



Ontologias: como e porquê criá-las

Julio Cesar Sampaio do Prado Leite
 Karin Koogan Breitman

Departamento de Informática PUC- Rio
 R. Marquês de São Vicente 225
 22453-900 - Rio de Janeiro, Brasil
 www.inf.puc-rio.br/~julio

JAI 1 1

Ontologia

- Estudo do que existe – *being*
- Platão – metafísica
- Aristóteles - 10 categorias
- Ontologia: século XVII

Onto (o que existe) + Logos (conhecimento sobre)

2

Categorias de Aristóteles

Substância	Um gato
Qualidade	O gato é preto
Quantidade	O gato tem 50cm de comprimento
Relação	O gato tem metade do tamanho de um...
Onde	O gato está em casa
Quando	O gato saiu ontem
Posição	O gato está sentado
Possuir	O gato tem um rato
Ação	O gato está correndo
Sentimento	O gato quer leite.

3

Árvore de Porfírio



Supreme genus: Substance
 Differentiae: material immaterial
 Subordinate genera: Body Spirit
 Differentiae: animate inanimate
 Subordinate genera: Living Mineral
 Differentiae: sensitive insentient
 Proximate genera: Animal Plant
 Differentiae: rational irrational
 Species: Human Beast
 Individuals: Socrates Plato Aristotle etc.

Figure 4 Tree of Aristotle's categories drawn by Porphyry

4

Ontologia x ontologia [Guarino98]

- Filosofia
 - Disciplina → Ontologia
- Ciência da Computação
 - Artefato → ontologia

5

Web Semântica

- "A Web Semântica é uma EXTENSÃO da web atual na qual é dada a informação um SIGNIFICADO bem definido, permitindo com que computadores e pessoas trabalhem em cooperação." Berners-Lee, Hendler e Lassila
- "The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation."

6



Web Semântica

- A maior parte dos recursos primários estão em linguagem natural, compreensíveis para humanos.
- Metadados.
- Combinar recursos primários com recursos de metadados.
- Metadados em formato padronizado podem ser entendidos por software e pessoas.



7



Web Semântica - Objetivos

“enriquecer a informação disponível com semântica que pode ser entendida por máquinas”

“fornecer acesso inteligente a informação heterogênea e distribuída, permitindo que produtos de software possam fazer uma intermediação entre as necessidades do usuário e as fontes de informação disponíveis.”



ontologia

8



Definição [Berners-Lee]

“an ontology is a document or file that formally defines the relationship among terms”

“ontologia é um documento ou arquivo que formalmente define os relacionamentos entre termos”

- *Ontologia = taxonomia + regras de inferência*



9



Definição [Gruber93]

“Uma ontologia é uma especificação formal explícita de uma conceitualização compartilhada.”

Conceitualização – modelo abstrato de como as pessoas pensam

Especificação explícita – os conceitos e relacionamentos utilizados neste modelo abstrato são fornecidos através de termos explícitos e bem definidos [Gruninger02]



10



Definição [Maedche02]

$O := \{C, \mathcal{R}, \mathcal{H}^c, rel, \mathcal{A}^o\}$ que consiste de:

- § Dois conjuntos disjuntos, C (conceitos/classes) and \mathcal{R} (relacionamentos)
- § Uma **hierarquia de conceitos**, \mathcal{H}^c : \mathcal{H}^c é um relacionamento direto $\mathcal{H}^c \subseteq C \times C$ chamado hierarquia de conceitos ou taxonomia. $\mathcal{H}^c(C_1, C_2)$ significa C_1 é um sub-conceito de C_2
- § Uma função $rel : \mathcal{R} \rightarrow C \times C$ que relaciona os conceitos de modo não taxonômico
- § Um conjunto de **axiomas** \mathcal{A}^o , expressos em uma linguagem lógica apropriada.



11



Para que serve uma ontologia?

Noy & McGuinness:

- Compartilhar entendimento comum de uma estrutura de informação por pessoas ou agentes de software
- Permitir o reuso de conhecimento de domínio
- Explicitar hipóteses sobre um domínio
- Separar conhecimento de domínio de conhecimento operacional
- Análise



12



Porque construir uma ontologia?

- Explicitar significado
- Integrar recursos na Web



interoperabilidade



13



Conceitos importantes

Conceito/Classe/Entidade

Objetos que possuem um conjunto de características que nos permitem classificá-los como sendo membros deste grupo.



14



Conceitos importantes

Propriedade/Atributo/Role/Slot

Característica



15



Conceitos importantes

Relacionamento/Restriction

Modo pelo qual duas ou mais classes estão relacionadas



16



Conceitos importantes

Generalização X Composição

- Ortogonalidade:
 - Generalização = Tipo de (*is a*)
 - Composição = Parte de (*part of*)

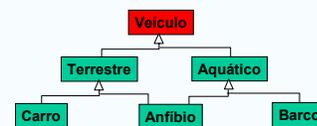


17



Generalização

tipo de



18

Composição

parte de

```

    graph TD
      Lampada[Lâmpada] ---|1| Base[Base]
      Lampada ---|1| Cobertura[Cobertura]
      Lampada ---|1| Interruptor[Interruptor]
      Lampada ---|1| Fiação[Fiação]
  
```

19

Contra exemplo [Welty02]

- Carro & Motor
 - Carro – propriedade – acomoda pessoas
 - Motor – propriedade – gera força rotacional
- Motor não acomoda pessoas...
não é um tipo de carro.

20

Exemplo

Taxonomia dos seres vivos

21

Estrutura de ontologias

- Artefatos com diversos graus de estruturação
 - Taxonomia – Yahoo
 - Metadata Schemes – Dublin core
 - Lógica

22

Tipos de ontologia

23

Enfoques

- Inteligência artificial – Engenharia do conhecimento
 - Esforços voltados ao mapeamento de domínios, criação de grandes bases de conhecimento para mapear o conhecimento humano
 - Construídas por experts
- Web semântica
 - Ontologias voltadas para aplicações específicas
 - Construídas por engenheiros de software

24



IA

- Enfoque top down
- Tempos antigos “*divide et impera*”
- *Sistema e sub-sistema* [von Bertalanffy]
- Dijkstra (programming considered a human activity)
- Especifica as partes individualmente
- Satisfeito? O problema está resolvido?
- Constrói as partes individualmente



se uma das partes ainda é complexa: subdividir

25



Dividir para Conquistar

- Enfoque top down
- Desvantagens (Jackson)
 - escolha mais arriscada possível
 - a maior decisão é qual divisão fazer
 - é tomada muito cedo
 - pouco conhecimento
 - baixa compreensão do problema
 - o mundo real não é organizado de modo hierárquico
 - estruturas paralelas e concorrentes



