

Centro Universitário Feevale

ONTOLOGIA DE DOMÍNIO PARA ANÁLISE DE *BLOGS*

Fernando Mertins

Prof. Orientador: Rodrigo Rafael Villarreal Goulart

Dezembro de 2007



Roteiro da Apresentação

- Introdução
- Objetivos
- Metodologia
- Ontologia
 - Construção
 - Experimentos
- Conclusões
- Trabalhos Futuros

Introdução

- Problemática e motivação
 - Principais mecanismos de busca da internet não consideram o significado da informação
 - Exemplos: *carvalho, manga, aparelho de porcelana*
 - Conseqüências: encontra itens irrelevantes e pode descartar itens importantes
- Web Semântica
 - Informação para máquinas e humanos
 - Ontologias são apenas um dos componentes

Introdução

- Ontologia (Ciência da Computação)

“Especificação formal e explícita de uma conceitualização compartilhada.” [Gruber]

- Componentes
 - Classes
 - Propriedades
 - Instâncias
 - Restrições

Introdução

- Ontologia (Ciência da Computação)

“Especificação formal e explícita de uma conceitualização compartilhada.” [Gruber]

- Componentes

- Classes
 - Definem conceitos do domínio;
- Propriedades
 - Organizadas em uma taxonomia;
- Instâncias
 - Relacionamento “tipo-de”, ou herança.
- Restrições

Introdução

- Ontologia (Ciência da Computação)

“Especificação formal e explícita de uma conceitualização compartilhada.” [Gruber]

- Componentes

- Classes

- Propriedades

- Instâncias

- Restrições

- Descrevem características e relacionamentos entre as classes;
- Tipo “objeto” ou “primitivo”.

Introdução

- Ontologia (Ciência da Computação)

“Especificação formal e explícita de uma conceitualização compartilhada.” [Gruber]

- Componentes

- Classes

- Propriedades

- Instâncias

- Restrições

- “Materialização” das classes;

- Exemplos específicos de indivíduos;

- Caracterizam uma Base de Conhecimento.

Introdução

- Ontologia (Ciência da Computação)

“Especificação formal e explícita de uma conceitualização compartilhada.” [Gruber]

- Componentes

- Classes
 - Restringem a interpretação dos conceitos;
- Propriedades
 - Diferentes tipos: valor, cardinalidade, disjunção, etc.
- Instâncias
- Restrições

Objetivos

- Objetivo geral

Construção e utilização de uma ontologia de domínio sobre super celulares (smartphones), para que seja possível analisar o conteúdo de blogs que possam influenciar a imagem das organizações.

- Pesquisa e esforços compartilhados com o grupo de pesquisa da Comunicação Social
 - Usuário especialista
 - Não faz parte da ontologia

Metodologia do Trabalho de Conclusão

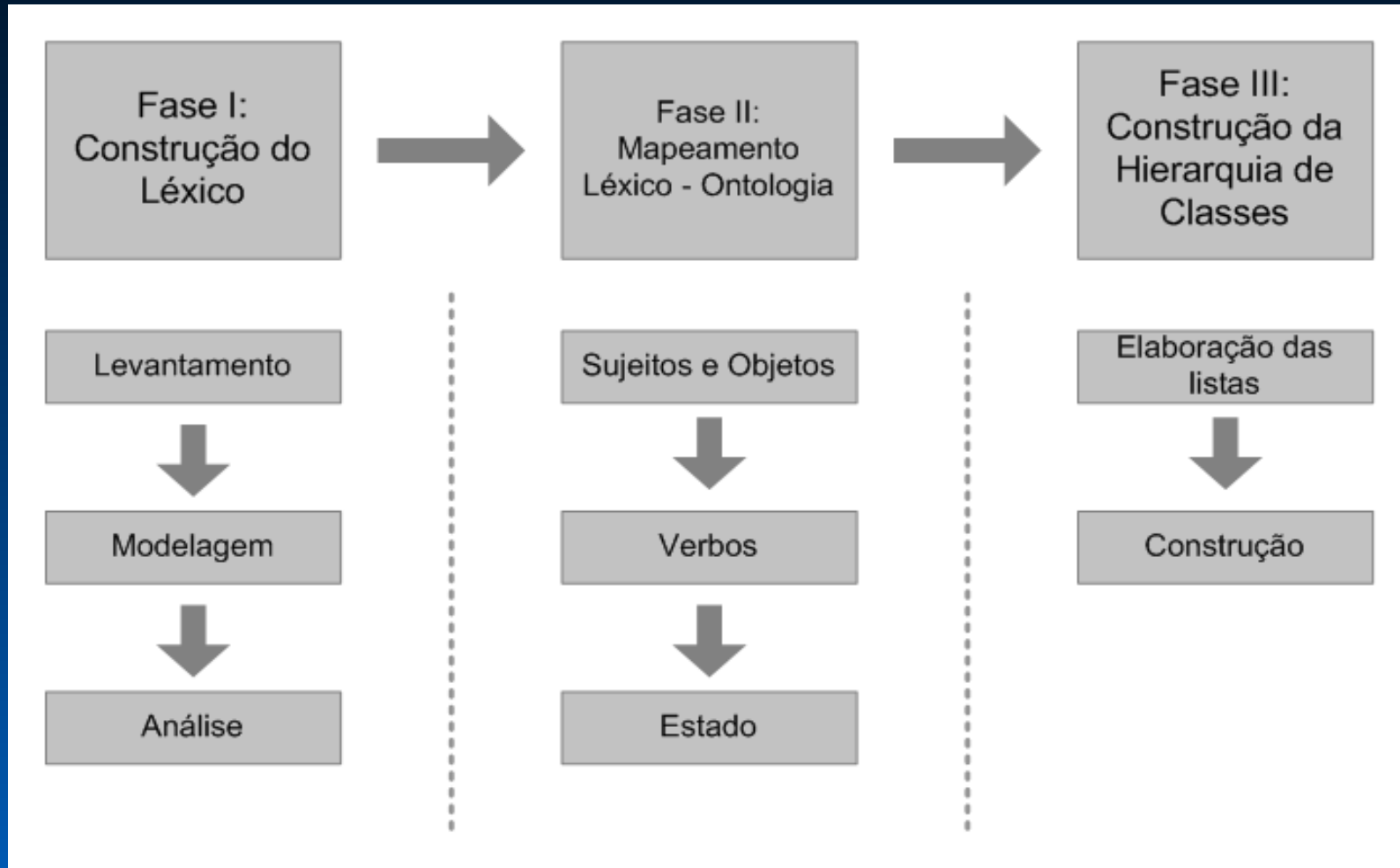
- Revisão bibliográfica inicial
 - Representação de conhecimento, ontologias, sistemas de lógica, ferramentas, linguagens, padrões e *blogs*
- Análise metodologias de construção de ontologias
 - *Ontology Development 101*
 - Projeto *TOVE*
 - **Léxico Ampliado da Linguagem – LAL**
- Modelagem \Rightarrow Léxico \Rightarrow Ontologia \Rightarrow Inferências

