN “9789897224386”, tem 688 paginas, é escrito em Português, tem data de edição 2018 e com o titulo original).

A editora “Quetzal Editores” vai ter como data propertie o seu ano de fundação.

Ponto nº8

Iremos adicionar á biblioteca central o exemplar da 1 edição do Memorial Do Convento.

Adicionado o Individual LivroMemorialDoConvento com Object Properties (As edições dele o Autor desta Publicação José Saramago (individual) e qual o tipo deste livro, neste caso é romance (individual)), foi também adicionado as Data Properties (tem Linguagem Português).

Adicionado o Individual Ed1MemorialDoConvento, que é uma edição original do LivroMemorialDoConvento. Esta edição vai ter como objectProperties(tem Editora Editorial Caminho (individual) e vai ter 1 exemplar (individuals). Tem como data properties (tem data de edição 1982 e com o título original).

A editora “Editorial Caminho” vai ter como data propertie o seu ano de fundação.

Ainda tem o Individual DasMemorial que é uma edição traduzida do LivroMemorialDoConvento. Esta edição vai ter como objectProperties(traduzido por Andreas_Klotsch (individual), tem Editora Hoffman Und Campe (individual). Tem como data properties (é escrito em Alemão, tem data de edição 2018 e com o titulo Das Memorial).

Ponto nº9

The Semantic Web(individual) irá ser um livro com vários editores (Chiara Ghidini, Olaf Hartig, e Maria Maleshkova (entre outros)), é do tipo Ciências da Computação, tem Editora Springer (object property). Alem disso tem data de edição de 2019 e título “The Semantic Web – ISWC 2019” (Data property). Este livro ira possuir dois Volumes (TheSemanticWebV1 e TheSemanticWebV2) cada um terá o seu numero de volume.

No caso do volume 1 este terá a Contribuição do “Uncovering the Semantics of Wikipedia Categories” da autoria de Nicolas Heist e Heiko Paulheim.

Ponto nº10

Através do Equivalent ([\(podeConsultar some Publicação\) and \(podeRequesitar some ExemplarDisp\)](#))

Podem consultar todas as publicações, mas só podem requisitar os exemplares Disponíveis, isto é, aqueles que não têm o estatuto restrito.

Ponto nº11

Para um utente conseguir requisitar no máximo 1 exemplar nos dividimos os exemplares requisitados dos exemplares que se pode requisitar, isto ajuda nos pois assim cada utente só pode ter uma requisição dentro dos exemplares requisitados, enquanto ele tiver não poderá requisitar mais nenhum mas quando já não possuir nenhum livro requisitado pode fazer de novo o mesmo processo de requisitar.

Também adicionamos o característica funcional nas object propertys das requisições (Requisitou,RequisitadoPor,podeRequesitar) para que possa haver no máximo uma relação de saída ao longo da propriedade para esse utente.

Resumindo se um Utente requisitou o livro x e esse livro tem la a sua requisição apenas quando ele o devolver é que irá poder requisitar de novo, caso contrário não vai poder requisitar mais nenhum.

Ponto nº12

Fizemos uma classe estatuto onde esta ira ter os individuals restrito, requisitado, livre e atrasado e será referida a cada exemplar através de uma object property temEstatuto, lembrando que é impossível um exemplar a ser requisitado com valor restrito, estará nos exemplares indisponíveis.

Ponto nº 13

Foi criado um individual utente com o nome de Amadeu e dissemos que o Amadeu Requisitou o exemplar do Memorial do Convento. Para adicionarmos também nos registos do exemplar temos de dizer que o exemplar foi RequisitadoPor amadeu.

Sendo o “Requisitou” e o “RequisitadoPor” inversos um do outro.

3 Inferências e Inconsistências

3.1

1. **Traduzidos** - Irá aparecer com o reasenor quais as edições traduzidas dos livros devido no equivalent to estar ([livros and éTraduzidoPara some xsd:string](#)), em cada versão traduzida temos um dataproperty que diz para que língua foi traduzida a versão, isto faz com que quando aquele equivalent é lido no reasoner vá buscar todas as versões que são traduzidas.

Descrição: livros and (temEdiçãoLivro some (éTraduzidoPara some xsd:string))

2. **EmPT500** – Basicamente só precisamos dizer que é um livro e tem o Numero de páginas maior que 500, para isto vamos ao nosso data property temNumeroPaginas e criamos um data type maisDe500Pg (xsd:int[> "500"^^xsd:int]).

Descrição: livros and (temEdiçãoLivro some (temNumeroPaginas some maisDe500Pg))

3. **Autor2Livros** - Basicamente só precisamos de ver qual o Autor que é autor publicação de mais do que um livro.

Descrição: autor and (éAutorPublicação min 2 livros)

4. **AutorEmpDeL** - O que nos fizemos foi, primeiro vimos os autores de contribuições em publicações periódicas de periodicidade diária em código, ([éAutorContribuição some \(éContribuição some \(temJornal some \(temPeriodicidade value diária\)](#)). E depois temos ainda de dizer que são autores de livros ([éAutorPublicação some livros](#)). Temos ainda de dizer que é um autor e a partir daí conseguimos ver todos os autores que correspondem ao desejado.

Descrição: autor and ((éAutorContribuição some (éContribuição some (éEdição some (temPeriodicidade value diária)))) and (éAutorPublicação some Publicação_não_periódica))

5. **Atrasados**- Através do ([\(RequesitadoPor some utente\) and \(temEstatuto value atrasado\)](#)) fomos buscar os exemplares que foram requisitados por alguém e que tinham o estatuto atrasado e depois fomos apenas buscar o utente que requisitou esse mesmo exemplar.