26



Ontologias enfoque Top-down

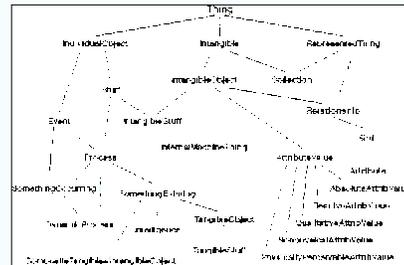
- Cyc upper ontology
 - Base de conhecimento com 3000 termos (termos mais gerais da realidade consensual dos humanos)
- WordNet
 - Banco de dados léxico para a língua inglesa com mais de 42.000 termos



27



CYC



28



Exemplo WordNet

WordNet 1.7.1 Search
 Search word:
 Results for "Synonyms, ordered by estimated frequency" search of noun
 "dessert"

1 sense of dessert

Sense 1
 dessert -- (a dish served as the last course of a meal)
 => course --
 (part of a meal served at one time; "she prepared a three course meal")

Return to [overview for dessert](#)
 Return to [WordNet home](#)



29



Web Semântica

- Grande número de pequenas ontologias interligadas
- Em alguns anos toda empresa, universidade, agência governamental terá seu conteúdo conectado a uma ontologia

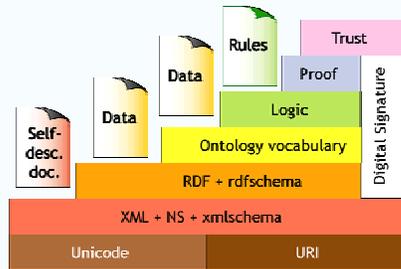
• James Hendler, *Agents and the Semantic Web*



30



Linguagens para Ontologia



31



Linguagens para ontologias na web semântica

- RDF(S) = RDF + RDF Schema-
- XOL (Ontology Exchange Language): <http://www.ai.sri.com/pkarp/xol/>
- SHOE: Simple HTML Ontology Extensions
- <http://www.cs.umd.edu/projects/plus/SHOE/>
- OIL (Ontology Inference Layer) <http://www.ontoknowledge.org/oil/>
- DAML+OIL <http://www.daml.org/2001/03/daml+oil-index>
- OWL (Web Ontology Language) <http://www.w3.org/TR/owl-features/>

32



Evolução

- XML sintaxe para estruturar documentos. Não impõem restrições semânticas.
- XML Schema – restringe a estrutura de documentos XML
- RDF modelo de dados para objetos e relacionamentos. Semântica simplificada
- RDF Schema – vocabulário para descrever propriedades e classes de RDF. Semântica para generalização.
- DAML+OIL, OWL – maior vocabulário para descrever classes e propriedades (relacionamentos, e.g., disjunção, cardinalidade, igualdade)

33



RDF

- RDF - Resource Framework Description – modelo de dados formal e sintaxe
- Codifica metadados em formato processável por máquinas.
- Interoperabilidade entre aplicações
- RDF(S) – primitivas para modelar ontologias simples
- Não é expressivo suficiente (não tem conectivos para negação, disjunção nem conjunção)

34



SHOE

SHOE – extensão de HTML para anotar páginas
 Desenvolvido na Universidade de Maryland
 Informação é embebida nas páginas
 Menos expressivo do que RDF(S)
 Manutenção é problemática
 Projeto foi abandonado

35



OIL

- Ontology inference layer
- Patrocinado pela Comunidade Européia através do On-to-knowledge project.
- Surgiu da necessidade de uma linguagem mais expressiva do que RDF
- Oferece ferramental de edição – O
 - OntoEdit, developed at the University of Karlsruhe
 - Protégé-2000, Stanford University
 - OILed, Manchester University
- Oferece suporte a verificação (pode ser mapeada para lógica de descrição) FaCT
- Fornece uma extensão para RDF e RDF(S) – ontologias em OIL são documentos válidos em RDF.

36



DAML + OIL

- DARPA Agent Markup Language (DAML)
- Desenvolvido pelo Defense Advanced Research Projects Agency (DARPA) em conjunto com o W3C
- DAML-ONT em Outubro de 2000.
- DAML+OIL Em Dezembro de 2000
- Pode ser mapeada para lógica de primeira ordem (escrita ANSI Knowledge Interchange Format (KIF))
- DARPA mantém uma enorme biblioteca digital de ontologias – www.daml.org/ontologies
- Favorita em razão a grande base instalada.



37



DAML+OIL

```

</daml:ontology>
<daml:Class rdf:about="file:C:\Users\anarosa\Cursos\Ontologias\sobremesa.daml#sobremesa">
  <rdf:label>sobremesa</rdf:label>
  <rdf:comment><![CDATA[]]></rdf:comment>
  <olied:creationDate><![CDATA[2003-08-13T20:10:07Z]]></olied:creationDate>
  <olied:creator><![CDATA[anarosa]]></olied:creator>
  <rdf:subClassOf>
    <daml:Restriction>
      <daml:Property
        rdf:resource="file:C:\Users\anarosa\Cursos\Ontologias\sobremesa.daml#Doce"/>
      <daml:hasClass rdf:resource="http://www.w3.org/2000/10/XMLSchema#boolean"/>
    </daml:Restriction>
  </rdf:subClassOf>
</daml:Class>
<daml:Class rdf:about="file:C:\Users\anarosa\Cursos\Ontologias\sobremesa.daml#base_biscoito">
  <rdf:label>base_biscoito</rdf:label>
  <rdf:comment><![CDATA[]]></rdf:comment>
  <olied:creationDate><![CDATA[2003-08-13T20:11:41Z]]></olied:creationDate>
  <olied:creator><![CDATA[anarosa]]></olied:creator>
  <rdf:subClassOf>
    <daml:Class rdf:about="file:C:\Users\anarosa\Cursos\Ontologias\sobremesa.daml#torta">
      </rdf:subClassOf>
  </rdf:subClassOf>

```



38



OWL

- OWL em Março de 2003
- Nova linguagem recomendada pelo W3C
- Revisão do DAML+OIL
- Possui três sublinguagens:
 - OWL Lite
 - Owl DL (Description Logics) mapeamento para DAML+OIL
 - Owl Full
- Não existem ferramentas de edição.



39



Engenharia de ontologias

- Disciplina criada com o objetivo de sustentar o desenvolvimento de ontologias
- Engloba as atividades de:
 - Projeto
 - Construção
 - Avaliação
 - Validação
 - Manutenção
 - Integração
 - Compartilhamento
 - Reutilização



40



Desenvolvimento de ontologias

- Desenvolvidas por especialistas
- Métodos disponíveis: Tove, Methontology, Noy & McGuiness ...
- Maioria é muito vaga ou pesada.