Construção da Ontologia

- Léxico Ampliado da Linguagem – LAL
 - Linguagem de representação simples
 - Universo de Informação (UdI)
 - Léxico: documento em hipertexto relacionando conceitos e regras
 - Entidades básicas: *termo, noção e impacto*
 - Classificação em *sujeito, objeto, verbo e estado*

Construção da Ontologia

- Visão geral da metodologia do LAL



Construção da Ontologia

- Fase I – Lista de Termos (*Brainstorming*)

Lista de Termos Candidatos (parcial)	
agenda	ouvir
antena	Nokia
Apple	número
atendida	online
bateria	Processador
bluetooth	planilha eletrônica
BlackBerry	SMS
cartão de memória	teclado QWERTY
câmera digital	tela
desligado	TIM
GSM	Treo
hardware	trocar de operadora
marca	USB
mp3	usuário
operadora	visualizar
Palm	Windows Mobile

Construção da Ontologia

- Fase I – Lista de Termos (*Brainstorming*)

Lista de Termos Candidatos (parcial)	
agenda	ouvir
antena	Nokia
Apple	número
atendida	online
bateria	Processador
bluetooth	planilha eletrônica
BlackBerry	SMS
cartão de memória	teclado QWERTY
câmera digital	tela
desligado	TIM
GSM	Treo
hardware	trocar de operadora
marca	USB
mp3	usuário
operadora	visualizar
Palm	Windows Mobile

*Lista de Termos inicial:
~320 termos*

*Lista de Termos final:
~180 termos*

Construção da Ontologia

- Fase I – Lista de Termos (coleta de documentos)

- Manuais*
- Especificações*
- Revistas*
- Textos de análise*



Fonte:
www.palm.com.br

Ontologia de Domínio

Construção da Ontologia

- Fase I – Léxico

Léxico de Smartphones (parcial)

[objeto] componente; hardware; peça

Noção: característica (funcionalidade) física de um smartphone.

Impactos: pode ser tela, teclado, tecla/botão, câmera, bateria, conexão de dados, memória, antena, processador, cartão SIM.

[sujeito] fabricante; marca

Noção: uma determinada empresa ou marca que fabrica smartphones.

Impactos: pode ser Nokia, Motorola, Apple, RIM ou Palm. Possui um ou vários modelos de smartphones. Possui concorrentes.

[verbo] falar; conversar

Noção: forma de comunicação entre dois usuários através de smartphones.

Impactos: ambos smartphones devem estar localizados na área de cobertura. Ambos smartphones devem estar ligados. Diminui a autonomia da bateria.

[estado] atendida

Noção: situação de uma ligação quando esta foi realizada com sucesso entre dois smartphones, do ponto de vista do usuário destinatário.

Impactos: contrário de não atendida.

[objeto] conexão de dados

Noção: componente que permite a troca de informações.

Impactos: pode ser fone de ouvido, USB ou Bluetooth.

[objeto] iPhone

Noção: modelo de smartphone.

Impactos: é fabricado pela Apple. Possui modelo único. É uma inovação.

Construção da Ontologia

- Fase I – Léxico

Léxico de Smartphones (parcial)
<p>[objeto] componente; hardware; peça Noção: característica (funcionalidade) física de um <u>smartphone</u>. Impactos: pode ser <u>tela</u>, <u>teclado</u>, <u>tecla/botão</u>, <u>câmera</u>, <u>bateria</u>, <u>conexão de dados</u>, <u>memória</u>, <u>antena</u>, <u>processador</u>, <u>cartão SIM</u>.</p>
<p>[sujeito] fabricante; marca Noção: uma determinada empresa ou marca que fabrica smartphones. Impactos: pode ser <u>Nokia</u>, <u>Motorola</u>, <u>Apple</u>, <u>RIM</u> ou <u>Palm</u>. Possui um ou vários <u>modelos</u> de <u>smartphones</u>. Possui <u>concorrentes</u>.</p>
<p>[verbo] falar; conversar Noção: forma de comunicação entre dois <u>usuários</u> através de <u>smartphones</u>. Impactos: ambos <u>smartphones</u> devem estar localizados na <u>área de cobertura</u>. Ambos <u>smartphones</u> devem estar <u>ligados</u>. Diminui a autonomia da <u>bateria</u>.</p>
<p>[estado] atendida Noção: situação de uma <u>ligação</u> quando esta foi realizada com sucesso entre dois <u>smartphones</u>, do ponto de vista do <u>usuário</u> destinatário. Impactos: contrário de <u>não atendida</u>.</p>
<p>[objeto] conexão de dados Noção: <u>componente</u> que permite a troca de <u>informações</u>. Impactos: pode ser <u>fone de ouvido</u>, <u>USB</u> ou <u>Bluetooth</u>.</p>
<p>[objeto] iPhone Noção: <u>modelo</u> de <u>smartphone</u>. Impactos: <u>é fabricado</u> pela <u>Apple</u>. Possui <u>modelo</u> único. É uma <u>inovação</u>.</p>

