



Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

**Um *Framework* de Alinhamento Ontológico
entre a TI e o Negócio de uma Organização**

Rafael Gostinski Ferreira

Monografia apresentada como requisito parcial
para conclusão do Mestrado em Informática

Orientadora
Prof.^a Dr.^a Célia Ghedini Ralha

Brasília
2007

Universidade de Brasília – UnB
Instituto de Ciências Exatas
Departamento de Ciência da Computação
Mestrado em Informática

Coordenadora: Prof.^a Dr.^a Alba Cristina Magalhães de Mello

Banca examinadora composta por:

Prof.^a Dr.^a Célia Ghedini Ralha (Orientadora) – CIC/UnB
Prof.^a Dr.^a Renata Mendes de Araújo – UNIRIO
Prof.^a Dr.^a Marisa Bräscher Basílio Medeiros – CID/UnB

CIP – Catalogação Internacional na Publicação

Rafael Gostinski Ferreira.

Um *Framework* de Alinhamento Ontológico entre a TI e o Negócio de uma Organização/ Rafael Gostinski Ferreira. Brasília : UnB, 2007.
194 p. : il. ; 29,5 cm.

Tese (Mestre) – Universidade de Brasília, Brasília, 2007.

1. Ontologia, 2. Alinhamento ontológico, 3. Modelo ITIL,
4. Tecnologia da Informação, 5. Arquitetura SOA, 6. Modelagem de Processos

CDU 004

Endereço: Universidade de Brasília
Campus Universitário Darcy Ribeiro – Asa Norte
CEP 70910-900
Brasília – DF – Brasil

Dedicatória

Dedico este trabalho aos meus pais, pois sempre me incentivaram e me apoiaram nos momentos difíceis. Ambos professores, desde cedo alertaram-me sobre a importância do estudo e da busca pelo conhecimento na formação moral e cultural do cidadão.

Agradecimentos

Agradeço a todos que me auxiliaram neste trabalho, em especial, à Professora Célia Ghedini Ralha pela oportunidade deste trabalho e pela confiança dispensada, ao Professor José Carlos Loureiro Ralha pelos ensinamentos na utilização do \LaTeX , ao Professor Guilherme Albuquerque Pinto pela ajuda na análise de complexidade dos algoritmos utilizados, e aos colegas da Tecnologia do Banco do Brasil Fernando Henrique, Daniel Oliveira, Antônio Carlos Bonatto, Giovane Longoni, Jeferson Adriano Martins, José de Ribamar, Kazuo Kotani, Aivaldo Costa Gomes, entre tantos outros, que contribuíram para uma melhor compreensão do ambiente de TI desta organização.

Resumo

As disciplinas de desenvolvimento de sistemas e Tecnologia da Informação colhem benefícios quando passam a adotar as técnicas de gestão por processos. Criam visibilidade, implantam conceitos de serviço e seu gerenciamento, além de criar enorme integração entre as diversas áreas usuárias, as áreas de negócio e o próprio time de tecnologia. No entanto, diante da infinidade de conceitos envolvidos entre o demandante de um serviço, seja ele um cliente externo ou um próprio usuário da organização, e o aparato tecnológico designado para atender essa demanda, torna-se imprescindível a existência de um vocabulário compartilhado; o qual, esteja baseado nas melhores práticas de gestão de TI preconizadas pelo modelo *Information Technology Infrastructure Library* - ITIL. Nossa proposta considera que a premissa de gestão por processos, unida à maior compreensão do modelo ITIL pode, por meio da construção de um portfólio ontológico dos conceitos envolvidos em todas as disciplinas do modelo, possibilitar uma linguagem comum às camadas de negócio e tecnologia; incluindo clientes, usuários e representantes do negócio no que se refere aos serviços providos pela organização. O objetivo é desenvolver, a partir da modelagem de processos organizacionais e da utilização de algoritmos de alinhamentos de ontologias, um *framework* ontológico integrado à arquitetura orientada a serviço que permita o alinhamento dos conceitos empregados pela área comercial e tecnológica de qualquer instituição.

Palavras-chave: Ontologia, Alinhamento ontológico, Modelo ITIL, Tecnologia da Informação, Arquitetura SOA, Modelagem de Processos

Abstract

System Development and Information Technology disciplines present benefits when adopting management techniques based on processes. They allow visibility, implant concepts of service and its management and besides create enormous integration among other areas such as business and the technology team. However the great amount of concepts between the organizational service and the technological apparatus, assigned to take care of this demand, urge the existence of a shared vocabulary; which can be based on the best management practices of IT expressed at the Information Technology Infrastructure Library - ITIL. Our proposal considers as premisses that the management based on process, together with a better understanding of ITIL model can, through the construction of an ontological framework relate the concepts used in different models bring benefit to the organization when integrated to the Service Oriented Architecture - SOA. This integrated ontological framework use a common language to business and technology layers relating customers, users and representatives of business together to provide best organizational services.

Keywords: *Ontology, Ontological Alignment, ITIL Model, Information Tecnology, SOA Architecture, Process Modeling*

Sumário

Lista de Figuras	10
Lista de Tabelas	12
Lista de Abreviaturas	13
Capítulo 1 Introdução	15
Capítulo 2 Fundamentação Teórica	17
2.1 Arquitetura SOA	17
2.2 Modelagem de Processos	21
2.3 Modelos de Processos de TI	26
2.3.1 ITIL	26
2.3.2 COBIT	35
2.3.3 CMMI	40
2.3.4 PMBOK	45
2.4 Ontologia	47
2.4.1 Critérios para a construção de ontologias	48
2.4.2 Tipos de Ontologias	49
2.4.3 Metodologias utilizadas para a construção de ontologias	52
2.4.4 Ferramentas para construção de ontologias	58
2.4.5 Linguagem OWL	60
2.4.6 Ferramentas para <i>Merging</i> e <i>Mapping</i> de ontologias	60
2.5 Sistemas de Recuperação de Informação	63
2.6 Algoritmos de Alinhamento	65
Capítulo 3 Trabalhos Correlatos	73
Capítulo 4 <i>Framework</i> Proposto	79
4.1 Proposta de Trabalho	79
4.2 Metodologia Utilizada	80
4.2.1 Identificação dos Domínios Ontológicos	82
4.2.2 Compreensão dos Domínios Ontológicos	83
4.2.3 Construção das Ontologias	86
4.2.4 Validação Estrutural	92
4.2.5 Construção OAN e Adição de Semântica	93
4.2.6 Validação Semântica	100

4.2.7 Alinhamento das ontologias de TI e de Negócio	104
Capítulo 5 Estudo de Caso e Análise dos Resultados	105
Capítulo 6 Conclusão e Trabalhos Futuros	115
Apêndice A Ontologia de TI	118
Apêndice B Ontologia de Negócio	170
Referências	188

Lista de Figuras

2.1	Esquema básico da Arquitetura SOA adaptado de (Corporation, 2006).	18
2.2	Estrutura do repositório de dados das ferramentas para modelagem de processos (Sordi, 2003).	24
2.3	Elementos de TI no modelo ITIL (itSMF, 2006).	33
2.4	Estrutura do COBIT (Institute, 2007).	37
2.5	Visão Geral do RUP (Jacobson et al., 1999).	43
2.6	Processos que compõem o MR-MPS (Softex, 2006).	46
2.7	Tipos de ontologias (Guarino, 1998).	50
2.8	Diferentes tipos de operações entre ontologias (Euzenat, 2004).	61
2.9	Medidas <i>Precision</i> e <i>Recall</i> (van Rijsbergen, 1975).	64
2.10	Como calcular <i>Precision</i> e <i>Recall</i> adaptado de (van Rijsbergen, 1975).	64
2.11	Curva de <i>Precision X Recall</i> para duas <i>Queries</i> Q1 e Q2 (van Rijsbergen, 1975).	65
2.12	Visão geral do processo de alinhamento (Ashpole et al., 2005).	66
2.13	Características e medidas de similaridade para diferentes tipos de entidades em NOM (Ashpole et al., 2005).	68
2.14	Características e medidas de similaridade para diferentes tipos de entidades em QOM (Ashpole et al., 2005).	69
2.15	Processo detalhado do algoritmo APFEL (Ashpole et al., 2005).	70
2.16	Métricas utilizadas para verificação da qualidade do alinhamento (Ashpole et al., 2005).	71
2.17	Análise de Complexidade do Algoritmo <i>Active</i> APFEL.	72
3.1	Ontologias dos países da Comunidade Européia (Herborn and Wimmer, 2006).	76
3.2	Passos para a construção do HLDO (Herborn and Wimmer, 2006).	77
4.1	<i>Framework</i> ontológico proposto.	81
4.2	Metodologia do Trabalho.	82
4.3	Principais objetos utilizados na modelagem de processos no <i>WBI Workbench</i>	84
4.4	Modelagem dos processos de planejamento e registro de incidentes.	87
4.5	Representação da metodologia utilizada (Noy and McGuinness, 2001).	88
4.6	Representação de hierarquia de classes da ontologia de TI definida.	89

4.7	Ilustração de <i>slots</i> da ontologia de TI definida.	90
4.8	Exemplo de instância da ontologia de TI definida.	91
4.9	Exemplo de <i>slot</i> da ontologia de TI definida.	92
4.10	Exemplos de consultas realizadas na ontologia de TI.	94
4.11	Validação da ontologia OAN dos domínios de Negócio e TI.	95
4.12	<i>Merging</i> de subconjuntos das ontologias A e B.	96
4.13	<i>Merging</i> de subconjuntos das ontologias de domínio.	97
4.14	Mapeamento dos subconjuntos fundidos na OAN.	98
4.15	Sintaxe no Protégé-OWL.	99
4.16	Adicionando semântica para utilização de mecanismos de inferências automatizados.	100
4.17	Inclusão de novos termos à ontologia.	101
4.18	Testes realizados pelo raciocinador <i>Racer Pro</i>	102
4.19	Menu OWL do Protégé.	103
5.1	Organograma da diretoria da instituição bancária.	106
5.2	Objeto de integração do trabalho.	106
5.3	Estabelecimento do ANS entre DICOM e DITEC.	107
5.4	Transações disponibilizadas pelo GEFIN.	108
5.5	Resultado da <i>f-measure</i> para os algoritmos utilizados.	109
5.6	Segurança (Visão de TI) x Normas Externas (Visão de Negócio).	111
5.7	Cliente Interno (Visão de TI) x Funcionário (Visão de Negócio).	112
5.8	Itens de configuração (Visão de TI) x Investimentos Equipamentos de TI (Visão de Negócio).	113
5.9	Serviços On-line (Visão de TI) x Gerenciador Financeiro (Visão de Negócio).	113

Lista de Tabelas

2.1	Iniciativas empresariais estruturadas a partir da análise de processos (Sordi, 2003).	22
2.2	Complexidades de algumas abordagens de alinhamento (Ashpole et al., 2005).	71
4.1	Codificação da modelagem dos processos de TI.	85
5.1	Resultados comparativos dos algoritmos utilizados.	109
5.2	Elementos alinhados pelo algoritmo <i>label PROMPT</i>	110
5.3	Outros alinhamentos obtidos com o Algoritmo <i>Active APFEL</i>	114

Lista de Abreviaturas

Acordo de Nível de Serviço (ANS)
Acordos de Nível Operacional (ANO)
Active Ontology Alignment (AOA)
Activity-based Costing (ABC)
Alignment Process Feature Estimation and Learning (APFEL)
American National Standards Institute (ANSI)
Atributos do Processo (AP)
Business Inteligence (BI)
Business Process Improvement (BPI)
Business Process Management (BPM)
Business Process Management System (BPMS)
Business Process Reengineering (BPR)
Business Register Interoperability Throughout Europe (BRITE)
Business Registration Process Ontologies (BRPO)
Business Service Mangement (BSM)
Capability Maturity Model Integration (CMMI)
Central Computer and Telecommunications Agency (CCTA)
Comitê de Controle de Mudança (CCM)
Configuration Managment Data Base (CMDB)
Contratos de Apoio (CA)
Control Objectives for Information and related Technology (COBIT)
Diretoria Comercial (DICOM)
Diretoria de Desenvolvimento de Aplicativos (DAA)
Diretoria de Tecnologia (DITEC)
Enterprise Service Bus (ESB)
European Examination Institute for Information Science (EXIM)
Framework for Ontology Alignment and Mapping (FOAM)
Generic Knowledge Base Editor (GKB-Editor)
Gerenciador Financeiro para clientes Pessoa Jurídica (GEFIN)
High Level Domain Ontology (HLDO)
Information Systems Audit and Control Foundation (ISACF)
Information Systems Examination Board (ISEB)
Information Technology Infrastructure Library (ITIL)
Institute of Electrical and Electronics Engineers (IEEE)
Integrated Product and Process Development (IPD)
International Electrotechnical Commission (IEC)

International Organization for Standardization (ISO)
Item de Configuração (IC)
Java Business Process Management (JbossjBPM)
Java Ontology Editor (JOE)
Microsoft Operation Framework (MOF)
Modelo de Melhoria de Processo de Software Brasileiro (MR-MPS)
Naïve Ontology Mapping (NOM)
Office for Government Commerce (OGC)
Ontologia de Alto Nível (OAN)
Ontological Constraints Manager (OCM)
Ontology Design Environment (ODE)
Project Management Body of Knowledge (PMBOK)
Project Management Institute (PMI)
Quick Ontology Mapping (QOM)
Rational Unified Process (RUP)
Recovering Architectural Assumption Method (RAAM)
Requisição de Mudança (RDM)
Resource Description Framework (RDF)
Resultados Esperados dos Atributos do Processo (RAP)
Return of Investment (ROI)
Service Oriented Architecture (SOA)
Software Engineering (SE)
Software Engineering Institute (SEI)
Supplier Sourcing (SS)
Tecnologia da Informação (TI)
Toronto Virtual Enterprise (TOVE)
Total Quality Management (TQM)
Unified Modeling Language (UML)
Web Ontology Language (OWL)
WebSphere Business Integration (WBI)

Capítulo 1

Introdução

Nas últimas duas décadas presenciamos grandes evoluções nas áreas de conhecimento relacionadas à gestão da tecnologia e da informação. Três dessas áreas são essenciais para promover mudanças na forma de planejar, usar e extrair benefícios da tecnologia de informação: a evolução das metodologias e dos padrões aceitos pelo mercado; a evolução tecnológica que permite intensa integração; e o uso de indicadores e dos conceitos de governança nas práticas de gestão.

Para uma visão prática, nos interessa entender que todo trabalho realizado nas empresas faz parte de algum processo. Não existe um produto ou um serviço oferecido por uma empresa sem um processo empresarial bem definido. Da mesma forma, não faz sentido existir um processo empresarial que não ofereça um produto ou um serviço ao cliente final.

As disciplinas de desenvolvimento de sistemas e Tecnologia da Informação - TI colhem benefícios quando passam a adotar as técnicas de gestão por processos em suas atividades. Criam visibilidade, implantam conceitos de serviços e seu gerenciamento, além de criar enorme integração entre as diversas áreas usuárias, as áreas de negócio e o próprio time de tecnologia.

Neste trabalho será dado um enfoque especial à questão da integração entre as áreas de negócio e de tecnologia. Segundo Luftman (2001) essa questão não é recente, afinal, há alguns anos, os investimentos em TI têm aumentado consideravelmente, porém, as empresas não atingem a liderança de mercado e a lucratividade esperada. Surge assim, uma situação conflitante e de desconfiança entre essas camadas da organização. A situação torna-se ainda mais grave diante da enorme dificuldade de se encontrar profissionais com conhecimento satisfatório em domínios tão distintos e complexos. O profissional de negócio precisa ter visão sistêmica de seu negócio, conhecer o contexto mercadológico da empresa, o portfólio de serviços disponíveis, atentar para novas oportunidades, riscos e principalmente às necessidades de seu cliente. Por outro lado, o profissional de TI, além de ter de dominar um conhecimento tecnológico em constante inovação atua num contexto ampliado, que pressupõe conhecimentos de Administração de Empresas, Economia, Ciências Sociais e Humanas.

Assim, existe uma infinidade de conceitos e informações envolvidas entre o demandante de um serviço, seja ele um cliente externo ou um usuário da própria organização, e o aparato tecnológico designado para atender essa demanda. Torna-se imprescindível, portanto, a existência de um vocabulário compartilhado, con-

sensual, o qual esteja baseado nas melhores práticas de gestão de TI, preconizadas por exemplo pelo modelo *Information Technology Infrastructure Library* - ITIL, e integrado à abordagem proposta pela arquitetura *Service Oriented Architecture* - SOA.

Diante da carência que as organizações tem de visualizar os seus processos internos e externos para melhor alocar os seus recursos e obter maior competitividade, o trabalho será baseado no conceito de Modelagem de Processos. O objetivo é desenvolver um *framework* ontológico que permita estabelecer uma linguagem comum às camadas de TI e negócio, incluindo clientes, usuários e representantes do negócio no que se refere aos serviços providos pela organização. Esse *framework* será composto pelas ontologias de domínio de TI e de Negócio e pela Ontologia de Alto Nível - OAN, que compreenderá termos das ontologias de domínio. O foco principal será dado à visão tecnológica através de componentes de TI representados pelas disciplinas de gestão preconizadas pelas melhores práticas do ITIL.

Este documento está estruturado da forma a permitir uma compreensão progressiva do *framework* proposto. Dessa forma, o Capítulo 2 apresenta aspectos teóricos utilizados neste trabalho, principalmente arquitetura SOA, modelagem de processos, alguns modelos que disciplinam os processos de TI, uma revisão breve dos conceitos, ferramentas, linguagens e operações ontológicas, uma revisão de Sistemas de Recuperação de Informação com foco em medidas como *precision* e *recall*. Uma revisão de trabalhos correlatos pode ser encontrada no Capítulo 3. No Capítulo 4 será apresentada a proposta do *framework* ontológico com a metodologia utilizada. A análise de resultados será apresentada no Capítulo 5, enquanto o Capítulo 6 conclui este trabalho e descreve alguns trabalhos futuros que podem dar continuidade a esta pesquisa.

Capítulo 2

Fundamentação Teórica

Neste estudo considerou-se a premissa da modelagem de processos para uma maior compreensão do modelo ITIL e da área de negócio utilizada como exemplo. Essa modelagem foi o ponto de partida para a construção de um portfólio formado por mais de uma ontologia, o qual pretende abranger conceitos envolvidos em cada uma das disciplinas dos modelos de TI e de negócio. Esse portfólio quando integrado a um *framework* orientado a serviços como pelo uso de uma arquitetura como a SOA permitirá uma linguagem comum às camadas de TI e de negócio bem como aos usuários dos serviços providos pelas organizações.

Este capítulo apresenta uma breve introdução aos conceitos relacionados a um *framework* integrador de conceitos. Outros aspectos teóricos serão apresentados como diversos *frameworks* de processos de TI como: ITIL, COBIT, CMMI, RUP, MPS-BR e PMBOK. Serão também tratados aspectos relacionados a ontologias e medidas de recuperação de informação.

2.1 Arquitetura SOA

Uma arquitetura de *software* envolve conceitos ligados a definições utilizadas pela ANSI/IEEE que afirmam que uma arquitetura trata basicamente de como os componentes fundamentais de um sistema se relacionam intrinsecamente e extrinsecamente. Uma arquitetura orientada a serviços ou SOA tem como seu componente fundamental o conceito de serviços.

Conhecida pelos especialistas como a nova tendência mundial, a arquitetura SOA desponta como o principal tópico de discussão em inúmeras organizações em torno do mundo. Essa arquitetura tem como objetivo principal garantir que qualquer aplicação seja implementada somente se estiver devidamente relacionada a um processo de negócio existente, provendo, dessa forma: agilidade para solucionar as demandas, redução de custos com desenvolvimento, simplicidade, padronização, reaproveitamento e compartilhamento de *software*.

A Figura 2.1 apresenta um esquema básico desta arquitetura segundo Corporation (2006). Nela podemos perceber o barramento *Enterprise Service Bus* - ESB responsável pela ligação dos elementos da visão tecnológica: componentes e recursos de TI; e a visão de negócio, que inclui os portfólios de negócios e serviços.

As principais características que compõem uma arquitetura SOA segundo Cou-

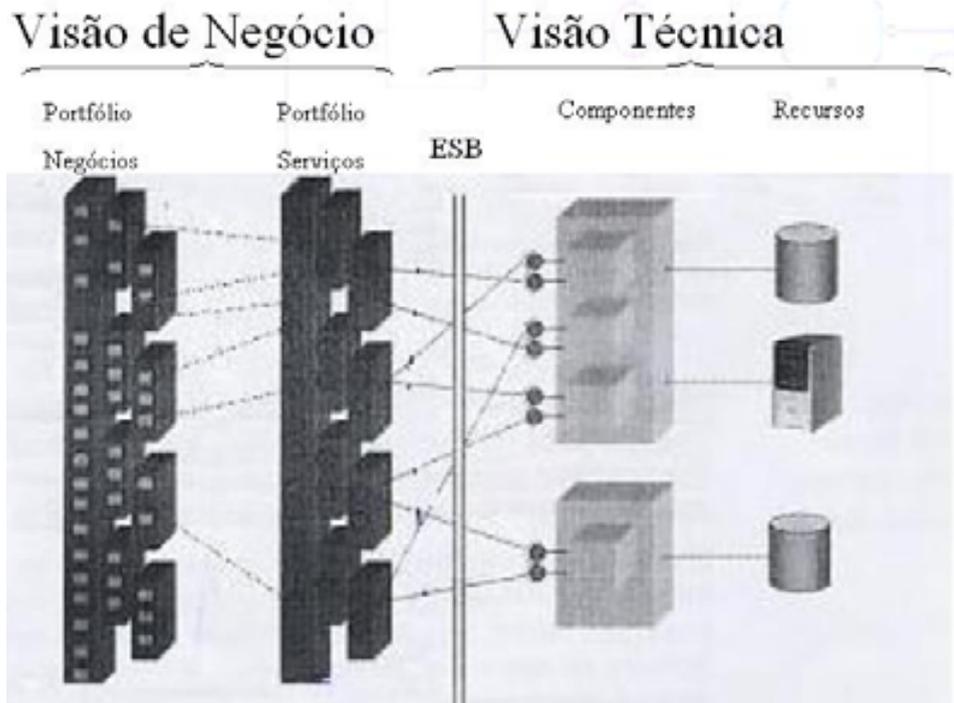


Figura 2.1: Esquema básico da Arquitetura SOA adaptado de (Corporation, 2006).

tinho (2004) são:

- Reuso de *software* - surgiu inicialmente por uma necessidade de economizar recursos de *hardware* (Clements, 1996). Há algumas décadas não havia memória suficiente nos dispositivos para armazenar muitas rotinas (segmento de código) e então, foi observado que era possível executar tarefas similares através de uma única sub-rotina parametrizada, e assim não desperdiçar o uso da escassa memória disponível. O reuso caixa-preta visa eliminar a necessidade do desenvolvedor conhecer detalhes de implementação dos componentes de *software* que farão parte do processo de reuso. Em vez disso, o reuso se dá através da descrição de interfaces ou contratos bem definidos que devem ser respeitados pela implementação a ser elaborada. O esforço tende a ser utilizado na nova implementação evitando-se a ocorrência de desperdícios e reutilizando-se a implementação de terceiros;
- Distribuição - Sistemas abertos (*open-systems*) são definidos como sistemas onde novas entidades podem dinamicamente passar a compor o sistema, deixar de existir ou ainda evoluir (Kielmann, 1996). Já um sistema aberto distribuído parte do mesmo princípio, porém leva em consideração o fato das entidades poderem estar localizadas em máquinas diferentes. As aplicações orientadas a serviços são por natureza aplicações distribuídas que fazem uso exaustivo dos conceitos de distribuição, seja por que são formadas por sistemas legados executados em diferentes plataformas ou por questões de escalabilidade, onde o desempenho do sistema é vital para a resolução do

problema proposto. Mais do que isso, geralmente aplicações orientadas a serviços são aplicações distribuídas abertas que fazem uso do registro de serviços;

- Heterogeneidade ambiental - O fenômeno da Internet apresentou uma nova realidade para os desenvolvedores de sistemas, principalmente os sistemas de informação cooperativos onde diferentes fontes de informação são agregadas com o objetivo de aumentar a quantidade de informações disponíveis para o usuário. Por ser uma rede de escala gigantesca, existe uma infinidade de recursos distintos que podem ser acessados através da rede. Tais recursos podem ser não só informações puras, mas também na forma de acesso a outros sistemas de informação (Meijler and Nierstrasz, 1997). Como atualmente até mesmo pequenas redes locais são heterogêneas, pois são compostas de plataformas de *hardware* e *software* distintas, a dificuldade da criação de sistemas de informação cooperativos aumenta consideravelmente. O cenário traçado tenta mostrar a necessidade de um sistema se comunicar mesmo em ambientes altamente heterogêneos. Além disso, os ambientes podem ser dinâmicos, onde não é possível saber durante a elaboração do sistema quais serão os componentes (em que plataformas estarão sendo executadas) que irão compor o sistema no futuro. Sistemas baseados em uma arquitetura orientada a serviços são tipicamente sistemas cooperativos abertos. Dessa natureza decorre a necessidade de que sistemas baseados em SOA ofereçam suporte a ambientes heterogêneos;
- Composição - Como atualmente os requisitos de um sistema mudam muito rapidamente, a orientação a objetos por si só não resolve o problema da composição de aplicações. Mesmo uma pequena alteração de código pode causar um grande impacto no sistema. O ideal perseguido é compor ou recompor aplicações sem tocar em código já constituído e devidamente testado. O desenvolvimento orientado a componentes disponibiliza uma maior flexibilidade, separando claramente algumas partes de uma aplicação (componentes) através de interfaces bem definidas (caixa-preta). Os primeiros casos de sucesso da composição de componentes foram os construtores de aplicação visuais como *Delphi* e *Visual Basic*, porém eles geralmente eram restritos a um certo domínio de aplicação. O *Delphi*, por exemplo, focava em aplicações de banco de dados para o sistema operacional *Windows* (Schneider et al., 1999). Como compor esses componentes de maneira simples e eficiente com a menor quantidade de código passou a ser uma área da Ciência da Computação bastante explorada. Alguns modelos formais e *frameworks* conceituais foram propostos e deram origem a uma série de linguagens específicas para composição de componentes como citato em (Sample et al., 1999);
- Coordenação - Em uma arquitetura SOA tão importante quanto a composição é a coordenação. A diferença entre composição e coordenação é que a coordenação está mais ligada com sincronização, ordenação e temporização; enquanto composição trata mais da combinação de elementos em

um todo. Porém, conforme cita Sample et al. (1999) a composição apropriada dos elementos é um pré-requisito para a coordenação. Na indústria, a coordenação está mais relacionada ao gerenciamento de processos de negócio *Business Process Management* - BPM, onde a linguagem usada tenta descrever intimamente como funciona o processo da corporação, sempre executando uma coordenação entre os agentes que executam cada passo no processo todo. Esse tipo de linguagem trata principalmente de questões como distribuição de tarefas e sincronização na execução das mesmas;

- **Dinamismo e Adaptabilidade** - Os sistemas abertos são dinâmicos por definição. Como aplicações orientadas a serviços demandam sistemas abertos, pode-se dizer que também são dinâmicas. Aplicações orientadas a serviços conseguem se adaptar às mudanças de requerimentos com facilidade. Os serviços podem entrar e sair do sistema em tempo de execução, tipicamente se comunicando com o registro de serviços através de mensagens de publicação. Além disso, geralmente os sistemas nunca fazem referência direta ao serviço e sim a uma interface para o serviço, possibilitando o dinamismo;
- **Estado** - O estado de um componente pode ser dividido em duas classes distintas: persistente ou conversacional. Estado persistente é definido quando existe a possibilidade de pelo menos uma das partes de persistir dados trocados na comunicação entre as partes. Quando dados de uma mensagem enviada a um determinado componente podem ser recuperados após a troca de mensagens em qualquer instante de tempo, pode-se dizer que esse dado foi persistido. Já o estado conversacional diz respeito ao estado dos componentes somente durante a comunicação ou conversação. Uma vez finalizada a conversação entre os componentes, o estado é desfeito e seus dados são liberados. Tipicamente, sistemas baseados em arquiteturas orientadas a serviços são construídos apenas com componentes que não possuem estado conversacional. Um dos motivos para essa escolha é que geralmente essas aplicações são distribuídas e por isso quanto menos mensagens forem trocadas entre os componentes, melhor desempenho o sistema vai obter;
- **Sincronia** - A troca de mensagens, seu modo de transmissão e a semântica são partes fundamentais em sistemas distribuídos, pois são a única maneira que os componentes possuem de se comunicar e sincronizar suas ações;
- **Robustez de Protocolos** - Um protocolo é um conjunto de regras formais que definem como os dados devem ser transmitidos entre as partes. Qualquer tipo de comunicação entre partes necessita de um protocolo, até mesmo a invocação de um método local é regida por um protocolo - neste caso trata questões como ordem dos parâmetros na pilha, localização do valor de retorno entre outros. Porém, em ambientes distribuídos, principalmente se forem heterogêneos, onde as partes estão conectadas através de uma rede de comunicação, os protocolos são mais relevantes. Questões como desempenho, robustez e segurança são muito importantes de serem consideradas nestes cenários.

A arquitetura SOA por meio de suas propriedades básicas revela-se, portanto, como uma abordagem que pretende garantir que qualquer aplicação seja implementada somente se estiver devidamente relacionada a um processo de negócio existente, ou seja, trata-se de uma ferramenta para o alinhamento estratégico entre o negócio e a TI de uma organização. Todavia, para se atingir esse alinhamento, torna-se necessário explorar o que eles têm em comum, isto é, a premissa de processos, os quais compreendem todas as atividades realizadas. Dessa forma, apresentaremos na próxima sessão uma visão geral de processos.

2.2 Modelagem de Processos

Administradores, engenheiros, analistas de sistemas, profissionais de recursos humanos, analistas de organização e métodos, entre outros profissionais, têm empregado técnicas, ferramentas, conceitos e métodos específicos para realizarem atividades correlacionadas às áreas de processos empresariais. Nesse cenário, ganha importância a necessidade de visualização e compreensão dos processos realizados. Assim, destaca-se a abordagem de modelagem de processos, afinal, segundo Abdala and Sant'Anna (2003) modelar um processo significa representá-lo através de uma coleção estruturada de elementos que descrevem suas características.

As diferentes áreas interpretam e analisam diversos aspectos do processo; num contexto mais amplo, tais aspectos devem ser totalmente coerentes e complementares, resultando numa visão corporativa e integral do processo. Essa visão ampla, integrada e coerente do processo de negócio, do ponto de vista corporativo, é prejudicada pelo distanciamento das áreas que atuam isoladamente sob alguns aspectos distintos da gestão do processo. Tal distanciamento pode ser percebido pela incompatibilidade de termos e conceitos empregados pelas diferentes áreas para identificar uma mesma característica do processo ou, ainda, pelo pouco compartilhamento e troca de dados sobre o mesmo processo.

Na Tabela 2.1 de Sordi (2003), destacamos diversas iniciativas de empresas que são estruturadas e analisadas em torno da abstração de processos de negócio. Essas iniciativas são implementadas nas empresas através de projetos, que geralmente incluem fases de trabalho abrangendo análise da situação atual do processo (*as-is*), concepção de uma nova proposta de projeto para o processo (*to-be*), e monitoramento da evolução deste na pós-implementação. Todas essas modernas iniciativas relacionadas a processos enfatizam e aprimoram em alguns aspectos a capacidade da empresa em realizar o gerenciamento de seus processos.

A idéia de processo, portanto, tem estado presente nos textos e nas discussões sobre Administração de Empresas nos últimos anos. É praticamente impossível evitar temas como Redesenho e Reengenharia de Processos, Organização por Processos e Gestão por processos. Essa idéia, no entanto, não é nova e tem raízes na tradição da engenharia industrial e no estudo dos sistemas sociotécnicos.

Entre todas as tecnologias empregadas nas empresas, a abordagem de processos tem importância especial para a TI. Além da sua utilização na automatização de tarefas e na própria execução dos processos, ela pode ser empregada em diversas atividades de apoio e gestão, tais como: visualização do processo, automatização do que é interessante automatizar, sincronização das atividades, coordenação dos

Tabela 2.1: Iniciativas empresariais estruturadas a partir da análise de processos (Sordi, 2003).

Iniciativas Relacionadas a Processos	Nome Mercadológico
Reengenharia de Processos	<i>Business Process Reengineering - BPR</i>
Melhoria de Processos	<i>Business Process Improvement - BPI</i>
Gerenciamento de Qualidade Total	<i>Total Quality Management - TQM</i>
Seis Sigma	<i>Six Sigma</i>
Análise de Cadeia de Valor	<i>Supply Chain Analysis</i>
Custo Baseado em Atividades	<i>Activity-based Costing - ABC</i>
Engenharia da Informação	<i>Information Eengineering</i>
Automação de Processos	<i>Workflow</i>
Inteligência de Negócio	<i>Business Intelligence - BI</i>
Gerenciamento Eletrônico de Documentos	<i>Document Management</i>

esforços, comunicação dos dados, entre outros. As organizações que conseguem visualizar de maneira ampla e integrada seus processos internos e externos, buscando sua contínua otimização para redução de custos, diminuição de tempo e melhoria da qualidade, estão aplicando técnicas de modelagem de processos. Segundo Sordi (2003), um dos pilares da Gestão por Processos é o *Business Process Management - BPM*, que surgiu em 2001 nos Estados Unidos como um conjunto de práticas e soluções feitas para sanar a demanda por mais controle, integração e transparência nos processos de toda a empresa.

Nesse contexto, a modelagem de processos passa a ser vista como um importante recurso, pois possibilita o alinhamento entre processos, informações e recursos. Essa abordagem será empregada no tratamento eficaz da informação, provendo tanto facilidades como respaldos formais à tomada de decisão, a qual será norteada pelos objetivos e critérios da instituição envolvida.

Tendo em vista a importância de modelagem de processos encontramos em Sawy (2002) as fases de modelagem de processos:

- criação da visão empresarial;
- identificação e compreensão dos processos existentes;
- redesenho de processos;
- implementação de processos;
- manutenção dos processos.

Nesse contexto, o uso de um *software* para modelagem de processos facilita a formalização dos processos para uma melhor análise dos mesmos. A partir da modelagem dos processos em uma ferramenta, é possível se ter uma visão sistêmica das atividades e dos recursos envolvidos. Por meio dessa visão, torna-se mais

fácil a identificação de gargalos, caminhos críticos, desperdícios de recursos por parte dos gestores, os quais estarão mais capacitados a tomar decisões e a propor, quando necessário, um redesenho de processos que melhor atenda aos objetivos da organização. Outra facilidade dos *softwares* de modelagem de processos é permitir, a um custo baixo, a identificação das consequências de implementações e alterações futuras nos processos organizacionais através do emprego de técnicas de simulação.