41



Método “Inicial”

- Identificação do Propósito
- Construção
 - Captura
 - Codificação (Formalização)
 - Integração
- Avaliação
- Documentação

Uschold, M., King, M.m 1995, Towards a Methodology for Building Ontologies, Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI'95



42



METHONTOLOGY

- Especificação
- Conceituação
- Integração
- Implementação (formalização)
- Avaliação

Fernández, M., Gómez-Pérez, A., Juristo, N., 1997, METHONTOLOGY: From Ontological Art Towards Ontological Engineering, Ontological Engineering - Working Notes, Stanford, California, March

43



Método de TOVE (TOronto Virtual Enterprise)

- Descrição de Cenários de Motivação
- Definição de Questões de Competência Informais
- Especificação em Lógica de Primeira Ordem
- Descrição Formal das Questões de Competência
- Especificação de Axiomas Formais
- Avaliação

Grüniger, M., Fox, M. S., 1995, Methodology for the Design and Evaluation of Ontologies, Technical Report, University of Toronto.

44



Processo de desenvolvimento de Ontologias

Protégé-2000:



Na realidade - um processo iterativo:



45



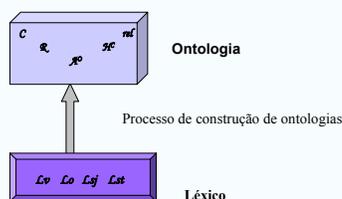
Construção de ontologias baseadas no léxico

- Pode ser realizada por não especialistas
- Agiliza o processo pois é baseado em um processo maduro para a construção de léxicos que já foi validado em inúmeros projetos reais. Modelagem leve (Lighweight).
- Foco na linguagem:
 - Esclarece a separação entre termos específicos de um domínio daqueles que fazem parte do vocabulário mínimo.

46



Do léxico para ontologia



47



Léxico Ampliado da Linguagem

O Léxico Ampliado da Linguagem (LAL) é uma representação utilizada na Engenharia de Requisitos. O Portal de Engenharia de Requisitos (<http://www.er.les.inf.puc-rio.br/>) assim define a área:

“A Engenharia de Requisitos, uma sub-área da Engenharia de Software, estuda o processo de definição dos requisitos que o software deverá atender. A área surgiu em 1993 com a realização do 1 International Symposium on Requirements Engineering. O processo de definição de requisitos é uma interface entre os desejos e necessidades dos clientes e a posterior implementação desses requisitos em forma de software.

Objetivo da ER: entender as necessidades e atender os desejos dos clientes sempre foi colocado como um dos maiores desafios da Engenharia de Software. A postura da Engenharia de Requisitos é a de prover ao Engenheiro de Software, métodos, técnicas e ferramentas que auxiliem o processo de compreensão e registro dos requisitos que o software deve atender. Diferentemente de outras sub-áreas da engenharia de software, a área de requisitos tem que lidar com conhecimento interdisciplinar envolvendo, muitas vezes, aspectos de ciências sociais e ciência cognitiva.”

48



Léxico Ampliado da Linguagem

O LAL [Leite 90] é uma linguagem de representação simples. Porque simples? Porque é composta de apenas 3 entidades básicas: **símbolo** [1], **noção** e **impacto**. Abaixo descrevemos sua sintaxe. O LAL é uma linguagem de representação da Engenharia de Requisitos que tem por objetivo mapear o vocabulário utilizado no Udl.

[1] Também referenciado como **termo**.

49



Léxico Ampliado da Linguagem

- Fundamentos:
 - A Teoria da Semiótica de Eco: a idéia de símbolos e sua contextualização.
 - A Escola de Vienna (A linguagem das observações de Carnap): a idéia de que os termos de uma linguagem devem ser ancorados em fenômenos observáveis.
 - Sociologia: a idéia de que a linguagem é uma reflexão da cultura.

50



Léxico Ampliado da Linguagem

- Fundamentos:
 - Aspectos Sociais da Computação (Kling): a interação entre atores sociais e sistemas computacionais.
 - Linguagens de Domínio (Neighbors): reutilização em alto nível de abstração, similar ao conceito de "little languages" de Bentley.
 - Hyperties (Shneiderman): uma das primeiras implementações de hipertexto.

51



Léxico Ampliado da Linguagem

- Baseado num sistema de códigos :
 - Símbolos
 - Noções (Denotação)
 - Impactos (Conotação)
- Cada símbolo é descrito por seus impactos e suas noções.
- Símbolo (termo) = entrada no léxico

52



Exemplo

Mesa/ Mesas

Noções:

Um dos componentes do restaurante.

Cada mesa possui um no. de mesa.

Impactos:

O cliente pode sentar na mesa.

O cliente pode trocar de mesa.

A mesa está aberta ou a mesa está fechada.

53



Exemplo

Garçon/Garçons

Noções:

Pessoa que trabalha no restaurante.

Responsável pela comunicação entre os clientes e o caixa.

Impactos

Realiza as tarefas: jogar a comanda, entregar o pagamento.

54



Exemplo

Abrir a mesa/ Abre mesa/ Abriu mesa

Noções:

Tarefa realizada pelo caixa.

Acontece quando o cliente senta na mesa e faz o pedido.

Caixa verifica se a mesa está fechada.

Caixa recebe a comanda e bota a comanda no escaninho.

Impactos

A mesa está aberta.

Se a mesa está aberta, o caixa não pode abrir a mesa. Caixa avisa o garçon que a mesa está aberta.

55



Léxico Ampliado da Linguagem

LAL: Representação de símbolos na linguagem de aplicação.

Sintaxe:

{Símbolo},ⁿ

Símbolo: entrada do léxico que tem um significado especial no domínio de aplicação.

Sintaxe:

{Nome},ⁿ + {Noção},ⁿ + {Impacto},ⁿ

Nome: identificação do símbolo. Mais de um símbolo representa sinónimos.

Sintaxe:

Palavra | Frase

Noção: denotação do símbolo. Tem que ser expresso usando referências a outros símbolos e usando o princípio do vocabulário mínimo.

Sintaxe:

Sentença

Impacto: conotação do símbolo. Tem que ser expresso usando referências a outros símbolos e usando o princípio do vocabulário mínimo.

Sintaxe:

Sentença

Nota: Sentença é composta de Símbolo e Não-Símbolos. Não-Símbolos devem pertencer a um vocabulário mínimo.

56



Léxico Ampliado da Linguagem



Figura 3.2 - Exemplos de entradas no LAL.

57



Símbolos do Léxico

Propriedades dos símbolos:

- Cada símbolo tem zero ou mais sinónimos.
- Cada símbolo tem uma ou mais noções.
- Cada símbolo pode ter um ou mais impactos.