Léxico final:
~19 páginas

Construção da Ontologia

- Fase II – Mapeamento
 - Ferramenta *Protégé OWL*
 - *OWL: Ontology Web Language*
 - OWL DL (lógica de descrição)
- Tabela de mapeamento

<i>Elementos do LAL</i>	<i>Elementos da Ontologia</i>
Objeto e Sujeito	Classe
Estado	Classe ou Propriedade
Verbo	Propriedade



Construção da Ontologia

- Fase II – Mapeamento
- Exemplo, impacto:

Fabricante *fabrica* Modelo



Domínio
(classe)



Propriedade



Faixa
(classes)

Construção da Ontologia

- Ontologia modelada no *Protégé* (propriedades)

The screenshot displays the Protégé ontology editor interface. The 'PROPERTY BROWSER' on the left shows a list of object properties for the 'Smartphone' project, with 'fabrica ↔ eh_fabricado_por' selected. The 'PROPERTY EDITOR' on the right shows the configuration for the 'fabrica' property, including a table for annotations and a section for domain and range settings.

PROPERTY BROWSER
For Project: Smartphone

Object Datatype Annotation All

Object properties

- emite_fatura ↔ eh_emitida_por_operadora
- paga ↔ eh_paga_por
- armazena ↔ eh_armazenada_em
- fabrica ↔ eh_fabricado_por**
- tem_tela
- possui_smartphone ↔ pertence_a
- executa_aplicativo
- atende_codigo_de_area ↔ eh_atendido_por
- eh_atendido_por_operadora ↔ atende_codigo_de_area
- fornece ↔ eh_fornecido_por
 - fornece_sinal
 - fornece_programa_de_relacionamento
 - fornece_servico
 - fornece_plano
- eh_fornecido_por ↔ fornece
- faz_backup_de
- escuta
- visualiza
- tem_conexao_de_dados
- envia
- fala_com
- disca

PROPERTY EDITOR
For Property: fabrica (instance of owl:ObjectProperty)

Property	Value	Lang
rdfs:comment		

Domain: Fabricante

Range: Modelo

Functional
 InverseFunctional
 Symmetric
 Transitive

Inverse: eh_fabricado_por

~90 propriedades

Construção da Ontologia

- Ontologia modelada no *Protégé* (classes e axiomas)

The screenshot displays the Protégé ontology editor interface. On the left, the 'SUBCLASS EXPLORER' shows an 'Asserted Hierarchy' for the project 'Smartphone'. The hierarchy includes classes like 'Operadora' (with subclasses AT_T, Brasil_Telecom, Claro, TIM, Vivo), 'Servico' (with subclasses Auto_atendimento, Jogo, Moblog), 'Plano' (with subclasses Plano_cartao, Plano_conta, Plano_empresarial), 'Sinal', 'Smartphone', 'Software' (with subclasses Acessorio, Aplicativo, Jogo), and 'Computador'. The main 'CLASS EDITOR' is set to edit the 'Smartphone' class. It shows a table of properties and values, with 'rdfs:comment' currently selected. Below the table, the 'Asserted Conditions' section lists several axioms for the 'Smartphone' class, such as '(executa_aplicativo **only** Aplicativo) and (executa_aplicativo **min** 3)', 'modelo **only** Modelo', 'modelo **exactly** 1', 'tem_bateria **min** 1', 'tem_memoria **min** 1', 'tem_processador **exactly** 1', 'tem_sistema_operacional **only** Sistema_operacional', 'tem_sistema_operacional **exactly** 1', and 'tem_tela **min** 1'. The class is also shown to be a subclass of 'owl:Thing' and to have the property 'pertence_a **only** Usuario'. A blue oval in the bottom right corner of the interface contains the text '~100 classes'.

Construção da Ontologia

- Ontologia modelada no *Protégé* (instâncias)

The screenshot displays the Protégé ontology editor interface. The top navigation bar includes tabs for Metadata (Smartphone.owl), OWLClasses, Properties, Individuals, and Forms. The main window is divided into two panes: the Instance Browser on the left and the Individual Editor on the right.

Instance Browser: Shows the class hierarchy for 'Smartphone'. Under the 'Asserted' tab, a list of instances is visible: celular_do_fernando, iPhone_da_Maria, iPhone_da_Sandra, N95_do_Pedro, and Treo_do_Rodrigo. The instance 'celular_do_fernando' is currently selected.

Individual Editor: Shows the configuration for the instance 'celular_do_fernando' (instance of Smartphone). The editor is organized into a grid of property-value pairs:

- esta_arranhado:** false
- esta_quebrado:** true
- fora_de_area:** false
- vibra:** true
- eh_sincronizado_cor:** (empty)
- modelo:** BlackBerry_Curve_8300_1
- pertence_a:** Fernando
- possui_plano:** (empty)
- tem_cartao_SIM:** (empty)
- tem_memoria:** CENTO_E_VINTE_MB
- tem_numero:** Numero_46
- tem_processador:** Ti_OMAP_235_MHz
- tem_sistema_operac:** blackberry
- tem_tecla:** (empty)
- tem_teclado:** (empty)
- utiliza_operadora:** (empty)
- recebe_ligacao:** (empty)
- recebe_mensagem:** (empty)
- tem_conexao_de_da:** USB_57, Bluetooth_55, Wireless_58, Fone_de_ouvido_56
- tem_tela:** Tela_normal_arranhada

A blue oval at the bottom right of the interface contains the text '~76 indivíduos'.