Existem diversos *softwares* para modelagem de processos, dentre eles podemos citar:

- *IBM WebSphere Business Integration - WBI Workbench* - ferramenta de fácil manipulação que permite analisar, simular e validar processos de negócio. Disponível em: <http://www14.software.ibm.com/webapp/>;
- *IBM WBI Advanced Modeller* - ferramenta integrada ao ambiente Eclipse de desenvolvimento que foi desenvolvida como uma atualização do *WBI Workbench*. No entanto, ela apresenta novos modos de visualização e a possibilidade de simulações de decisões com probabilidade estatística de ocorrência. Disponível em: www.ibm.com;
- *Java Business Process Management (JbossjBPM)* - ferramenta para desenvolvimento de modelagens de processos em ambiente Java. Ela não permite simulações, mas é uma ferramenta de *software* livre que permite integração com múltiplas plataformas. Disponível em: <https://jbpm.dev.java.net/>;
- *PArchitect* - ferramenta de patente brasileira para modelagem de processos. Apresenta como diferencial a simulação de processos em períodos pré-definidos de tempo (*time slice*). Devido a essa propriedade pode ser utilizada para simulação de processos de diferentes áreas como a comercial e a biológica. Disponível em: www.p3lab.com;
- *IDS-Scheer ARIS* - conjunto de ferramentas integradas de modelagem de processos. Através de filtros e métodos gráficos simples, ela possibilita ao usuário um método eficiente e rápido de documentação e estruturação do conhecimento. Disponível em: <http://www.ids-scheer.com>.

Neste trabalho, utilizamos a ferramenta proprietária *WBI Workbench* versão 4.2.3 através do convênio *Academic Initiative* da IBM.

As ferramentas de análise e projeto de processos diferem muito entre si, sobretudo pelo conjunto de objetos tratados de forma diferenciada em seus repositórios de dados. Ferramentas oriundas da área de qualidade possuem um grupo de objetos diferentes das ferramentas concebidas para o trabalho de análise e especificação de sistemas de informação, assim como aquelas voltadas aos trabalhos de reengenharia que apresentam outro enfoque. Portanto, uma atividade muito importante, segundo Sordi (2003), é definir os desafios da solução *Business Process Management System* - BPMS na organização, quais facilidades de análise e monitoramento do processo se esperam na execução diária do processo. Essa

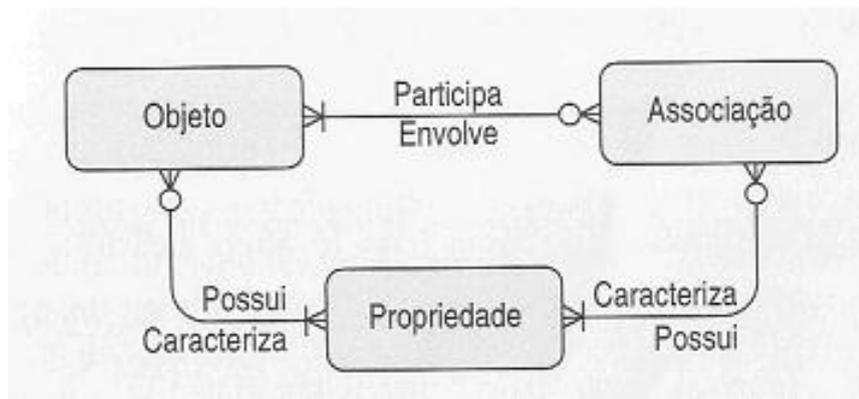


Figura 2.2: Estrutura do repositório de dados das ferramentas para modelagem de processos (Sordi, 2003).

definição auxiliará a especificar os objetos mínimos requeridos para o repositório de dados de uma ferramenta BPMS. Projetos de BPMS são de natureza corporativa; sua correta implementação elimina problemas de especificações de processos incoerentes.

A abordagem tradicional das empresas em analisar e acompanhar seus processos envolve diversas ferramentas, utilizando diferentes repositórios de dados. A área de análise de sistema, por exemplo, possui sua ferramenta específica, enquanto a de organização e métodos possui outra, assim como a área de qualidade e tantos outros grupos que estudam e analisam processos. Há muitas ferramentas e bases de dados distintas, o que não proporciona à empresa o compartilhamento de dados e a geração de uma forte base de conhecimento sobre o processo; pelo contrário, essa situação acaba por gerar muitas informações incoerentes sobre o mesmo processo, incoerências lógicas, difíceis de serem analisadas e percebidas, que se revelam apenas quando da ocorrência de inconsistências e erros durante a operação do processo.

Analisar a estrutura de dados apresentada pelo repositório das ferramentas utilizadas para modelagem do processo é um aspecto central para verificar a aderência delas às necessidades da solução BPMS pretendida pela empresa. A maioria das ferramentas apresenta uma proposta de estruturação de dados mediante três mecanismos básicos: objetos, associações e propriedades, conforme apresentado na Figura 2.2 encontrada em (Sordi, 2003).

Os objetos do repositório de dados podem ser bastante amplos e diversificados. Agrupamo-los em algumas categorizações a título de organização e melhor exposição e compreensão:

- objetos estratégicos: objetivo, fator crítico de sucesso, problema, processo, hipótese;
- objetos corporativos: unidade organizacional, *site*/localização;
- objetos humanos: competências, grupos de trabalho, profissionais;

- objetos operacionais: processo, atividade, regra, entidade de evento, entidade externa;
- objetos tecnológicos: sistema, tela, servidor, banco de dados.

Os objetos associam-se entre si; algumas dessas associações são bastante utilizadas e recebem inclusive nomes para representá-las; é o caso do fluxo de dados, controle que interliga processos de um mesmo nível.

A maioria das associações são criadas à medida que os modelos vão sendo elaborados. Seguem alguns exemplos de associações:

- Objetivo versus Unidade Organizacional;
- Processo versus Unidade Organizacional;
- Processo versus Problema;
- Objetos versus Hipótese;
- Processo versus Entidade;
- Objetivo versus Processo;
- Processo versus Processo;
- Processo versus Sistema;
- Processo versus Grupo de Trabalho.

Tanto objetos quanto associações possuem diversas propriedades que os caracterizam. Essas propriedades podem ser genéricas a todos os objetos e associações, como é o caso das propriedades relacionadas à auditoria do sistema: identificação do usuário que criou o objeto atual, data e horário da criação. Alguns exemplos de propriedades específicas de objetos:

- objeto processo tem como propriedades: frequência, tempo, custo;
- objeto objetivo tem como propriedades: importância (estratégica, tática, operacional);
- objeto unidade organizacional tem como propriedade: responsável.

Alguns exemplos de propriedades específicas de associações:

- associação entre unidade organizacional versus processo tem como propriedade: natureza do relacionamento (executa/monitora/planeja);
- associação entre processo versus entidade tem como propriedade: certeza da manipulação de dados (cria, lê, altera e exclui).

A qualidade da estrutura de dados relativa ao processo implica diretamente a capacidade das ferramentas em dispor algoritmos que gerem análises para versionamento da complementação e qualidade dos modelos criados, bem como para gestão de produtos preliminares para as etapas seguintes do ciclo de gestão do processo. Como exemplos de produtos gerados da base de dados do processo, podemos citar:

- algoritmo de simulação do processo que gera relatórios de análise com maior ou menor nível de detalhamento, dependendo das informações contidas no modelo de dados do processo. Para atividades do processo, esse modelo contempla atributos de recursos alocados, quantidade do recurso, tempo estimado, atributos de custo para cada recurso, entre outros dados que influenciam diretamente na capacidade de simulação;
- algoritmo para geração de *template* inicial para ferramenta de automação *workflow* com base na descrição lógica contida no modelo do processo. O modelo lógico do processo diferencia uma atividade manual de uma atividade implementada por um *software*, identifica pontos de interação humana com *software*, entre outras informações que podem ser úteis na montagem do *template* do *workflow*.

Os processos estão portanto muito presentes no contexto das organizações, afinal, não há nenhum serviço oferecido ao cliente que não seja ou integre um processo organizacional. Nesse ambiente, a modelagem de processos torna-se uma importante ferramenta que, além de facilitar a compreensão dos processos, permite uma melhor utilização dos recursos envolvidos. No que se refere aos processos de TI, a preocupação com a melhoria e a gestão dos mesmos, reflete-se nos diversos Modelos de Processos de TI que surgiram nos últimos anos. Passaremos a discutir na próxima seção alguns destes modelos.

2.3 Modelos de Processos de TI

Nesta seção apresentaremos os modelos de referências de processos de TI mais relacionados ao *framework* proposto. Como o objetivo do trabalho é estabelecer uma linguagem comum às camadas de TI e negócio, discutiremos modelos que envolvem áreas como: infra-estrutura tecnológica, desenvolvimento de sistemas, governança corporativa entre outras.

2.3.1 ITIL

O modelo ITIL foi concebido na década de 80 pela *Central Computer and Telecommunications Agency* - CCTA do Governo Britânico com intuito de otimizar seus processos internos. No início dos anos 90, dada a sua grande capacidade de adaptação a quaisquer estruturas, tornou-se *best practice* e foi adotado por diversas empresas européias. Em 1996, tornou-se referência mundial quando foi adotado por companhias da Austrália, Canadá, Estados Unidos e diversos países da Ásia.

No ano de 2000, foi criado o *Office for Government Commerce* - OGC com intuito de regulamentar o uso do ITIL em âmbito mundial. Nesse mesmo ano a Microsoft adotou o ITIL para desenvolver o seu modelo estendido denominado *Microsoft Operation Framework* - MOF (Corporation, 2000). Em 2001, foram criados o *European Examination Institute for Information Science* - EXIM e o *Information Systems Examination Board* - ISEB, órgãos certificadores dessa metodologia.

A biblioteca de infra-estrutura de TI denominada ITIL é composta por um conjunto consistente de melhores práticas mundiais que objetiva alinhar os serviços de TI aos requisitos de negócios através da gestão de qualidade de seus componentes e serviços. O Modelo ITIL é dividido basicamente em dois grandes grupos:

- Suporte a serviços de TI - determina os meios pelos quais os serviços serão oferecidos e gerenciados. Contém as seguintes gerências:
 - Incidentes;
 - Problemas;
 - Mudanças;
 - Configuração;
 - Implantação;
 - *Service Desk*.
- Prestação de serviços de TI - serviços que serão oferecidos aos clientes e usuários. As gerências incluídas neste grupo são:
 - Disponibilidade;
 - Nível de Serviço;
 - Continuidade de Serviços de TI;
 - Capacidade;
 - Financeira para Serviços de TI.

Faremos uma breve descrição de cada uma das disciplinas do modelo ITIL conforme (itSMF, 2006):

- Gerência de Incidentes - tem como objetivo o pronto restabelecimento e normalização da operação do serviço de TI minimizando os impactos adversos na operação do negócio e finalmente garantindo a manutenção dos mais altos níveis de qualidade de serviço.
 - Escopo: Tratamento de eventos que não fazem parte do comportamento padrão dos serviços de TI e que causem ou possam causar interrupções ou redução de qualidade do serviço prestado.
 - Atividades Operacionais do Processo:
 - * Detectar e registrar de incidentes;
 - * Abrir requisição de serviço;

- * Analisar, classificar, priorizar (impacto e urgência) e suporte inicial;
 - * Investigar, diagnosticar e resolver;
 - * Resolver e restaurar os níveis normais de operação (*recovery*);
 - * Fechar de incidentes;
 - * Fazer gestão da resolução do incidente (acompanhamento, comunicação e repasse).
- Gerência de Problemas - tem como objetivo minimizar os impactos negativos de Incidentes e Problemas no negócio, causados por erros dentro da infra-estrutura de TI, e prevenir a recorrência de incidentes relacionados a esses erros.
 - Escopo: Controle dos problemas, controle dos erros, gestão proativa.
 - Atividades Operacionais do Processo:
 - * Identificação e registro;
 - * Classificação;
 - * Identificação do Erro;
 - * Avaliação Erro;
 - * Fechamento do Erro;
 - * Acompanhamento da resolução do Erro;
 - * Análise de tendências;
 - * Direcionamento das ações preventivas;
 - * Revisão dos principais Problemas;
 - * Informações gerenciais;
 - * Métricas e Indicadores;
 - * Auditorias periódicas.
 - Gerência de Mudanças - tem como objetivo aumentar a produtividade no tratamento das Mudanças e reduzir seus riscos e impactos mediante padronização.
 - Escopo: Mudanças que afetem *hardware*, *software*, aplicações e toda a documentação e procedimentos associados com a operação, suporte e manutenção da infra-estrutura de TI.
 - Atividades Operacionais do Processo:
 - * Registrar as Requisições de Mudanças;
 - * Verificar a qualidade do registro da Requisição de Mudança - RDM;
 - * Priorizar e categorizar as Requisições de Mudanças;
 - * Avaliar impactos e recursos;
 - * Autorizar as Mudanças;
 - * Agendar as Mudanças;

- * Coordenar a construção, os testes e a implementação das Mudanças;
 - * Coordenar a construção, os testes e a implementação das Mudanças Emergenciais;
 - * Acompanhar a construção, os testes e a implementação das Mudanças;
 - * Avaliar a efetividade da RDM, após a implementação;
 - * Revisar do processo de Mudança;
 - * Fechar Mudança;
 - * Reunir periodicamente o Comitê de Controle de Mudança - CCM.
- Gerência de Liberação - tem como objetivo planejar e controlar a instalação de *hardware* e *software*, usando os serviços dos processos de Gestão de Configuração e de Gestão de Mudança, para garantir que as mudanças sejam rastreáveis, seguras e que apenas versões testadas, autorizadas e corretas sejam implantadas, além de assegurar proteção aos ambientes de processamento e seus serviços.
 - Escopo: Planejamento, desenho, construção, configuração e testes de *hardwares* e *softwares* para criar um conjunto de componentes de implantação para o ambiente operacional.
 - Atividades Operacionais do Processo:
 - * Planejar implantação definindo: conteúdo, fases, cronogramas, recursos, papéis e responsabilidades, plano de retorno, critérios de qualidade e critérios de aceitação;
 - * Desenhar, construir e configurar componentes da implantação;
 - * Fazer testes de aceitação em conjunto com equipes de negócio para avaliar funcionalidades do novo sistema, processo de instalação e plano de contingência;
 - * Detalhar a implantação definindo: cronogramas dos eventos e reuniões gerenciais, recursos papéis e responsabilidades, plano de comunicação para usuários e equipes de suporte, lista de itens de configuração para instalação e remoção, compra de equipamentos;
 - * Comunicar, preparar e treinar as equipes de atendimento ao usuário, bem como as equipes de suporte, sobre o planejado e como eles serão afetados pela nova implantação;
 - * Distribuir e instalar as versões de *software* do ambiente de desenvolvimento para o ambiente de teste e de lá para o de operação.
 - Gerência de Configuração - tem como objetivo o provimento de um modelo lógico de infra-estrutura e serviços para fornecer informação segura e atualizada sobre os itens de configuração - IC em uso, assegurando o inter-relacionamento com as demais disciplinas de gerenciamento de serviços de TI.

- Escopo: Identificação, registro e relatório de componentes de TI incluindo versão, elementos constituintes e relacionamentos. Os itens que devem estar sob controle da gerência de configuração incluem *hardware*, *software* e documentação associada (arquitetura técnica de produtos e versões de acordos).
- Atividades Operacionais do Processo:
 - * Identificar item de configuração: Seleção e identificação das estruturas de configuração para toda a infra-estrutura de ICs incluindo o proprietário, seus inter-relacionamentos e a documentação de configuração;
 - * Controlar ICs: Garantir que apenas ICs identificados e autorizados sejam registrados e manipulados;
 - * Inventariar a situação da configuração: Emissão de relatório com dados históricos e correntes durante o ciclo de vida de cada IC;
 - * Verificar e auditar configuração: Revisões e auditorias para verificar a existência física de cada IC e se estão corretamente registrados no sistema de gerenciamento de configuração;
 - * Arquivamento, limpeza (checagem de redundâncias) e *backup's*;
 - * Serviços gerais sobre configuração (relatórios, consultorias e análises).
- Função *Service Desk* - tem como objetivo proporcionar um ponto único de contato com os clientes. É o responsável por facilitar a recuperação dos níveis operacionais de serviços com o mínimo de impacto ao cliente segundo os níveis de serviços acordados e as prioridades negociais estabelecidas.
 - Escopo: Prover um ponto de contato vital e diário entre clientes, usuários, serviços de TI e terceira instância dos grupos de suporte. Desse modo, o *Service Desk* estende a escala de serviços e oferece uma abordagem global-focalizada, possibilitando que os processos de negócios sejam integrados à infra-estrutura de gerenciamento de serviços.
 - Atividades Operacionais do Processo:
 - * Receber os primeiros chamados dos clientes;
 - * Registrar e acompanhar incidentes, requisição de serviços e reclamações de clientes/usuários;
 - * Efetuar avaliação inicial dos incidentes/pedidos, tentando resolvê-los ou consultando alguém que o possa, baseado em procedimentos previamente acordados (rotinas estabelecidas);
 - * Coordenar a segunda e terceira instância dos grupos de suporte;
 - * Manter clientes informados sobre o status e progresso do seu pedido, bem como comunicar aos clientes, as mudanças planejadas de curto prazo nos níveis de serviço acordados;
 - * Controlar o ciclo de vida do pedido/incidente relatado, incluindo fechamento, verificação e confirmação com solicitante;

- * Fornecer informações gerenciais e recomendações para a melhoria dos níveis de serviço, identificando novas oportunidades de negócio e redução dos custos de infra-estrutura de TI;
 - * Identificar e comunicar necessidades de informação, treinamento e a instrução de clientes.
- Gerência de Disponibilidade - tem como objetivo otimizar a capacidade da infra-estrutura de TI, seus serviços e a estrutura organizacional da área de TI, para fornecer um nível de disponibilidade sustentável e de custo justificável, permitindo com isso que o negócio alcance seus objetivos.
 - Escopo: Tudo o que tange à desenho, implementação, mensuração e gerenciamento da disponibilidade da infra-estrutura de TI para assegurar o atingimento das metas estabelecidas de disponibilidade para os negócios da empresa. Deve ser aplicada a todos os novos Serviços de TI e aos Serviços onde os Requisitos de Nível de Serviço - RNS ou Acordo de Nível de Serviço - ANS tenham sido estabelecidos. Além disso, considera todos os aspectos que podem afetar a Disponibilidade, tais como: treinamento, capacitação, políticas, eficiência de processos, procedimentos e ferramentas.
 - Atividades Operacionais do Processo:
 - * Determinar os requisitos de disponibilidade dos serviços de TI para suportar os negócios;
 - * Determinar funções vitais de negócio e seus impactos decorrentes de falhas em componentes da Infra-estrutura de TI;
 - * Definir metas de disponibilidade, confiabilidade e manutenibilidade dos componentes da Infra-estrutura de TI estabelecidas que sustentam os serviços de TI;
 - * Identificar os custos e benefícios associados com melhorias propostas;
 - * Estabelecer métricas e relatórios de disponibilidade, Confiabilidade e Manutenibilidade na visão de negócio, usuários e organização de TI factíveis;
 - * Monitorar e analisar as tendências de disponibilidade, confiabilidade e manutenibilidade dos componentes de TI;
 - * Rever os índices de disponibilidade identificando os níveis inatingíveis, Investigar as razões pelas quais os níveis ficaram abaixo do acordado, produzir e manter planos para melhorias proativas da infra-estrutura de TI necessárias à disponibilidade.
 - Gerência de Nível de Serviço - tem como objetivo manter e melhorar a qualidade de serviços de TI, através de um ciclo contínuo.
 - Escopo: Estabelecer Acordo de Nível de Serviço - ANS para todos os serviços de TI disponíveis, estabelecer Acordos de Nível Operacional com os provedores internos e estabelecer Contratos de Apoio com os provedores externos.

- Atividades Operacionais do Processo:
 - * Monitoração e elaboração de relatórios;
 - * Reuniões de revisão de serviços;
 - * Programa de melhoria de serviços;
 - * Manutenção de Acordos de Nível de Serviço, Acordos de Nível Operacional - ANO e Contratos de Apoio - CA.
- Gerência de Continuidade - tem como objetivo suportar todos os processos de Gestão da Continuidade de Serviços assegurando que as facilidades tecnológicas e de prestação de serviços requeridos, incluindo computadores, redes, aplicações, telecomunicações, suporte técnico e atendimento, possam ser recuperadas dentro dos requisitos acordados para manutenção da capacidade operacional do negócio.
 - Escopo: Acompanhar os serviços de TI necessários para suportar os processos de negócios críticos e analisar os riscos que podem causar séria interrupção dos processos de negócio.
 - Atividades Operacionais do Processo:
 - * Realizar análise de requisitos e definição de estratégias para continuidade dos serviços de TI;
 - * Preparar os planos de continuidade e recuperação para os Serviços de TI;
 - * Fornecer treinamento para o pessoal envolvido;
 - * Testar e verificar o plano de continuidade;
 - * Implementar melhorias no planos de continuidade e recuperação;
 - * Avaliar os planos de continuidade e recuperação.
- Gerência de Capacidade - tem como objetivo conhecer os requisitos do negócio (a demanda pela entrega de serviços), o funcionamento da organização (a capacidade atual de entrega de serviços) e a infra-estrutura de TI (os meios para a entrega de serviços), para assegurar que a capacidade atual e futura atenda aos aspectos de desempenho requeridos pelo negócio com a melhor relação custo-benefício.
 - Escopo: O processo de Gestão da Capacidade deve ser o foco de todos os assuntos relativos ao desempenho e capacidade da TI. O processo deve circundar ambos ambientes operacional e de desenvolvimento.
 - Atividades Operacionais do Processo:
 - * Monitoração da capacidade;
 - * Análise;
 - * Ajuste;
 - * Implementação;
 - * Armazenamento dos dados da Gestão da Capacidade;
 - * Gestão da Demanda;

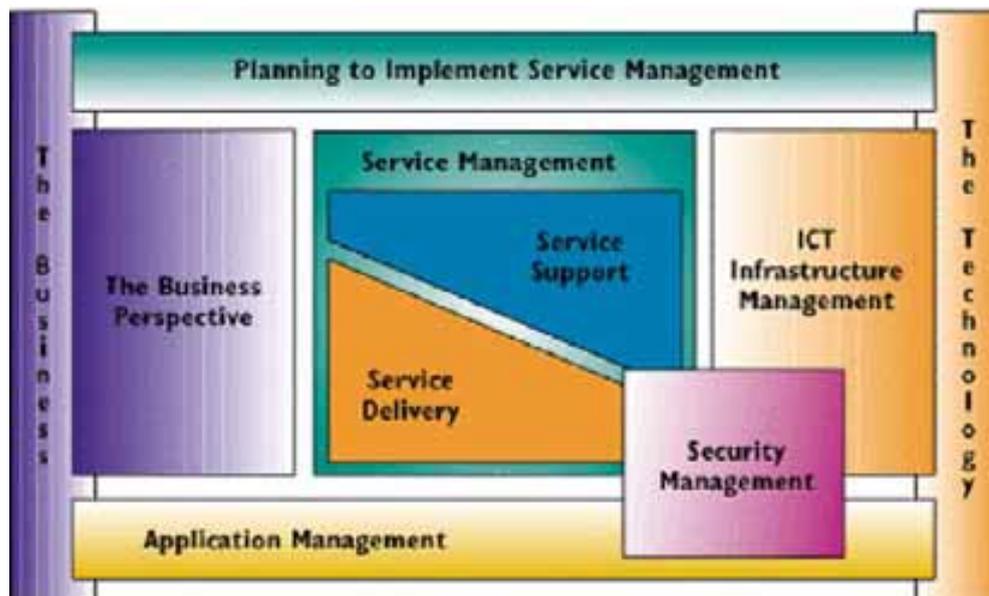


Figura 2.3: Elementos de TI no modelo ITIL (itSMF, 2006).

- * Modelagem;
 - * Dimensionamento da Aplicação;
 - * Realizar o Planejamento da Capacidade.
- Gerência Financeira - tem como objetivo contabilizar os gastos dos serviços de TI e atribuir esses custos aos demandantes dos serviços.
 - Escopo: Apoiar a criação do orçamento de TI, criar meios de contabilização financeira dos recursos de TI e apoiar cobrança financeira do uso dos recursos de TI.
 - Atividades Operacionais do Processo:
 - * Orçamentação;
 - * Contabilização Financeira;
 - * Cobrança;
 - * Reportar.

A Figura 2.3 ilustra os dois grandes grupos *Service Support* e *Service Delivery*, os quais contém as 11 gerências descritas anteriormente, sendo que a Gerência de *Service Desk* desempenha uma função operacional que integra todas as demais gerências. A gerência de segurança, da mesma forma, é responsável por garantir a continuidade e a proteção das informações do serviço e de seus usuários, minimizando os danos ao ambiente organizacional. Podemos também dizer que as gerências de suporte de serviço estão relacionadas com os processos operacionais, enquanto os de entrega de serviços com os táticos.

ITIL versão 3.0

Segundo Castro (2007) a TI amadureceu a um ritmo elevado, e com os novos modelos de negócio e o desenvolvimento das tecnologias, o que se dizia que eram as melhores práticas, provavelmente agora pode-se chamar de boas práticas. Com isso, houve a necessidade de atualizar o ITIL, de forma a assegurar que ele vá de encontro às necessidades das comunidades de hoje. Os processos com que as organizações estão trabalhando atualmente continuarão a fazer parte do novo ITIL. No entanto, os processos de entrega de serviços e suporte a serviços estão integrados num só ciclo de vida de serviços. Isto reflete melhor como a gestão de serviços é aplicada e assim a sua implementação torna-se mais fácil.

O processo de atualização para a versão 3.0, iniciado em 2004 inclui participação de consultores, fornecedores e comunidades de usuários. Os textos novos foram escritos por cinco pares de autores, maioria dos Estados Unidos e Reino Unido, e enviados para 700 revisores para edição.

De acordo com a nova versão estão incluídos os novos livros conforme seguem: Estratégia de Serviços, *Design* de Serviços, Transição de serviços, Operações de Serviços e Melhorias Contínuas. A seguir, descreveremos brevemente o conteúdo de cada um dos livros novos:

- **Estratégia de Serviços (*Service Strategy*):** esse livro aborda principalmente as estratégias, políticas e restrições sobre os serviços. Inclui também temas como reação de estratégias, implementação, redes de valor, portfólio de serviços, gerenciamento, gestão financeira e Retorno Operacional de Investimentos ou *Return of Investment - ROI*;
- **Transição de Serviços (*Service Transition*):** o volume apresenta um novo conceito sobre o sistema de gerenciamento do conhecimento dos serviços. Também inclui abordagem sobre mudanças, riscos e garantia de qualidade. Os processos endereçados são planejamento e suporte, gerenciamento de mudanças, gerenciamento de ativos e configurações, entre outros;
- **Operações de Serviços (*Service Operations*):** operações cotidianas de suporte são o mote principal desse livro. Existe foco principal em gerenciamento de *Service Desk* e requisições de serviços, separadamente de gerenciamento de Incidentes e de Problemas, que também têm espaço;
- **Melhorias Contínuas de Serviços (*Continual Service Improvement*):** a ênfase do volume está nas ações planejar, fazer, checar e agir, de forma a identificar e atuar em melhorias contínuas dos processos detalhados nos quatro livros anteriores. Melhorias nesses aspectos também levam a serviços aprimorados aos clientes e usuários.

O Modelo ITIL serviu como base para a compreensão e elaboração da ontologia de domínio de TI fornecendo elementos para a modelagem de processo. Todavia, como pretendemos alinhar esse domínio ao de negócio, apresentaremos um modelo que baseia-se justamente no alinhamento entre o negócio e a TI para capacitar às organizações a executarem seus processos em conformidade com os princípios de governança corporativa.

2.3.2 COBIT

Atualmente, é impossível imaginar uma empresa sem uma forte área de TI, para manipular os dados operacionais e prover informações gerenciais aos executivos para tomadas de decisões. A criação e manutenção de uma infra-estrutura de TI, incluindo profissionais especializados requerem altos investimentos. Algumas vezes a alta direção da empresa coloca restrições aos investimentos de TI por duvidarem dos reais benefícios da tecnologia. Entretanto, a ausência de investimentos em TI pode ser o fator chave para o fracasso de um empreendimento em mercados cada vez mais competitivos. Por outro lado, alguns gestores de TI não possuem habilidade para demonstrar os riscos associados ao negócio sem os corretos investimentos em TI. Para melhorar o processo de análise de riscos e tomada de decisão é necessário um processo estruturado para gerenciar e controlar as iniciativas de TI nas empresas, o qual possibilite o retorno de investimentos e a adição de melhorias nos processos empresariais. Esse movimento foi denominado como Governança em TI ou *IT Governance*.

O termo *IT governance* pode ser definido como uma estrutura de relações e processos que dirige e controla uma organização a fim de atingir seu objetivo de adicionar valor ao negócio através do gerenciamento balanceado do risco com o retorno do investimento em TI. No mundo globalizado, a informação e a tecnologia que suportam o negócio organizacional representa o seu mais valioso recurso. Além disso, num ambiente de negócios altamente competitivo e dinâmico é requerido uma excelente habilidade gerencial, onde TI deve suportar as tomadas de decisão de forma rápida, constante e com custos cada vez mais baixos.

Não existe dúvida sobre o benefício da tecnologia aplicada aos negócios. Entretanto, para serem bem sucedidas, as organizações devem compreender e controlar os riscos associados no uso das novas tecnologias. O modelo internacional *Control Objectives for Information and related Technology* - COBIT pode ser citado como um recurso eficiente para auxiliar o gerenciamento e controle das iniciativas de TI nas empresas.

Conceituação

O COBIT é um guia para a gestão de TI recomendado pela Fundação americana *Information Systems Audit and Control Foundation* - ISACF que inclui recursos tais como um sumário executivo, um *framework*, controle de objetivos, mapas de auditoria, um conjunto de ferramentas de implementação e um guia com técnicas de gerenciamento. As práticas de gestão do CobiT são recomendadas pelos peritos em governança de TI que ajudam a otimizar os investimentos de TI e fornecem métricas para avaliação dos resultados (Institute, 2007).

O COBIT é orientado ao negócio e independe da infra-estrutura de TI adotada na empresa. Ele fornece informações detalhadas para gerenciar processos baseados em objetivos de negócio. O COBIT é projetado para auxiliar três audiências distintas:

- Gerentes que necessitam avaliar o risco e controlar os investimentos de TI em uma organização;

- Usuários que precisam ter garantias de que os serviços de infra estrutura de TI estão sendo bem gerenciados. Para fornecer produtos e serviços aos usuários internos e externos à organização;
- Auditores que podem se apoiar nas recomendações do COBIT para avaliar o nível de qualidade de gestão de TI e aconselhar melhorias de controle na organização.

Estrutura

Conforme o Institute (2007), o COBIT está dividido em quatro domínios:

- Planejamento e organização;
- Aquisição e implementação;
- Entrega e suporte;
- Monitoração.

A Figura 2.4 ilustra a estrutura do COBIT com os quatro domínios, onde existe uma clara interligação com os processos de negócio da organização. Os mapas de controle fornecidos pelo COBIT auxiliam os auditores e gerentes a manter controles suficientes para garantir o acompanhamento das iniciativas de TI e recomendar a implementação de novas práticas, se necessário. O ponto central é o gerenciamento da informação com os recursos de TI para garantir o negócio da organização. Cada domínio cobre um conjunto de processos para garantir a completa gestão de TI, totalizando 34 processos. A seguir listaremos os 34 processos subdivididos de acordo com os quatro domínios do COBIT.

- Planejamento e Organização:
 1. Define o plano estratégico de TI;
 2. Define a arquitetura da informação;
 3. Determina a direção tecnológica;
 4. Define a organização de TI e seus relacionamentos;
 5. Gerencia os investimentos de TI;
 6. Gerencia a comunicação das direções de TI;
 7. Gerencia os recursos humanos;
 8. Assegura o alinhamento de TI com os requerimentos externos;
 9. Avalia os riscos;
 10. Gerencia os projetos;
 11. Gerencia a qualidade.
- Aquisição e implementação:

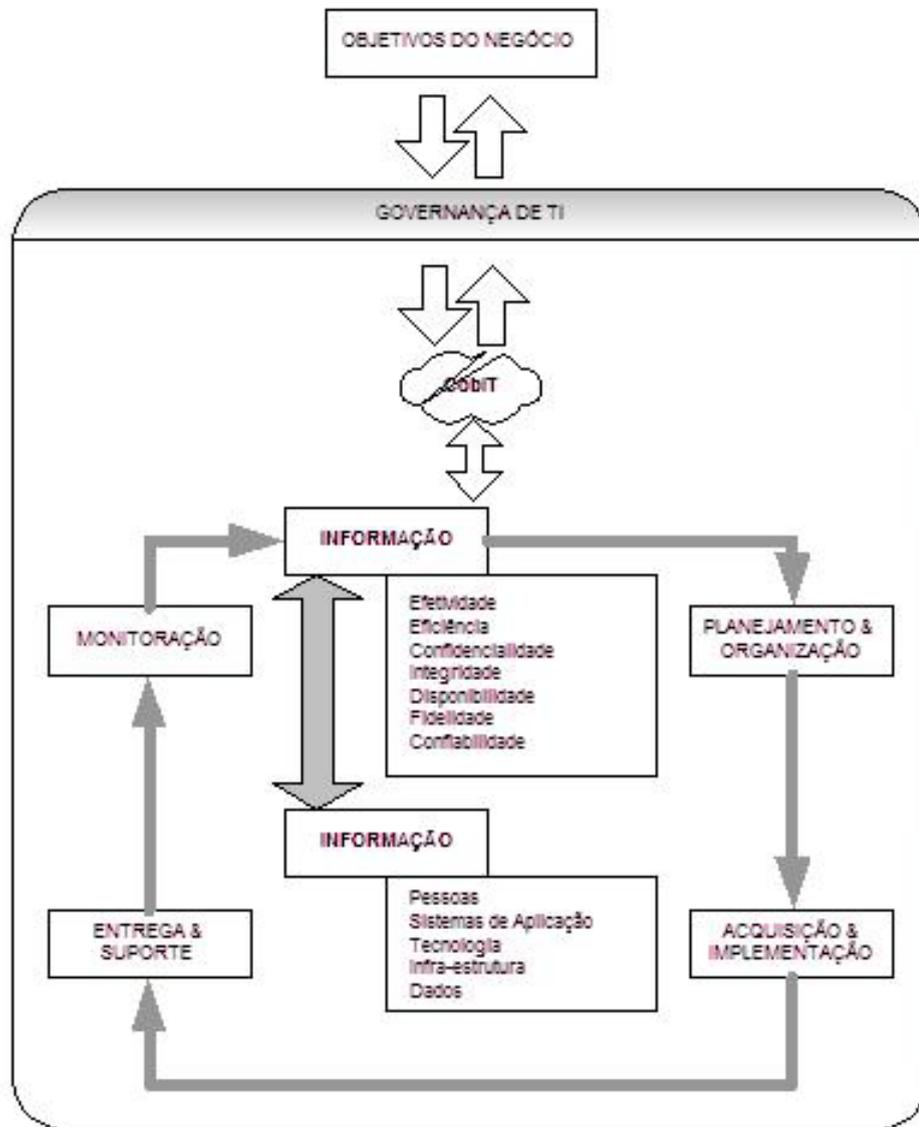


Figura 2.4: Estrutura do COBIT (Institute, 2007).