Descrição: utente and (Requisitou some ((RequesitadoPor some utente) and (temEstatuto value atrasado)))

6. **Populares**- Fomos ver quais os exemplares que tinham mais do que 3 requisições na data property temNumeroRequisições (para vermos se era maior que 3 tivemos que fazer um datatype NumeroMaiorQue3) e que estavam com o estatuto atrasado ou requisitado.

Descrição: ExemplarDisp and (((temEstatuto value atrasado) or (temEstatuto value requisitado)) and (temNumeroRequisições some NumeroDeReqMaior3))

1. **coAutores**- fomos ver os autores de um livro e se existissem 2 e um fosse diferente do outro entao eles seriam coAutores.

Descrição: $\text{éAutorPublicação}(?x, ?l) \wedge \text{éAutorPublicação}(?y, ?l) \wedge \text{differentFrom}(?x, ?y) \rightarrow \text{coAutores}(?x, ?y)$

Ou através do rules:

$\text{éAutorPublicação}(?x, ?l), \text{éAutorPublicação}(?y, ?l), \text{DifferentFrom}(?x, ?y) \rightarrow \text{coAutores}(?x, ?y)$

2. **emLinguasDiferentes**- fomos ver se ambos tinham linguagem, se eram diferentes um do outro e se pertenciam a mesma edição, caso isso acontecesse então eram em Linguas Diferentes.

Descrição: $\text{temLinguagem}(?x, ?lp) \wedge \text{temLinguagem}(?y, ?ld) \wedge \text{differentFrom}(?lp, ?ld) \wedge \text{temEdiçãoLivro}(?p, ?y) \wedge \text{temEdiçãoLivro}(?p, ?x) \rightarrow \text{emLinguasDiferentes}(?x, ?y)$

Ou através do rules:

$\text{temLinguagem}(?x, ?lp), \text{temLinguagem}(?y, ?ld), \text{DifferentFrom}(?lp, ?ld), \text{temEdiçãoLivro}(?p, ?y), \text{temEdiçãoLivro}(?p, ?x) \rightarrow \text{emLinguasDiferentes}(?x, ?y)$

3.2

1. Pela publicação periódica ser a única com periodicidade e ser disjoint with a não periódica, sempre que se tentar num livro por um object property temPeriodicidade irá dar erro ao correr o reasoner. Exemplo: Caso adicionarmos no LivroOdisseia Periodicidade por exemplo semanal ao ligar o reasoner dará erro.
2. Irá dar erro pois o object property temEdiçãoOriginal e éEdiçãoOriginal é do tipo funcional para que possa haver no máximo uma relação de saída. Logo sempre que se tente adicionar mais uma edição original tanto do lado da edição (éEdiçãoOriginal) como do lado do livro(temEdiçãoOriginal) irá dar erro. Exemplo: se no exemplar exemplo que nos criamos para fazer testes adicionarmos éEdiçãoOriginal do livroMemorialDoConvento quando ligarmos o reasoner irá dar erro.
3. É impossível alguém requisitar o mesmo exemplar de livro que já está requisitado, porque além de ele estar já no estado requisitado através do funcional cada livro só pode ser requisitado por max 1 utente, só pode haver no máximo uma relação de saída, logo dará erro. Exemplo: Se pusermos por exemplo o Utente João a requisitar um livro já requisitado (MemorialDoConventoEX1) que neste caso foi pelo amadeu irá ser uma inconsistência, logo dará erro quando for ligado o reasoner.

4. Só se pode requisitar livros que tenham estatuto requisitado, livre ou atrasado, logo sempre que alguém tente por uma requisição no livro restrito o reasoner provocara logo um erro.Exemplo: Se no exemplar exemplo que criamos puser estatuto Restrito e tentar fazer RequesitadoPor irá dar erro ao ligar o reasoner. Como os exemplares são os únicos que podem ter atribuições como RequesitadoPor e Requisitou se tentarmos requisitar uma publicação não periódica também dará erro. Exemplo: Se tentarmos requisitar a edição1 com o utente João adicionado por nos por exemplo irá dar erro através do reasoner.

Propriedades/classes Criadas

Criei um novo utente João só para ver o caso dos Inferências atrasado e para ver se as propriedades de que mais do que um utente a requisitar o mesmo exemplar daria mesmo erro ou não. Fizemos alterações pondo o estado do exemplar 1 da odisseia como atrasado e requisitado também pelo João para conseguirmos ver alguns resultados, deixamos este utente porque achamos que é mais fácil para a visualização tanto das inferências como das inconsistências.

Dividimos nos livros as edições traduzidas e não traduzidas para ficar mais fácil de perceber, mas as edições traduzidas é diferente da classe (Traduzidos) pois a edição insere apenas as edições que foram traduzidas, os traduzidos são quais os livros que têm essas edições traduzidas.

Foram adicionados equivalents para o reasenor chegar as instâncias sozinho em praticamente tudo apesar de não ser pedido achamos que ficava mais completo.

Criamos um ExemplarRestrito para ver se caso ele tivesse com estatuto restrito e tentassem requisitar se daria erro, e deu.

Criamos uma ediçãoExemplo para ver se caso fosse posto como uma 2 edição Original se o mesmo daria erro ou não, e deu.

Achamos por bem por apenas os exemplares da biblioteca dentro da biblioteca e dividir esses exemplares em exemplares que já foram requisitados e os que ainda podem ser. Os exemplares requisitados são os exemplares que tenham estatuto value atrasado ou requisitado e os prontos a requisitar que tenham o estatuto value livre. Por isso a inferência dos populares foi para essa secção(ExemplarRequesitado) porque foi o que achamos mais indicado.