Realização de Experimentos


- OK, temos uma base de conhecimento, e agora?

Inferir conhecimento utilizando a ontologia, de forma a auxiliar a realização de buscas mais inteligentes na blogosfera, por meio de API's de busca como a Technorati.

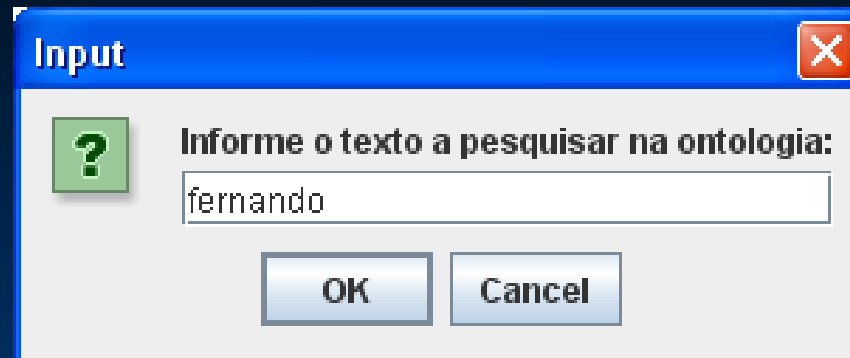
- Exemplos

- Considerar empresas concorrentes ao realizar busca na internet
- Dado um termo qualquer, encontrar na ontologia relacionamentos com fabricantes

Realização de Experimentos

- Apoio TCC anterior (KERN, 2007)
 - Programação de algoritmos em *Java*
 - Utilização do *framework* semântico *Jena*
 - Utilização do classificador (*reasoner*) *Pellet*
- Classes e métodos em O.O. para manipular a base de conhecimento
- 
- API's

Realização de Experimentos



```
*** INÍCIO DO PROCESSAMENTO ***  
  
LISTANDO INDIVÍDUOS ENCONTRADOS  
=====
```

[Parcial] Indivíduo: celular_do_fernando (Smartphone)
==> Fabricante: RIM_do_Brasil

[Exato] Indivíduo: Fernando (Usuario)
==> Fabricante: RIM_do_Brasil

```
*** PROCESSAMENTO CONCLUÍDO ***
```

Conclusões

- Estudo das tecnologias relacionadas a ontologias
- Estudo das limitações atuais de busca e funcionamento dos *blogs*
- **Criação da ontologia sobre *smartphones***
- Utilização da ontologia através de experimentos de inferência em *Java* que apóiam as buscas na internet
- Principais contribuições
 - Ontologia propriamente dita
 - Detalhamento e especificação de todas as etapas da construção

Trabalhos Futuros

- Aprimoramento constante da ontologia
- Adição de novos indivíduos
- Realização de inferências mais sofisticadas em conjunto com usuário especialista em crises de imagem