1. Identifica as soluções de automação;
 2. Adquire e mantém os softwares;
 3. Adquire e mantém a infra-estrutura tecnológica;
 4. Desenvolve e mantém os procedimentos;
 5. Instala e certifica softwares;
 6. Gerencia as mudanças.
- Entrega e suporte:
 1. Define e mantém o Acordo de Nível de Serviço;
 2. Gerencia os serviços de terceiros;
 3. Gerencia a performance, o desempenho e a capacidade do ambiente;
 4. Assegura a continuidade dos serviços;
 5. Assegura a segurança dos serviços;
 6. Identifica e aloca custos;
 7. Treina os usuários;
 8. Assiste e aconselha os usuários;
 9. Gerencia a configuração;
 10. Gerencia os problemas e incidentes;
 11. Gerencia os dados;
 12. Gerencia a infra-estrutura;
 13. Gerencia as operações.
 - Monitoração:
 1. Monitora os processos;
 2. Analisa a adequação dos controles internos;
 3. Provê auditorias independentes;
 4. Provê segurança independente.

Evolução e Benefícios

A primeira publicação do COBIT foi em 1996 enfocando o controle e análise dos sistemas de informação. Sua segunda edição em 1998 ampliou a base de recursos adicionando o guia prático de implementação e execução. A edição atual, já coordenada pelo instituto americano *IT Governance Institute*, introduz as recomendações de gerenciamento de ambientes de TI dentro do modelo de maturidade de governança.

Na era da dependência eletrônica dos negócios e da tecnologia, as organizações devem demonstrar controles crescentes em segurança. Cada organização deve compreender seu próprio desempenho e deve medir seu progresso. O *benchmarking* com outras organizações deve fazer parte da estratégia da empresa para

conseguir a melhor competitividade em TI. As recomendações de gerenciamento do COBIT com orientação no modelo de maturidade em governança auxiliam os gerentes de TI no cumprimento de seus objetivos alinhados com os objetivos da organização.

Os *guidelines* de gerenciamento do COBIT focam na gerência por desempenho. Seus indicadores-chaves identificam e medem os resultados dos processos, avaliando seu desempenho e alinhamento com os objetivos dos negócios da organização.

O grande diferencial do COBIT é sua orientação para negócios, o que vem atender as seguintes demandas:

- Da administração e gerência, visando equilibrar os riscos e os investimentos em controles no ambiente dinâmico de TI;
- Dos usuários, que dependem dos serviços de TI e seus respectivos controles e mecanismos de segurança para realizar suas atividades;
- Dos auditores, que podem utilizá-lo para validar suas opiniões ou para recomendar melhorias dos controles internos à administração.

Fatores Críticos de Sucesso

Segundo Institute (2007), os fatores críticos de sucesso definem os desafios mais importantes ou ações de gerenciamento que devem ser adotadas para colocar sobre controle a gestão de TI. São definidas as ações mais importantes do ponto de vista do que fazer a nível estratégico, técnico, organizacional e de processo.

Os indicadores de objetivos definem como serão mensurados os progressos das ações para atingir os objetivos da organização, usualmente expressos nos seguintes termos:

- Disponibilidade das informações necessárias para suportar as necessidades de negócios;
- Riscos de falta de integridade e confidencialidade das informações;
- Eficiência nos custos dos processos e operações;
- Confirmação de confiabilidade, efetividade e conformidade das informações.

Portanto, os indicadores de desempenho definem medidas para determinar como os processos de TI estão sendo executados e se eles permitem atingir os objetivos planejados; são os indicadores que definem se os objetivos serão atingidos ou não; são os indicadores que avaliam as boas práticas e habilidades de TI.

A seguir desenvolvermos uma revisão de alguns modelos relacionados ao processo de desenvolvimento de *software* e suas principais características.

2.3.3 CMMI

A qualidade de *software* é uma área de conhecimento da Engenharia de *Software* que objetiva garantir a qualidade no processo de desenvolvimento de *software* através da definição e normatização de processos de desenvolvimento. Apesar dos modelos aplicados na garantia da qualidade de *software* atuarem principalmente no processo, o principal objetivo é garantir um produto final que satisfaça às expectativas do cliente, dentro daquilo que foi definido em termos de requisitos. A seguir descreveremos alguns modelos de qualidade e maturidade para desenvolvimento de *software*.

O *Capability Maturity Model Integration* - CMMI é um modelo de referência que contém práticas (Genéricas ou Específicas) necessárias à maturidade em disciplinas específicas como: *Software Engineering* - SE, *Integrated Product and Process Development* - IPPD, *Supplier Sourcing* - SS.

Desenvolvido pelo *Software Engineering Institute* - SEI dos Estados Unidos da América, o CMMI é uma evolução do *Capability Maturity Model* - CMM e procura estabelecer um modelo único para o processo de melhoria corporativo, integrando diferentes modelos e disciplinas. Trata-se de um guia destinado a melhorar os processos organizacionais e a habilidade desses em gerenciar o desenvolvimento, a aquisição e a manutenção de produtos e serviços. O CMMI organiza as práticas, que já são consideradas efetivas, em uma estrutura que visa auxiliar a organização a estabelecer prioridades para melhoria e também fornece um guia para a implementação dessas melhorias.

Existem dois tipos de representação no CMMI: em estágios ou contínua. A representação em estágios define um conjunto de áreas de processo para definir um caminho de melhoria para a unidade organizacional, descrito em termos de níveis de maturidade. A representação contínua permite que uma organização selecione uma área de processo específica e melhore com relação a esta área. A representação contínua usa níveis de capacidade para caracterizar melhoria relacionada a uma área de processo.

O CMMI está dividido em cinco estágios:

- Realizado: Estágio inicial;
- Gerenciado: Gerenciamento de requisitos, planejamento de projeto, monitoramento e controle de projeto, gerenciamento de fornecedores, medição e análise, garantia da qualidade do processo e do produto, gerenciamento de configuração;
- Definido: Desenvolvimento de requisitos, solução técnica, integração do produto, verificação e validação, foco no processo organizacional, definição do processo organizacional, treinamento organizacional, gerenciamento de riscos, gerenciamento integrado do projeto, análise da decisão e resolução;
- Quantitativo: Gerenciamento quantitativo do projeto, performance do processo organizacional;
- Otimizado: Análise causal e resolução, inovação organizacional e implantação.

Na sequência passaremos a apresentar outro modelo relacionado ao desenvolvimento de *software*.

2.3.3.1 RUP

O *Rational Unified Process - RUP* é um processo proprietário de Engenharia de *Software* criado pela *Rational Software Corporation* que fornece técnicas a serem seguidas pelos membros da equipe de desenvolvimento de *software* com o objetivo de aumentar a sua produtividade.

O RUP usa a abordagem da orientação a objetos em sua concepção e é projetado e documentado utilizando a notação *Unified Modeling Language - UML* para ilustrar os processos em ação (Jacobson et al., 1999). É um processo considerado pesado e preferencialmente aplicável a grandes equipes de desenvolvimento e a grandes projetos, porém o fato de ser amplamente customizável torna possível que seja adaptado para projetos de qualquer escala. Para a gerência do projeto, o RUP provê uma solução disciplinada de como assinalar tarefas e responsabilidades dentro de uma organização de desenvolvimento de *software*. Ele oferece uma abordagem baseada em disciplinas para atribuir tarefas e responsabilidades dentro de uma organização de desenvolvimento. Sua meta é garantir a produção de *software* de alta qualidade que atenda às necessidades dos usuários dentro de um cronograma e de um orçamento previsíveis.

Segundo Jacobson et al. (1999), o RUP reúne alguns das melhores práticas em desenvolvimento de *software* moderno e as coloca à disposição dos projetos e organizações, a saber:

- Desenvolvimento iterativo de *software*: é a realização do *software* em várias iterações, identificando riscos para o projeto, desenvolvendo soluções para os riscos selecionados e verificando a eliminação dos riscos ao fim de cada iteração;
- Gerenciamento de requisitos: descreve como extrair, organizar e documentar funcionalidades exigidas. É utilizada nesta etapa a noção de casos de uso e cenários para capturar exigências funcionais;
- Arquitetura baseada em componentes: descreve como projetar uma arquitetura flexível, que acomode mudanças e seja intuitivamente compreensível, promovendo efetivamente a reutilização de *software*;
- Modelagem visual do *software*: é a utilização de elementos gráficos e diagramas na modelagem de *software*, isto é, a representação dos elementos estruturais e comportamentais do sistema através de modelos visuais;
- Verificação da qualidade de *software*: é a constante preocupação com a qualidade dos artefatos da aplicação e o *software* propriamente dito. Atividades de garantia da qualidade devem ser realizadas durante o processo de desenvolvimento;
- Controle de mudanças do *software*: normalmente os requisitos de uma aplicação mudam e controlar as mudanças dos requisitos e do *software* são importantes para o sucesso do processo de desenvolvimento.

O RUP também define três importantes aspectos do processo de desenvolvimento de *software*, que são: o processo deve ser Guiado pelos Casos de Uso, Centrado na Arquitetura, e Iterativo e Incremental.

Guiado por Casos de Uso - Um *software* é construído para servir aos usuários. Desta forma, para se construir um *software* de sucesso, os analistas devem descobrir e atender aos requisitos dos usuários através das funcionalidades do sistema.

Centrado na Arquitetura - A arquitetura de sistema é utilizada pelo RUP como um artefato primário para provar conceitos, construir e evoluir o sistema em desenvolvimento.

O papel da arquitetura é fornecer uma visão geral do sistema antes da construção propriamente dita. A arquitetura envolve os aspectos dinâmicos e estáticos mais importantes da aplicação e é desenvolvida principalmente a partir de requisitos definidos pelos usuários nos casos de uso, mas sofre fortes influências de vários outros fatores, como: plataformas (arquitetura dos computadores, sistemas operacionais, protocolos de comunicação), componentes reutilizáveis, sistemas legados, requisitos não funcionais, dentre outros.

A arquitetura é descrita através de diferentes visões do sistema: a visão de lógica, a visão de implementação, a visão de processo, a visão de implantação e a visão de caso de uso. Porém, estas visões apresentam somente os elementos que são importantes para a aplicação, os outros detalhes são simplesmente omitidos.

Iterativo e Incremental - Desenvolver um *software* é uma grande empreitada que pode demorar meses ou anos. É prático dividir o trabalho em partes menores ou mini-projetos. Cada mini-projeto é uma iteração que resulta em um incremento, ou seja, é a repetição de um conjunto de atividades que gera um acréscimo de funcionalidade ao sistema. Entretanto, nem todo incremento é aditivo em termos de funcionalidade, ele pode simplesmente substituir um projeto superficial por um mais detalhado.

As principais características do desenvolvimento iterativo são: resolução dos principais riscos antes da realização de grandes investimentos, permitindo o *feedback* do usuário desde cedo, e realização da integração e do teste de forma contínua, tornando possível a disponibilização de implementações parciais.

Como vantagens do desenvolvimento iterativo e incremental, têm-se que os riscos mais comprometedores são atacados primeiro e que, em casos de falhas, o sistema pode ser interrompido nas fases iniciais sem uma grande perda de tempo e investimento; e que uma porção do sistema pode ser desenvolvida e entregue enquanto uma outra parte do mesmo ainda está em desenvolvimento.

O ciclo de vida do RUP apresenta-se dividido em duas dimensões, as quais refletem as duas visões em que um sistema pode ser descrito: componentes dinâmicos, que representam o tempo e mostram os aspectos dinâmicos, como ciclos, fases, iterações e marcos, e componentes estáticos, que representam os aspectos estáticos, como: trabalhadores, atividades, artefatos e fluxos.

A Figura 2.5 mostra a arquitetura geral do RUP com suas duas dimensões:

- o eixo horizontal representa o tempo e mostra os aspectos do ciclo de vida do processo à medida que se desenvolve;

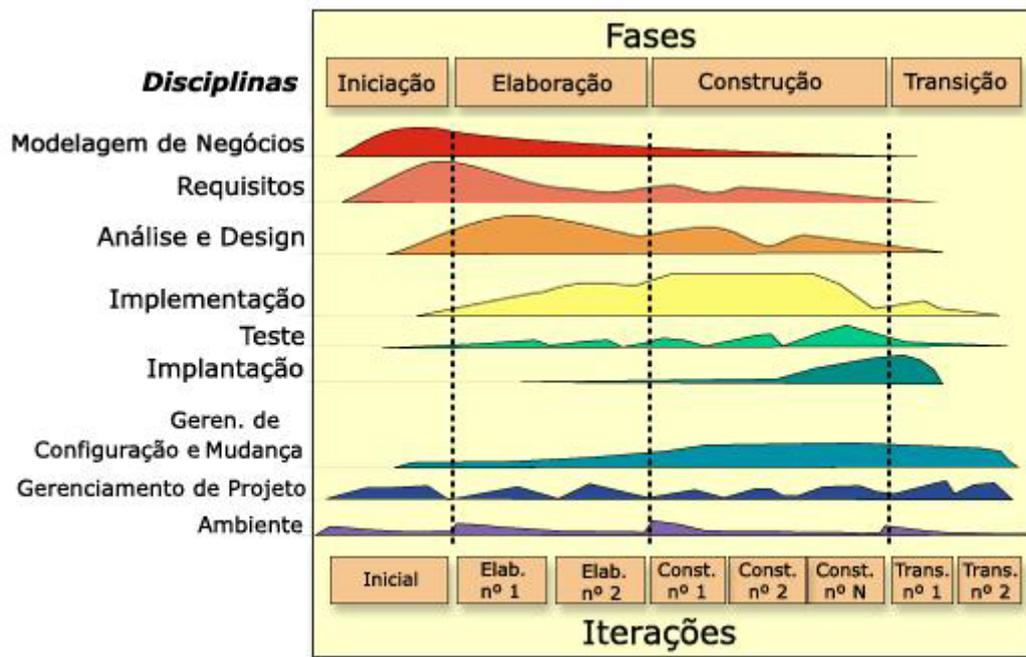


Figura 2.5: Visão Geral do RUP (Jacobson et al., 1999).

- o eixo vertical representa as disciplinas, que agrupam as atividades de maneira lógica, por natureza.

A seguir veremos um modelo de qualidade de processo voltada para a realidade do mercado de pequenas e médias empresas de desenvolvimento de *software* no Brasil. Trata-se de um modelo inspirado no CMMI e baseado na realidade do mercado brasileiro.

2.3.3.2 MPS-BR

Atualmente, o MPS.BR já é uma realidade no Brasil e, dentro de alguns anos, deve se expandir para outros países. O Modelo de Referência MR-MPS segundo Softex (2006) define níveis de maturidade que são uma combinação entre processos e sua capacidade. A definição dos processos segue a forma apresentada na Emenda 1 da ISO/IEC 12207, declarando o propósito e os resultados esperados de sua execução. Isso permite avaliar e atribuir graus de efetividade na execução dos processos. As atividades e tarefas necessárias para atender ao propósito e aos resultados esperados não são definidas neste guia, devendo ficar a cargo dos usuários do MR-MPS. A capacidade do processo é a caracterização da habilidade do processo para alcançar os objetivos de negócio, atuais e futuros; estando relacionada com o atendimento aos atributos de processo associados aos processos de cada nível de maturidade.

- Níveis de maturidade:

Os níveis de maturidade estabelecem patamares de evolução de processos, caracterizando estágios de melhoria da implementação de processos na organização. O nível de maturidade em que se encontra uma organização permite prever o seu desempenho futuro ao executar um ou mais processos.

O MR-MPS define sete níveis de maturidade:

- A (Em Otimização);
- B (Gerenciado Quantitativamente);
- C (Definido);
- D (Largamente Definido);
- E (Parcialmente Definido);
- F (Gerenciado);
- G (Parcialmente Gerenciado).

A escala de maturidade se inicia no nível G e progride até o nível A. Para cada um destes sete níveis de maturidade é atribuído um perfil de processos que indicam onde a organização deve colocar o esforço de melhoria. O progresso e o alcance de um determinado nível de maturidade MPS se obtém quando são atendidos os propósitos e todos os resultados esperados dos respectivos processos e dos atributos de processo estabelecidos para aquele nível.

A divisão em estágios, embora baseada nos níveis de maturidade do CMMI tem uma graduação diferente, com o objetivo de possibilitar uma implementação e avaliação mais adequada às micros, pequenas e médias empresas. A possibilidade de se realizar avaliações considerando mais níveis também permite uma visibilidade dos resultados de melhoria de processos em prazos mais curtos.

- Processo:

Os processos no MR-MPS são descritos em termos de propósito, resultados e informações adicionais. O propósito descreve o objetivo geral a ser atingido durante a execução do processo. Os resultados esperados do processo estabelecem os resultados a serem obtidos com a efetiva implementação do processo. Estes resultados podem ser evidenciados por um artefato produzido ou uma mudança significativa de estado ao se executar o processo. As informações adicionais são referências que podem ajudar na definição do processo pela organização. Normalmente, elas citam o processo ou sub-processo da NBR ISO/IEC 12207 e a área de processo do CMMI que estão relacionados ao processo do MR-MPS. Estas referências fornecem descrições de atividades, tarefas e melhores práticas que podem apoiar a definição e implementação do processo nas organizações.

Os processos são agrupados, por uma questão de organização, de acordo com a sua natureza, ou seja, o seu objetivo principal no ciclo de vida de *software*. Esse agrupamento resultou em três classes de processos, que são:

- Processos fundamentais - atendem o início e a execução do desenvolvimento, operação ou manutenção dos produtos de *software* e serviços correlatos durante o ciclo de vida de *software*;
- Processos de apoio - auxiliam um outro processo e contribuem para o sucesso e qualidade do projeto de *software*;
- Processos organizacionais - uma organização pode empregar estes processos em nível corporativo para estabelecer, implementar e melhorar um processo do ciclo de vida.

Os processos que compõem o MR-MPS são os descritos na Figura 2.6.

- Capacidade do Processo:

A capacidade do processo é representada por um conjunto de atributos de processo descrito em termos de resultados esperados. A capacidade do processo expressa o grau de refinamento e institucionalização com que o processo é executado na organização. No MPS, à medida que a organização evolui nos níveis de maturidade, um maior nível de capacidade para desempenhar o processo deve ser atingido pela organização.

O atendimento aos atributos do processo - AP, através do atendimento aos resultados esperados dos atributos do processo - RAP é requerido para todos os processos no nível correspondente ao nível de maturidade, embora eles não sejam detalhados dentro de cada processo.

Os níveis são acumulativos, ou seja, se a organização está no nível F, esta possui o nível de capacidade do nível F que inclui os atributos de processo dos níveis G e F para todos os processos relacionados no nível de maturidade F (que também inclui os processos de nível G). Isto significa que, ao passar do nível G para o nível F, os processos do nível de maturidade G passam a ser executados no nível de capacidade correspondente ao nível F.

A capacidade do processo no MPS possui cinco atributos de processos que são: AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP 3.2. Cada AP está detalhado em termos de resultados esperados do atributo de processo para alcance completo do atributo de processo e pode ser encontrado em (Softex, 2006).

2.3.4 PMBOK

Em um mercado cada vez mais globalizado e competitivo que tem levado as organizações a viverem em permanente estado de mudança, temos presenciado uma busca incessante das empresas no uso de melhores práticas de gerenciamento de Projetos. Essa busca tem sido incentivada e facilitada pelo *Project Management Institute - PMI* organização referência mundial em Gerenciamento de Projetos. Criado na Pensilvânia, EUA em 1969 é uma instituição sem fins lucrativos dedicada ao avanço do estado da arte em gerenciamento de projetos e seu principal compromisso é promover o profissionalismo e a ética em gestão de projetos. Atualmente o PMI está representado no Brasil por seções regionais (*Chapters*), formadas por brasileiros voluntários em vários de nossos Estados.

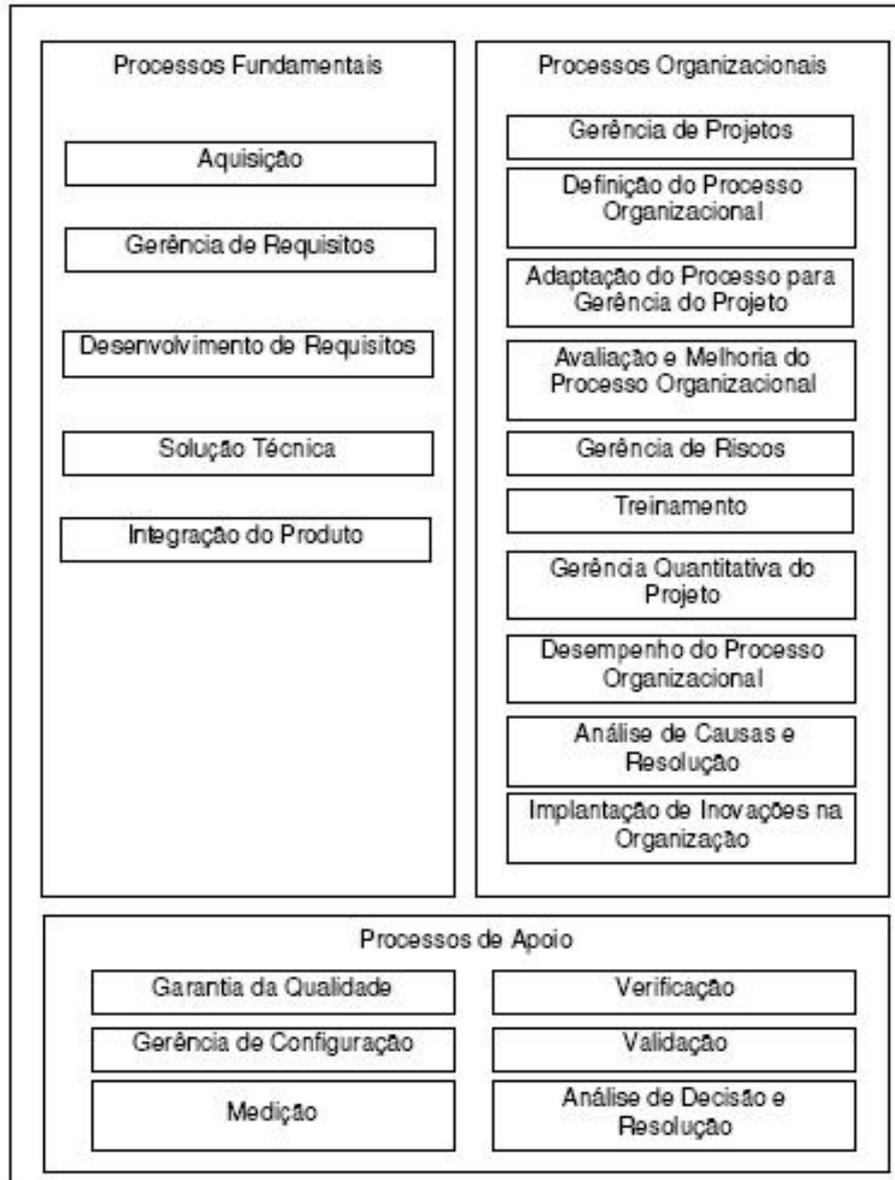


Figura 2.6: Processos que compõem o MR-MPS (Softex, 2006).

Uma das grandes contribuições do PMI, para a divulgação das boas práticas de gerenciamento de projetos, foi a publicação de um documento denominado *A guide to the Project Management Body of Knowledge - PMBOK*. Publicado pela primeira vez em 1987, se encontra atualmente na 3ª edição lançada em outubro de 2004, possuindo tradução para vários idiomas, inclusive o Português, sob o título Um Guia de Conjunto de Conhecimentos em Gerenciamento de Projetos.

O PMBOK sugere quais processos devem ser executados, durante o gerenciamento de projetos, nas áreas de Escopo, Tempo, Custo, Recursos Humanos, Comunicação, Risco, Aquisições e Qualidade, propondo também um conjunto de processos para a integração dessas áreas. Esse guia tem sido a principal fonte de informações para que as empresas melhorem os seus processos de gerenciamento.

Verificamos nas seções anteriores a existência de diversos modelos de processos de TI. Todos essas abordagens descreverem uma série de conceitos que são fundamentais para uma compreensão completa da visão da área de TI nas organizações. Todavia, esses conceitos precisam ser alinhados e estruturados não só entre as equipes de tecnologia, mas também entre elas e a equipe de negócio para que a empresa obtenha resultados satisfatórios. A situação torna-se ainda mais complexa, quando consideramos também o vocabulário específico da camada negocial das organizações. Para estruturarmos essa grande quantidade de termos envolvidos, utilizaremos a abordagem de ontologias descrita na próxima seção.

2.4 Ontologia

O termo ontologia será empregado neste trabalho como a definição de regras que regulam a combinação entre os termos e as relações. As relações entre os termos são criadas por especialistas e os usuários formulam consultas usando os conceitos especificados. Uma ontologia define assim uma linguagem (conjunto de termos) que será utilizada para formular consultas (Guarino, 1998). Desse modo, as ontologias se apresentam como um modelo de relacionamento de entidades em um domínio particular do conhecimento. O objetivo de sua construção é a necessidade de um vocabulário compartilhado onde as informações possam ser trocadas e também reutilizadas pelos usuários de uma comunidade, sejam eles humanos ou agentes inteligentes.

Várias definições surgiram na literatura que enriquecem esta definição ou sugerem outras. O termo ontologia deriva do grego *onto*, ser, e *logia*, discurso escrito ou falado. Esse termo Ontologia, com letra maiúscula, tem sido empregado ao longo da história pela filosofia e estuda as teorias sobre a natureza da existência. Para Araújo (2006) ontologia é a representação de um vocabulário, freqüentemente especializado em algum domínio ou assunto importante. Mais precisamente, não é o vocabulário que qualifica uma ontologia, mas os conceitos que os termos do vocabulário transmitem. Então, transferindo os termos de uma ontologia de uma linguagem para outra, por exemplo, do Inglês para o Francês, não muda o conceito ontológico. Almeida (2003) em seu artigo adotou a definição de Borst (1997) a qual define ontologia como uma especificação formal e explícita de uma conceitualização compartilhada. Nessa definição, formal significa legível para computadores; especificação explícita diz respeito a conceitos, propriedades,

relações, funções, restrições e axiomas explicitamente definidos e manipulados por computadores; compartilhado quer dizer conhecimento consensual e conceitualização diz respeito a um modelo abstrato de algum fenômeno do mundo real. Destas observações pode-se concluir que a importância de uma ontologia é esclarecer a estrutura de um conhecimento. Determinado o domínio, a ontologia é o sistema de representação do conhecimento deste domínio. Então, para se estruturar um sistema de representação de conhecimento eficiente e seu vocabulário basta realizar uma análise ontológica formal do domínio. Uma das definições de ontologia que é frequentemente encontrada e estudada na literatura, e que será adotada nesse trabalho, é a apresentada por Gruber (2005): Ontologia é uma especificação formal e explícita de uma conceitualização [...]

A construção de ontologias é um trabalho bastante complexo que exige critérios e metodologias para que sejam desenvolvidas, por isso se faz necessário o entendimento de alguns componentes básicos. Segundo Novello (2003) fazem-se necessários cinco elementos básicos para se formalizar o conhecimento através de ontologias:

- Conceito: É a representação de algo, ou de qualquer coisa, acerca do domínio em questão. Sendo que as propriedades de um conceito são denominadas atributos. Por exemplo, um conceito poderia ser uma pessoa, tendo como atributo a idade;
- Relacionamentos: são as integrações entre os conceitos do domínio. Nessas relações, podem-se definir as cardinalidades. Por exemplo, pode-se ter um relacionamento entre pessoas e universidade, através da relação *estuda em*;
- Funções: são relações especiais onde o enésimo elemento da relação é único para os $n - 1$ elementos precedentes. As funções são definidas formalmente como: $F : C_1 \times C_2 \times C_3 \times \dots \times C_{n-1} \times C_n$;
- Axiomas: elas modelam sentenças que são sempre verdadeiras. Por exemplo, define-se que a idade de uma pessoa corresponde ao ano atual subtraindo a data de seu nascimento;
- Instâncias: representam os elementos de uma ontologia, ou seja, são as representações dos conceitos e relações que foram estabelecidas pela ontologia.

2.4.1 Critérios para a construção de ontologias

Independente do domínio, a construção de uma ontologia requer um trabalho complexo e dispendioso, portanto qualquer apoio pode ser de grande valia. Gruber (2005) propõe um conjunto de critérios que devem ser observadas no desenvolvimento de uma ontologia visando com isso atingir maior eficiência na proposta de construir um conhecimento compartilhado e a interoperabilidade entre os programas de base nas conceitualizações compartilhadas. Os critérios são os seguintes:

- Clareza: uma ontologia deve, efetivamente, comunicar o significado pretendido na definição dos termos. Suas definições devem ser objetivas e independentes do contexto social ou computacional. Formalismo é um meio

para esse fim. Uma definição deve ser usada quando puder ser declarada em axiomas lógicos. Onde for possível uma definição completa é preferida em relação a uma definição parcial e todas as definições devem ser documentadas com linguagem natural;

- **Coerência:** Uma ontologia deve ser coerente, isto é, as inferências devem ser consistentes com as definições. Pelo menos as definições axiomáticas devem ser logicamente consistentes. Coerência também deve ser aplicada para os conceitos que são definidos informalmente, como aqueles descritos em documentos de linguagem natural e exemplos. Se a sentença que pode ser inferida a partir de axiomas contradiz uma definição ou exemplo dado informalmente, então a ontologia é incoerente;
- **Extensibilidade:** Uma ontologia deve ser projetada para antecipar os usos de um vocabulário compartilhado, ou seja, uma ontologia deve oferecer um conceito fundamentado por uma gama de tarefas antecipadas e a representação deve ser hábil para que se possa estender e especializar a ontologia. Em outras palavras, uma ontologia deve ser capaz de definir novos termos para usos especiais baseado em um vocabulário já existente, de um modo que não requeira a revisão de definições existentes;
- **Tendências de codificação mínimas:** A conceitualização deve ser especificada no nível do conhecimento sem depender de uma codificação particular. Uma tendência de codificação resulta quando escolhas de representação são feitas puramente para a conveniência de notação ou implementação. Assim, essa tendência de codificação deve ser minimizada porque os agentes que compartilham conhecimento podem ser implementados em diferentes sistemas e estilos de representações;
- **Compromisso ontológico mínimo:** Uma ontologia deve requerer o mínimo compromisso ontológico, suficiente para atender a intenção da atividade compartilhada do conhecimento. Uma ontologia deve fazer poucas afirmações possíveis sobre o mundo que está sendo modelado, permitindo que as partes comprometidas com a ontologia fiquem livres para especializar e instanciar a ontologia, sempre que necessário.

2.4.2 Tipos de Ontologias

As considerações a seguir são sugeridas por Guarino (1998) para o desenvolvimento de diferentes tipos de ontologias de acordo com o nível de generalidade, conforme demonstra a Figura 2.7.

- **Ontologia de alto nível:** Descreve conceitos de forma bem geral como espaço, tempo, material, objeto, evento, ação, entre outros, os quais são independentes de um problema ou domínio particular.
- **Ontologias de domínio e ontologias de tarefas:** Descreve, respectivamente, o vocabulário relacionado a um domínio genérico (como medicina ou automóveis) ou uma tarefa genérica (como diagnóstico ou vendas), especializando os termos na ontologia de alto nível.

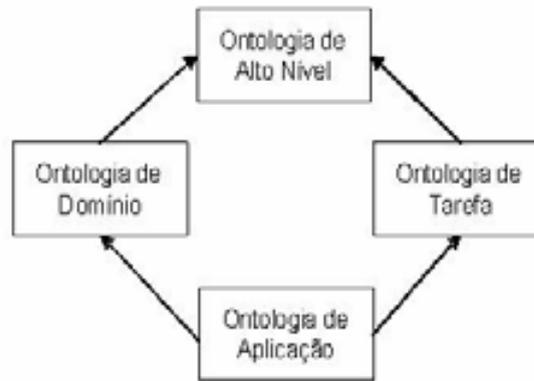


Figura 2.7: Tipos de ontologias (Guarino, 1998).

- Ontologias de aplicação: Descreve conceitos dependendo de um domínio particular e de uma tarefa, as quais são sempre especializações de ambas as ontologias relacionadas. Esses conceitos sempre correspondem aos papéis desempenhados pela entidade de domínio enquanto atuam numa certa atividade, como unidade de reposição ou componente de espera. Por exemplo, ajuda para o diagnóstico de doenças mentais.

As ontologias não apresentam sempre a mesma estrutura, dependem sempre da proposta de cada uma. Mas existem algumas características e componentes básicos comuns que são encontrados em muitas delas fazendo com que possuam semelhanças entre suas funções. Almeida (2003) diferencia alguns tipos de ontologias quanto às funções, ao grau de formalismo de seu vocabulário, à aplicação, à estrutura e ao conteúdo da conceitualização. A relação a seguir coloca resumidamente essas diferenças.

Segundo Mizoguchi et al. (1995) as ontologias podem ser classificadas quanto à função em:

- Ontologias de domínio: são reutilizáveis no domínio e fornecem vocabulário sobre conceitos, seus relacionamentos, sobre atividades e regras que os governam;
- Ontologias de tarefas: Fornecem um vocabulário sistemático de termos, especificando tarefas que podem ou não estar no mesmo domínio;
- Ontologias gerais: Incluem um vocabulário relacionado a coisas, eventos, tempo, espaço casualidade, comportamento, funções, entre outros.

Segundo Uschold and Grüninger (1996) as ontologias podem ser classificadas quanto ao formalismo em:

- Ontologias altamente informais: Expressa livremente em linguagem natural;
- Ontologias semi-informais: Expressa em linguagem natural de forma restrita e estruturada;

- Ontologias semi formais: Expressa em uma linguagem artificial definida formalmente;
- Ontologia rigorosamente formal: Os termos são definidos com semântica, teoremas e provas.

Também segundo Uschold and Grüninger (1996) as ontologias podem ser classificadas quanto à aplicação em:

- Ontologia de autoria neutra: Um aplicativo é escrito em uma única língua e depois convertido para uso em diversos sistemas, reutilizando-se as informações;
- Ontologias como especificação: Cria-se uma ontologia para um domínio, a qual é usada para documentação e manutenção no desenvolvimento de softwares;
- Ontologia de acesso comum à informação: Quando o vocabulário é inacessível, a ontologia torna a informação inteligível, proporcionando conhecimento compartilhado dos termos.

Segundo Heijst et al. (1997) as ontologias podem ser classificadas quanto ao conteúdo em:

- Ontologias terminológicas: Especificam termos que serão usados para representar o conhecimento em um domínio (por exemplo os léxicos);
- Ontologias de informação: Especificam a estrutura de registros de bancos de dados (por exemplo os esquemas de bancos de dados);
- Ontologias de modelagem do conhecimento: Especificam conceitualizações do conhecimento, têm uma estrutura interna semanticamente rica e são refinadas para o uso do domínio do conhecimento que descrevem;
- Ontologia de aplicação: Contêm as definições necessárias para modelar o conhecimento em uma aplicação;
- Ontologia de domínio: Expressam conceitualizações que são específicas para um determinado domínio do conhecimento;
- Ontologias genéricas: Similares às ontologias de domínio, mas os conceitos que as definem são considerados genéricos e comuns a vários campos;
- Ontologias de representação: Explicam as conceitualizações que estão por trás dos formalismos de representação do conhecimento.