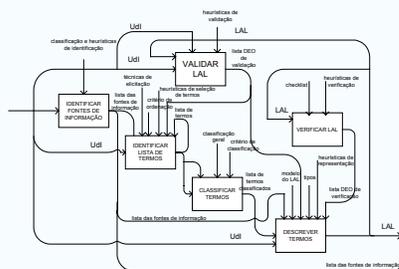
Conceitos importantes:

- Circularidade
- Vocabulário Mínimo

58



Processo de Construção do LAL



59



Processo de Construção do LAL

- Realizar entrevistas não-estruturadas com atores do domínio.
- Listar todas as palavras ou frases "significativas".
- Realizar entrevistas estruturadas, a lista serve de guia.
- Pergunte aos atores o significado do símbolo
- Escrever sentenças curtas e simples distinguindo entre noções e impactos.
- Orações simples.
- Voz ativa.
- Forma verbal.
- Questione o significado diferenciando entre noções e impactos
- Faça a representação do léxico.
- Verifique o Léxico usando inspeções
- Valide o resultado com uma entrevista estruturada ou uma reunião

60



Regras de Formação do LAL

	NOÇÃO	IMPACTO
Sujeito	Quem é o sujeito?	Quais ações são feitas?
Verbo	Quem faz, quando acontece e que procedimentos estão envolvidos	Quais são os impactos da ação no ambiente (LAI), isto é, que outras ações também ocorrem, e quais são os estados resultantes.
Objeto	Define o objeto e identifica outros objetos com os quais ele se relaciona.	Ações que são aplicadas ao objeto.
Estado	O que significa e quais ações levaram a este estado.	Identifica outros estados e ações que podem ocorrer a partir do estado aqui descrito.

©Julio Cesar Sampaio do Prado Leite

61



Verificação

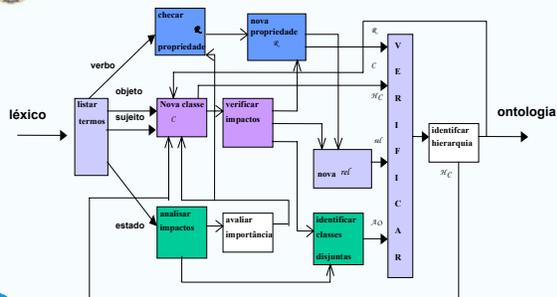
GRUPO	DEFEITOS
Descrição	<ul style="list-style-type: none"> • Símbolos mal descritos • Símbolos incompletos • Descrição incompatível com o tipo
Classificação	<ul style="list-style-type: none"> • Classificação incorreta
Identificação	<ul style="list-style-type: none"> • Símbolos omitidos • Símbolos incorretos • Símbolos incorretamente incluídos
Referência	<ul style="list-style-type: none"> • Falta de referências a outros símbolos • Mal uso de símbolos

Tabela 2 Classificação de Defeitos [Kaplan 00]

62



Processo



63



Processo

- Listar os termos alfabeticamente de acordo com seu tipo (verbo, objeto, sujeito ou estado)
- Fazer 3 listas: conceito (classe) (C), propriedade (R) and axiomas (A⁰). Na lista de classes cada entrada terá um nome, descrição (língua natural) e uma lista contendo de zero ou mais *ref* (função que relaciona o conceito em questão a outros, de maneira não taxonômica). As entradas na lista de axiomas terão nomes (labels) somente.
 - Utilizando a lista de símbolos do léxico classificados como sujeito ou objeto, para cada termo:
 - Adicione uma nova classe a lista de classes. O nome da classe é o símbolo do léxico propriamente dito. A descrição da classe é a noção do termo.
 - Para cada impacto,
 - Checar se já faz parte da lista de propriedades da ontologia.
 - Caso não faça parte da lista (a propriedade não existe), adicione uma nova propriedade na lista (de propriedades). O nome da propriedade deve ser baseado no verbo utilizado para descrever o impacto.
 - Verificar consistência.
 - Na lista de classes adicione uma nova *ref* para a classe em questão. A *ref* é formada pela classe + a propriedade (definida em 3.1.1.1) + outra classe relacionada o (esta classe é o objeto direto/indireto do verbo utilizado no impacto do símbolo do léxico. Usualmente é um termo do próprio léxico e aparece sublinhado).

64



Processo

- Checar se existem indicativos de negação no vocabulário mínimo que relacionem duas ou mais classes. Verificar se estas classes possuem um relacionamento do tipo *disjuntas* (exemplo macho e fêmea).
 - Se verdadeiro, adicionar o *disjoint* a lista de axiomas.
 - Verificar consistência.
 - Utilizando a lista de símbolos classificados como tipo verbo, para cada termo:
 - Checar se já faz parte da lista de propriedades da ontologia.
 - Caso não faça parte da lista (a propriedade não existe), adicione uma nova propriedade na lista (de propriedades). O nome da propriedade é o símbolo do léxico propriamente dito.
 - Verificar consistência.
 - Utilizando a lista de símbolos classificados como tipo estado, para cada termo:
 - Para cada impacto
 - Tentar identificar a importância relativa do termo para a ontologia. Esta estratégia é similar a utilização de questões de competência proposta em [Gruninger95]. Estas questões são obtidas através do refraseamento dos impactos de cada símbolo em perguntas iniciadas por quando, onde, o quê, quem, porque, e como.

65



Processo

- Checar se existem indicativos de negação no vocabulário mínimo que relacionem duas ou mais classes. Verificar se estas classes possuem um relacionamento do tipo *disjunto* (exemplo macho e fêmea).
 - Se verdadeiro, adicionar o *disjoint* a lista de axiomas.
 - Caso o termo seja central a ontologia, classifique-o como classe (C).
 - Caso contrário (o termo não é central para a ontologia) classifique-o como propriedade (R).
 - Verificar consistência.

66

Exemplo do processo

Substituto tipo: sujeito

Noções:
Pessoa que está presente na reunião no lugar de outra. Ele ou ela foram escolhidos pela pessoa convidada para a reunião.

Impactos:
Se não pode estar presente na reunião, manda notificação. Se puder estar presente na reunião, manda confirmação. E aprovação/ participação da reunião.

Símbolo do Léxico

Processo

3.1 Adicione uma nova classe a lista de classes. O nome da classe é o símbolo do léxico propriamente dito. A descrição da classe é a noção do termo.
3.1.1 Para cada impacto,

Substituto

Descrição: (Documentação em DAML+ OIL): Pessoa que está presente na reunião no lugar de outra. Ele ou ela foram escolhidos pela pessoa convidada para a reunião.

ontologia

Exercício: Elaborar a ontologia de sobremesas do restaurante Couve Flor

- Oferece 4 tipos de sobremesas: tortas, pudins, musses e frutas
 - As tortas podem ser de chocolate ou frutas. Podem ter um ou mais recheios e cobertura opcional.
 - Somente são servidas musses de frutas
 - As frutas, servidas naturais ou em forma de compota são: morango, abacaxi, maçã.
 - Algumas sobremesas podem ser servidas quentes.