Centro Universitário Feevale

ONTOLOGIA DE DOMÍNIO PARA ANÁLISE DE *BLOGS*

Disponível em: <http://www.digitins.com.br/fernando/ontologia/Smartphone.owl>

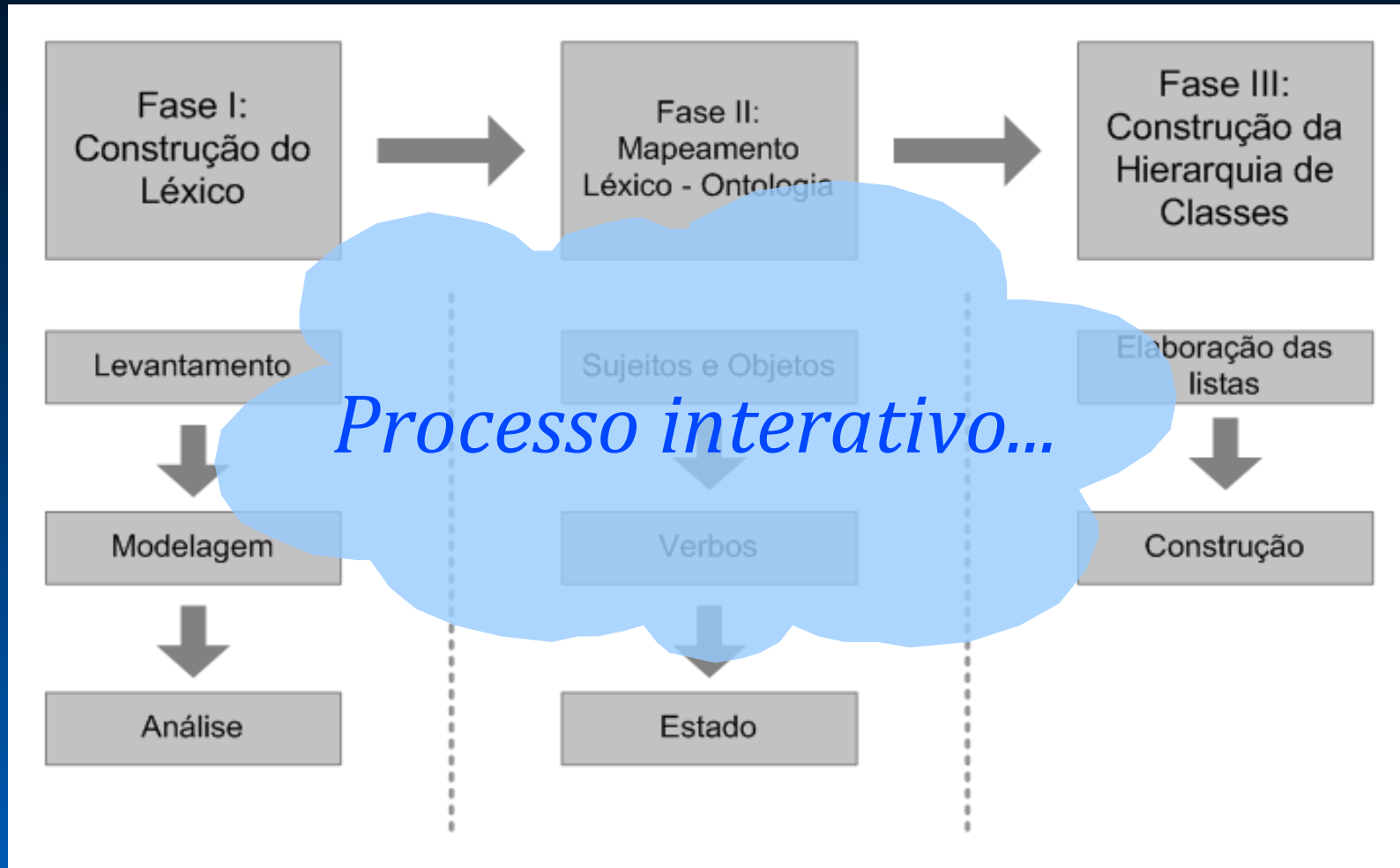
Fernando Mertins

Prof. Orientador: Rodrigo Rafael Villarreal Goulart

Dezembro de 2007

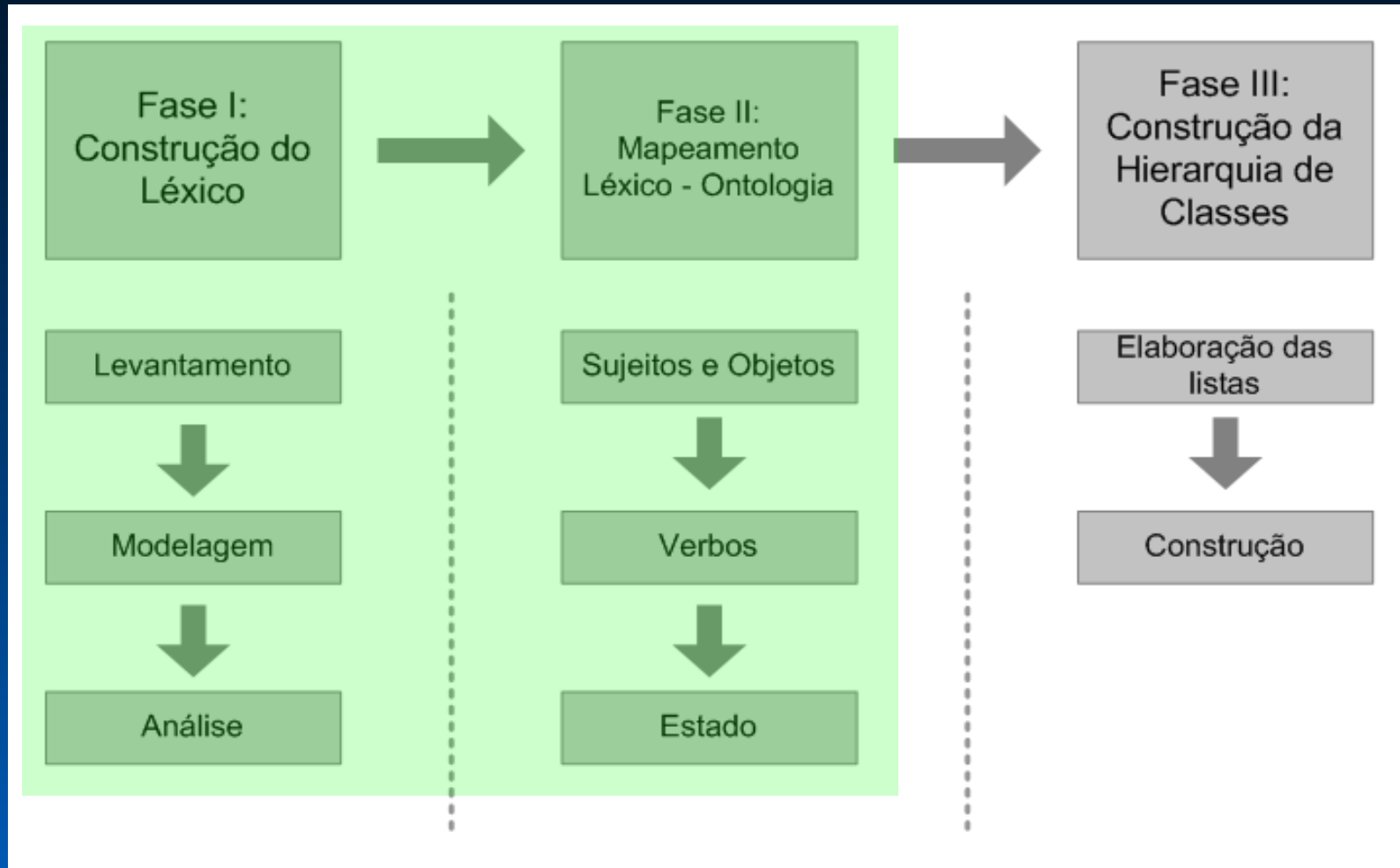
Construção da Ontologia

- Visão geral da metodologia do LAL



Construção da Ontologia

- Principal abordagem do LAL



```
/**
 * Lista as classes da ontologia, para cada classe lista suas propriedades.
 * IMPORTANTE: utilizar apenas com o modelo criado a partir do reasoner OWL
 * do próprio Jena.
 * @param model
 */
public void jenaListaClasses(OntModel modelo) {

    String textoUsuario = JOptionPane.showInputDialog("Informe o texto a pesquisar na ontologia:");

    System.out.println("LISTANDO INDIVÍDUOS ENCONTRADOS");
    System.out.println("=====");

    for (ExtendedIterator i = modelo.listNamedClasses(); i.hasNext();) {
        OntClass classe = (OntClass) i.next();

        // Para cada classe, recupera apenas as instâncias diretas (desconsidera instâncias de subclasses).
        for (ExtendedIterator j = classe.listInstances(true); j.hasNext();) {
            Individual ind = (Individual) j.next();

            if (ind.getLocalName().equalsIgnoreCase(textoUsuario)) {