A seguir descreveremos algumas metodologias para a construção de ontologias encontradas na literatura.

2.4.3 Metodologias utilizadas para a construção de ontologias

Ontoclean

A metodologia OntoClean é baseada em noções formais, que são gerais o bastante para serem usadas em qualquer esforço de ontologia, independentemente de um domínio particular. Ela utiliza estas noções para definir um conjunto de metapropriedades que, por sua vez, são usadas para caracterizar aspectos relevantes do significado pretendido das propriedades, classes e relações que formam uma ontologia (Guarino and Welty, 2002). Em adição, as metapropriedades impõem várias restrições na estrutura taxonômica de uma ontologia, que ajudam a avaliar as escolhas feitas.

As noções básicas do OntoClean são: essência, rigidez, identidade e unidade. Uma propriedade não é essencial se ela ocorre como verdadeira acidentalmente. Tomemos como exemplo a propriedade *ser dura*, aplicada a esponjas. Algumas esponjas podem ser duras mas isto não torna *ser dura* uma propriedade essencial de esponja. Uma forma especial de essência é rigidez. Uma propriedade é rígida se ela é essencial para todas as suas instâncias; uma instância de uma propriedade rígida não pode deixar de ser uma instância desta propriedade em um mundo diferente (Guarino and Welty, 2002).

Há propriedades que não são essenciais para suas instâncias. Propriedades que são essenciais para algumas entidades e não são essenciais para outras são chamadas semi-rígidas e as que nunca são essenciais são chamadas anti-rígidas (Guarino and Welty, 2002). *Ser dura* é semi-rígida, pois há instâncias que devem ser duras (e.g. martelo) e instâncias que devem ser duras mas podem não ser (e.g. esponjas).

As metapropriedades impõem restrições sobre a relação de subordinação. Uma destas restrições é que propriedades anti-rígidas não podem ter propriedades rígidas como suas subordinadas (e.g: a classe pessoa não pode ser subordinada à classe estudante pois esta é anti-rígida e aquela, rígida). Toda instância de estudante pode deixar de ser um estudante enquanto que uma pessoa não pode deixar de ser uma pessoa.

Dois noções filosóficas são usadas nesta metodologia: identidade e unidade (Guarino and Welty, 2002). Identidade refere-se ao problema ser capaz de reconhecer entidades individuais no mundo como sendo as mesmas ou diferentes. Unidade se refere a ser capaz de reconhecer todas as partes que formam uma entidade individual. Os critérios de identidade são condições usadas para determinar igualdade. Unidade refere-se ao problema de descrever a forma como as partes de um objeto estão ligadas, de forma a saber o que é parte do objeto e o que não é, e sob quais condições o objeto é um todo. Para algumas classes, todas as suas instâncias são todo, para outras nenhuma das suas instâncias são todo. E.g. a classe água não representa objetos todo. Já oceano pode ser uma classe que representa objetos todo, já que uma instância dessa classe é uma entidade única.

O OntoClean anexa para cada propriedade (classe) em uma ontologia, metapropriedades adequadas que descrevem seu comportamento com respeito às noções ontológicas anteriormente descritas. Seus benefícios incluem (Guarino

and Welty, 2002):

1. Identificação de um *backbone* taxonômico, que consiste de todas as propriedades rígidas na ontologia;
2. Descoberta de uso inconsistente de subordinação na taxonomia.

Uschold e King

A metodologia de Uschold e King Tetlow et al. (2005) foi desenvolvida com base na experiência adquirida na construção da *Enterprise Ontology* e propõe os seguintes passos:

1. Identificação do propósito - justificativa da construção da ontologia e seus objetivos;
2. Construção da ontologia - dividida em três passos:
 - (a) Captura da ontologia:
 - i. Identificação dos conceitos chave e os relacionamentos no domínio de interesse, ou seja, o escopo;
 - ii. Produção de definições de texto não ambíguas para os conceitos e relacionamentos;
 - iii. Identificação dos termos que se referem aos conceitos e relacionamentos.
 - (b) Codificar a ontologia - representar o conhecimento adquirido no passo anterior em uma linguagem formal;
 - (c) Integrar ontologias existentes - verificar a necessidade de utilização de ontologias existentes.
3. Avaliação - os autores propõem um julgamento técnico das ontologias, o ambiente de *software* associado, e a documentação, que pode ser especificações de requisitos e questões de competência;
4. Documentação - recomenda que diretrizes sejam estabelecidas para documentar ontologias, determinando o tipo e o propósito da ontologia.

TORonto Virtual Enterprise (TOVE)

Esta metodologia foi desenvolvida por Gruninger e Fox Gruninger and Fox (1995), e é baseada na experiência do desenvolvimento da ontologia do projeto TOVE, da Universidade de Toronto, dentro do domínio de processos de negócios e modelagem de atividades. Ela constrói um modelo lógico de conhecimento que será especificado pelo significado da ontologia (López, 1999). Esse modelo não é construído diretamente. Primeiro, é feita uma descrição informal das especificações para serem conhecidas pela ontologia e então, essa descrição é formalizada (López, 1999). São propostos os seguintes passos:

1. Captura dos cenários de motivação. De acordo com os autores da metodologia, o desenvolvimento das ontologias é motivado pelos cenários que surgem na aplicação. Os cenários são problemas armazenados ou exemplos que não são adequadamente focalizados pelas ontologias existentes. Um cenário de motivação também proporciona um conjunto de soluções intuitivamente possíveis para os problemas do cenário;
2. Formulação das questões de competência. São baseadas nos cenários obtidos no passo anterior e podem ser consideradas como exigências na forma de questões. Uma ontologia deve ser capaz de representar estas questões usando sua terminologia e caracterizar as respostas para as questões usando axiomas e definições. Estas questões são questões de competência informais, já que não são expressas em uma linguagem formal de ontologia. As questões servem como restrições sobre o que a ontologia pode ser, e são usadas para avaliar se os propósitos ontológicos estão de acordo com os requisitos;
3. Especificação da terminologia da ontologia dentro de uma linguagem formal:
 - (a) Obter a terminologia informal. O conjunto de termos usado pode ser obtido das questões de competência informais, já definidas. Estes termos servem de base para a especificação da ontologia em uma linguagem formal;
 - (b) Especificação da terminologia formal. A terminologia da ontologia é especificada usando formalismo. Estes termos permitirão que as definições e restrições sejam posteriormente expressadas por meio de axiomas.
4. Formulação das questões de competência formais usando a terminologia da ontologia. Após a terminologia da ontologia ser definida, as questões de competência são definidas formalmente;
5. Especificação dos axiomas e definições para os termos na ontologia dentro de uma linguagem formal. Os axiomas em uma ontologia especificam as definições dos termos na ontologia e restrições sobre sua interpretação; eles são definidos como sentenças de primeira ordem. O desenvolvimento de axiomas para a ontologia, verificando as questões de competência formais é um processo iterativo;
6. Estabelecer condições para caracterizar a completude da ontologia. Deve-se definir as condições sobre as quais as soluções para as questões de competência são completas.

METHONTOLOGY

O METHONTOLOGY permite a construção de ontologias no nível de conhecimento e inclui: a identificação do processo de desenvolvimento da ontologia, um ciclo de vida baseado na evolução dos protótipos, e técnicas particulares para execução de cada atividade. Este modelo é apoiado pelo *Ontology Design Environment* - ODE (López, 1999).

- Processo de Desenvolvimento da ontologia

O processo de desenvolvimento da ontologia define as atividades executadas quando construindo ontologias. Segundo López (1999) existem três categorias de atividades:

1. Atividades de Gerenciamento de Projeto - incluem planejamento, controle e garantia de qualidade. O planejamento identifica quais tarefas serão executadas, como elas serão organizadas, quanto tempo e quais os recursos são necessários para sua conclusão. O controle garante que as tarefas planejadas serão concluídas da forma que elas foram projetadas para serem desempenhadas. A garantia da qualidade garante que a qualidade de cada produto resultante (ontologia, *software* e documentação) é satisfatória;
2. Atividades Orientadas a Desenvolvimento - incluem especificação, conceitualização, formalização e implementação. Especificação define o objetivo da ontologia, em quais situações pretende-se utilizá-la e identifica o conjunto de termos a serem representados, juntamente com suas características. A fase de conceitualização organiza e converte a visão do domínio percebida informalmente em uma especificação semi-formal, usando um conjunto de representações intermediárias que o especialista no domínio e o ontologista (desenvolvedor da ontologia) podem entender. A formalização transforma o modelo conceitual em um modelo formal ou semi-computável. Implementação constrói modelos computáveis em uma linguagem computacional. A manutenção atualiza e corrige a ontologia;
3. Atividades de Suporte - incluem uma série de atividades, desempenhadas ao mesmo tempo que as atividades orientadas ao desenvolvimento, sem as quais a ontologia não pode ser construída. Elas incluem aquisição do conhecimento, avaliação, integração, documentação e gerenciamento de configuração. Aquisição do conhecimento adquire o conhecimento através de um dado domínio. Avaliação faz um julgamento técnico das ontologias, seus ambientes de *software* e documentação associados com respeito a cada fase e entre as fases dos seus ciclos de vida. Integração de ontologias é necessária quando a nova ontologia que está sendo construída reusa outras ontologias que já estão disponíveis. A documentação detalha, claramente e exaustivamente, cada uma das fases concluídas e os produtos gerados. Gerenciamento de configuração registra todas as versões da documentação, *software* e código de ontologia para controle de mudanças.

- Ciclo de vida da ontologia

Ele identifica o conjunto de estágios através do qual a ontologia segue durante seu tempo de vida, descreve quais atividades serão desempenhadas em cada estágio e como os estágios estão relacionados (López, 1999).

Método 101

Este método utiliza um enfoque iterativo para o desenvolvimento de ontologia, ou seja, ele parte de uma versão inicial da ontologia, que será revisada e refinada aos poucos. Noy and McGuinness (2001) enfatizam algumas regras consideradas fundamentais no projeto de ontologia:

1. Não há um modo correto de modelar um domínio - há alternativas viáveis;
2. O desenvolvimento de ontologia é necessariamente um processo iterativo;
3. Conceitos em ontologia devem estar ligados a objetos e relacionamentos em seu domínio de interesse;

Levando isto em consideração, o método é baseado em sete passos.

1. Determinar o domínio e escopo da ontologia. Para começar definindo o escopo e domínio da ontologia são apresentadas algumas das questões básicas:
 - Qual o domínio que a ontologia cobrirá?
 - Qual será o uso da ontologia?
 - Para quais tipos de perguntas a informação na ontologia deve proporcionar respostas?
 - Quem usará e manterá a ontologia?

As respostas para estas perguntas podem mudar durante o processo de projeto da ontologia, mas em qualquer momento pode ajudar a definir o escopo do modelo. Uma das formas de determinar o escopo da ontologia é esboçar uma lista de questões que uma base de conhecimento baseada na ontologia deve capaz de responder: as questões de competência (Grüninger and Fox, 1995). Estas questões servirão para testar a ontologia posteriormente e podem ser apenas um esboço, não precisam ser exaustivas.

2. Considerar o reuso de ontologias existentes. O reuso de ontologias pode ser um requisito se o novo sistema interage com outras aplicações que já tem ontologias ou vocabulários controlados. Podemos encontrar bibliotecas de ontologias reusáveis na web, tais como Ontolingua. Através do reuso de ontologias é possível refinar e estender as fontes existentes para o nosso domínio particular.
3. Enumerar os termos importantes da ontologia. Os termos sobre os quais podem ser feitas declarações devem ser listados, junto com suas propriedades. Por exemplo, em uma ontologia sobre vinhos, termos importantes relacionados a vinhos podem ser: localização, cor e uva. Inicialmente é importante ter uma lista de termos, sem se preocupar com sobreposição entre os conceitos que eles representam, relações entre os termos e as propriedades que eles podem ter. Os próximos dois passos - definição da hierarquia de classes e das propriedades dos conceitos - estão intimamente interligados. Geralmente, nós criamos algumas definições de conceitos na hierarquia e descrevemos propriedades desses conceitos; depois definimos mais conceitos e assim por diante.

4. Definir classes e a hierarquia de classes. Há vários enfoques no desenvolvimento de hierarquia, um deles descrito na seção 2.4.3:
 - Um processo de desenvolvimento *top-down* começa com a definição dos conceitos mais gerais no domínio e posteriormente, é feita a especialização dos conceitos;
 - Um processo de desenvolvimento *bottom-up* começa com a definição das classes mais específicas, com subsequente agrupamento destas classes em conceitos mais gerais;
 - Um processo de desenvolvimento de combinação é formado pela combinação dos dois enfoques anteriores. Nós definimos os conceitos que mais se sobressaem primeiro e depois os generalizamos e os especializamos adequadamente.

Qualquer que seja o enfoque escolhido, geralmente começamos definindo classes. Da lista formada no passo 3, nós selecionamos os termos que descrevem os objetos que possuem uma existência independente. Esses termos serão classes na ontologia. Nós organizamos as classes em uma taxonomia hierárquica, verificando que: se a classe A é uma superclasse de B, então toda instância de B é também uma instância de A. Neste caso, a classe B representa um conceito que é “um tipo de” A.

5. Definir as propriedades das classes (*slots*). Após definirmos algumas classes, nós devemos descrever a estrutura interna dos conceitos. Os termos que sobraram da lista de termos do passo 3 provavelmente são propriedades. Para cada propriedade na lista, nós devemos determinar quais as classes que ela descreve. Todas as subclasses de uma classe herdam os *slots* desta classe.
6. Definir os valores das propriedades (*facet*s). Os *slots* podem ter diferentes facetas descrevendo o tipo de valor, valores permitidos, o número de valores (cardinalidade), e outras características que os valores dos *slots* podem ter. Um *slot* pode ter como tipo um valor que é uma instância, cujo escopo é uma classe específica.
7. Criar instâncias. O último passo é criar instâncias individuais das classes na hierarquia. Para isto, é necessário: escolher uma classe, criar uma instância dessa classe e preencher os valores dos *slots*.

Neste trabalho utilizaremos esta metodologia por apresentar passos mais bem definidos e explicados, de forma que é possível segui-los com facilidade. Além dessas características, este método considera o reuso de ontologias existentes. Como nesta pesquisa partimos da ontologia de Gerência de Versionamento descrita em Porto and Bruggemann (2006), construída nesta metodologia, optamos por continuar nesta mesma abordagem de construção ontológica.

2.4.4 Ferramentas para construção de ontologias

Algumas ferramentas têm sido utilizadas para auxiliar o desenvolvimento e a construção de ontologias. Segundo Almeida (2003) critérios devem ser definidos para que as ferramentas de construção de ontologias possam ser comparáveis. Como afirma Campos (2004) um modelo conceitual deve ser visto, também, como um espaço comunicacional em que transpomos o mundo fenomenal para um espaço de representação. E a ontologia utiliza-se da representação gráfica, como ferramenta para garantir um projeto lógico mais bem estruturado de um sistema. Em geral, as ferramentas utilizam linguagens de representação para a construção das ontologias. A seguir algumas ferramentas para a construção, uso e edição de ontologias e suas descrições baseado no trabalho de (Almeida, 2003):

- *Ontolingua*: Conjunto de serviços que possibilitam a construção de ontologias compartilhadas entre grupos. Permite acesso a uma biblioteca de ontologias, tradutores para linguagens e um editor para criar e navegar pela ontologia (Farquhar et al., 1996);
- *Generic Knowledge Base Editor - GKB-Editor*: Ferramenta para navegação e edição de ontologias por meio de sistemas de representação baseados em frames. Oferece interface gráfica, em que os usuários podem editar diretamente a base de conhecimento e selecionar a parte que é de seu interesse (Paley and Karp, 1997);
- *Java Ontology Editor - JOE*: Ferramenta para construção e visualização de ontologias. Proporciona gerenciamento do conhecimento em ambientes abertos, heterogêneos e com diversos usuários. As ontologias são visualizadas como um diagrama entidade-relacionamento, como o gerenciador de arquivos do MS Windows ou como uma estrutura em árvore (Mahalingam and Huhns, 1997);
- *OntoEdit*: É um ambiente gráfico para edição de ontologias que permite inspeção, navegação codificação e alteração de ontologias. O modelo conceitual é armazenado usando um modelo de ontologia que pode ser mapeado em diferentes linguagens de representação. As ontologias são armazenadas em bancos relacionais e podem ser implementadas em XML, Flogic, RDF(S) e DAML+OIL (Maedche et al., 1997);
- *Ontological Constraints Manager - OCM*: É uma ferramenta para verificar a consistência de ontologias em relação a axiomas ontológicos. É composto por duas ferramentas de edição que possibilitam verificar a ocorrência de conflitos (Kaifoglou, 2001);
- *VOID*: Ambiente para navegação, edição e gerenciamento de ontologias. Por meio de simulação, possibilita o estudo de questões teóricas, como organização de bibliotecas de ontologias e tradução entre diferentes formalismos (Heijist et al., 1997);
- *WebODE*: Ambiente para engenharia ontológica que dá suporte à maioria das atividades de desenvolvimento de ontologias. A integração com outros

sistemas é possível, importando e exportando ontologias de linguagens de marcação (Arpírez et al., 2001);

- *WebOnto*: Ferramenta que possibilita a navegação, criação e edição de ontologias, representadas na linguagem de modelagem OCML. Permite o gerenciamento de ontologias por interface gráfica, inspeção de elementos, verificação da consistência da herança e trabalho cooperativo. Possui uma biblioteca com mais de cem ontologias (Domingues et al., 2001);
- *Text-to-onto*: Proporciona um ambiente para o aprendizado e construção de ontologias a partir de textos. Os textos podem ser em linguagem natural ou formatados em HTML. O sistema é composto por um módulo de gerenciamento de textos e um extrator de informações. Os resultados são armazenados em XML (Maedche et al., 1997);
- *Protégé 2000*: É um ambiente interativo para projeto de ontologias, de código aberto, que oferece uma interface gráfica para edição de ontologias e uma arquitetura para a criação de ferramentas baseadas em conhecimento. A arquitetura é modulada e permite a inserção de novos recursos (Noy et al., 2000a). Essa ferramenta foi desenvolvida na Universidade de Stanford e está disponível para utilização gratuitamente. Ele possui uma arquitetura de metaclasses, documentos de formato padrão usados para definir novas classes em uma ontologia, que o tornam facilmente extensível e permite o seu uso juntamente com outros modelos de conhecimento. Os objetivos básicos dessa ferramenta são: apresentar interoperabilidade com outros sistemas de representação do conhecimento e facilidade de uso e configuração. Para o desenvolvimento da ontologia proposta nesse trabalho será utilizado o Protégé 2003 versão 1.8 (<http://protege.stanford.edu>). O Protégé, dentre as diversas ferramentas disponíveis no mercado para a construção de ontologias, foi escolhido para a implementação desse trabalho. As duas principais funcionalidades que levaram a essa escolha foram: a interface gráfica que permite que o desenvolvedor se concentre na modelagem das entidades sem se preocupar com o formato do arquivo ou linguagem de saída e a diversidade de modos de visualização. Além dessas funcionalidades, o Protégé mostrou ser uma ferramenta de uso fácil e intuitivo. Uma vasta documentação a respeito do uso e de exemplos de utilização são encontrados na Web (<http://protege.stanford.edu/plugins/prompt/prompt.html>). Um outro ponto importante para a escolha do Protégé como ferramenta para esse trabalho é a funcionalidade de *plugin Prompt* Noy and Musen (2001), o qual foi fundamental para realização de operações entre as ontologias de TI e de Negócio tais como o *merge* e o alinhamento baseado em *label*.

Atualmente, boa parte dos esforços de pesquisa tem sido dirigida para ontologias de sistemas. Modernas ferramentas de desenvolvimento de ontologias, como o Protégé (com o *Plugin OWL*), permitem aos usuários encontrar erros e detectar inconsistências, semelhante a um depurador em um ambiente de programação. Além disto, o Protégé gera código a partir de uma ontologia em OWL, criando classes correspondentes em Java.

2.4.5 Linguagem OWL

A *Ontology Web Language* - OWL é uma linguagem de marcação semântica para publicação e compartilhamento de ontologias na *Web*. OWL foi desenvolvida como uma extensão do vocabulário RDF e derivada da linguagem ontológica para Web DAML+OIL (Miller, 2005).

A OWL representa o significado de termos em vocabulários e relacionamentos entre eles, facilitando a interpretação do conteúdo Web por máquinas, já que oferece um vocabulário adicional com uma semântica formal. E.g: relações entre classes, cardinalidade, características de propriedades. A OWL é uma extensão do RDF. Enquanto o RDF promove a associação e integração de dados distribuídos, a OWL possibilita o raciocínio sobre esses dados (Niemann et al., 2005).

OWL é dividida em três sub-linguagens, de acordo com o nível de formalidade exigido e a liberdade dada ao usuário para a definição de ontologias: OWL *Lite*, OWL DL e OWL *Full*.

Sintaticamente, OWL Lite é a sub-linguagem mais simples que é indicada para usuários que precisam de uma hierarquia de classificação e restrições simples. Pode ser útil em migrações de taxonomias existentes para OWL, e possui complexidade formal mais baixa do que OWL DL.

OWL DL proporciona expressividade máxima e garantia de decidibilidade e completude computacional. Esta linguagem inclui todos os construtores OWL, mas adiciona algumas restrições. As restrições mais significativas são: uma classe não pode ser um indivíduo ou propriedade e uma propriedade não pode ser um indivíduo ou classe. OWL DL possui esse nome devido à sua correspondência com *Description Logics* - DL, um campo de pesquisa da lógica formal sobre o qual se baseia a linguagem OWL (McGuinness and van Harmelen, 2004).

A OWL Full provê expressividade máxima, além da sintaxe livre do RDF, mas não provê qualquer garantia computacional. A principal característica da OWL Full em comparação com a OWL DL e OWL Lite é que uma classe, por definição uma coleção de indivíduos, pode ser um indivíduo de outra classe, como em RDF Schema. Segundo Gasevic et al. (2006), OWL *Full* é uma extensão da OWL DL, que é uma extensão da OWL *Lite*; desta forma, toda ontologia OWL *Lite* é uma ontologia OWL DL e OWL *Full*, e toda ontologia OWL DL é uma ontologia OWL *Full*.

2.4.6 Ferramentas para *Merging* e *Mapping* de ontologias

Podemos classificar as ferramentas que encontram correspondências entre ontologias em 4 grupos segundo a operação realizada:

- Ferramentas para *merging* de duas ontologias para criar uma nova (E.g: OntoMerge (Dou et al., 2003));
- Ferramentas para definição de uma função de transformação que transforma uma ontologia em outra (Chalupsky, 2000);
- Ferramentas para definir um mapeamento entre conceitos em duas ontologias através da identificação de pares de conceitos relacionados (e.g: AnchorPrompt, GLUE Doan et al. (2002), OBSERVER (Mena, 1996));

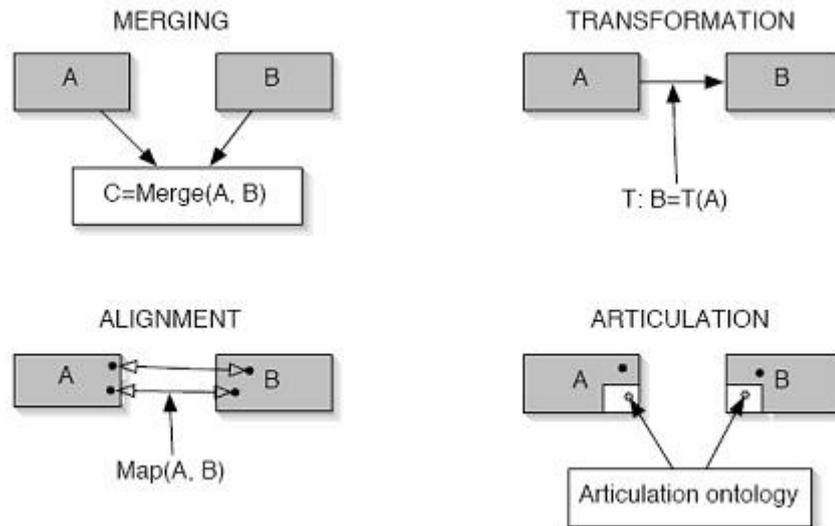


Figura 2.8: Diferentes tipos de operações entre ontologias (Euzenat, 2004).

- Ferramentas para definição de regras de mapeamento que relacionam somente partes relevantes de ontologias de origem (Mitra et al., 2000).

Outra maneira de categorizar as ferramentas é através do tipo de *input* que a ferramenta utiliza em sua análise:

- Nome de classes e definições de linguagem natural (E.g: ferramentas desenvolvidas para alinhamento semi-automático de domínios ontológicos em uma ontologia central (Hovy, 2000));
- Hierarquia de classes (e.g: Chiamera (McGuinness et al., 2000));
- Hierarquia de classes, *slots* e facets (e.g: AnchorPrompt Noy et al. (2000b), ONION (Mitra et al., 2000));
- Instâncias de classes (e.g: GLUE (Doan et al., 2002));
- Descrições de classes (e.g: ferramentas baseadas na descrição lógica como OBSERVER (Mena, 1996)).

A seguir serão descritas algumas ferramentas que foram mencionadas em maiores detalhes:

- Chimaera (McGuinness and van Harmelen, 2004) é uma ferramenta de merging interativa baseada no editor do ontologia do Ontolingua (Farquhar et al., 1996). Ela permite que um usuário utilize conjuntamente ontologias desenvolvidas em diferentes formalismos. O usuário pode solicitar uma análise ou uma orientação de Chimaera em qualquer momento do processo de merging. A ferramenta aponta, então, os locais onde a sua atenção é requerida. A única relação taxonômica que Chimaera considera é a relação

de subclasse e superclasse. Trata-se de uma ferramenta similar ao iPrompt. Entretanto, desde que se utiliza somente de uma hierarquia da classe em sua análise, falta muitas das correspondências que o iPrompt identifica. Estas correspondências incluem sugestões para *mergear slots* com nomes similares que são adicionadas às classes já *mergeadas*. Além disso, Chimaera utiliza-se da interface do ontolinguagem para edição de classes, a qual é de difícil interação e manipulação;

- Em OntoMerge (Dou et al., 2003), uma ontologia *mergeada* corresponde a união de duas ontologias de origem e um conjunto de axiomas. A primeira etapa no OntoMerge baseia-se em uma tradução de ambas as ontologias em uma representação sintática comum desenvolvida pelos projetistas da ferramenta. A seguir, são definidos axiomas os quais contém termos de ambas as ontologias. Finalmente um mecanismo de inferência estabelece conclusões baseadas nas sentenças definidas nas ontologias de origem o que gera, conseqüentemente novos termos na ontologia *mergeada*. Essa ferramenta pode ser utilizada conjuntamente com outras abordagens como iPrompt ou AnchorPrompt, pois permite o desenvolvimento de axiomas auxiliares que contribuem para uma maior compreensão do domínio ontológico;
- Em Ontomorph (Chalupsky, 2000) define um conjunto de operadores de transformação que podem ser aplicados a uma ontologia. Um especialista humano, através de uma lista inicial de matches e das ontologias de origem, define um conjunto de operadores que necessitam ser aplicados às ontologias iniciais com o objetivo de resolver diferenças entre eles. A seguir o Ontomorph aplica os operadores automaticamente para a construção da nova ontologia;
- GLUE é um sistema que emprega técnicas de aprendizagem para encontrar *mappings* (Doan et al., 2002). Essa ferramenta baseia-se em múltiplos learners que exploram as informação extraídas das instâncias criadas e da estrutura taxonômica das ontologias;
- O sistema OBSERVER (Mena, 1996) utiliza descrição lógica para responder queries e identificar mapeamentos possíveis entre duas ou mais ontologias. Primeiramente, os usuários definem um conjunto de relações entre as ontologias. A seguir, o sistema auxilia o usuário na tarefa através da identificação de sinônimos nas ontologias de origem. Uma vez definidos os mapeamentos, os usuários podem estabelecer queries sobre suas ontologias iniciais;
- O sistema ONION (Mitra et al., 2000) é baseado na representação algébrica de ontologias. Conseqüentemente, essa abordagem fornece ferramentas para a definição de relacionamentos algébricos entre as ontologias. Além desse recurso, ela utiliza-se de técnicas lexicais e gráficas para a sugestão de articulações entre duas ontologias ou mais ontologias.

Verificamos nessa seção o grande potencial da abordagem ontológica na organização dos diversos conceitos envolvidos em um determinado domínio. Todavia, para que seja realizada a recuperação e a verificação da qualidade desses termos

que representam o conhecimento de um determinado domínio, torna-se necessária uma revisão de Sistemas de Informação com foco em medidas de recuperação de informação que apresentaremos na próxima seção.

2.5 Sistemas de Recuperação de Informação

De acordo com van Rijsbergen (1975) quando falarmos de Sistemas de Recuperação de Informação (SRI) queremos dizer, tão somente, Sistemas Automáticos de Recuperação de Informação.

O objetivo de um SRI é, dado um conjunto de documentos e uma requisição do usuário, devolver o máximo de documentos relevantes e o mínimo de documentos não relevantes possível. A requisição do usuário é uma especificação de certas características desejadas nos documentos. O sistema procura retornar os documentos que se encaixem nessa especificação, na esperança de que eles sejam relevantes para o usuário. Poderíamos dizer que trata-se de um sistema de recuperação de documentos, ao invés de informação, no entanto o segundo é o mais utilizado.

Nessa descrição fica claro que documentos não-relevantes podem emergir no resultado, o que significa que alguma imprecisão é esperada. Documentos que talvez fossem relevantes ao usuário eventualmente também podem ter ficado de fora. Isso também é esperado, afinal, sabemos que caracterizar a intenção do usuário não é trivial. Podemos então simplificar o objetivo de um SRI dizendo que ele deve facilitar o acesso do usuário a informações que, se supõe, sua base de dados de documentos contenha, ou então informá-lo da impossibilidade da recuperação.

Cada uma das fases do processamento dos documentos, além do algoritmo de busca utilizado, afetam a performance geral dos SRI, o que suscita a necessidade de avaliarmos esses métodos, e o sistema como um todo, em relação a sua efetividade. Pode-se avaliar SRI em relação a diferentes aspectos. Em van Rijsbergen (1975) são citados seis aspectos no quais os SRIs podem ser avaliados:

- A cobertura de material relevante pela coleção;
- O tempo de resposta;
- A forma de apresentação dos resultados;
- O esforço despendido pelo usuário para obter a informação desejada;
- A capacidade do sistema em retornar material relevante;
- A precisão com que distingue documentos relevantes dos não-relevantes.

Avaliando a Efetividade do SRI

Avaliar quanto a efetividade é avaliar se um sistema em questão está conseguindo alcançar o seu objetivo. Para avaliar um SRI quanto à sua efetividade, devemos utilizar as medidas de *Precision* e *Recall*.

$$P = \frac{|A \cap B|}{|B|} \quad \textit{Precision}$$

$$R = \frac{|A \cap B|}{|A|} \quad \textit{Recall}$$

Figura 2.9: Medidas *Precision* e *Recall* (van Rijsbergen, 1975).

		Relevante	Não-Relevante	
Retornado	$A \cap B$	$A' \cap B$	B	
Não-Retornado	$A \cap B'$	$A' \cap B'$	B'	
		A	A'	N

Figura 2.10: Como calcular *Precision* e *Recall* adaptado de (van Rijsbergen, 1975).

Precision é a proporção entre o número de documentos relevantes existentes no resultado e o número de documentos efetivamente retornados. É uma medida da capacidade do sistema de distinguir material relevante do não-relevante.

Recall é a proporção entre o número de documentos relevantes retornados e o número de documentos relevantes que existem na coleção. É uma medida da capacidade do sistema de retornar o máximo de material relevante possível para o usuário.

Essas duas medidas nos dão uma idéia da efetividade do sistema em satisfazer as requisições do usuário. As vantagens de se utilizar *Precision* e *Recall* é que são as medidas mais utilizadas para se avaliar SRIs, e por isso mais amplamente compreendidas (van Rijsbergen, 1975).

Para entendermos como se calculam essas quantidades, as quais estão representadas na Figura 2.9, precisaremos do auxílio da Figura 2.10. O conjunto A representa o conjunto de todos os documentos relevantes a uma determinada consulta e A' os documentos não-relevantes, B os documentos recuperados e B' os não recuperados, e finalmente N é o número total de documentos existentes na coleção.

Elas são grandezas inversamente proporcionais, pois se aumentarmos o número de documentos relevantes que o sistema retorna, certamente aumentaremos também o número de documentos não-relevantes retornados, o que diminui a precisão do sistema. O contrário também é verdade, como mostra a Figura 2.11.

A curva de *Precision* e *Recall* é construída para uma determinada requisição.

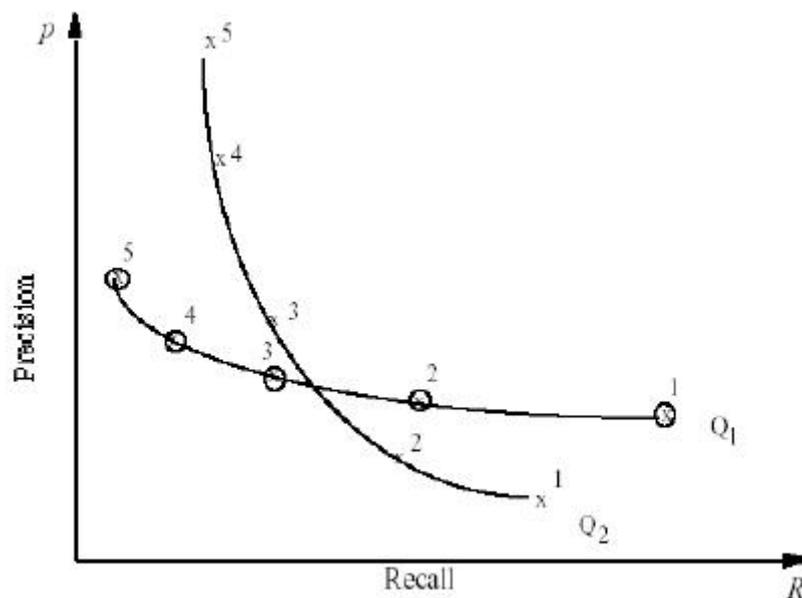


Figura 2.11: Curva de *Precision X Recall* para duas *Queries* Q1 e Q2 (van Rijsbergen, 1975).

Podemos querer avaliar o sistema quanto ao resultado de várias consultas, para que possamos tirar conclusões estatísticas mais confiáveis. Como cada resultado pode trazer um número diferente de documentos, para se utilizar o cálculo do *Recall* é necessário introduzir a noção de interpolação entre as várias curvas levantadas. A Figura 2.11 mostra a curva de *Precision x Recall* para duas *queries* distintas.