68

Léxico das sobremesas

<p>Sobremesa tipo: objeto</p> <p>Noções: Alimento doce geralmente servido após a refeição.</p> <p>Impactos: Pode ser torta, musse, pudim ou fruta.</p>	<p>Base Biscoito/ Torta de biscoito/Biscoito/Biscoitos tipo: objeto</p> <p>Noções: Sobremesa de corte cuja massa é elaborada com biscoitos ao invés de farinha de trigo.</p> <p>Impactos: Tem massa de biscoito.</p>	<p>Pudim tipo: objeto</p> <p>Noções: Sobremesa láctea.</p> <p>Impactos: É servido frio. Tem calda de fruta.</p>	<p>Doce tipo: estado</p> <p>Noções: Temperaturas entre 5° e 15°</p> <p>Impactos: Sobremesas frias são mantidas na geladeira.</p>
<p>Torta/Tortas tipo: objeto</p> <p>Noções: Sobremesa de corte de sabores variados.</p> <p>Impactos: Pode ser de fruta ou chocolate. Pode ter cobertura. Pode ser servida fria ou quente. Pode ter um ou mais recheios.</p>	<p>Base Massa/ Torta de Massa/Massa tipo: objeto</p> <p>Noções: Sobremesa de corte cuja massa é elaborada com farinha de trigo.</p> <p>Impactos: Tem massa de pão de ló.</p>	<p>Fruta/Frutas tipo: objeto</p> <p>Noções: Sobremesa natural perecível.</p> <p>Impactos: Pode ser maçã, morango e abacaxi.</p>	<p>Musse tipo: objeto</p> <p>Noções: Sobremesa leve.</p> <p>Impactos: Sabores de frutas. É apresentada fria.</p>
<p>Natural tipo: estado</p> <p>Noções: Estado da fruta quando nenhum processo foi aplicado.</p> <p>Impactos: Servida fria</p>	<p>Escolher sobremesa tipo: verbo</p> <p>Noções: Ato de selecionar a sobremesa desejada.</p> <p>Impactos: Escolha restrita a tortas, musses, frutas ou pudim.</p>	<p>Compota tipo: objeto</p> <p>Noções: Sobremesa feita a partir de frutas.</p> <p>Impactos: É cozida.</p>	<p>Quente tipo: estado</p> <p>Noções: Temperaturas entre 15° e 30°</p> <p>Impactos: Sobremesas são aquecidas no microondas.</p>

69

Léxico das sobremesas

<p>Fria tipo: estado</p> <p>Noções: Temperaturas entre 5° e 15°</p> <p>Impactos: Sobremesas frias são mantidas na geladeira.</p>	<p>Quente tipo: estado</p> <p>Noções: Temperaturas entre 15° e 30°</p> <p>Impactos: Sobremesas são aquecidas no microondas.</p>	<p>Doce tipo: estado</p> <p>Noções: Sensação gustativa</p> <p>Impactos: Todas as sobremesas são doces</p>	<p>Musse tipo: objeto</p> <p>Noções: Sobremesa leve.</p> <p>Impactos: Sabores de frutas. É apresentada fria.</p>
<p>Natural tipo: estado</p> <p>Noções: Estado da fruta quando nenhum processo foi aplicado.</p> <p>Impactos: Servida fria</p>	<p>Escolher sobremesa tipo: verbo</p> <p>Noções: Ato de selecionar a sobremesa desejada.</p> <p>Impactos: Escolha restrita a tortas, musses, frutas ou pudim.</p>	<p>Compota tipo: objeto</p> <p>Noções: Sobremesa feita a partir de frutas.</p> <p>Impactos: É cozida.</p>	<p>Torta Musse tipo: objeto</p> <p>Noções: Sobremesa leve feita de base de massa e recheada de musse.</p> <p>Impactos: É apresentada fria.</p>

70

Processo

1. Listar os termos alfabeticamente de acordo com seu tipo (objeto, sujeito, verbo ou estado)

Objeto	Sujeito	Verbo	Estado
Base biscoito		Escolher sobremesa	Doce
Base massa			Fria
Compota			Natural
Fruta			Quente
Musse			
Pudim			
Sobremesa			
Torta musse			
Torta			

71

Processo

2. Fazer 3 listas: conceito (classe) (C), propriedade (R) and axiomas (A⁰).
Na lista de classes cada entrada terá um nome, descrição (linguagem natural) e uma lista contendo de zero ou mais rel (função que relaciona o conceito em questão a outros, de maneira não taxonômica).
As entradas na lista de axiomas terão nomes (labels) somente.

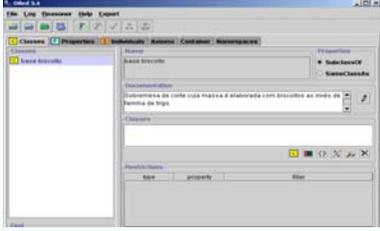
Classe	Propriedade	Axioma
Nome: Descrição: Restrições:	Nome:	Nome:
Nome: Descrição: Restrições:	Nome:	Nome:
Nome: Descrição: Restrições:	Nome:	Nome:

72

Processo

3. Utilizando a lista de símbolos do léxico classificados como sujeito ou objeto, para cada termo:

- 3.1 Adicione uma nova classe a lista de classes. O nome da classe é o símbolo do léxico propriamente dito. A descrição da classe é a noção do termo. 3.1.1 Para cada impacto,



Base Massa/ Torta de Massa/Massa
tipo: objeto

Noções:
Sobremesa de corte cuja massa é elaborada com farinha de trigo.

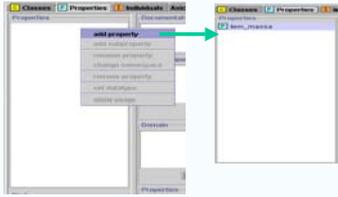
Impactos:
Tem massa de pão de ló.

73

Processo

3.1.1.1 Checar se já faz parte da lista de propriedades da ontologia.

3.1.1.2 Caso não faça parte da lista (a propriedade não existe), adicione uma nova propriedade na lista (de propriedades). O nome da propriedade deve ser baseado no verbo utilizado para descrever o impacto



Base Massa/ Torta de Massa/Massa
tipo: objeto

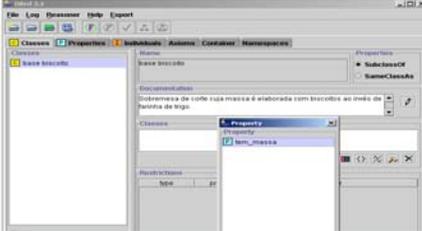
Noções:
Sobremesa de corte cuja massa é elaborada com farinha de trigo.

Impactos:
Tem massa de pão de ló.

74

Processo

3.1.1.3 Na lista de classes adicione uma nova *ref* para a classe em questão. A *ref* é formada pela classe + a propriedade (definida em 3.1.1.1) + outra classe relacionada o (esta classe é o objeto direto/indireto do verbo utilizado no impacto do símbolo do léxico. Usualmente é um termo do próprio léxico e aparece sublinhado).