                System.out.println("[Exato] Indivíduo: " + ind.getLocalName() + " (" + classe.getLocalName() +
                    analisaIndivíduo(ind, classe, modelo);
                System.out.println("");
            } else if (ind.getLocalName().toLowerCase().indexOf(textoUsuario.toLowerCase()) >= 0) {
                System.out.println("[Parcial] Indivíduo: " + ind.getLocalName() + " (" + classe.getLocalName() +
                    analisaIndivíduo(ind, classe, modelo);
                System.out.println("");
            }

            this.listaIndivíduos.clear();
        }
    }
}
```

```

public void analisaIndividuo(Individual ind, OntClass classe, OntModel modelo) {
    if (listaIndividuos.contains(ind)) {
        // Individuo já foi processado.
    } else {
        listaIndividuos.add(ind);

        // Percorre as propriedades da classe, verificando se alguma delas
        // possui como "range" a classe Fabricante.
        for (ExtendedIterator i = classe.listDeclaredProperties(); i.hasNext();) {
            OntProperty ontProp = (OntProperty) i.next();

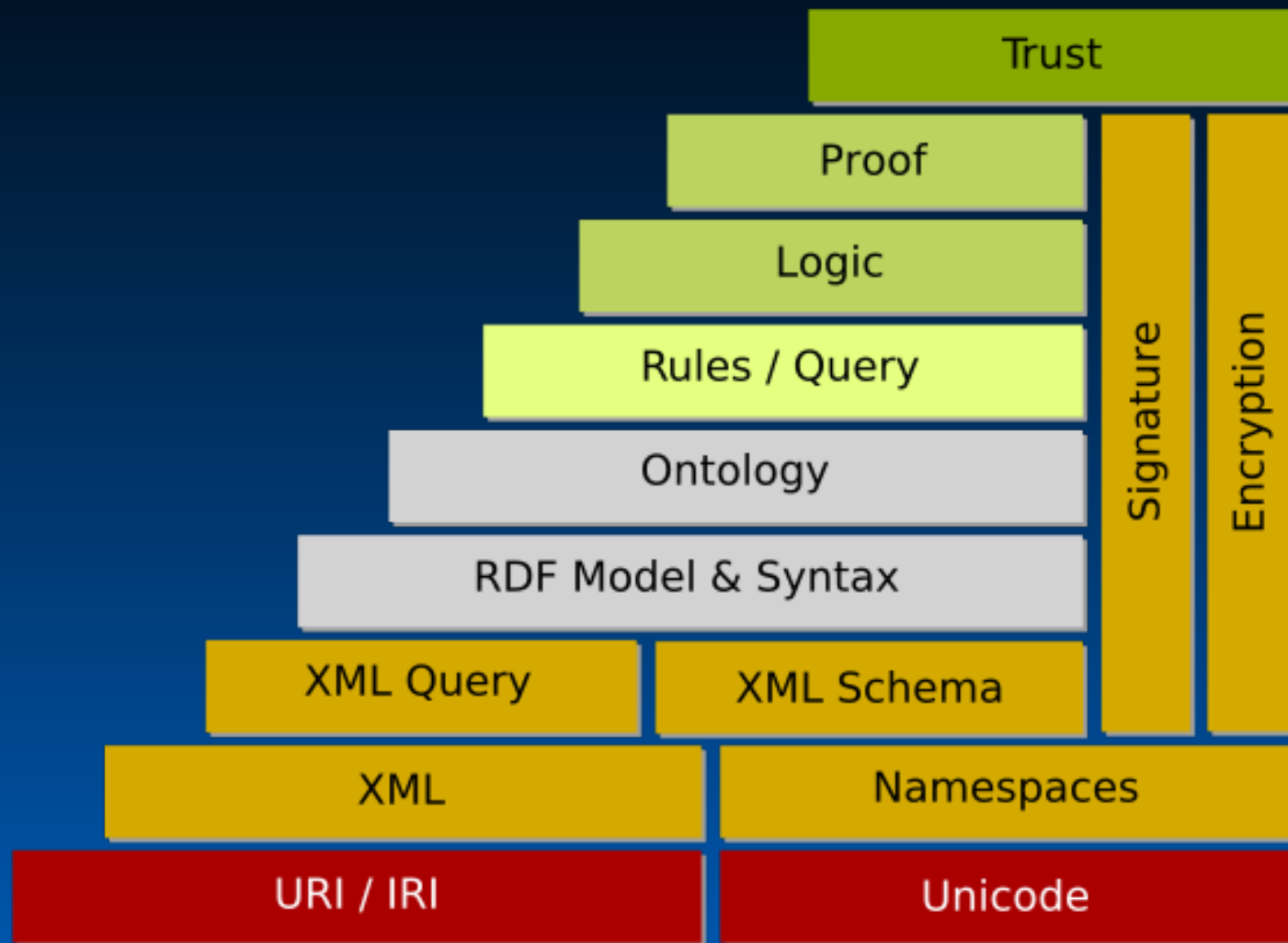
            //if (ontProp.isObjectProperty() && !ontProp.isInverseFunctionalProperty()) {
            if (ontProp.isObjectProperty()) {