Para se comparar dois SRIs utilizando-se essas medidas, convencionou-se alguns níveis de *Recall*, por exemplo 0; 0,2; 0,5; 0,7; 1, e então a *Precision* é calculada para cada um dos níveis estabelecidos. O sistema que apresentar as melhores medidas de *Precision* para os mesmos níveis de *Recall* é considerado de maior efetividade.

Neste trabalho, foram utilizadas as medidas de *precision* e *recall* para avaliarmos a efetividade dos resultados apresentados pelos algoritmos de alinhamentos que passaremos a discutir.

2.6 Algoritmos de Alinhamento

Nesta seção apresentaremos as abordagens de alinhamento ontológico que foram utilizadas no alinhamento das ontologias de domínio estudadas e que permitiram a construção do *framework* proposto.

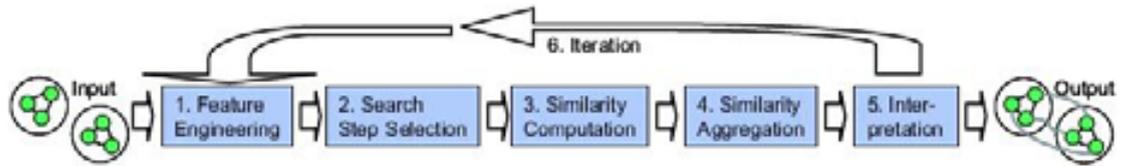


Figura 2.12: Visão geral do processo de alinhamento (Ashpole et al., 2005).

FOAM

O *Framework for Ontology Alignment and Mapping* - FOAM é uma ferramenta para alinhamento automático ou semi-automático entre duas ou mais ontologias *OWL*. Ela é baseada em heurísticas de similaridade de entidades individuais, as quais podem ser conceitos, relações e instâncias.

Na abordagem adotada pelo FOAM são utilizadas técnicas de alinhamento como *Quick Ontology Mapping - QOM* Ehrig and Staab (2004) e *PROMPT* Noy and Musen (2001), as quais apresentam em comum seis passos principais em seu algoritmo conforme apresentado em (Ashpole et al., 2005):

1. *Feature Engineering* - seleção de partes de toda a definição ontológica para descrever um trecho específico. Isto inclui características básicas como labels e disposições estruturais como também características mais complexas com o as restrições usadas em *OWL*;
2. *Search Step Slection* - escolher duas entidades das duas ontologias para comparação (e_1, e_2);
3. *Similarity Assessment* - indicar a similaridade para uma dada característica de duas entidades (e.g.: $\text{sim superconcept}(e_1, e_2) = 1.0$);
4. *Similarity Aggregation* - agregação de múltiplas similaridades determinadas para um par de entidades em uma única medida;
5. *Interpretation* - utilização de um threshold e uma estratégia de interpretação para propor um alinhamento ($\text{align}(e_1) = e'_2$). Essa estratégia pode incluir uma validação do usuário;
6. *Iteration* - como a similaridade de um alinhamento influência a similaridade dos pares de entidades vizinhas; a similaridade é propagada entre termos vizinhos.

Finalmente, são dados como resultados pares de entidades alinhadas. A Figura 2.12 ilustra a visão geral do processo de alinhamento proposto em (Ashpole et al., 2005).

Abordagens utilizadas para alinhamento;

As abordagens utilizadas neste trabalho e que estão presentes no FOAM foram:

- *PROMPT/Label-Approach* (Noy and Musen, 2001);
- *Quick Ontology Mapping* - QOM;
- *Active Ontology Alignment* - AOA;
- *Alignment Process Feature Estimation and Learning* - APFEL;
- *Active APFEL*.

A seguir descreveremos as principais características das abordagens citadas.

- *PROMPT/Label-Approach*:

Antes de utilizarmos abordagens mais avançadas para a computação dos alinhamentos entre as ontologias, testamos uma abordagem mais simples e direta, a qual é baseada unicamente nos rótulos das entidades comparadas. Para essa abordagem foi utilizado *PROMPT-tool* (Noy and Musen, 2003). O algoritmo *PROMPT*[12] de Noy and Musen (2001), que está disponível como um *plugin* do Protégé, é uma ferramenta que provê uma abordagem semi-automática para o *merging* e o alinhamento ontológico. Juntamente com *ONION* de Mitra et al. (2000) é uma das primeiras ferramentas desenvolvidas para o alinhamento ontológico. Depois de ter identificado os alinhamentos através do *matching* dos rótulos, o usuário é questionado a marcar quais os alinhamentos que devem ser realmente *mergeados* levando em consideração as possíveis inconsistências apresentadas pela ferramenta. Nesse trabalho focamos nas ações realizadas para identificar os possíveis candidatos passíveis de alinhamento. Dessa forma, nessa etapa, não foram considerados os passos de *merging*.

- *Naïve Ontology Mapping* - NOM:

Segundo Ashpole et al. (2005), a abordagem NOM trata de um método básico da comparação baseado em etiquetas e em semelhança de *strings* para computar a similaridade entre as entidades, a qual recebe maior peso quanto maior for o grau de similaridade. Através de computações sucessivas e da aplicação de um *threshold* fixo para descartar falsas evidências de similaridade, o algoritmo, a partir dos pares já encontrados, utiliza medidas estruturais de similaridade mais sofisticadas, conforme indica a Figura 2.13: Segundo Ashpole et al. (2005) esse algoritmo apresenta a mesma complexidade da abordagem *Anchor-PROMPT* Noy et al. (2000b), a qual não pudemos aplicar nesse trabalho devido à indisponibilidade do respectivo *plugin* para o Protégé.

- *Quick Ontology Mapping* - QOM:

Esta abordagem proposta em Ashpole et al. (2005) tem como enfoque a preocupação em torno da eficiência dos algoritmos de alinhamento. Ela

Comparing	No.	Feature	Similarity Measure
Concepts	1	(label, X_1)	string(X_1, X_2)
	2	(identifier, X_1)	explicit(X_1, X_2)
	3	(X_1 , sameAs, X_2) relation	object(X_1, X_2)
	4	(direct relations, Y_1)	set(Y_1, Y_2)
	5	all (inherited relations, Y_1)	set(Y_1, Y_2)
	6	all (superconcepts, Y_1)	set(Y_1, Y_2)
	7	all (subconcepts, Y_1)	set(Y_1, Y_2)
	8	(subconc., Y_1) / (superconc., Y_2)	set(Y_1, Y_2)
	9	(superconc., Y_1) / (subconc., Y_2)	set(Y_1, Y_2)
	10	(concept siblings, Y_1)	set(Y_1, Y_2)
	11	(instances, Y_1)	set(Y_1, Y_2)
Relations	1	(label, X_1)	string(X_1, X_2)
	2	(identifier, X_1)	explicit(X_1, X_2)
	3	(X_1 , sameAs, X_2) relation	object(X_1, X_2)
	4	(domain, X_{d1}) and (range, X_{r1})	object(X_{d1}, X_{d2}), (X_{r1}, X_{r2})
	5	all (superrelations, Y_1)	set(Y_1, Y_2)
	6	all (subrelations, Y_1)	set(Y_1, Y_2)
	7	(relation siblings, Y_1)	set(Y_1, Y_2)
	8	(relation instances, Y_1)	set(Y_1, Y_2)
Instances	1	(label, X_1)	string(X_1, X_2)
	2	(identifier, X_1)	explicit(X_1, X_2)
	3	(X_1 , sameAs, X_2) relation	object(X_1, X_2)
	4	all (parent-concepts, Y_1)	set(Y_1, Y_2)
	5	(relation instances, Y_1)	set(Y_1, Y_2)
Relation-Instances	1	(domain, D_1) and (range, R_1)	object(D_1, D_2), (R_1, R_2)
	2	(parent relation, Y_1)	set(Y_1, Y_2)

Figura 2.13: Características e medidas de similaridade para diferentes tipos de entidades em NOM (Ashpole et al., 2005).

Comparing	No.	Feature	Similarity Measure
Concepts	1	(label, X_1)	string(X_1, X_2)
	2	(identifier, X_1)	explicit(X_1, X_2)
	3	(X_1, sameAs, X_2) relation	object(X_1, X_2)
	4	(direct relations, Y_1)	set(Y_1, Y_2)
	5a	(relations of direct superconc., Y_1)	set(Y_1, Y_2)
	6a	(direct superconcepts, Y_1)	set(Y_1, Y_2)
	7a	(direct subconcepts, Y_1)	set(Y_1, Y_2)
	8a	(subconc., Y_1) / (superconc., Y_2)	set(Y_1, Y_2)
	9a	(superconc., Y_1) / (subconc., Y_2)	set(Y_1, Y_2)
	10	(concept siblings, Y_1)	set(Y_1, Y_2)
	11a	(direct instances, Y_1)	set(Y_1, Y_2)
Relations	1	(label, X_1)	string(X_1, X_2)
	2	(identifier, X_1)	explicit(X_1, X_2)
	3	(X_1, sameAs, X_2) relation	object(X_1, X_2)
	4	(domain, X_{d1}) and (range, X_{r1})	object($X_{d1}, X_{d2}, X_{r1}, X_{r2}$)
	5a	(direct superrelations, Y_1)	set(Y_1, Y_2)
	6a	(direct subrelations, Y_1)	set(Y_1, Y_2)
	7	(relation siblings, Y_1)	set(Y_1, Y_2)
	8a	(direct relation instances, Y_1)	set(Y_1, Y_2)
Instances	1	(label, X_1)	string(X_1, X_2)
	2	(identifier, X_1)	explicit(X_1, X_2)
	3	(X_1, sameAs, X_2) relation	object(X_1, X_2)
	4a	(direct parent-concepts, Y_1)	set(Y_1, Y_2)
	5	(relation instances, Y_1)	set(Y_1, Y_2)
Relation-Instances	1	(domain, X_{d1}) and (range, X_{r1})	object($X_{d1}, X_{d2}, X_{r1}, X_{r2}$)
	2	(parent relation, Y_1)	set(Y_1, Y_2)

Figura 2.14: Características e medidas de similaridade para diferentes tipos de entidades em QOM (Ashpole et al., 2005).

surgiu a partir da otimização do efetivo, mas ineficiente algoritmo NOM. A idéia central desta abordagem é a utilização de heurísticas para diminuir o número dos candidatos passíveis de alinhamento. Um exemplo de heurística utilizada é a comparação de somente *labels* próximas ou vizinhas considerando-se a estrutura da ontologia, assim, diminuir-se-ia o número de pares de entidades a serem submetidos às sucessivas computações do algoritmo. A Figura 2.14 indica as características e medidas de similaridade para diferentes tipos de entidades. Nela se percebe a modificações de algumas características da Figura 2.13 (indicadas pelo índice 'a' na coluna No.) as quais denotam considerações à cerca da eficiência do algoritmo.

- *Active Ontology Alignment* - AOA:

A Active Ontology Alignment descrita em Ashpole et al. (2005) trata-se de uma abordagem semi automática que considera a opinião do usuário para a validação do alinhamento, ou seja, a idéia é incluir a participação do usuário durante a execução de algum algoritmo automático de alinhamento. Uma

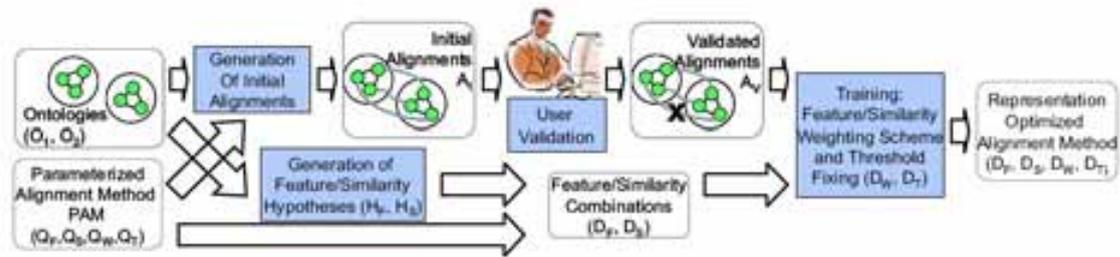


Figura 2.15: Processo detalhado do algoritmo APFEL (Ashpole et al., 2005).

vez concluído o cálculo da similaridade de determinado par de entidades, é perguntado ao usuário se ele confirma ou não a validade do alinhamento proposto pelo algoritmo automatizado. A resposta do usuário para o alinhamento proposto é então armazenada com o correspondente valor 1 ou 0 conforme a resposta seja afirmativa (*true*) ou negativa (*false*). Essas respostas são essenciais para que nas iterações subseqüentes elas sejam utilizadas no cálculo de uma nova e mais correta lista de alinhamentos automáticos.

- *Alignment Process Feature Estimation and Learning - APFEL:*

O algoritmo APFEL Ashpole et al. (2005), o qual está detalhado na Figura 2.15, tem como principal característica a existência de uma etapa de treinamento, ou seja, a utilização de alinhamentos pré definidos pelo usuário para serem considerados em proposições futuras de alinhamentos.

Essa abordagem tem como premissa a utilização do conhecimento que o usuário ou especialista tenha a cerca dos termos de ambos os domínios ontológicos os quais se deseja alinhar.

Assim, o FOAM quando configurado para a utilização dessa abordagem, requer como entrada, além do par de ontologias, um arquivo contendo alinhamentos prévios de termos pertencentes às duas ontologias.

- *Active APFEL:*

A abordagem *Active APFEL* proposta em Ashpole et al. (2005) trata-se de um método híbrido o qual apresenta as características simultâneas das abordagens AOA e APFEL . Assim, essa abordagem considera a participação do usuário para a validação dos alinhamentos propostos dos termos das ontologias de domínio fontes, o que implica a característica de semi automaticidade dessa abordagem; bem como considera a necessidade da utilização do conhecimento que o usuário ou especialista tenha a cerca dos termos de ambos os domínios ontológicos os quais se deseja alinhar. O FOAM, quando configurado para essa abordagem, requer como entrada, além do par de ontologias, um arquivo contendo alinhamentos prévios de termos pertencentes às duas ontologias. Além disso, ele considera, em tempo de execução, a opinião do usuário quanto aos alinhamentos sugeridos.

$$\text{Recall: } r = \frac{\# \text{ correct_found_alignments}}{\# \text{ all_correct_alignments}}$$

$$\text{Precision: } p = \frac{\# \text{ correct_found_alignments}}{\# \text{ all_found_alignments}}$$

$$\text{F-Measure: } f = \frac{(b^2+1)pr}{b^2p+r} \quad b=1$$

Figura 2.16: Métricas utilizadas para verificação da qualidade do alinhamento (Ashpole et al., 2005).

Descrição da Funções f , p , r

Foram utilizadas métricas padronizadas para a recuperação da informação com a finalidade de verificar a qualidade das diferentes abordagens descritas em (Ashpole et al., 2005). A Figura 2.16 indica as medidas de $recall(r)$, $precision(p)$ e $f\text{-measure}(f)$ utilizadas:

Consideramos a $f\text{-measure}$ com a mais relevante nesse estudo pelo fato representar o resultado balanceado das medidas r e p . Além dessas medidas foram utilizados os seguintes parâmetros:

- *Time*: tempo requerido para descobrir os alinhamentos. O tempo utilizado para interação com o usuário é desconsiderado;
- *User Validated Pairs*: número de pares validados pelo usuário em tempo de execução. Nas abordagens totalmente automatizadas, esse valor é 0.

Um comparativo das complexidades dos algoritmos envolvidos pode ser verificado na Tabela 2.2.

Tabela 2.2: Complexidades de algumas abordagens de alinhamento (Ashpole et al., 2005).

Algoritmo	Complexidade
<i>Label/PROMPT</i>	$O(n^2.1)$
<i>Anchor-PROMPT</i>	$O(n^2.log^2(n))$
GLUE	$O(n^2)$
NOM	$O(n^2.log^2(n))$
QOM	$O(n.log(n))$

Conforme percebemos na Figura 2.17, o algoritmo *Active APFEL* executa passos totalmente automatizados, representados pelo algoritmo QOM, e passos semi-automatizados, os quais são executados por meio da iteração com o usuário (algoritmo AOA) e por uma máquina de aprendizado (algoritmo APFEL). Como

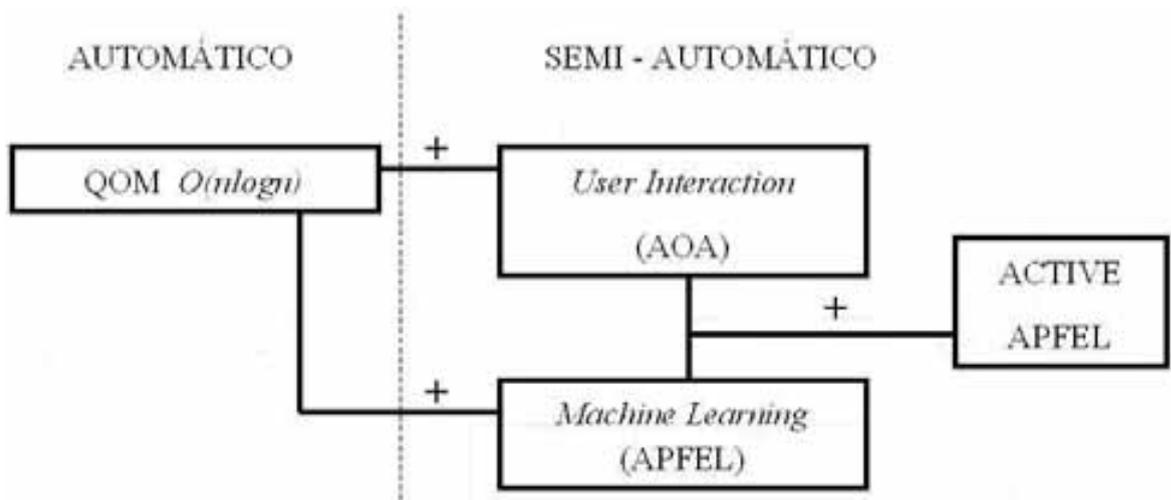


Figura 2.17: Análise de Complexidade do Algoritmo *Active APFEL*.

não encontramos em Ashpole et al. (2005) qual é a técnica de aprendizado utilizada pelo algoritmo APFEL, nem temos condições de analisar a complexidade de um algoritmo que envolve a interação com o usuário, não apresentamos, a exemplo de Ashpole et al. (2005), a complexidade dos algoritmos APFEL e *Active APFEL* neste trabalho.

Este capítulo apresentou um levantamento dos principais fundamentos teóricos relacionados ao *framework* ontológico proposto neste trabalho. No próximo capítulo, será realizada uma breve revisão de literatura de trabalhos correlatos que servem como premissa a fundamentação teórica apresentada.

Capítulo 3

Trabalhos Correlatos

Um importante ponto na busca pela compreensão dos processos e atividades inerentes ao corpo funcional de uma organização está relacionado com a extração e a representação do conhecimento implícito dos membros de uma organização. Alguns trabalhos relacionados mostram claramente esta preocupação:

- van Vliet et al. (2006) propôs o método denominado *Recovering Architectural Assumption Method* - RAAM para recuperar e modelar explicitamente as decisões tomadas durante o processo de desenvolvimento de *software*. Para realizar a recuperação e a identificação dos processos executados, o método provê um *tool box* na qual realiza a análise de diferentes fontes, tais como: entrevistas com funcionários, relatórios financeiros, documentação, controle de versionamento, análise de código fonte entre outros. Como resultado da aplicação do *tool box* é obtida uma lista de suposições ou *assumptions* que indicam com foram tomadas as principais decisões arquiteturais que nortearam a implementação dos *software*. O estudo revelou que o método é poderoso para a realização da recuperação de conhecimentos implícitos. Todavia, mostrou-se frágil na tarefa inicial de encontrar as pessoas certas para realizar um entendimento das *assumptions* relevantes;
- O *Knowledge Evolution Cycle* proposto em Rus and Lindvall (2003) descreve uma maneira para a partir do conhecimento tácito evoluir para o conhecimento explícito. Nessa pesquisa, o autor defende a premissa de que o conhecimento humano é o principal ativo de muitas companhias. Dessa forma, ele deve ser preservado e disseminado de um nível individual para um nível organizacional. A pesquisa, que também considera passos suportados pelo método RAMM van Vliet et al. (2006), demonstrou que o gerenciamento do conhecimento envolve muitos aspectos tais como: sócio-culturais, organizacionais e tecnológicos. A exemplo de nossa proposta de estudo, o foco do trabalho de Rus and Lindvall (2003) foram os aspectos tecnológicos. Todavia, também consideramos o conhecimento pertinente a área de negócio da organização;
- Segundo Bass et al. (2003), uma arquitetura de *software* manifesta antecipadamente as decisões de projeto, tais decisões determinam o desenvolvimento

do sistema, o *deployment* e a evolução. Geralmente essas suposições ou (*assumptions*) não são modeladas explicitamente, desse modo, torna-se mais difícil em um momento futuro encontrar as razões que levaram a essa ou aquela decisão de projeto. Dessa forma, o método proposto, assume como base um acesso livre ao código fonte do produto e às estatísticas do processo de desenvolvimento;

- Maier et al. (2002) apresenta o projeto de uma metodologia geral que servirá de base para, a partir de uma Intranet comum e solução de arquivos, chegarmos a uma estrutura de suporte de TI. Nesse estudo é descrita uma abordagem em direção ao suporte de conhecimento dos funcionários em uma organização, o qual é baseado em uma firme interconexão de documentos, conhecimento, organização e processos.

Uma outra preocupação das organizações é a estratégia de negócio, preparação tecnológica e performance organizacional. Nesse sentido, o estudo de Croteau and Bergeron (2006) possibilitou a identificação dos perfis de *deployment* tecnológico associados com os vários tipos de estratégias de negócio. Esse artigo também propiciou um maior entendimento do impacto da disciplina de Sistemas de Informação na performance organizacional conforme recomendado por DeLone and McLean (1992), assim como a interação entre estratégia de negócio e outros componentes organizacionais como o *deployment* tecnológico (Miller, 1998).

Visando a otimização dos processos Niessink and van Vilet (1999) propuseram um modelo chamado *IT Service CMM* o qual pode ser usado para estabelecer o nível de maturidade dos processos de TI e identificar direções para o atingimento de melhorias dos serviços. Esse modelo surgiu a partir de esforços para desenvolver um *framework* que visava prover qualidade aos serviços de TI das organizações. Segundo os autores, o modelo proposto preenche a lacuna entre *frameworks* de gerenciamento de qualidade baseado no TQM e as melhores práticas preconizadas pelo ITIL. A mais importante lição aprendida nessa pesquisa é que as melhorias dos serviços de TI podem ser consideradas um sucesso se algumas condições organizacionais forem atendidas, ou seja, algumas práticas necessitam de outras práticas como pré requisito, como exemplo, podemos citar:

- o gerenciamento de problemas necessita de um gerenciamento consistente de incidentes;
- a implementação de um catálogo de serviço necessita de informações históricas de ANS e performance.

Todavia, outras práticas podem ser introduzidas em qualquer organização de serviços de TI, tais como:

- *Service Evaluations*;
- *Service Specification and Reporting*.

Todas essas lições são refletidas no IT Service CMM

Da mesma forma, para otimizar os processos e identificar pontos de melhoria, bem como planos para atingí-los, van der Raadt et al. (2005) propõe um modelo

multidimensional baseado no alinhamento e na maturidade de processos para o estabelecimento de uma arquitetura de *software*. Esse modelo considera que o alinhamento entre os processos, bem como a maturidade dos mesmos são complementares para o desenvolvimento de um arquitetura de *software*. Segundo os autores, os modelos de maturidade vêem o alinhamento como uma subvariável que explica o nível de maturidade de uma arquitetura. Os modelos de alinhamento, por sua vez, enxergam a maturidade como uma subvariável que permite o estabelecimento do alinhamento. A pesquisa também menciona que a correlação entre o alinhamento e a maturidade torna-se ainda mais evidente quando consideramos o fato de que ambas as abordagens consideram variáveis como: governança, processos, comunicação e escopo. Em nossa proposta, também nos fundamentamos nessas variáveis, especialmente no que se refere à governança, através dos conceitos apresentados pelo modelo COBIT, e à gestão por processos de TI, por meio das melhores práticas preconizadas pelo modelo ITIL.

Diante do contexto da constante busca pela qualidade dos serviços e do alinhamento entre o negócio e a área de TI, merecem destaque duas grandes abordagens o modelo ITIL e a arquitetura SOA, cuja integração tem revelado grandes progressos nessa direção, conforme é demonstrado em (Shum and Dhillon, 2005). Nesse estudo verificou-se que a metodologia *Business Service Management* - BSM, a qual está baseada nas melhores práticas do ITIL, possibilita à organização a gerenciar TI a partir de uma perspectiva do negócio. Através da identificação e do mapeamento dos processos críticos de negócio, os quais são subjacentes a infraestrutura de TI e serviços, o BSM conecta serviços de negócio chave aos serviços de TI que os gerenciam, tais como: roteadores, servidores e aplicações. Enquanto o BSM realiza o mapeamento, a arquitetura SOA atua como um framework que conecta a infra-estrutura. Assim, podemos citar que as abordagens BSM, SOA e ITIL são complementares, e que quando integradas obtém-se um incremento na agilidade de TI.

Objetivando a integração de conceitos entre diferentes linguagens, as ontologias vêm sendo exploradas no mundo da pesquisa. Para facilitar a construção de ontologias, sejam eles de negócio, de gestão de TI ou de desenvolvimento de *software*, a técnica de modelagem de processos é de grande valia. Assim, o trabalho correlato mais relacionado a esta proposta foi o realizado em (Herborn and Wimmer, 2006). Nessa pesquisa foi desenvolvido um *framework* para a construção de uma ontologia única para os Registros de Negócios na *European Community* - EC. Trata-se de uma pesquisa conduzida conjuntamente com o projeto *Business Register Interoperability Throughout Europe* - BRITE e que está disponível em: <http://www.ecgi.org/brite>. Esse trabalho utilizou-se inicialmente da modelagem de processos para compreender as peculiaridades regionais de cada um dos países europeus envolvidos. O objetivo era possibilitar a interoperabilidade entre diferentes sistemas tecnológicos legados e aspectos de nível organizacional e semântico peculiares de cada nação. O *framework* desenvolvido, ajudou a transpor barreiras tais como incompatibilidade de sistemas, divergências de formatação de dados, questões de linguagens, além de diferenças de políticas e de estratégias que dificultavam o processo de registro de organizações que atuam na EC. A Figura 3.1 indica o objeto estudado por (Herborn and Wimmer, 2006). Nela percebemos que em cada país da EC havia, no órgão governamental de registro, um

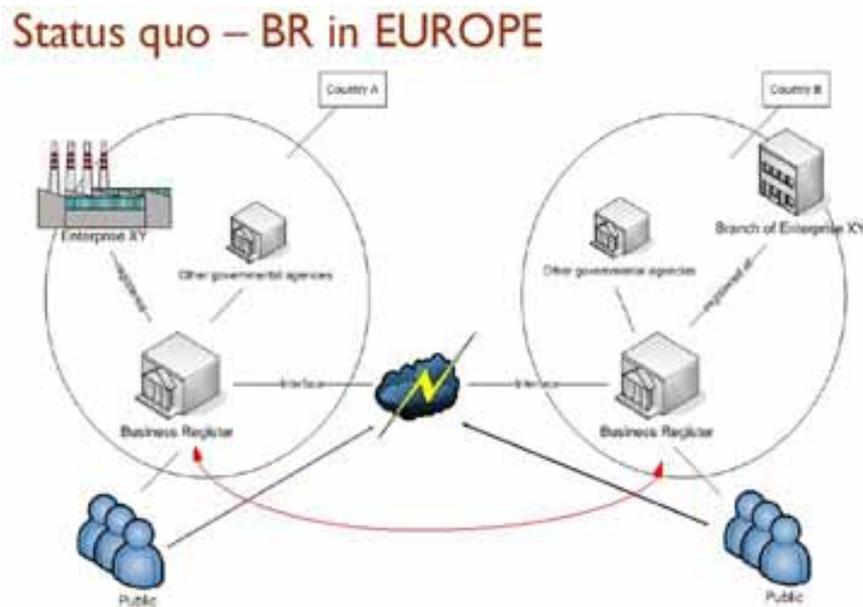


Figura 3.1: Ontologias dos países da Comunidade Europeia (Herborn and Wimmer, 2006).

processo particular de registro que dificultava o reconhecimento dessa organização em um outro país. Assim, ao se cadastrar, no país A, a empresa XY no órgão governamental de registro, havia um problema de interface entre esse órgão e o órgão respectivo do país B, o que ocasionava um *branch* no reconhecimento dessa empresa no país B. Essa situação estava causando grandes problemas comerciais em um mercado emergente e com grande potencial como o Mercado Comum Europeu.

O *framework* proposto em Herborn and Wimmer (2006) é composto por nove passos, a saber:

1. Definir o *High Level* do Domínio Ontológico - HLDO - no caso uma única ontologia de domínio que comporte os termos dos diversos países da EC. A Figura 3.2 apresenta os passos utilizados em Herborn and Wimmer (2006) para a construção da ontologia de alto nível;
2. Identificar domínios ontológicos de interesse - consiste na identificação da ontologia de domínio particular do processo de registro de negócios de cada país;
3. Selecionar subconjuntos de vocabulário do domínio de interesse - que corresponde à seleção dos termos relevantes para a composição da HLDO;
4. *Merging* dos subconjuntos de vocabulários nacionais - que corresponde a intercalação dos termos de cada país;
5. Fazer o mapeamento dos subconjuntos *mergeados* no HLDO - que consiste na correta inclusão do subconjunto de termos de cada país na HLDO;

6. Adicionar semântica ao HLDO - que consiste na adição de significado lógico aos termos da HLDO;
7. Identificação dos processos ontológicos de *Business Registration Process Ontologies* - BRPO - consiste na identificação da ontologia de processo particular do processo de registro de negócios de cada país;
8. Desenvolver o high level de BRPO - no caso uma única ontologia de processo que comportasse o registro de negócio de toda a EC;
9. Integrar os processos - que corresponde à integração dos diversos processos nacionais em um macro-processo de registro de negócio que permeie toda a EC.

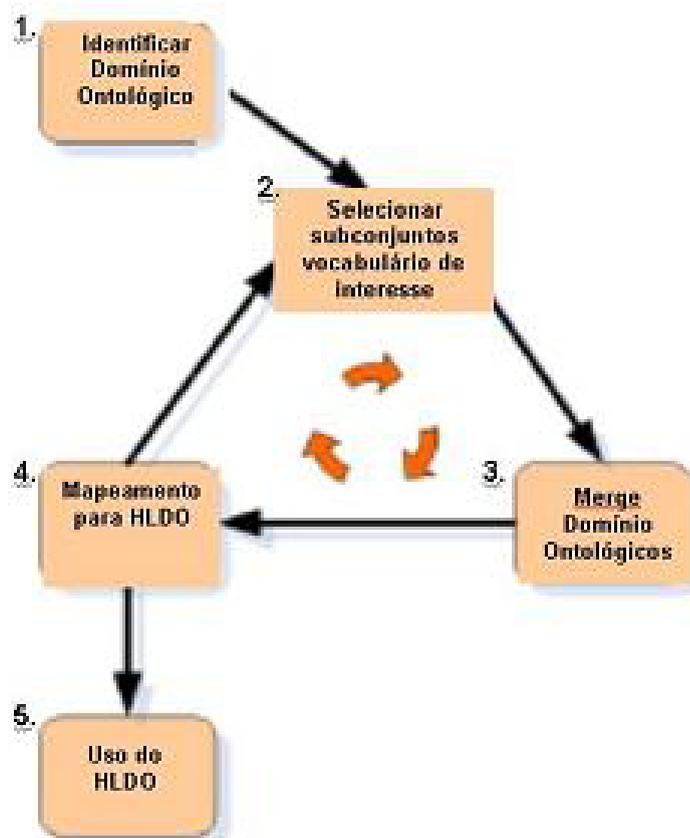


Figura 3.2: Passos para a construção do HLDO (Herborn and Wimmer, 2006).

Da mesma forma que as pesquisas de van Vliet et al. (2006), Bass et al. (2003) e Maier et al. (2002) apresentadas neste capítulo, o presente estudo também tem a preocupação em realizar a identificação e a explicitação do conhecimento envolvido nos mais diversos processos organizacionais. Somente através do mapeamento desse conhecimento é que a organização poderá elevar a maturidade de seus processos, sejam eles processos de TI, conforme indicado por Niessink and van Vliet (1999) ou processos de Negócio. Assim, para se atingir o objetivo de

alinhar o negócio e a TI de uma empresa são utilizados como premissa os conceitos recomendados pelo modelo ITIL e pela arquitetura SOA, conforme indicado em (Shum and Dhillon, 2005).

Como ponto de partida para este trabalho foi utilizado o *framework* proposto por Herborn and Wimmer (2006), uma vez que nele também foi considerada a abordagem de modelagem de processos como etapa necessária à compreensão e a implementação da ontologia do domínio envolvido. Todavia, no estudo citado, os domínios correspondiam aos países da comunidade europeia enquanto que em nossa pesquisa os domínios correspondem aos ambientes de TI e de negócio de uma organização. Assim, enquanto a abordagem de Herborn and Wimmer (2006) objetiva o alinhamento entre as peculiaridades legais, lingüísticas e tecnológicas dos países integrantes da comunidade europeia, nosso estudo se propõe, através da utilização de uma abordagem ontológica, a alinhar os conceitos envolvidos nos processos de TI e de negócio de uma organização.

Para a construção de um *framework* ontológico amplo, que contemple os diversos termos utilizados pelas áreas usuárias, de negócio e o próprio time de tecnologia torna-se necessário não só utilizar as abordagens CMM como em Nissink and van Vilet (1999) ou ITIL e SOA como em (Shum and Dhillon, 2005). Assim, como nossa proposta objetiva elucidar a infinidade de conceitos existentes entre os técnicos de TI que se empenham em controlar seu ambiente operacional, os responsáveis pelo desenvolvimento de *software* que buscam otimizar suas aplicações e técnicas de desenvolvimento e a camada negocial que visa melhor satisfazer seus clientes e acionistas, nosso *framework* se propõe a compartilhar e utilizar termos de outros modelos reconhecidos internacionalmente, tais como: COBIT, CMMI, RUP, MPS-BR e PMBOK.

Capítulo 4

Framework Proposto

Neste capítulo apresentaremos inicialmente o *framework* ontológico proposto para permitir uma linguagem comum às camadas organizacionais, em especial as camadas de TI e de negócio da organização. Na sequência, detalharemos a sequência de passos utilizada para a construção do *framework*.