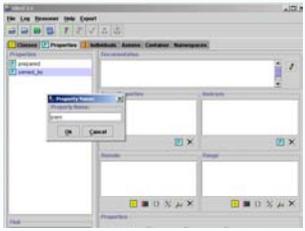
75

Process

4. Using the list of lexicon terms classified as type verb, for each term:

4.1.1 Check the relation list for a relation that expresses it.

4.1.1.1 If there is none, add new a relation to the relation list. The relation name is the lexicon term itself.



Pays
type: verb

Notions:
an action performed by the customer after the customer informs the total, the customer handles the payment to the restaurant.

Behavioral Responses:
if the amount is more than the total, the assistant has to provide change.
a receipt is printed after the payment.

Example 76

76

Processo

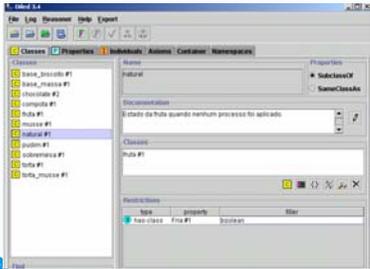
5. Utilizando a lista de símbolos classificados como tipo estado, para cada termo:

- 5.1.1 Para cada impacto
 - 5.1.1.1 Tentar identificar a importância relativa do termo para a ontologia. Esta estratégia é similar a utilização de questões de competência proposta em [Gruninger95]. Estas questões são obtidas através do refaseamento dos impactos de cada símbolo em perguntas iniciadas por quando, onde, o quê, quem, porque, e como.

77

Estado → Classe

5.1.2 Caso o termo seja central a ontologia, classifique-o como classe (C).



Natural
tipo: estado

Noções:
Estado da fruta quando nenhum processo foi aplicado.

Impactos:
Servida frut

78

Estado Propriedade

5.1.3 Caso contrário (o termo não é central para a ontologia) classifique-o como propriedade (Ø).

Fria
tipo: estado
Noções:
 Temperaturas entre 5° e 15°
Impactos:
 Sobremesas **frias** são mantidas na geladeira.

Estrutura da ontologia

```

graph TD
    sobremesa#1 --> fruta#1
    sobremesa#1 --> compota#1
    sobremesa#1 --> natural#1
    sobremesa#1 --> musse#1
    sobremesa#1 --> torta_musse#1
    sobremesa#1 --> pudim#1
    sobremesa#1 --> flan#1
    sobremesa#1 --> torta#1
    torta#1 --> base_biscoito#1
    torta#1 --> base_massa#1
  
```

Ferramenta C&L português

Informações sobre o léxico

Nome: quanto
Temperatura: Temperaturas entre 15 graus e 30 graus Celsius.
Classificação: 111.120
Impacto(s): Sobremesas são aquecidas no microondas.
Simbolismo:

Alterar Léxico Remover Léxico

Cenários e termos do léxico que referenciam este

Cenários: tortas
Termos: torta

Veja o código fonte!

Ferramenta C&L português

Conceitos: quanto

Verbo: sobremesas são aquecidas no microondas.

Impacto: sobremesas são aquecidas no microondas.

O predicado do verbo existe na lista abaixo?

sim não

Se não existe, ele pertence a lista de elementos da nossa nomenclatura?

sim não

LAL Ontologia

Adicionando Sujeitos e Objetos

- Para cada conceito, separar seu sujeito, verbo e predicado (esse procedimento se repetirá para todos os impactos do conceito).
- Processo de separação do Verbo:
 - Verificar se o verbo está cadastrado na lista de verbos da Ontologia.
 - Em caso afirmativo, selecioná-lo, marcar a opção "sim" e clicar em "Inserir Verbo".
 - Caso contrário, marcar a opção "não", digitá-lo e clicar em "Inserir Verbo".
 - Para finalizar, clicar em "Próximo Passo >>>".

LAL Ontologia

Adicionando Sujeitos e Objetos

- Processo de separação do Predicado:
 - Verificar se o predicado está cadastrado na lista de predicados da Ontologia.
 - Em caso afirmativo, selecioná-lo, marcar a opção "sim" e clicar em "Inserir Predicado".
 - Caso contrário, marcar a opção "não".

LAL Ontologia

Adicionando Sujeitos e Objetos

- Processo de separação do **Predicado**:
 - Verificar se o predicado está cadastrado na lista de elementos do namespace da Ontologia.
 - Em caso afirmativo, selecioná-lo e clicar em “Inserir Predicado”.
 - Caso contrário, digitá-lo e clicar em “Inserir Predicado”.
 - Para finalizar, clicar em “Próximo Passo >>”



85

LAL Ontologia

Adicionando Sujeitos e Objetos

- Processo de inserção de **Axioma**:
 - Verificar se o termo está cadastrado na lista do vocabulário mínimo
 - Em caso afirmativo, selecioná-lo e clicar em “Inserir Axioma”.
 - Caso contrário, digitá-lo e clicar em “Inserir Axioma”.
 - Para finalizar, clicar em “Próximo Passo >>”



86

LAL Ontologia

Adicionando Estados

- Verificar se o termo é Classe ou Propriedade
- Marcar uma das opções e clicar em “OK”.



87

LAL Ontologia

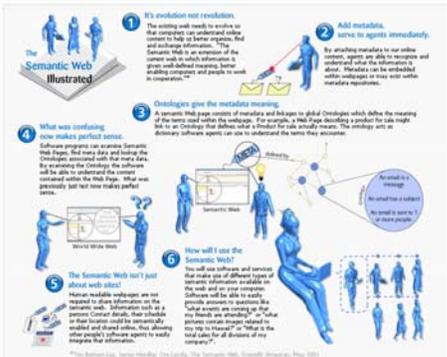
Criando a Hierarquia de Sub-Conceptos

- Escolher um dos conceitos da coluna da esquerda (à direita da árvore da ontologia) e relacioná-lo com todos os seus possíveis sub-conceitos na coluna da direita.
- Após ter sido feita a seleção, pressionar o botão “Incluir Relação”.
- Após incluir todas as relações desejadas, clicar em “Finalizar Algoritmo”.



88

The Semantic Web Illustrated



- It's evolution not revolution.** The existing web needs to evolve so that computers can understand what content is for better organization, find and exchange information. The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation.
- Add metadata.** By attaching metadata to our online content, agents are able to recognize and understand what the content is about. Metadata can be embedded within webpages or may refer to metadata repositories.
- Ontologies give the metadata meaning.** A semantic Web page consists of metadata and links to global Ontologies which define the meaning of the terms used within the webpage. For example, a Web page describing a product for sale might use an Ontology that defines what is meant by the terms "product". The ontology acts as dictionary software agents can use to understand the terms they encounter.
- What was confusing, now makes perfect sense.** Software programs can examine Semantic Web pages, find more data and linking the information associated with our web pages. By examining the Ontology, the software will be able to understand the content contained within the Web Page. What was previously just text now makes perfect sense.
- The Semantic Web isn't just about web sites!** Human readable webpages are not required to drive information on the Semantic Web. Information such as a person's Contact details, their schedule or their location could be semantically enabled and shared online. Their sharing programs that information.
- How will we use the Semantic Web?** We will use software and services that make use of different types of semantic information available on the web and on our computers. Software will be able to notify us of new information. For example, if our favorite sports are coming up, the software could notify us. Other software could be used to help us find information related to things we are interested in. "What is the total sales for all divisions of the company?"