                // Como é uma propriedade object-type, recupera a classe range dela (apenas uma).
                OntResource recursoAux = ontProp.getRange();
                OntClass classeRange = (OntClass) recursoAux.asClass();

                if (classeRange.getLocalName() != null) {
                    RDFNode node = ind.getPropertyValue((Property) ontProp);
                    Resource recursoNode = (Resource) node;

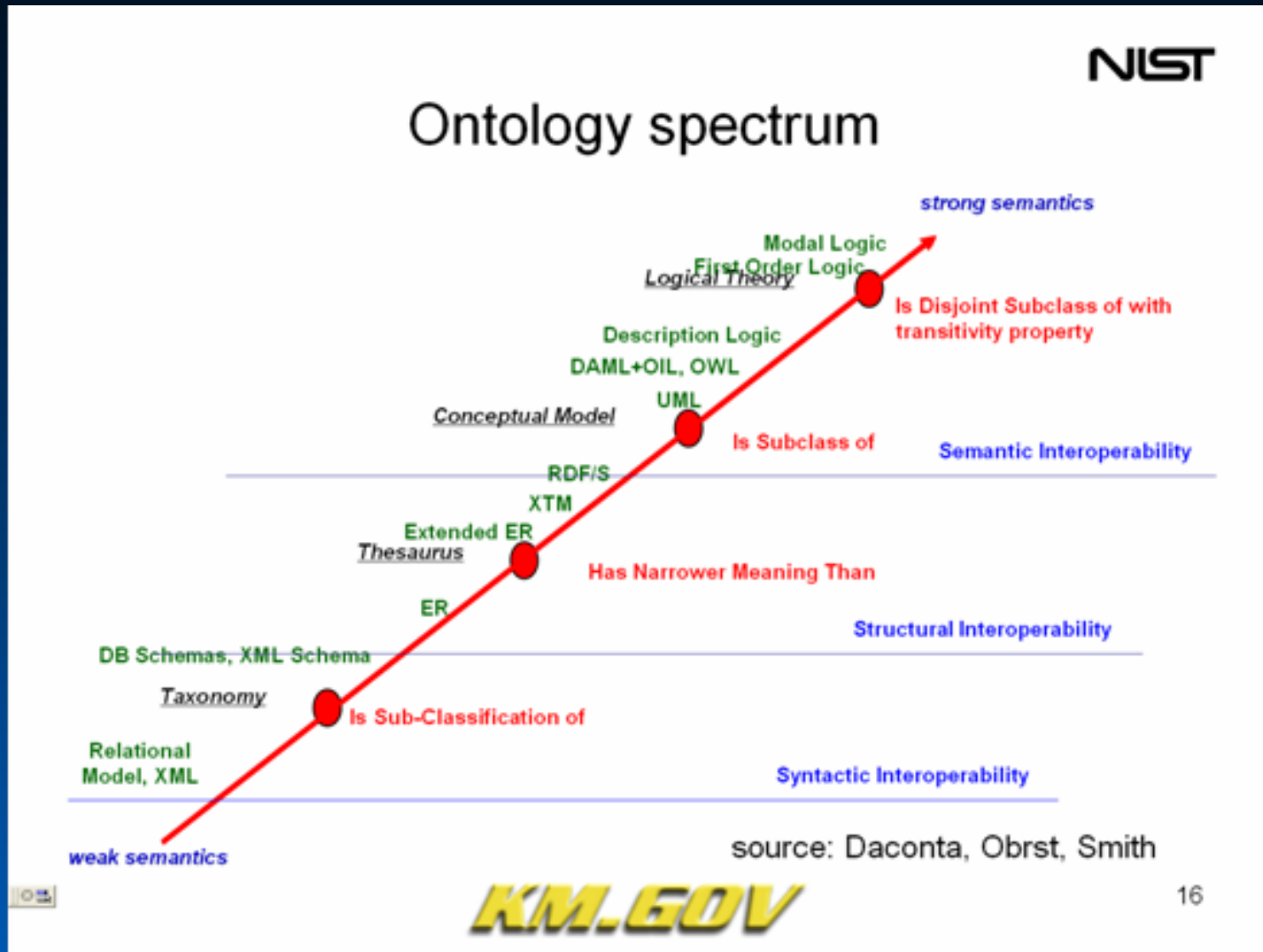
                    // Se o range for a classe Fabricante, então deve recuperar o indivíduo Fabricante relacionado
                    if (classeRange.getLocalName().equals("Fabricante")) {
                        if (recursoNode != null) {
                            System.out.println(" ==> Fabricante: " + recursoNode.getLocalName());
                        }
                    }
                    // Não é um Fabricante, então inicia recursividade.
                } else {
                    if (recursoNode != null) {
                        Individual novoInd = modelo.getIndividual(recursoNode.toString());
                        OntClass novaClasse = modelo.getOntClass("http://localhost/ontologia/Smartphone.owl#Fabricante");
                        analisaIndividuo(novoInd, novaClasse, modelo);
                    }
                }
            }
        }
    }
}