4.1 Proposta de Trabalho

Este trabalho propõe um *framework* ontológico que possibilita uma linguagem comum às camadas de negócio e TI permitindo o alinhamento entre estas duas áreas. A Figura 4.1 ilustra o *framework* proposto onde se destacam duas visões, a saber:

- A visão tecnológica que tem como foco as disciplinas de Gestão preconizadas pelas melhores práticas do modelo ITIL. Nessa visão destaca-se a Ontologia de TI cuja composição basea-se nos conceitos e termos propostos pelo modelo ITIL;
- A visão de negócio que é baseada na representação das atividades que compõem os processos de negócio da organização. Este processo obedece a um conjunto de regras e objetivos que denominamos Componentes de Negócio. Essa visão também compreende um repositório de dados que contém informações dos Negócios e Serviços oferecidos pela organização, a partir do qual será desenvolvida a Ontologia do Negócio.

Destacam-se também no *framework* proposto o Portfólio Ontológico da Organização e a OAN. O Portfólio Ontológico da Organização é um repositório que compreende as diversas ontologias da área negocial e da área tecnológica da organização. Seu objetivo é estabelecer uma base de dados compartilhada que contenha todo o vocabulário utilizado nos diversos ambientes organizacionais. Diferentemente do Portfólio Ontológico, a OAN consiste em uma única ontologia construída a partir das ontologias particulares de negócio e de TI. Ela descreve, conforme definido em Guarino (1998), conceitos de forma ampla, ou seja, conceitos gerais de um domínio maior e independente, no caso o domínio organizacional.

Seu objetivo é prover, em uma única estrutura ontológica, a concatenação de termos e relacionamentos específicos das ontologias de negócio e TI.

O *framework* proposto, a exemplo do trabalho de Herborn and Wimmer (2006), é estruturado em camadas, a saber:

- Aplicação - Corresponde à infra-estrutura tecnológica destinada ao suporte e execução dos diversos processos de TI. Ela contém aplicativos, servidores, *main frames*, terminais, dispositivos de rede, balanceadores de cargas, entre outros;
- Ontologia de Domínio - Contém as visões estudadas, suas respectivas ontologias e processos, e o Portfólio Ontológico da Organização
- OAN - Corresponde à Ontologia de Alto Nível e aos Macro Processos da Organização (Meta Ontologia). Os Macro Processos representam os processos responsáveis por gerar os produtos e serviços destinados aos clientes finais da Organização, ou seja, destinam-se ao atendimento dos objetivos e finalidades básicas da instituição.

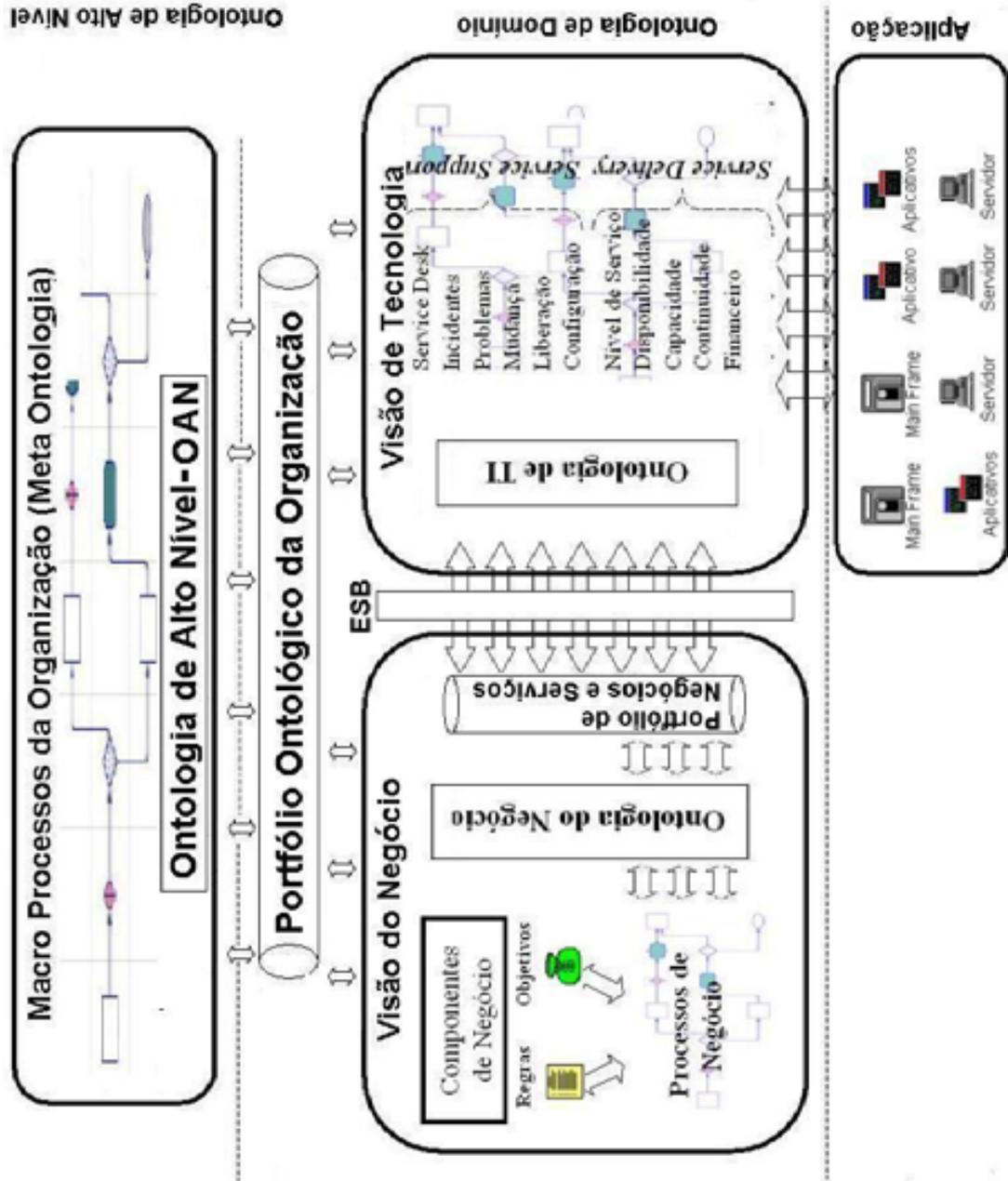
É interessante perceber que a camada Ontologia de Domínio, onde estão os domínios a serem alinhados, contém o ESB que, conforme Corporation (2006), corresponde a um barramento responsável pela ligação dos elementos da visão tecnológica: componentes e recursos de TI; e a visão de negócio: portfólio de negócios e portfólio de serviços.

O foco do estudo foi dado à visão tecnológica através de componentes de TI representados pelas disciplinas de Gestão preconizadas pelas melhores práticas do ITIL. Será utilizada também uma proposta entre TI e o negócio organizacional que possa trazer proveito para uma melhor integração entre essas áreas organizacionais.

4.2 Metodologia Utilizada

Esta seção descreverá a metodologia de desenvolvimento do *Framework* Ontológico proposto. A metodologia consiste de 7 etapas, a saber:

1. Identificação dos Domínios Ontológicos;
2. Compreensão dos Domínios Ontológicos;
3. Construção das Ontologias;
4. Validação Estrutural;
5. Construção da OAN e Adição de Semântica;
6. Validação Semântica;
7. Alinhamento Ontológico.



Ontologia de Alto Nível

Figura 4.1: *Framework* ontológico proposto.

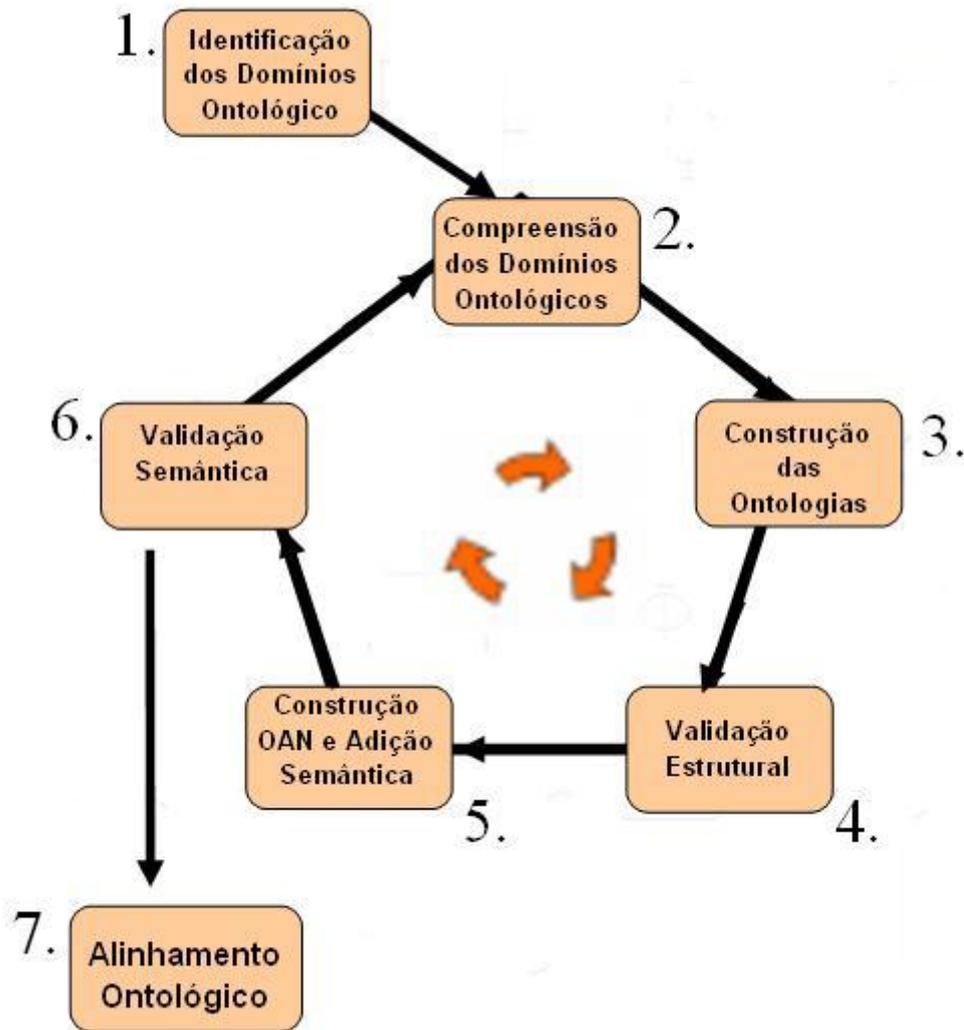


Figura 4.2: Metodologia do Trabalho.

A Figura 4.2 apresenta a metodologia e nas subseções seguintes descrevermos como foram realizadas cada uma das etapas propostas. Descreveremos também ao longo desta seção algumas características particulares da organização estudada, no caso uma instituição bancária.

4.2.1 Identificação dos Domínios Ontológicos

Como o objetivo deste trabalho é permitir uma linguagem comum às camadas de negócio e de TI da organização, os domínios estudados são os processos de TI, os quais estão baseadas nas melhores práticas do ITIL e os processos de negócio de uma organização. Diante da diversidade de processos tecnológicos e negociais de uma organização, optamos por estabelecer o escopo da seguinte maneira:

- Domínio de TI - tem como base as 11 disciplinas de Gestão preconizadas pelas melhores práticas do modelo ITIL. Como o objetivo deste trabalho é

o alinhamento dos domínios de negócio e TI e as atividades da Gerência de Nível de Serviço permeiam também a área negocial, essa disciplina recebeu um destaque especial. Afinal, segundo itSMF (2006), o Gerenciamento do Nível de Serviço eficaz aperfeiçoa o desempenho do negócio e permite a criação de uma relação eficaz entre a organização de TI e seus clientes;

- Domínio de Negócio - tem como base um serviço de negócio oferecido pela organização sobre a qual há um ANS definido. Afinal, segundo itSMF (2006), o ANS descreve os serviços em termos não técnicos, sintonizados com a percepção do cliente, e durante o período do acordo ele serve como padrão para medir e ajustar os serviços de TI. Serviços mais específicos, associados às atividades de negócios, são acordados num nível mais baixo da organização, por exemplo com o gerenciamento da unidade de negócio, com o dono do orçamento ou com o representante do cliente.

4.2.2 Compreensão dos Domínios Ontológicos

O *framework* ontológico proposto neste trabalho, representado na Figura 4.1, baseou-se no conceitos de Modelagem de Processos, uma vez que a necessidade das organizações é de visualizar os seus processos internos e externos para melhor alocar seus recursos e obter maior competitividade. Desta forma, a compreensão dos domínios ontológicos baseou-se na premissa de modelagem de processos como base para o entendimento das visões envolvidas. A partir dessas modelagens pode-se identificar quais são os termos que deverão ser selecionados para compor a ontologia.

O método utilizado para a modelagem de processo foi o *top-down*. A ferramenta utilizada para a documentação de processos foi o *WBI Workbench* da IBM. Na modelagem dos processos, estão representados: as atividades que são executadas em determinado fluxo de processo, os recursos utilizados, os relacionamentos (entradas e a saídas) entre os processos, as tomadas de decisão e os *steakholders* envolvidos. A figura 4.3 apresenta a simbologia adotada pela ferramenta *WBI Workbench*.

Na modelagem dos processos de TI foi utilizada a seguinte codificação para facilitar a organização dos inter-relacionamentos entre as disciplinas: XX.YY.ZZ onde,

XX	=	Processo
YY	=	Atividade
ZZ	=	Saída

Como exemplo podemos citar:

- SD.01.01 é Saída 01 da atividade 01 de *Service Desk*;
- LB.04.08 è Saída 08 da atividade 04 de Liberação.

A Tabela 4.1 apresenta as siglas dos Processos com as respectivas cores utilizadas nas modelagens.

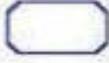
Nome do Objeto	Forma	Objeto
Conector	Seta	
Tarefa	Retângulo Arredondado	
Entrada/Saída	Círculo	
Processo	Quadrado	
Entidade Externa	Elipse	
Processo Externo	Elipse dentro de Elipse	
Decisão	Losango	
Múltipla Escolha	Pequeno Círculo	
Parada	Sinal de Pare	

Figura 4.3: Principais objetos utilizados na modelagem de processos no *WBI Workbench*.

Tabela 4.1: Codificação da modelagem dos processos de TI.

Processos	Sigla	Cor
Service Desk	SD	Azul Marinho
Configuração	CF	Amarela
Mudança	MD	Roxa
Problemas	PR	Vermelho Escuro
Incidentes	IN	Vermelho Claro
Liberação	LB	Verde Escuro
Capacidade	CP	Laranja
Nível de Serviço	NS	Marrom
Financeira	FC	Branca
Disponibilidade	DS	Rosa
Continuidade	CN	Azul
Gestão de Projetos	GP	Rosa
Processos Externos	EE	Cinza

A Figura 4.4 exemplifica a modelagem de alguns dos processos envolvidos na Disciplina de Gerenciamento de Incidentes do modelo ITIL, bem como os seus relacionamentos com as demais disciplinas do modelo e agentes externos, tais como clientes e usuários. Nela podemos observar que o processo de Identificar/Registrar Incidentes da Gestão de Incidentes (IN 01) recebe entre outras entradas, Requisições de Serviços (EE 01.01) provenientes de usuários, clientes e analistas (EE.01) e Informações sobre ICs (CF 08.03) provenientes do processo Gerar Informações (CF 08) da Gestão de Configuração. Também nesse simples exemplo podemos perceber uma grande variedade de conceitos relacionados com a área de TI, tais como: ICs, CMDB, Detalhes da ocorrência, Requisição de Serviço, Usuário, Clientes, Equipe de Suporte, Planos de Contingência testados, *Service Desk*, Aceite, Homologação, Monitoração, entre outros.

É interessante notar que os relacionamentos expressos na Figura 4.4 estão em conformidade com as relações da Disciplina de Incidentes com outros processos conforme estipulado em itSMF (2006), a saber:

- Relação entre Gerenciamento da Configuração e Incidentes - A base de dados da Gerência de Configuração - CMDB define as relações entre recursos, serviços, usuários e Níveis de Serviço. Por exemplo, o Gerenciamento da Configuração mostra quem é o responsável por um componente da infraestrutura, e assim os incidentes relacionados a esse componente específico podem ser direcionados mais eficientemente. Além disso ele ajuda a decidir quanto a soluções de contorno adequadas, como por exemplo desviar fila de impressão e transferir os usuários para outro servidor. Durante o registro do incidente os detalhes da configuração são ligados ao registro do incidente, para fornecer melhores informações sobre o erro. Onde for necessário pode-se atualizar na Base de Dados do Gerenciamento da Configuração o *status* dos componentes pertinentes;

- Relação entre Gerenciamento de Liberação e Incidentes - Os incidentes podem ser resolvidos implementando-se liberações, como por exemplo, instalando-se um monitor no ambiente de produção. Todavia, durante o processo de Aceite e Homologação podem ser detectados incidentes ou falhas que devem ser registradas pela disciplina de Incidentes. O Gerenciamento de Incidentes, por sua vez, fornecerá informações sobre esses incidentes para o Gerenciamento de Mudanças, que conseqüentemente determinará novas liberações;
- Relação entre *Service Desk* e Incidentes - A Central de Serviços freqüentemente é responsável por contatos com fornecedores de manutenção, o que inclui o reparo e substituição de impressoras, estações de trabalho e, em alguns casos, equipamento de telecomunicação. Esse tipo de manutenção pode abranger o tratamento dos incidentes assim como mudanças e requisições de serviço. A Central de Serviços também pode ter acesso a ferramentas, eventualmente utilizáveis para a avaliação do impacto de falhas que afetam recursos e equipamentos essenciais, como roteadores, servidores, e portais, sistemas de missão crítica, aplicações e banco de dados. Muitas vezes essas ferramentas podem detectar falhas e informar o Gerenciamento de Incidentes automaticamente quando uma falha ocorreu ou está ameaçando ocorrer.

É importante notar que esta etapa de Cormpreensão dos Domínios Ontológicos baseada na modelagem de processos foi utilizada especialmente na compreensão do domínio de TI da organização. Afinal, por questões de sigilo corporativo, as regras, estratégias e recursos que norteiam o domínio de negócio, bem como a modelagem de processos correspondentes, não puderam ser elucidadas no presente estudo. Desse modo, para uma maior compreensão da visão negocial, nos baseamos nas informações referentes a produtos e serviços encontrados na *intranet* da organização e em acordos de níveis de serviço previamente formalizados entre as camadas de TI e negócio da empresa.

4.2.3 Construção das Ontologias

A principal preocupação na construção das ontologias de domínio de Negócio e de TI foi seguir a metodologia de construção proposta em Noy and McGuinness (2001), a qual está representada na Figura 4.5.

A seguir, descrevemos os procedimentos realizados para a execução de cada uma das etapas durante o uso do método 101:

- Passo 1 - Determinar o domínio e escopo da ontologia

Durante este trabalho realizamos estudos de alinhamento ontológico entre os domínios de negócio e TI de uma organização financeira. No domínio tecnológico nós temos como escopo os conceitos abrangidos pelas disciplinas de gerenciamento propostas pelo modelo ITIL, mais especificamente a disciplina de Nível de Serviço conforme apresentado na seção 2.3. Já o domínio negocial compreende conceitos relacionados a estratégias, regras, objetivos e políticas organizacionais, mais especificamente o Gerenciador Financeiro da

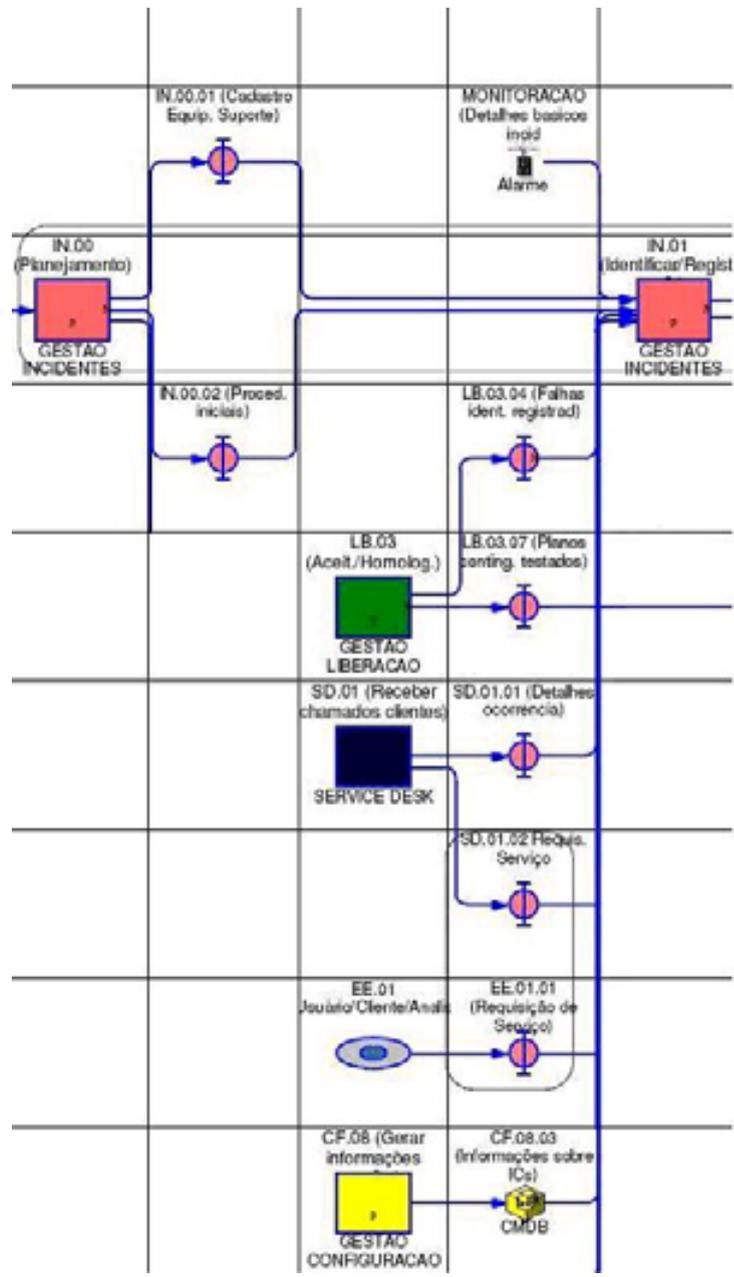


Figura 4.4: Modelagem dos processos de planejamento e registro de incidentes.

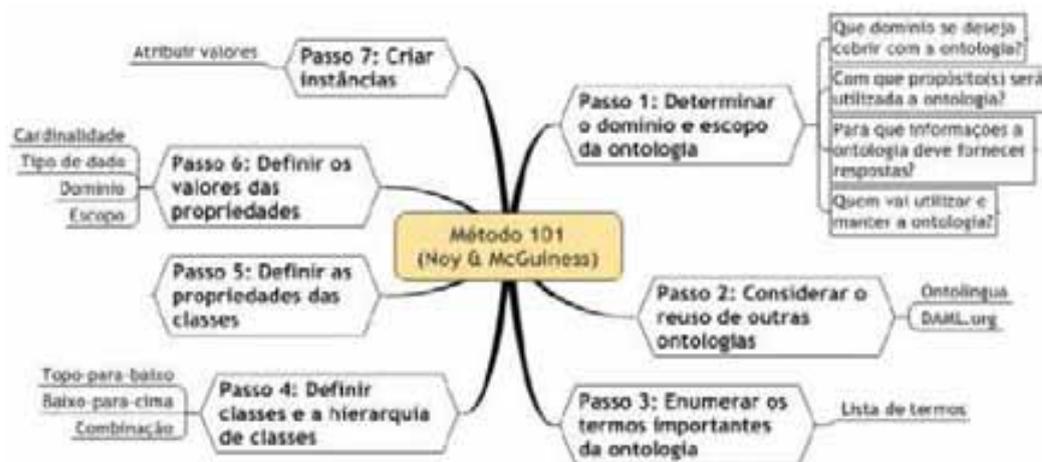


Figura 4.5: Representação da metodologia utilizada (Noy and McGuinness, 2001).

instituição. As ontologias serão utilizadas pela organização com o propósito de ter acesso a um vocabulário compartilhado entre as diversas camadas que a compõe, sejam elas estratégicas, táticas ou operacionais. Tal vocabulário proverá uma padronização entre os diversos conceitos utilizados pela organização e tornará os processos internos mais ágeis e eficazes. Além disso, contribuirá para um atendimento mais disciplinado dos clientes e usuários da instituição;

- Passo 2 - Considerar o reuso de outras ontologias

Nesse trabalho considerou-se o reuso de ontologia existentes, uma vez que a ontologia da visão de TI, inspirada no modelo ITIL partiu da ontologia da disciplina de Liberação (versionamento) proposta em (Porto and Bruggemann, 2006). A partir da ontologia de Liberação, expandiu-se o escopo para abranger as demais disciplinas do modelo ITIL. Devido à indisponibilidade de uma ontologia de Negócio pré-existente não se considerou o reuso para esta ontologia;

- Passo 3 - Enumerar os termos importantes da ontologia

Nessa fase foram levantados os principais termos abordados nas duas ontologias. Esse levantamento levou em consideração os conceitos que foram identificados na modelagem de processos realizada. Assim, na ontologia de TI foram enumerados termos como: Gerenciamento de Incidentes, ICs, CMDB, *hardware*, Mudança, Plano de Liberação, RDMassociada, Requisitos de Liberação, ANS, Número de bem, Número do Chamado, Criptografia, Data Final da Mudança, Erro Conhecido, Impacto, Indicadores de Desempenho, Gerenciado pelo, Disponibilidade Acordada, Duração do Acordo, etc. Já na ontologia de Negócio, foram enumerados termos como: Orçamento, Receitas, Órgãos Reguladores, Lucros, Investimentos, Serviço de Negócio, Nome do Produto, Pagamento de Salários, Políticas, Retorno

Operacional de Investimento, Licitações, Horário de Disponibilidade Acordado, Grau de Satisfação, Políticas, entre outros;

- Passo 4 - Definir as classes e suas hierarquias

Dos termos enumerados na etapa anterior foram selecionados apenas aqueles que representam classes, ou seja, foram desconsiderados os atributos e seus relacionamentos. Realizada a seleção, as classes foram adicionadas à ontologia de maneira hierárquica, segundo a abordagem *top dow*, conforme podemos observar na Figura 4.6;

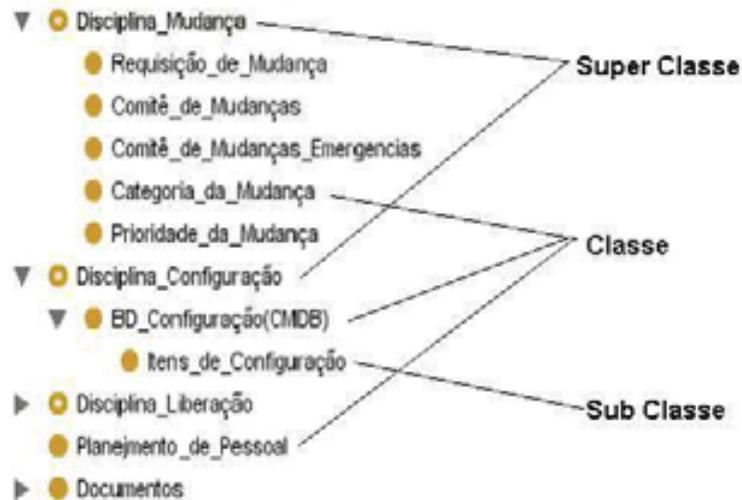


Figura 4.6: Representação de hierarquia de classes da ontologia de TI definida.

- Passo 5 - Definir as propriedades das classes (slots)

Nessa etapa, foram definidas as propriedades de cada uma das classes enumeradas na etapa anterior denominadas (*slots*). Assim, conforme podemos observar na Figura 4.7, a classe Disciplina-Mudança tem como atributos: Categorizada, Descrição-da-Mudança, Documentação-Impacto, Impacto-Mudança, Possui-gerente-mudança, Prioridade-Mudança, Vinculada;

- Passo 6 - Definir os valores das propriedades

Nessa fase foram atribuídas as facetas dos *slots* ou características de cada atributo definido no passo anterior. Entre esses atributos podemos citar o nome, o tipo de valor do atributo, o significado (documentação) e o valor *default*. A Figura 4.9 representa o *slot* Já-ocorreu-Incidente, o qual indica se já ocorreu ou não algum incidente com um determinado item de configuração, seja ele *hardware* ou *software*. Podemos observar que esse atributo é do tipo booleano, possui cardinalidade mínima e máxima igual a um e tem valor *default* igual a *false*;

- Passo 7 - Criar Instâncias



Figura 4.7: Ilustração de *slots* da ontologia de TI definida.

Na última etapa do método utilizado foram criadas instâncias para cada um dos conceitos envolvidos, ou seja, objetos reais que compõem as ontologias de Negócio e de TI. Por exemplo, para a classe Itens-de-Configuração, foram criadas 25 instâncias conforme ilustra a Figura 4.8. Nela podemos observar que a instância de número 30 trata-se de uma impressora.

Uma outra preocupação que norteou a construção das ontologias de Negócio e de TI foi ater-se aos princípios citados por Gruber (2005) e descritos no capítulo 2, dentre os quais podemos destacar:

- clareza;
- legibilidade;
- mínima codificação;
- mínimo compromisso ontológico.

Em relação à clareza, tentou-se definir apenas o que se presume ser útil para o entendimento da ontologia e para possíveis aplicações.

O quesito legibilidade pode ser verificado através do fato de que as terminologias dos conceitos definidos são os próprios termos definidos nas documentações oficiais utilizadas com relação à TI nas fontes originais da própria organização com relação ao negócio. Dessa forma, ao analisar as ontologias pode-se ter um mapeamento direto e intuitivo para os conceitos reais, sem a necessidade de uma interpretação ou associação mais complexa.

A mínima codificação pode ser vista em entidades definidas como Disciplina de Configuração e Pessoa, por exemplo. A intenção foi especificar conceitos genéricos quando os mesmos ainda não tinham uma função muito clara dentro da aplicação real, permitindo uma extensibilidade por parte dos usuários da ontologia. Esse princípio é limitado pela clareza. Entidades como ICs e Gerente de Negócio foram definidos da forma mais específica possível, pois são entidades que definem a estrutura principal dos conceitos envolvidos e definidos nas ontologias de TI e de Negócio.

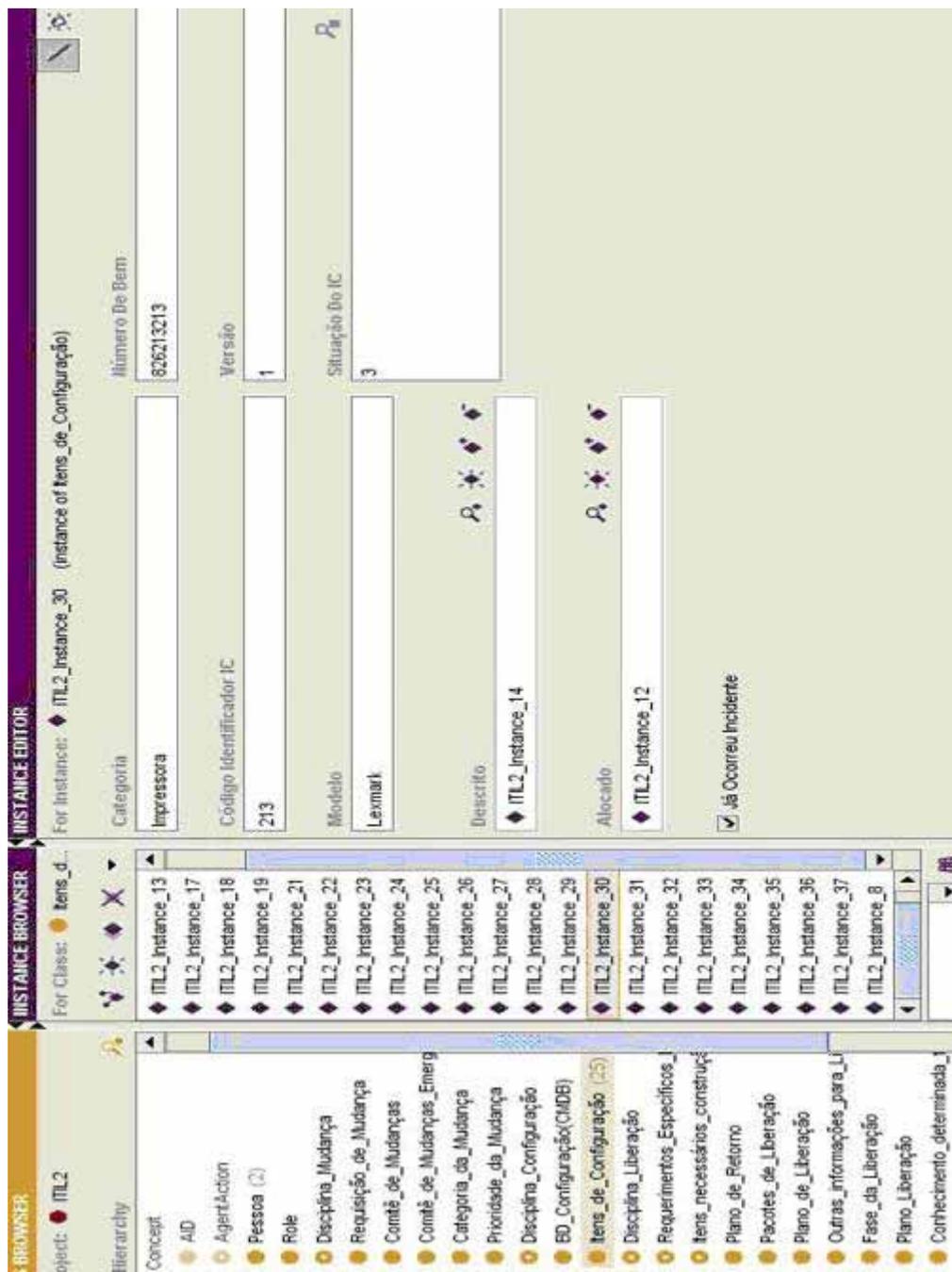


Figura 4.8: Exemplo de instância da ontologia de TI definida.

Ao se tratar do princípio definido como mínimo compromisso ontológico, objetivou-se gerar uma ontologia que permita que as partes interessadas fiquem livres para especializar e instanciar os conceitos sempre que necessário. Alguns conceitos, inclusive, nem possuem atributos, tais como Orçamento e Desastre. Dessa forma os usuários da ontologia têm a possibilidade de definir os atributos que mais interessam para a sua área de interesse. Por exemplo, na entidade Pessoa, não foram definidos atributos como: filiação, endereço, data de nascimento. Apesar de serem atributos válidos para a entidade Pessoa, pode ser que não exista interesse em compartilhá-los entre as diversas áreas da organização.



Figura 4.9: Exemplo de *slot* da ontologia de TI definida.

Os conceitos contidos na ontologia da TI, os quais foram baseados no modelo ITIL foram obtidas de fontes, tais como Corporation (2000) e (itSMF, 2006). Já os conceitos utilizados na ontologia de Negócio, foram obtidos de fontes como a Intranet corporativa, a base de conhecimentos e a página na Internet da organização analisada estudo de caso deste trabalho. As ontologias de TI e de Negócio podem ser verificadas respectivamente nos anexos A e B.