89

Busca de Informação

- Utilizar ontologias para reduzir o universo de opções em buscas na web
 - Ontologias restringem o significado de termos utilizados em buscas por palavras chave
 - Auxiliam na montagem da query

AVI

90



Outros usos

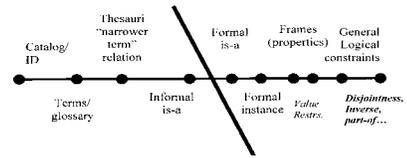
- Ciência da Informação
- Gestão do Conhecimento
- Engenharia de Software

91



Diferenciar

- tesouro, vocabulário, dicionário, léxico
- taxonomia
- ontologia



92



Ciência da Informação

- O termo *thesaurus* é definido pela UNESCO (<http://www.ulcc.ac.uk/unesco/index.htm>) como “a structured and defined list of terms which standardizes words used for indexing.” Ainda segundo a UNESCO, um tesouro geralmente caracteriza-se por:
 - fornecer um termo preferido;
 - agrupar termos relacionados em uma hierarquia, indicando quais são os termos mais gerais e quais são os mais específicos;
 - acrescentar referências cruzadas entre grupos de termos relacionados;
 - permitir atualização para refletir mudanças no uso dos termos do tesouro.

93



Gestão do Conhecimento

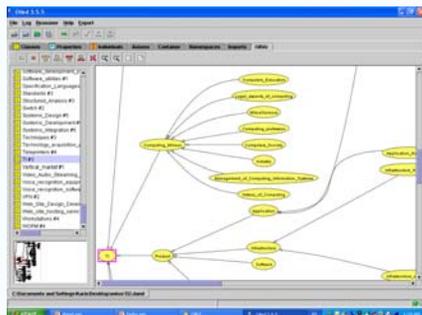
- Ferramenta (Exemplo) <http://www.stratify.com/>

“Stratify stands out because its new solution moves beyond the bits & bytes of taxonomies, classification and metadata.”
Winner of K&RWorld's prestigious *Tread-setter of the Year* award
- Conteúdo (Exemplo) <http://www.itscout.org/itscout/>
 - FlashAtlas includes taxonomies
 - IT Infrastructure
 - Software Development
 - Business Intelligence
 - COTS Applications

94



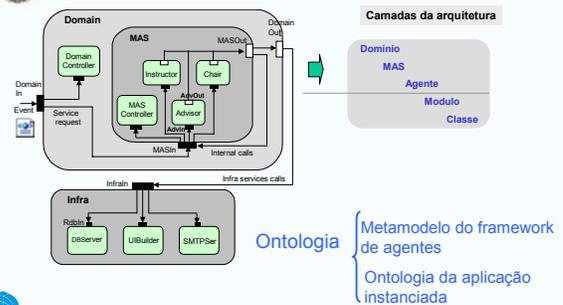
Gestão do Conhecimento



95



Engenharia de Software



96

Engenharia de Software

- <Domain_name="AcademicApplications">
- <Interface>
- + <Provided>
- + <Required>
- <Interface>
- <Service>
- + <Service_name="Receive
Advisement Requests">
- + <Service_name="Provide Request
Status">
- </Service>
- <Instance>
- + <DomainController>
- + <MAS_name="Electronic
Advisement">
- </Instance>
- <Domain_name>

XML
Hierarchical Structure - ontology

97

Metamodel Class

98

Referências

- [Berners-Lee02] – Berners-Lee, T.; Lassila, O. Hendlar, J. – The Semantic Web – Scientific American - <http://www.scientificamerican.com/2001/0501issue/0501berners-lee.html>
- [Fernandez-Lopez97]- M. Fernandez, A. Gomez-Perez, and N. Juristo. METHONTOLOGY: From Ontological Arts Towards Ontological Engineering. In Proceedings of the AAAI97 Spring Symposium Series on Ontological Engineering, Stanford, USA, pages 33–40, March 1997.
- [Wetly02] - Wetly, C.; Guarino, N. - Evaluating Ontological Decisions with ONTOCLEAN - Communications of the ACM - Vol. 45 No. 2 - February 2002 pp.61-65
- [Gruninger02] – Gruninger, M.; Lee, J. – Ontology applications and design - Communications of the ACM - Vol. 45 No. 2 - February 2002 pp.39-41
- [Maedche02] – Maedche, A. – Ontology Learning for the Sematic Web – Kluwer Academic Publishers – 2002.
- [Noy01] – Noy, N.; McGuinness, D. – Ontology Development 101 – A guide to creating your first ontology – KSL Technical Report, Stanford University, 2001.
- [Gruber93] – Gruber, T.R. – A translation approach to portable ontology specifications – Knowledge Acquisition – 5: 199-220
- [Hendler01] - Hendler, J. – Agents and the Semantic Web – IEEE Intelligent Systems – March/April - 2001. pp.30-37
- [Guarino98] – Guarino, N. – Formal Ontology and information systems – In Proceedings of the FOIS'98 – Formal Ontology in Information Systems, Trento, 1998.

99

Bibliografia recomendada

Livros

- [Davies03] – Davies, J., Fensel, D.; Hamellen, F.V., editors – Towards the Semantic Web: Ontology Driven Knowledge management – Wiley and Sons – 2003.
- [Fensel01] – Fensel, D. – Ontologie: a silver bullet for knowledge management and electronic commerce – Springer, 2001
- [Fensel03] – Fensel, D.; Wahlster, W.; Berners-Lee, T., editors – Spinning the Semantic Web – MIT Press, Cambridge Massachusetts, 2003.
- [Geroimenko03] – Geroimenko, V.; Chen, C.; editors – Visualizing the Semantic Web: XML Based Internet and Information Visualization – Springer, 2003.
- [Hjelm01] - Hjelm, H. – Creating the Semantic Web with RDF – Wiley- 2001
- [Maedche02] – Maedche, A. – Ontology Learning for the Sematic Web – Kluwer Academic Publishers – 2002.
- [Sowa00] – Sowa, J. F. – Knowledge Representation: Logical, Philosophical and Computational Foundations – Brooks/Cole Books, Pacific Grove, CA – 2000.