```

Web Semântica

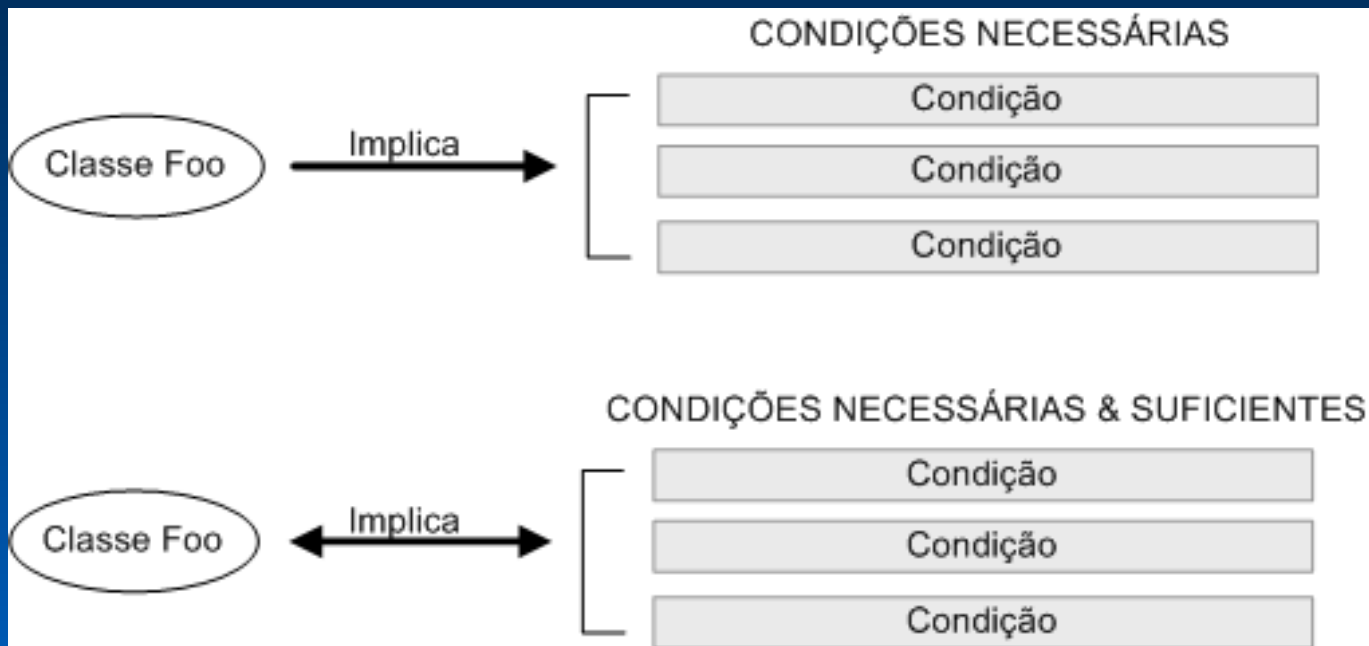


Espectro de Ontologias



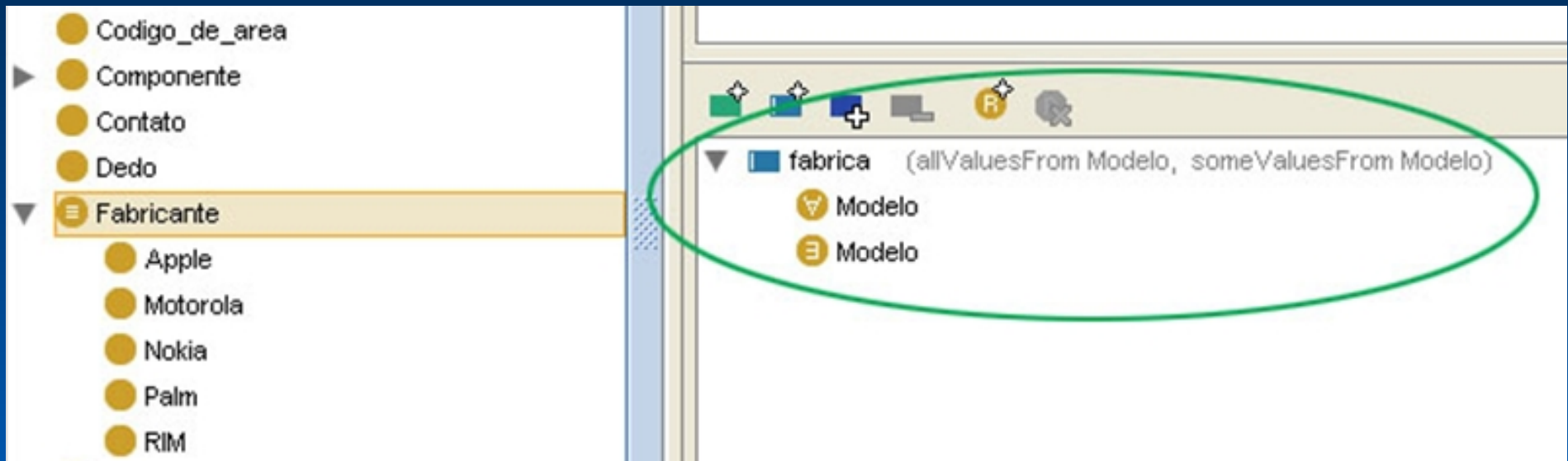
Restrições

- Se algo é instância desta classe, então é **necessário** atender a esta condição.
- Dado um indivíduo qualquer, se ele satisfizer as condições, então obrigatoriamente ele **deve** ser uma instância da classe.



Restrições

- Axioma de fechamento
- *Open World Assumption*



Fato (tripla)

- Recurso + Propriedade + Valor
- (Sujeito + Predicado + Objeto)