4.2.4 Validação Estrutural

Uma vez construídas as ontologias de Negócio e de TI, as mesmas passaram por um processo de validação. Nessa validação foram verificados aspectos relacionados à estrutura e à consistência e classificação das ontologias.

A validação estrutural das ontologias foi realizada através de consultas (*queries*) realizadas entre os termos da ontologia. Essa validação partiu da premissa de que se os termos das ontologias estivessem adequadamente interconectados, as consultas realizadas sobre os termos retornariam instâncias válidas segundo as especificações da consulta. A Figura 4.10 exemplifica a validação estrutural realizada para a ontologia de TI. Nessa figura está representada uma consulta que tem como objetivo identificar quais são os itens de configuração localizados em um determinado ambiente cuja categoria é impressora, o modelo é *Lexmark* e

que já apresentou incidente. Como resultado da consulta, foram encontrados seis instâncias que atendem essas condições conforme *search results* da mesma figura.

Neste trabalho, também realizamos a validação estrutural da OAN, ou seja, verificamos se a ontologia composta por elementos das duas ontologias de domínio Negócio e TI mantinham a interoperabilidade e a coerência. Para a validação da OAN, cuja construção detalhada será descrita na próxima seção, foram realizados os mesmos procedimentos descritos para a validação das ontologias de domínio. Diferentemente das consultas realizadas durante a fase de validação das ontologias de domínio, as consultas realizadas sobre a OAN permitem verificar a interoperabilidade de termos originários de duas visões diferentes, ou seja, de dois domínios com suas respectivas particularidades, mas que precisam estar alinhados para o sucesso de uma organização. A Figura 4.11 apresenta uma consulta envolvendo termos que pertenciam originariamente às ontologias de negócio e de TI. Nela podemos observar que os serviços de negócio: Serviço Governo e Serviço PF são suportados pelo serviço de TI: Impressões em Geral.

Muitas outras consultas foram realizadas envolvendo vocabulários da área de negócio e da área de TI, tais como: incidentes que afetam determinado serviço de negócio, itens de configuração que suportam determinado serviço de negócio, itens de configuração e custo associado, plano de continuidade de terminado negócio, ANS assinado por Gerente de Negócio, entre outros.

A validação estrutural das ontologias, através de consultas no Protégé, apresentou-se como uma ferramenta de grande potencial para realizar o gerenciamento dos recursos de TI. Afinal, neste estudo, foram realizadas inúmeras consultas que são de grande utilidade dos gestores de TI nas organizações tais como: modelos de ICs localizados por setor, ICs que já apresentaram incidentes, liberações associadas a mudanças, estados dos itens de configuração, localização dos chamados por situação, número de chamados por atendentes, problemas associados a incidentes, funcionários responsáveis por mudanças e liberações, nível de suporte de determinada ocorrência, serviços de TI afetados por determinado incidente, entre outros.

Todavia, o foco do nosso trabalho é estabelecer um alinhamento ontológico entre as camadas de negócio e TI. Assim, apesar da grande variedade de consultas que podem ser criadas envolvendo os aspectos estruturais dos domínios tecnológicos e negociais, dificilmente serão atingidas todas as possibilidades possíveis de pesquisas envolvendo os termos dessas ontologias. Dessa forma, tornou-se necessária uma validação das ontologias baseada em outras abordagens como a verificação semântica que passaremos a apresentar.

4.2.5 Construção OAN e Adição de Semântica

Esse trabalho é baseado na idéia de que uma ontologia de alto nível de uma organização compreende as ontologias de domínio de TI e Negócio. Duas ontologias que, apesar de apresentarem visões diferentes, devem apresentar coerência semântica para o cumprimento dos objetivos estratégicos da organização, garantindo, dessa forma, uma maior competitividade no mercado.

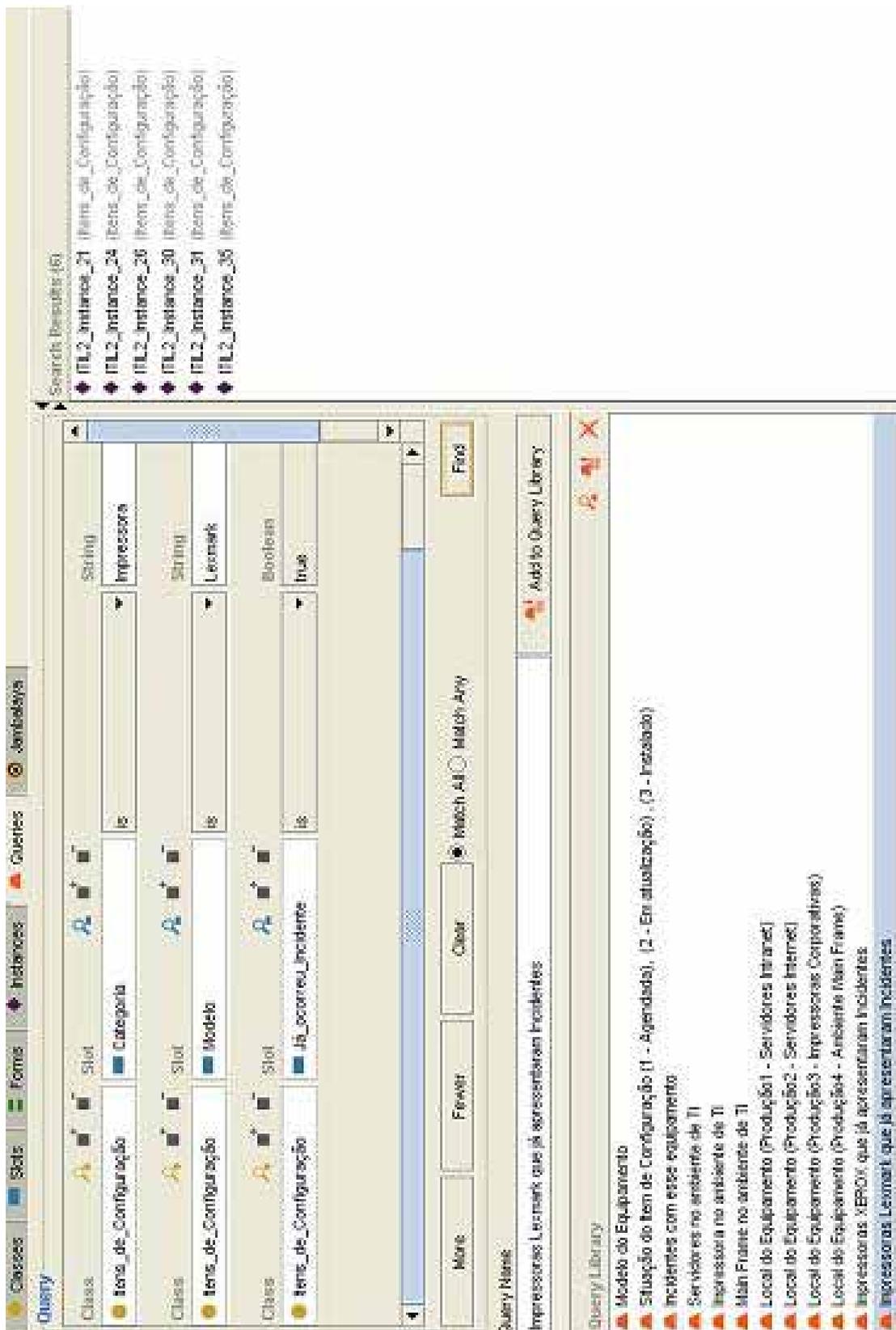


Figura 4.10: Exemplos de consultas realizadas na ontologia de TI.

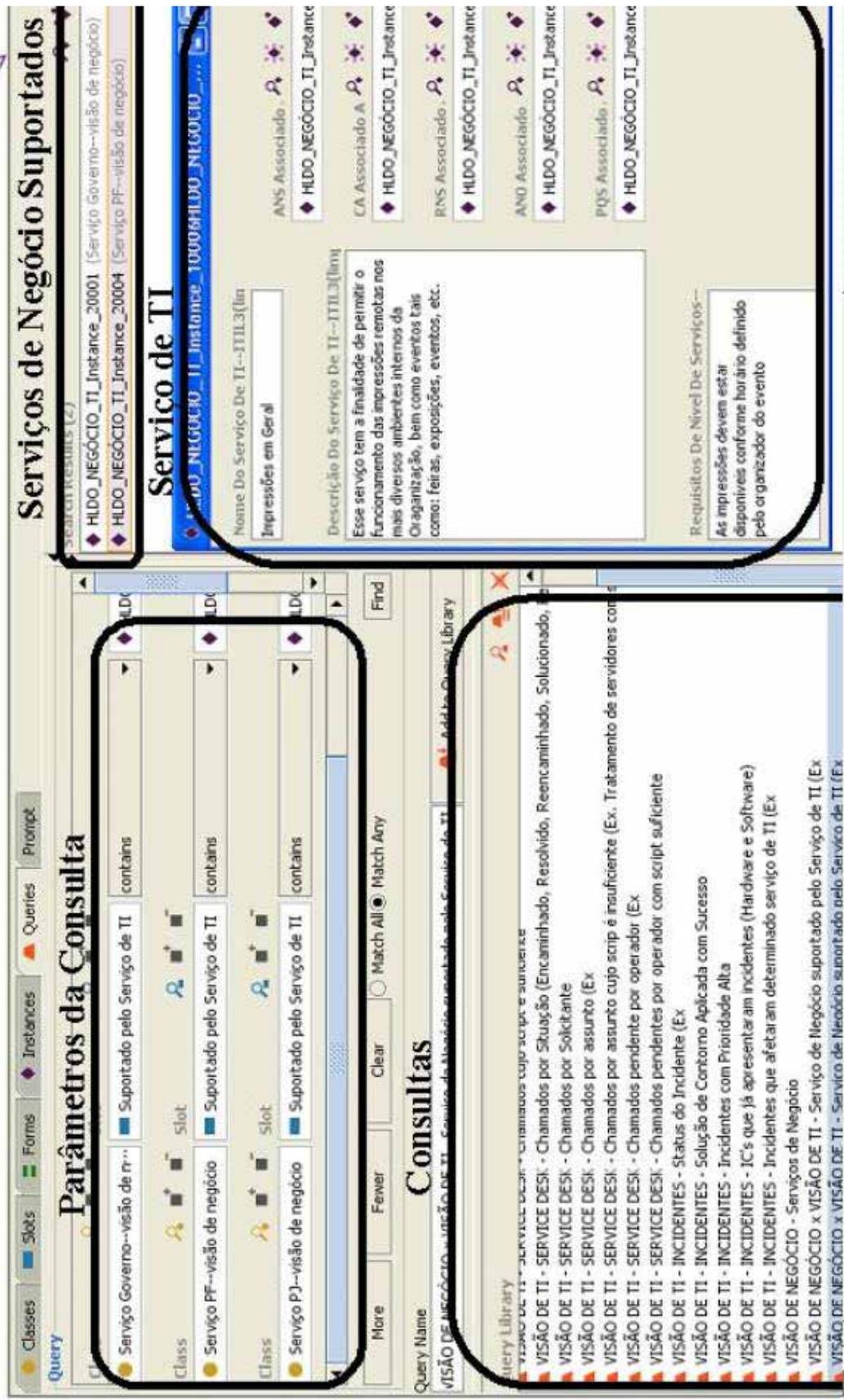


Figura 4.11: Validação da ontologia OAN dos domínios de Negócio e TI.

Construção OAN

Para a construção da OAN ou HLDO foram utilizados os seguintes passos do método proposto em Herborn and Wimmer (2006) (vide Figura 3.2):

- 1 - Identificar domínios ontológicos de interesse;
- 2 - Selecionar subconjuntos de vocabulário do domínio de interesse;
- 3 - *Merging* dos subconjuntos de vocabulários dos domínios;
- 4 - Fazer o mapeamento dos subconjuntos *mergeados* na HLDO;
- 5 - Adicionar semântica à HLDO.

Os dois primeiros passos da metodologia já foram realizados pela método proposto em Noy and McGuinness (2001) e utilizado neste trabalho conforme apresentado nas subseções anteriores.

O passo 3 consiste em construir uma ontologia combinando todos subconjuntos de vocábulos relevantes de ambos os domínios, sejam eles pertencentes ao domínio de negócio ou ao domínio de TI. É interessante notar que o *merge* inclui também a eliminação de duplicidades de entidades redundantes. Assim, classes comuns às duas ontologias, tais como Pessoa de TI e Pessoa de Negócio, foram integradas em apenas uma classe Pessoa na OAN conforme ilustrado na Figura 4.12.

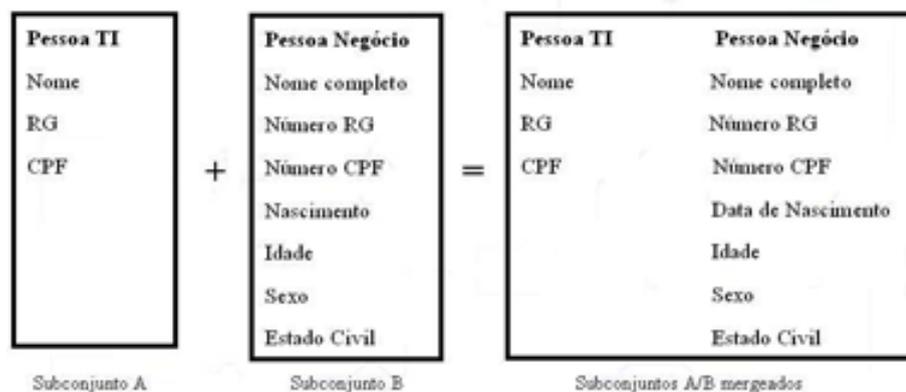


Figura 4.12: *Merging* de subconjuntos das ontologias A e B.

O passo 4 tem a finalidade de validar a nova a ontologia em relação às ontologias de domínios de Negócio e de TI pré-existentes após termos definido a OAN através do *merging* de diferentes subconjuntos de termos. Nessa etapa os termos e suas relações nos diferentes domínios precisam ser harmonizados.

Para os passos 3 e 4 foi utilizado o *plugin* Prompt (Noy and Musen, 2001). As figuras 4.13 e 4.14 demonstram a realização das etapa 3 e 4.

Na Figura 4.13, verificamos que o passo 3 executado pelo *plugin* Prompt é realizado com a intervenção do usuário, o qual através de operações *merge* e *copy* sugeridas pela ferramenta inicia a construção da OAN.

Ontologias de Domínio

Razões para sugestão selecionada

Name	Arg1	Arg2	Params
copy	● Gerenciamento do Nível de Serviço ITIL3(limpo)		
copy	● Fornecedores visão de negócio		params = {subs}
copy	● Cliente Externo (Usuário_Final) ITIL3(limpo)		params = {subs}
copy	● Conselho Deliberativo (DIRETORIA) visão		params = {subs}
merge	● Clientes visão de negócio ● Cliente ITIL3(limpo)		
copy	● Regras Internas visão de negócio		
copy	● Governo visão de negócio		
copy	● Pessoa Física visão de negócio		
copy	● Pessoa Jurídica visão de negócio		
copy	● Programa de Aperfeiçoamento de Serviço		
copy	● Provedor ITIL3(limpo)		

Reason for selected suggestion

- Gerenciamento do Nível de Serviço ITIL3(limpo) is a subclass of ● Concept--ITIL3(limpo)
- Gerenciamento do Nível de Serviço ITIL3(limpo) is a superclass of ● Requisitos de Nível de Serviços (RNS)--ITIL3(limpo)
- Gerenciamento do Nível de Serviço ITIL3(limpo) is a superclass of ● Catálogo de Serviços--ITIL3(limpo)
- Gerenciamento do Nível de Serviço ITIL3(limpo) is a superclass of ● ACORDO DE NÍVEL DE SERVIÇO
- Gerenciamento do Nível de Serviço ITIL3(limpo) is a superclass of ● Acordo de Nível Operacional (ANO)--ITIL3(limpo)
- Gerenciamento do Nível de Serviço ITIL3(limpo) is a superclass of ● Contrato de Apoio (CA)--ITIL3(limpo)
- Gerenciamento do Nível de Serviço ITIL3(limpo) is a superclass of ● Programa de Qualidade de Serviço (POS)--ITIL3(limpo)

Figura 4.13: *Merging* de subconjuntos das ontologias de domínio.

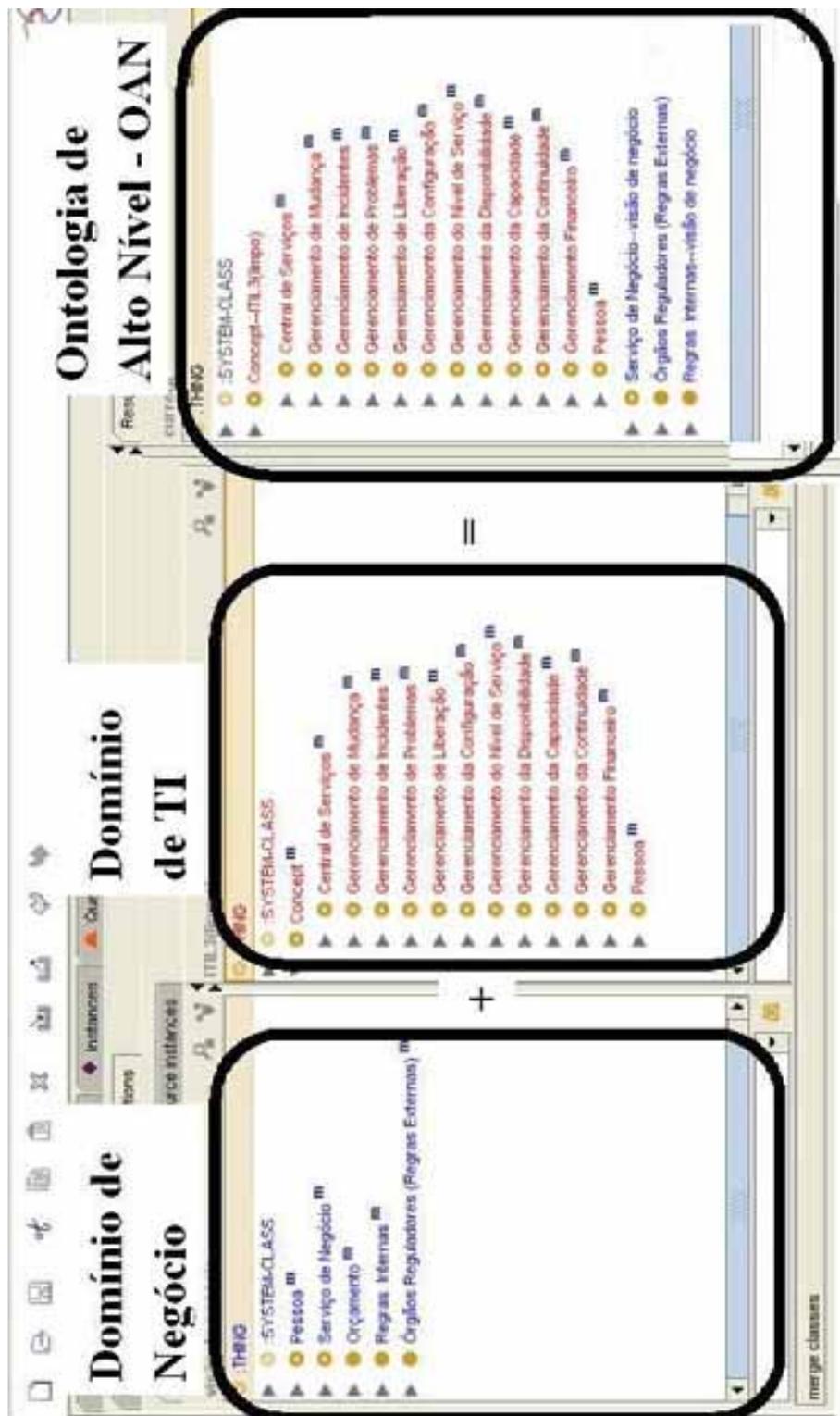


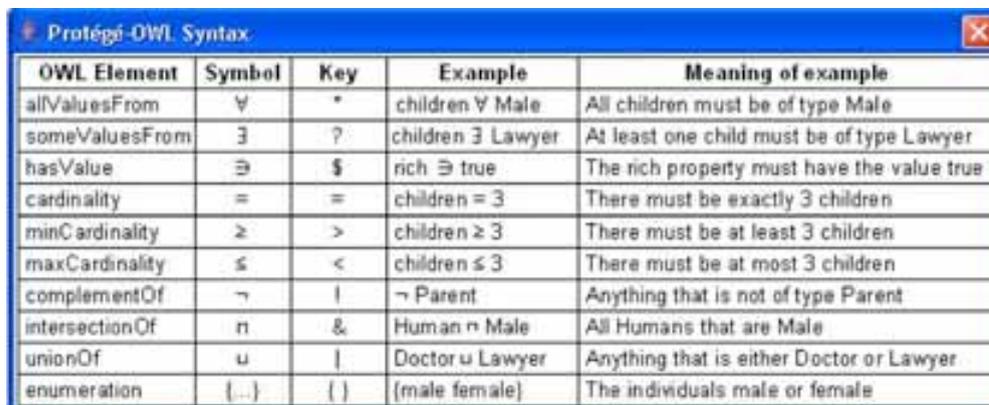
Figura 4.14: Mapeamento dos subconjuntos fundidos na OAN.

Na Figura 4.14, percebemos que uma vez concluído o *merge* entre os termos dos domínios de Negócio e de TI, eles são *setados* com o índice 'm' (ontologias de domínio à esquerda). Como resultado dessa operação, obtemos a OAN (à direita do sinal de =), a qual é constituída do mapeamento dos subconjuntos das ontologias de domínio.

O Passo 5, adicionar semântica a HLDO, consiste em adicionar significado aos termos dessa ontologia. É interessante notar que neste trabalho, diferentemente da metodologia proposta em Herborn and Wimmer (2006), também foi adicionada semântica às ontologias de domínio.

Adição de Semântica

As razões para a adição de semântica são muitas. Primeiro de tudo, temos que essa semântica possibilitará a utilização de axiomas envolvendo conjuntos de dados independentes. Para adicionar semântica às ontologias, elas tiveram de ser convertidas para o formato OWL (vide Seção 2.4.5). Na Figura 4.15 podemos verificar os principais símbolos da sintaxe OWL na ferramenta Protégé.



OWL Element	Symbol	Key	Example	Meaning of example
allValuesFrom	\forall	*	children \forall Male	All children must be of type Male
someValuesFrom	\exists	?	children \exists Lawyer	At least one child must be of type Lawyer
hasValue	\exists	\$	rich \exists true	The rich property must have the value true
cardinality	=	=	children = 3	There must be exactly 3 children
minCardinality	\geq	>	children \geq 3	There must be at least 3 children
maxCardinality	\leq	<	children \leq 3	There must be at most 3 children
complementOf	\neg		\neg Parent	Anything that is not of type Parent
intersectionOf	\cap	&	Human \cap Male	All Humans that are Male
unionOf	\cup		Doctor \cup Lawyer	Anything that is either Doctor or Lawyer
enumeration	{...}	{ }	{male female}	The individuals male or female

Figura 4.15: Sintaxe no Protégé-OWL.

Além disso, a semântica adicionada irá suportar a construção dos mecanismos de inferência automatizados, os quais permitirão a verificação das condições necessárias e suficientes para determinar a consistência semântica de um determinado termo, vide Figura 4.16.

A Figura 4.16 demonstra que para um determinado Item de Configuração - IC ser consistente é necessário e suficiente que ele esteja cadastrado em um Banco de Dados de Configuração - CMDB e que esteja associado a um determinado serviço de TI.

Finalmente, mecanismos de busca avançados podem ser aplicados para validação de resultados de inclusão de novos termos à ontologia conforme apresentando na Figura 4.17.

Através da Figura 4.17, percebemos que caso seja adicionado um novo termo chamado Cadeira, e o classificarmos como IC, perceberemos que essa classificação não será consistente, pois esse novo termo não apresenta a condição requerida aos ICs de estar associado a um determinado serviço de TI.

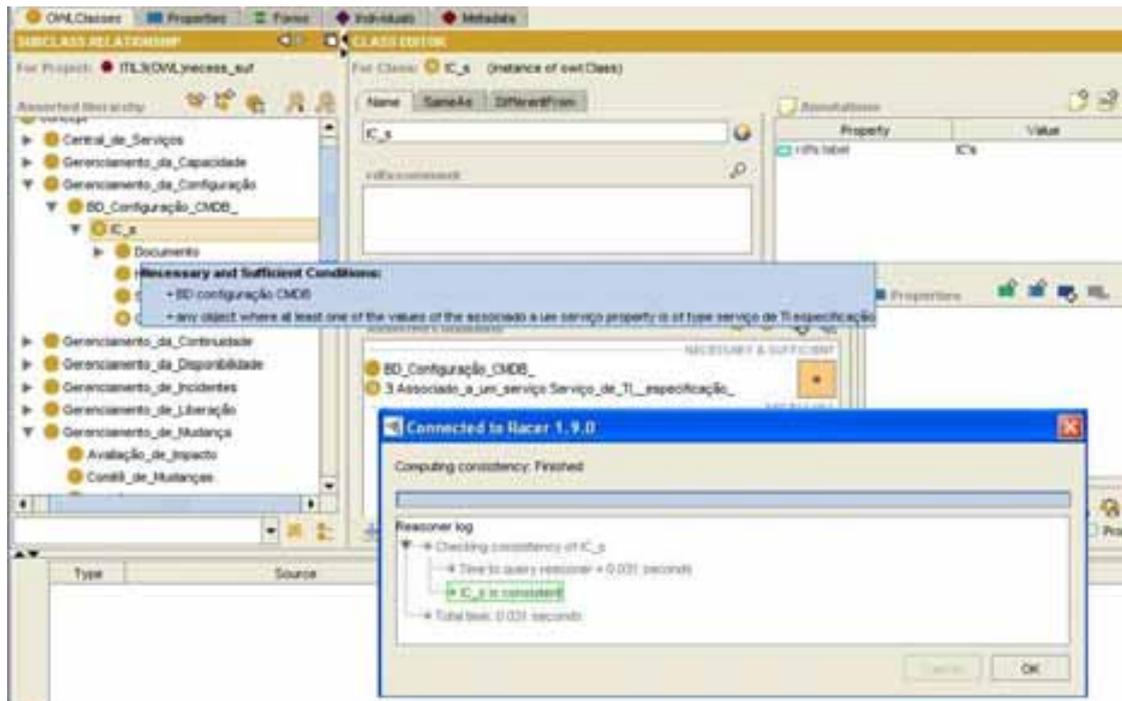


Figura 4.16: Adicionando semântica para utilização de mecanismos de inferências automatizados.

No estudo de caso apresentado em Herborn and Wimmer (2006), o objetivo era a construção de uma ontologia única para os registros de negócio entre os países pertencentes à comunidade européia, ou seja, os domínios envolvidos apresentavam semânticas similares. Diferentemente dessa abordagem, nossa proposta foca na integração de domínios reconhecidamente contrastantes e muitas vezes conflituosos. Assim, verificou-se que o *merge* utilizado no trabalho citado apresenta fragilidades quando queremos alinhar domínios tão distintos como a área comercial e a área tecnológica de uma organização. Tornou-se necessário então, utilizarmos uma abordagem focada no alinhamento ontológico.

4.2.6 Validação Semântica

Para o processo de validação da consistência e classificação das ontologias, elas foram convertidas para o formato OWL DL, que segundo Brockmans et al. (2006) trata-se de uma linguagem que expressa um formalismo com semântica bem definida, sobre a qual pode ser realizado raciocínio automatizado.

Segundo Stevens et al. (2002), o raciocinador realiza os seguintes serviços:

- analisa cada classe para verificar a consistência com a ontologia. A consistência pode ser verificada pela definição de classes disjuntas. Se definimos duas classes A e B como sendo disjuntas e colocamos a classe C como subclasse de A e de B, o raciocinador determinará uma inconsistência;
- verifica a classificação da hierarquia, levando em conta todas as restrições



Figura 4.17: Inclusão de novos termos à ontologia.

dadas na ontologia. Isto pode revelar novas subordinações entre as classes bem como erros de modelagem, que precisam ser corrigidos. Além disso, em alguns raciocinadores (e.g. RacerPro), dada a descrição de um indivíduo, é possível computar as classes das quais esse indivíduo é uma instância.

A verificação da consistência e classificação das ontologias geradas foi realizada através do raciocinador com licença educacional *RacerPro 1.9.0* (<http://www.racer-systems.com>). Para utilizar os recursos de raciocínio no Protege, executou-se o raciocinador com o ambiente ontológico inicializado no Protégé. Após a inicialização do aplicativo, bastou configurar os testes a serem realizados e clicar na opção *Check consistency*, no menu OWL, para checar a consistência da ontologia. Para verificar a classificação, utilizamos a opção *Classify taxonomy*, do mesmo menu. Os testes podem ser observados na Figura 4.18 e o menu é apresentado na Figura 4.19.

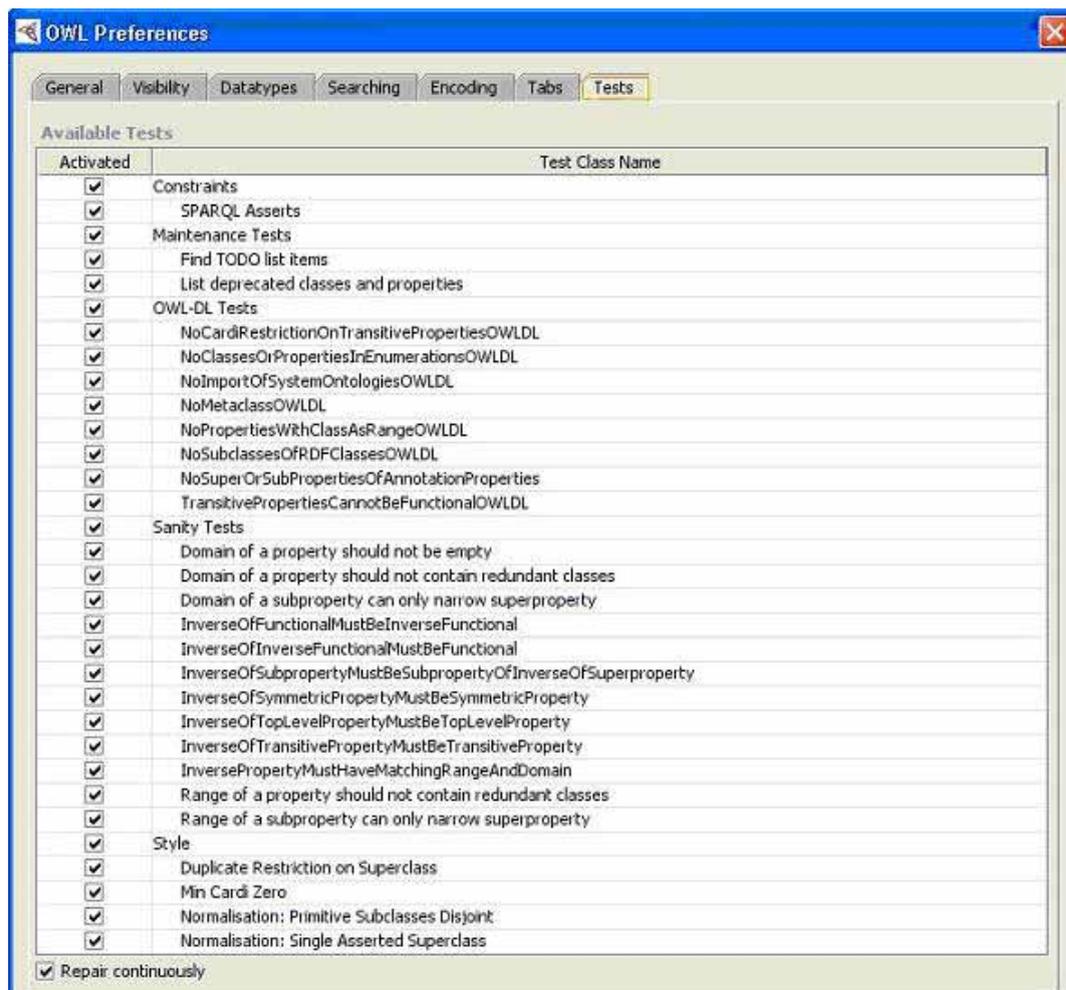


Figura 4.18: Testes realizados pelo raciocinador *Racer Pro*.

4.2.7 Alinhamento das ontologias de TI e de Negócio

Segundo Euzenat (2004) o problema do alinhamento de ontologias pode ser descrito na seguinte expressão: Dadas duas ontologias, cada qual com um conjunto de entidades discretas (as quais podem ser classes, propriedades, regras ou predicados) encontrar os relacionamentos existentes entre essas entidades. Para proceder o alinhamento ontológico entre os domínios de TI e de Negócio foi utilizado o *Framework for Ontology Alignment and Mapping* - FOAM proposto em (Ashpole et al., 2005). Essa abordagem foi utilizada por possuir uma versão disponível para *download* em: <http://www.aifb.uni-karlsruhe.de/WBS/meh/foam>. Além da facilidade de uso, esse *framework* disponibiliza a utilização de diferentes algoritmos para o alinhamento entre ontologias. O FOAM, os algoritmos adotados, bem como as métricas de alinhamento utilizadas neste trabalho podem ser encontradas na seção 2.6.

Capítulo 5

Estudo de Caso e Análise dos Resultados

Neste capítulo faremos uma breve introdução sobre a organização estudada na qual utilizamos o *framework* ontológico proposto. Na sequência discutiremos os resultados obtidos.

Estudo de Caso

No estudo de caso apresentado em Herborn and Wimmer (2006), o objetivo era a construção de uma ontologia única para os registros de negócio entre os países pertencentes à comunidade européia, ou seja, os domínios envolvidos apresentavam fundos semânticos similares, conforme indica a Figura 3.1. Diferentemente dessa abordagem, nossa proposta foca a integração de domínios com semânticas distintas e muitas vezes conflituosas que são a camada de TI e a camada de negócio da organização, no caso ilustrado neste trabalho por uma instituição bancária.

A Figura 5.1 contém um organograma da presidência da instituição bancária estudada (disponível em www.bb.com.br). Nela percebemos que o maior nível hierárquico da organização é composto pela presidência e por sete vice-presidências. Das quais, seis estão relacionadas às atividades negociais da organização e apenas a Vice-Presidência de Tecnologia e Logística está relacionada às atividades tecnológicas.

As vice-presidências que correspondem às camadas de Negócio subdividem-se em várias diretorias: Diretoria Comercial - DICOM, Diretoria de Agronegócios, Diretoria de Varejo, Diretoria de Comércio Exterior, entre outras. Por outro lado, a Vice-Presidência de Tecnologia subdivide-se na Diretoria de Tecnologia - DITEC, responsável pelas soluções de infra-estrutura tecnológica, e Diretoria de Desenvolvimento de Aplicativos - DAA, responsável pelo desenvolvimento de soluções de *software* para a organização. A Figura 5.2 apresenta as diretorias da organização que este estudo se propõe a integrar através de um processo que permeia a DICOM e a DITEC (Figura 5.3).