100

Bibliografia recomendada

Métodos

- [Fernandez-Lopez97]- M. Fernandez, A. Gomez-Perez, and N. Juristo. METHONTOLOGY: From Ontological Arts Towards Ontological Engineering. In Proceedings of the AAAI97 Spring Symposium Series on Ontological Engineering, Stanford, USA, pages 33–40, March 1997.
- [Gómez-Pérez98] – Gómez-Pérez, A. – Knowledge sharing and reuse – in The Handbook of Applied Expert Systems. CRC Press, 1998
- [Gruninger95] – Gruninger, M.; Fox, M. – Methodology for the Design and Evaluation of Ontologies: Proceedings of the Workshop on basic Ontological Issues in Knowledge Sharing – IJCAI-95 Canada, 1995.
- [Noy01-a] – Noy, N.; McGuinness, D. – Ontology Development 101 – A guide to creating your first ontology – KSL Technical Report, Stanford University, 2001.
- [Sure03] – Sure, Y.; Studer, R. – A methodology for Ontology based knowledge management in Davies, J., Fensel, D.; Hamellen, F.V., editors – Towards the Semantic Web: Ontology Driven Knowledge management – Wiley and Sons – 2003. pp. 33-46.
- [Ushold96] - Ushold, M.; Gruninger, M. – Ontologies: Principles, Methods and Applications. Knowledge Engineering Review, Vol. 11 No. 2 – 1996. pp. 93-136

101

Bibliografia recomendada

Ferramentas

- [OIEd site] - <http://oied.man.ac.uk/>
- [Erdmann02] – Erdmann, M.; Angele, J.; Staab, S.; Studer, R. - OntoEdit: Collaborative Ontology Development for the Semantic Web - Proceedings of the first International Semantic Web Conference 2002 (ISWC 2002), June 9-12 2002, Sardinia, Italia
- [Bechhofer01] - Sean Bechhofer, Ian Horrocks, Carole Gobin, Robert Stevens. OilEd: a Reason-able Ontology Editor for the Semantic Web. Proceedings of KR2001, Joint German/Austrian conference on Artificial Intelligence, September 19-21, Vienna. Springer-Verlag LNAI Vol. 2174, pp 396–408. 2001.
- [McGuinness02] – McGuinness, D.; Fikes, R.; Rice, J.; Wilder, S. – An Environment for Merging and Testing Large Ontologies – Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR-2000), Breckenridge, Colorado, April 12-15, San Francisco: Morgan Kaufmann. 2002. pp.483-493.
- [Chimaera00] – Chimaera ontology environment - <http://www.sanford.edu/software/chimaera>
- [FACT] – Fast Classification of Terminologies – TOOL - <http://www.cs.man.ac.uk/~horrocks/FaCT/>
- [Noy01-b] – Noy, N.; Sintek, M.; Decker, S.; Crubezy, R.; Fergerson, R.; Musen, A. – Creating Semantic Web Contents with Protégé 2000 – IEEE Intelligent Systems Vol. 16 No. 2, 2001. pp. 60-71

102



Bibliografia recomendada

Linguagens

- [Farquhar97] - Farquhar, A. - Ontologia tutorial - <http://ksl.stanford.edu/ontology/edu/ontology.html>
- [Genesereth91] - M. R. Genesereth: Knowledge interchange format. In J. Allen, R. Fikes, and E. Sandewall, editors, Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference (KR'91). Morgan Kaufmann Publishers, San Francisco, California, 1991
- [Heflin01] - Heflin, J.; Hendler, J. - A portrait of the Semantic Web in Action - IEEE Intelligent Systems - March/April - 2001. pp.54-59.
- [Hendler00] - Hendler, J.; McGuinness, D. - The DARPA agent Markup Language. IEEE Intelligent Systems. Vol 16 No 6, 2000. pp.67-73.
- [McGuinness03] - McGuinness, D.; Harmelen, F. - OWL Web Ontology Overview - W3C Working Draft 31 March 2003



103



Bibliografia recomendada

- [Buranarach 01] - Buranarach, M. The Foundation for Semantic Interoperability on the World Wide Web - Doctoral thesis - Department of Information Science and Telecommunications - School of Information Sciences - University of Pittsburgh - November 8, 2001.
- [Everett02] - Everett, J.O; Bobrow, D.; Stolle, R.; Crouch, R.; Paiva, V.; Condoravdi, C.; van der Berg, M.; Polanyi, L. - Making Ontologies Work for Resolving Redundancies Across Documents - Communications of the ACM, Vol. 45 No. 2 - February 2002
- [Gandon02] - Gandon, F. - Ontology Engineering: a synthesis - Project Acacia - INRIA Technical Report 4396 - March 2002 - 181 pages
- [Gruninger02] - Gruninger, M.; Lee, J.; - Introduction to the Ontology Application and Design section - guest editors - Communications of the ACM - February, Vol. 45, No.2 February 2002 - pp.39-41.
- [Guarino02] - Guarino, N.; Welty, C.; - Evaluating Ontological Decisions with Ontoclean- Communications of the ACM, Vol. 45 No. 2 - February 2002 - pp.61-65



104



Bibliografia recomendada

- [Horrocks99] - I. Horrocks, U. Sattler, and S. Tobies: Practical reasoning for expressive description logics. In H. Ganzinger, D. McAllester, and A. Voronkov, editors, Proceedings of the 6th International Conference on Logic for Programming and Automated Reasoning (LPAR'99), number 1705 in Lecture Notes in Artificial Intelligence, pages 161-180. Springer-Verlag, 1999.
- [Lamsweerde95] - Goal-directed elaboration of requirements for a meeting scheduler: problems and lessons learnt - in the *Proceedings of the Second IEEE International Symposium on Requirements Engineering (RE'95)* - York, March 27 to 29 - IEEE Computer Society Press, 1995 - pp.194-203
- [Leite90] - Leite, J.C.S.P.; Franco, A. P. - O uso de hipertexto na elicitação de linguagens de aplicação - em Anais do 4º Simpósio Brasileiro de Engenharia de Software - editado pela Sociedade Brasileira de Computação - pp.124-133 - 1990.
- [Leite93] - Leite, J.C.S.P.; Franco, A.P.M.Franco - A Strategy for Conceptual Model Acquisition. Proceedings of the IEEE International Symposium on Requirements Engineering, IEEE Computer Society Press, Pags. 243-246, San Diego 1993. [Potts97] - Potts, C. - Requirements Models in Context - Proceedings of the third IEEE Symposium on Requirements Engineering RE97 - Annapolis, Maryland - 1997 - pp.102-104
- [Prieto-Diaz91] - Prieto-Diaz, R., and Arango, G. (eds.) *Domain Analysis and Software Systems Modeling*. Los Alamitos, Ca.: IEEE Computer Society Press, 1991



105



Sites interessantes

- W3C
 - <http://www.w3.org>
- Curso de Web Semântica
 - <http://www.cse.lehigh.edu/~heflin/courses/sw-fall01/>
- Projeto OnToKnowledge
 - <http://www.ontoknowledge.org>
- OIL
 - <http://www.ontoknowledge.org/oil/>
- Prof. James Hendler
 - <http://www.cs.umd.edu/~hendler/>



106