Trata-se do processo que disponibiliza um serviço, o qual está definido no ANS assinado entre essas duas diretorias conforme demonstra a Figura 5.3. É interessante notar que essas duas diretorias apresentam respectivamente a visão



Figura 5.1: Organograma da diretoria da instituição bancária.

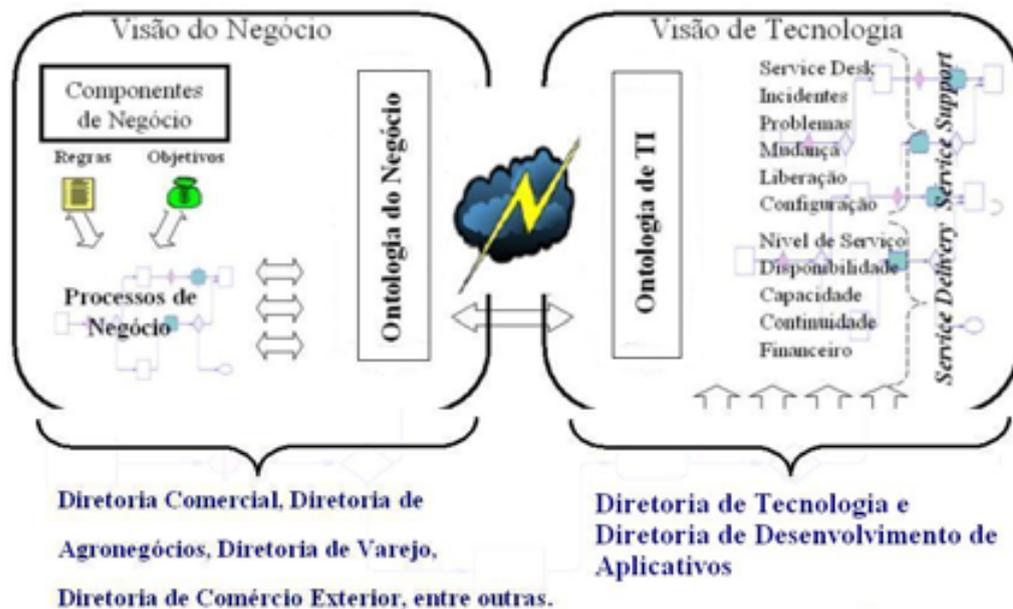


Figura 5.2: Objeto de integração do trabalho.

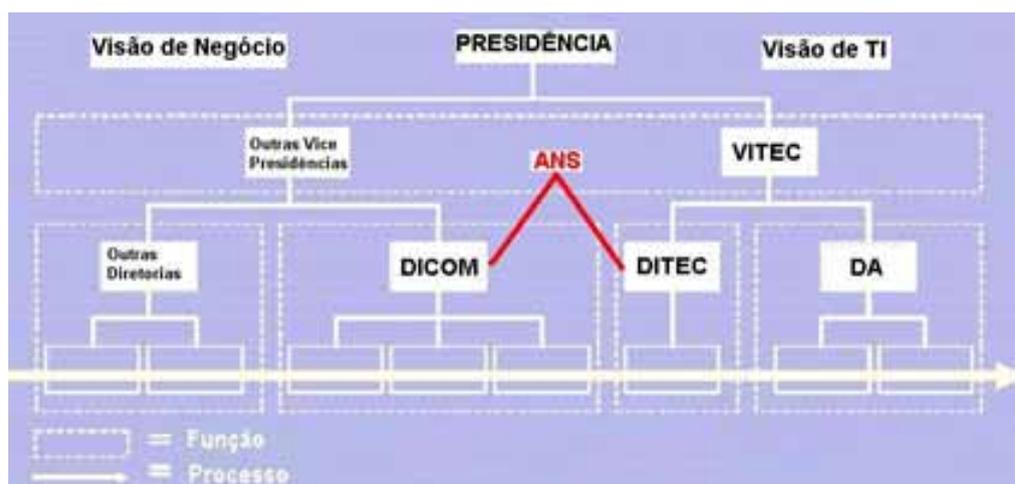


Figura 5.3: Estabelecimento do ANS entre DICOM e DITEC.

de TI e a visão de negócio sobre o serviço oferecido aos cliente.

O serviço de negócio selecionado para o estudo de caso deste trabalho corresponde ao aplicativo Gerenciador Financeiro para clientes Pessoa Jurídica - GEFIN desta instituição. Como foi estudada uma instituição financeira, o GEFIN trata-se de um serviço destinado a pessoas jurídicas clientes da organização que possibilita a execução de operações remotas e em tempo real, tais como: verificação de saldos e extratos, agendamentos de pagamentos, débitos em conta, controle de pagamentos de pessoal, entre outras. A Figura 5.4 apresenta a *interface* e uma visão geral da funcionalidades do aplicativo GEFIN (disponível em: www.bb.com.br).

Por questões de sigilo corporativo, as regras e estratégias que norteiam esse processo negocial, bem como a modelagem de processos correspondente não foram realizadas no presente estudo. Assim, para uma maior compreensão da visão negocial, nos baseamos nas informações referentes a produtos e serviços encontrados na *Intranet* da organização e no ANS do GEFIN.

Conforme citamos no Capítulo 4 a visão de TI tem como base as 11 disciplinas de Gestão preconizadas pelas melhores práticas do modelo ITIL. Como o objetivo deste trabalho é o alinhamento dos domínios de negócio e TI e as atividades da Gerência de Nível de Serviço permeiam também a área negocial, essa disciplina recebeu um destaque especial. Afinal, segundo itSMF (2006), o Gerenciamento do Nível de Serviço eficaz aperfeiçoa o desempenho do negócio e permite a criação de uma relação eficaz entre a organização de TI e seus clientes.

Análise dos Resultados

Neste trabalho, a exemplo de Ashpole et al. (2005), a principal medida de avaliação da qualidade do alinhamento é a *f-measure* (medida f). Isso indica que mesmo que a *precision* (medida p) ou o *recall* (medida r) apresentem um resultado mais alto em uma determinada abordagem, a análise definitiva do algoritmo

http://www44.bb.com.br/portaVemp/mpe/dwn/MPE.swf - Microsoft Internet Explorer fornecido por Banco do Br...

Gerenciador Financeiro Empresa XYZ Ltda Ouvidoria 00 Rede de Atendimento Sair

Micro e Pequenas Empresas Sua conta | Página inicial | Novidades | Demonstração | Ajuda

Consultas | Transferências | Pagamentos | Recebimentos | Investimentos | Empréstimos | Internacional | Outras opções

Acesso rápido Pagamento de salários | Transações pendentes | Gerenciamento de usuários | Cartão | Transferência de arquivos

Extrato conta corrente | Transferência TED | Pagamento de títulos | Cartão portador | Transferência entre contas

Boa noite EDINALDO, seu último acesso foi em 03/01/2002 às 16:37:28, sessão número 5.489

Acesse aqui o seu Fluxo de Caixa

Saldo Consolidado

Previsão de saldo para 25/07/2005

Agência	Conta corrente	Saldo cta corrente	Saldo cta investimento
8800-4	10024-2	12.451,24 C	412,34 C

Seguro Ouro Empresarial
Contrate um seguro para sua empresa e ganhe um sistema inteiramente grátis para protegê-la ainda mais.

Domicílio Visa
Centralize as vendas de sua empresa com cartões VISA no Banco do Brasil e obtenha acesso a vários produtos e serviços, taxas especiais e muitas outras vantagens.

0800 7290500 manual do usuário instruções para instalação configuração de impressão suporte técnico telefones de acesso

Ao efetivar o seu acesso, você tem à sua disposição o Gerenciador Financeiro Micro e Pequenas Empresas. O aplicativo possui um menu desenvolvido de acordo com o perfil de sua empresa, de forma a propiciar praticidade e agilidade na realização das transações. A cada sessão aberta você também recebe informações sobre data e hora do último acesso efetuado.

Menu

Concluído Intranet local

Figura 5.4: Transações disponibilizadas pelo GEFIN.

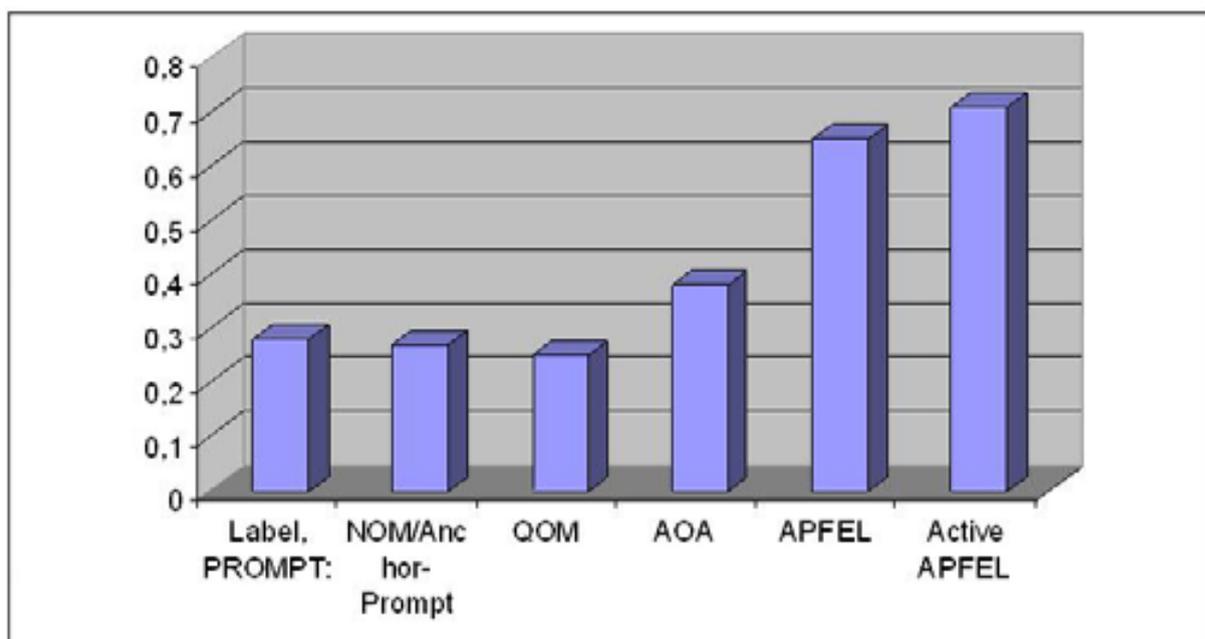


Figura 5.5: Resultado da f -measure para os algoritmos utilizados.

de alinhamento levará em consideração somente o valor encontrado na função f . Assim, na abordagem *label PROMPT* por exemplo, foi verificada a maior precisão, mas um *recall* relativamente baixo em relação aos demais algoritmos. O resultado da f -measure para cada uma das estratégias está detalhado na Figura 5.5. Já o resultado detalhado: automatização, questões, tempo, alinhamento prévio e medida f , das abordagens utilizadas pode ser verificado na Tabela 5.1.

O algoritmo *label PROMPT*, ao mesmo tempo que provê alinhamentos perfeitos devido à sua característica de basear-se unicamente no *label* dos conceitos, perde todas os alinhamentos estruturais. Trata-se de um algoritmo extremamente rápido que não requer mais de uma iteração. A Tabela 5.2 apresenta alguns conceitos que foram alinhados por esse algoritmo entre os domínios de negócio e de

Tabela 5.1: Resultados comparativos dos algoritmos utilizados.

Estratégia	Total Automat.	Questões	Tempo msec	Alinham. Prévios	F-Measure (f)
<i>Label/PROMPT</i>	SIM	0	219.0	NÃO	0.28
NOM	SIM	0	906.0	NÃO	0.27
QOM	SIM	0	329.0	NÃO	0.25
AOA	NÃO	55	5240.0	NÃO	0.58
APFEL	NÃO	0	719.0	SIM	0.65
<i>Active APFEL</i>	NÃO	49	5960.0	SIM	0.71

TI da organização.

Tabela 5.2: Elementos alinhados pelo algoritmo *label PROMPT*.

Visão de TI	Visão de Negócio
Nome	Nome
Pessoa	Pessoa
Orçamento	Orçamento
Cargo	Cargo
Serviço	Serviço

Pela análise dos resultados do algoritmo, percebemos uma certa fragilidade nessa abordagem. Afinal, o alinhamento baseado unicamente em *labels* pode apresentar algumas anormalidades quando considerados domínios tão diferentes. Imagine por exemplo uma ontologia referente à atividade industrial (indústria têxtil) e outra à atividade agrícola (árvores frutíferas). Caso houvesse o conceito manga nas duas ontologias esse algoritmo retornaria como válido o alinhamento dos conceitos manga (parte de uma peça de vestuário) e manga (fruto).

Os aspectos estruturais das ontologias de domínio foram exploradas na abordagem NOM, o que representou um significativo aumento no tempo da execução do algoritmo conforme demonstrado na Tabela 5.1. Todavia, como os domínios ontológicos possuíam estruturas completamente diferentes, essa abordagem não apresentou qualquer melhoria em relação a abordagem *label PROMPT*. A abordagem NOM, segundo Ashpole et al. (2005), apresenta o mesmo desempenho do algoritmo *Anchor-PROMPT* descrito em Noy and Musen (2001) e apresenta melhores resultados no alinhamento de domínios ontológicos estruturalmente semelhantes. Desta forma, este algoritmo não foi utilizado no alinhamento dos domínios estudados.

Conforme esperado, o algoritmo QOM é consideravelmente o mais rápido entre as abordagens que não consideram unicamente o *label* dos termos das ontologias, vide Tabela 5.1. Segundo Ashpole et al. (2005), esse algoritmo, da mesma forma que o NOM, explora as características estruturais das ontologias de domínio porém é executado de maneira otimizada, pois somente realiza a comparação de termos que apresentam um certo grau de similaridade, o que diminui consideravelmente o número de comparações realizadas. Todavia, como as estruturas das ontologias de domínio consideradas neste trabalho são consideravelmente diferentes, a abordagem QOM também não apresentou resultados satisfatórios.

Obviamente, conforme podemos perceber na Tabela 5.1, a adição de *inputs* do usuário durante o processo de alinhamento, melhora a qualidade do algoritmo. Dessa forma, com a utilização da abordagem AOA foram geradas 55 questões sobre possíveis alinhamentos ao usuário. Essa interação do usuário no processo de alinhamento provocou um esperado aumento no tempo de execução, todavia resultou em um considerável incremento na qualidade do alinhamento. É importante considerar nessa abordagem, o fato de que nem sempre o usuário estará capacitado a decidir por si próprio a validade ou não do alinhamento proposto.

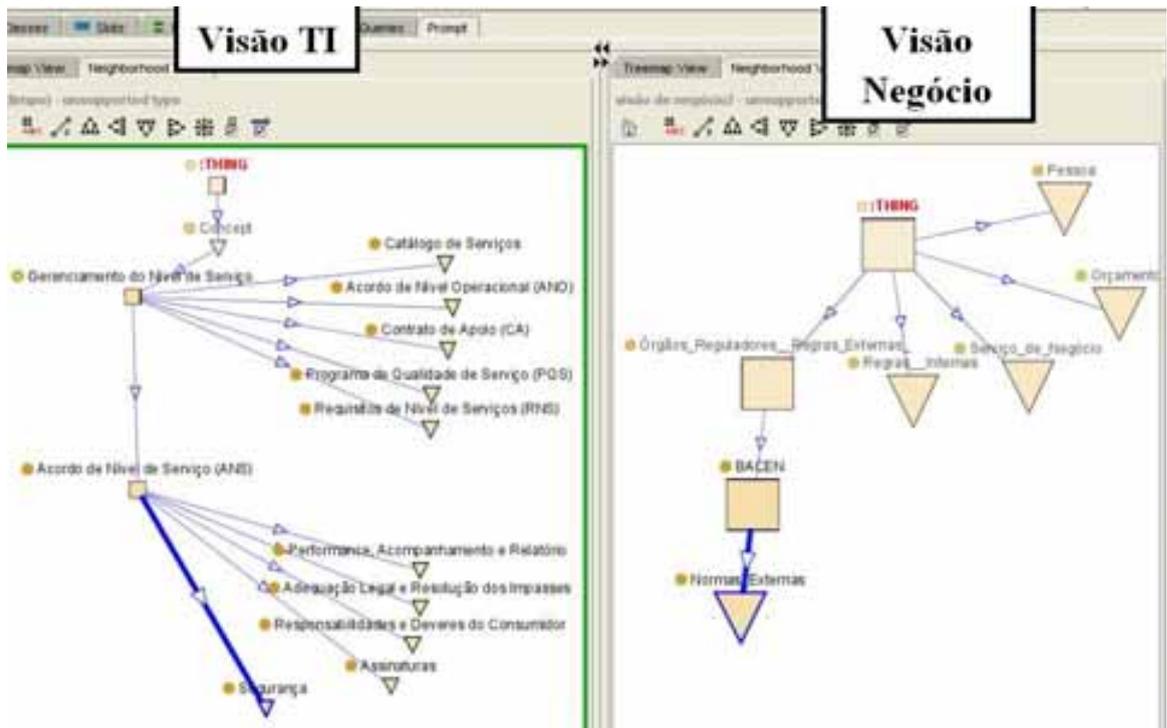


Figura 5.6: Segurança (Visão de TI) x Normas Externas (Visão de Negócio).

Assim, não necessariamente, a participação do usuário no processo garantirá uma qualidade ao processo. Em muitos casos, dependendo da complexidade das ontologias de domínio, a qualidade do alinhamento pode diminuir consideravelmente.

Também podemos perceber na Tabela 5.1 que a utilização de alinhamentos prévios como no caso do algoritmo APFEL, também provocam um incremento na medida f . Neste estudo, o algoritmo foi inicializado com 6 alinhamentos prévios envolvendo termos das ontologias de Negócio e de TI. Para o sucesso da abordagem, também é interessante que o usuário tenha um certo conhecimento dos domínios ontológicos envolvidos para propor alinhamentos iniciais corretos. Do contrário, a qualidade do treinamento do algoritmo e conseqüentemente dos alinhamentos resultantes podem decair e apresentar resultados indesejados.

Finalmente, o algoritmo *Active APFEL* a exemplo da pesquisa realizada em Ashpole et al. (2005) foi o que apresentou o melhor resultado expresso pela medida f no alinhamento entre os domínios ontológicos estudados. Esse algoritmo, ao utilizar simultaneamente as características individuais das abordagens AOA e APFEL foi o que apresentou um melhor resultado de alinhamento entre os termos de Negócio e de TI da organização estudada. A utilização de alinhamento prévio de 6 termos dos domínios ontológicos implicou a redução de 6 questões imputadas ao usuário. A medida f obtida, bem como o tempo de execução do algoritmo podem ser verificados na Tabela 5.1. Alguns alinhamentos obtidos através da abordagem *Active APFEL* estão descritos na Figuras 5.6.

Na Figura 5.6 temos que o algoritmo *Active APFEL* propôs o alinhamento do termo Segurança na visão de TI com o termo Normas Externas na visão

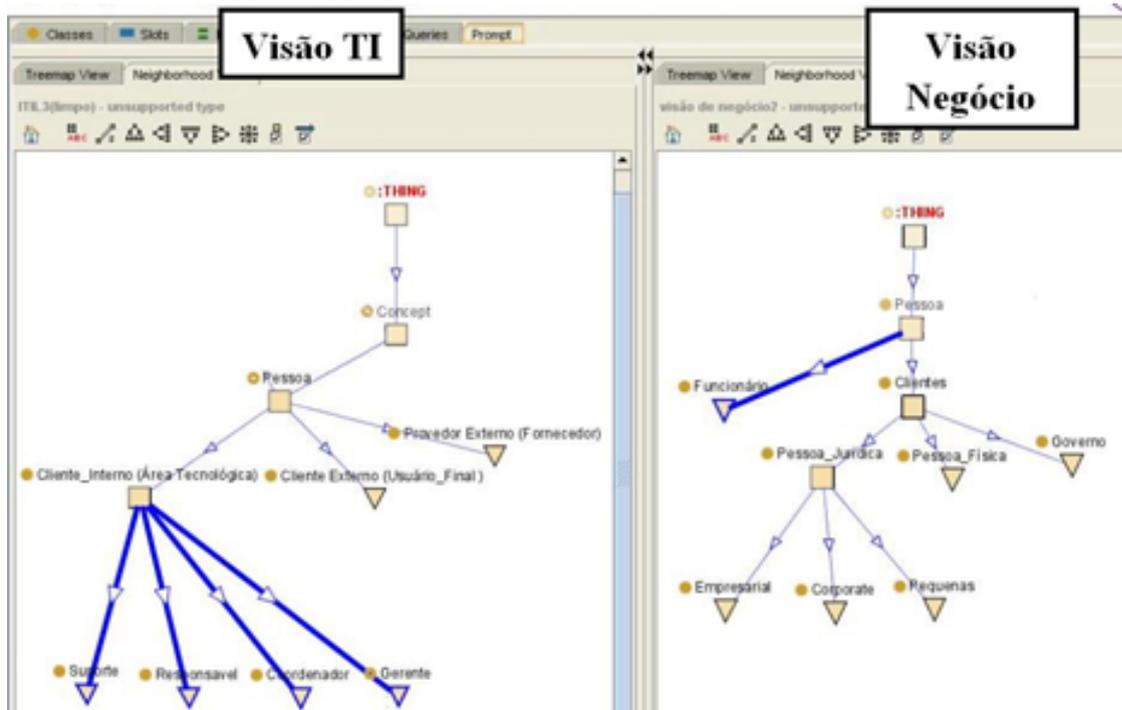


Figura 5.7: Cliente Interno (Visão de TI) x Funcionário (Visão de Negócio).

de Negócio. Esse alinhamento é bastante razoável, uma vez que os requisitos de segurança de um determinado serviço de TI devem estar em acordo com as normas externas de segurança da entidade fiscalizadora das instituições bancárias, no caso o Banco Central (BACEN).

Na Figura 5.7 identificamos o alinhamento entre o conceito clientes internos na área de TI e o conceito Funcionário na área de Negócio. De fato, segundo itSMF (2006), os clientes internos da área de TI, dividem-se em Suporte, Responsável, Coordenador e Gerente, os quais, na visão de negócio, são todos funcionários da organização.

Na Figura 5.8 identificamos o alinhamento entre os conceitos *software*, *hardware* e documentos na área de TI e o conceito Investimentos Equipamentos TI na área de Negócio. Esse alinhamento também é bastante razoável, afinal segundo itSMF (2006), os itens de configuração podem ser elementos de *hardware*, *software* ou documentos. No entanto, para a visão de negócio, não há o conceito de itens de configuração nem suas subdivisões de forma detalhadas. Para a área de negócio, esses elementos representam investimentos em equipamentos de TI.

Na Figura 5.9 temos que o algoritmo *Active APFEL* propôs o alinhamento do termo Serviços On-line na visão de TI com o termo GERENCIADOR FINANCEIRO na visão de Negócio. Esse alinhamento é bastante razoável, uma vez que entre os serviços de TI prestados pela organização, nós temos os serviços ON-LINE, que são aqueles executados em tempo real. Na ontologia de Negócio não há esse conceito, no entanto, entre os serviços de negócio prestados às pessoas jurídicas, há os serviços de auto-atendimento e, entre esses serviços, há o conceito de Gerenciador Financeiro, que é um serviço de negócio executado em tempo real.

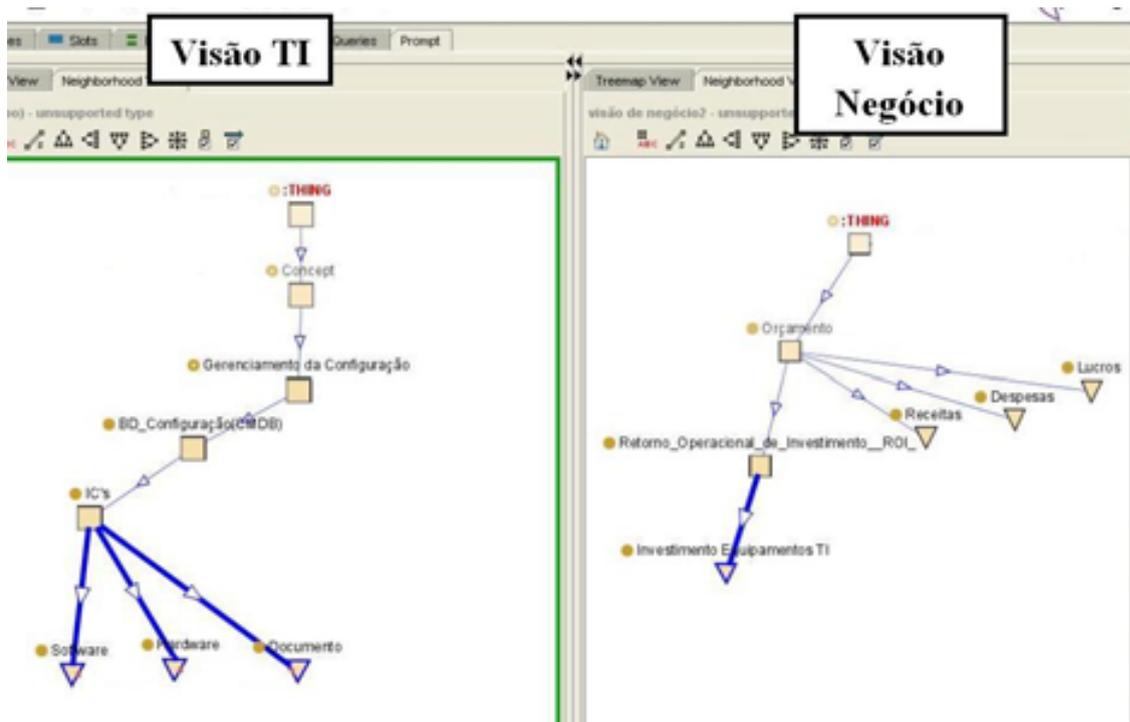


Figura 5.8: Itens de configuração (Visão de TI) x Investimentos Equipamentos de TI (Visão de Negócio).

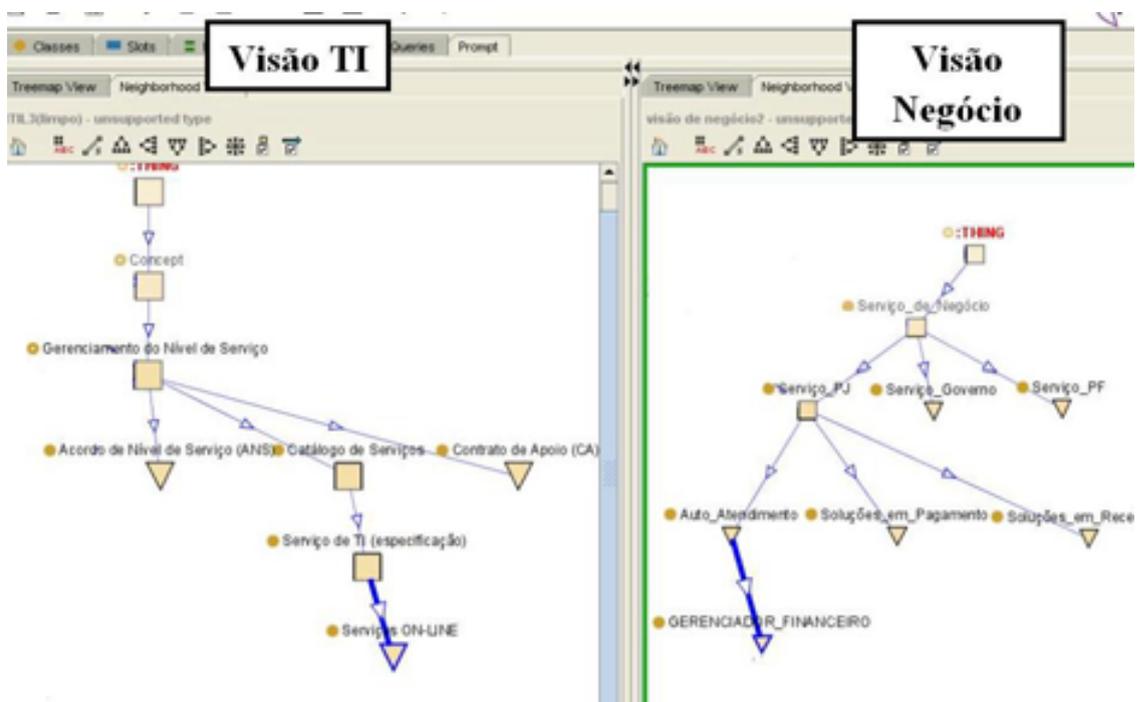


Figura 5.9: Serviços On-line (Visão de TI) x Gerenciador Financeiro (Visão de Negócio).

Os alinhamentos apresentados nas figuras, bem como outros alinhamentos obtidos com a utilização do Algoritmo *Active* APFEL, podem ser verificados na Tabela 5.3

Tabela 5.3: Outros alinhamentos obtidos com o Algoritmo *Active* APFEL.

Ontologia de TI	Ontologia de Negócio
Cadastro CPF/CNPJ	CPF/CNPJ
Cliente Externo	Empresarial
Cliente Externo	Cooperate
Cliente Interno	Funcionário
Descrição do Serviço de TI	Descrição da Norma
Documento	Investimento Equipamentos TI
Documento	Lucros
Gerenciamento da Configuração	Gerenciamento dos Dados
Gerenciamento da Continuidade	Plano de Contingência
Gerenciamento de Incidentes	Incidentes
Gerenciamento Financeiro	orçamento
<i>Hardware</i>	Investimento Equipamentos TI
Itens de configuração	Investimentos Equipamentos de TI
membro comite mud	Funcionário
membro comite mud emer	Funcionário
segurança	Normas Externas
Serviços On-line	Gerenciador Financeiro
<i>Software</i>	Investimento Equipamentos TI
Suporte	Funcionário

Percebemos a partir da análise dos resultados apresentados neste capítulo que algoritmos de alinhamento totalmente automatizados como *Label PROMPT*, *NOM*, *QOM* apesar de apresentarem tempo de execução inferior, deixam a desejar, quando as ontologias utilizadas como *input* tratam de domínios distintos e com disposições estruturais muito divergentes. Assim, algoritmos que envolvem a interação com o usuário como *AOA*, *APFEL* e *Active APFEL* resultam em pares de termos mais satisfatórios para os domínios que o *framework* ontológico se propõe a alinhar. Verificamos também que as áreas de negócio e de tecnologia da organização estudada, apesar de apresentarem perspectivas e termos diferentes sobre um determinado serviço, trabalham de maneira complementar para atingir os objetivos organizacionais. Assim, embora cada domínio tivesse um vocabulário específico, percebemos que é possível estabelecer um alinhamento entre os termos negociais e tecnológicos da empresa.

Capítulo 6

Conclusão e Trabalhos Futuros

Diante da infinidade de conceitos envolvidos entre o demandante de um serviço, seja ele um cliente externo ou um próprio usuário da organização, e o aparato tecnológico designado para atender essa demanda, torna-se imprescindível a existência de um vocabulário compartilhado entre os diversos atores. Este trabalho descreve, assim, um *framework* ontológico que, além de representar uma abordagem de ontologia de processos como método de melhoria nos padrões de gestão de TI, possibilita uma maior integração entre as diversas camadas que compõem a organização.

Utilizando-se de conceitos e padrões gerenciais com funções bem definidas (domínio ITIL) desenvolveu-se um portfólio ontológico com o objetivo de identificar os relacionamentos entre os principais conceitos desse modelo. Esse portfólio ontológico exerce uma importante função de elo entre os conceitos da Arquitetura Orientada a Serviços e o modelo ITIL.

Afinal, de acordo com os princípios gerais, ITIL tem o papel de criar operações eficientes de TI, as quais incluem a criação de diretrizes estruturais para os procedimentos e a elaboração de um processo efetivo de governança para melhor suportar os objetivos de negócios. Já a arquitetura SOA foi criada para atingir objetivos semelhantes, fornecendo diretrizes técnicas para ajudar desenvolvedores de *software* a desenhar aplicações que melhor sirvam tanto à organização como um todo, quanto aos usuários individuais.

Apesar dos dois objetivos comuns, existe um abismo entre esses dois esforços em muitas organizações. O obstáculo mais significativo é a distância psicológica e as barreiras estruturais e conceituais entre as operações de tecnologia e os times desenvolvedores de *software*. Geralmente, o que se vê são técnicos de TI empenhados em controlar seu ambiente operacional, enquanto os responsáveis pelo desenvolvimento de *software* estão preocupados em criar novas habilidades de aplicações. Uma longa história de trabalhos separados criou apreensões suficientes entre os dois grupos, que têm dificuldade em separar as diferenças e atingir um objetivo comum.

Nesse trabalho, verificamos que a modelagem dos processos executados pela organização, além de permitir o estabelecimento da visão sistêmica e da formalização das atividades, pode ser utilizada como o elo que permeia os diversos setores organizacionais. Além disso, demonstramos que o emprego da abordagem ontológica para o detalhamento de um vocabulário compartilhado, permite

estruturar um sistema de representação de conhecimento favorável não só entre os analistas, desenvolvedores e gestores de TI, mas também entre os usuários e os responsáveis pelo negócio da organização.

Constatamos também, que o *framework* ontológico desenvolvido, bem como a utilização de algoritmos de alinhamentos ontológicos, especialmente o *Active APFEL*, revelaram a possibilidade de um alinhamento harmonioso entre os conceitos e termos envolvidos em domínios tão conflitantes como são as camadas de negócio e de TI de uma organização. Além do alinhamento e do mapeamento de termos obtidos, verificamos que a abordagem ontológica proposta provê uma série de outras aplicações, entre as quais podemos citar:

- Ambiente de gestão e controle dos recursos de TI através das consultas (*queries*) disponibilizadas;
- Dispositivo para a gestão dos recursos humanos através da identificação das habilidades e capacidades necessárias à execução dos processos;
- Ferramenta para a explicitação do conhecimento implícito dos funcionários;
- Repositório formal e centralizado de conceitos empregados pelas camadas organizacionais;
- Recurso para a apresentação dos conceitos envolvidos no ambiente organizacional e para o treinamento de novos funcionários;
- Linguagem comum que atue como facilitador para a comunicação e divulgação no âmbito organizacional;
- Base de informações para o desenvolvimento de *softwares* e aplicativos.

Todavia, é importante notar que a implantação do *framework* proposto neste trabalho nas organizações é facilitada pelo grau de maturidade e cultura organizacionais. Assim, as empresas que detenham um certo nível de definição e documentação de seus processos possibilitarão uma modelagem mais realista dos mesmos. Da mesma forma, as organizações que adotem a prática de formalização de seus acordos em ANS proporcionarão uma maior qualidade na explicitação dos conceitos envolvidos. Por fim, quanto maior for o grau de comprometimento dos níveis estratégicos, táticos e operacionais da organização, maior será a consistência e a responsabilidade na manutenção e evolução das estruturas ontológicas implementadas.

Como estudos futuros serão focados os seguintes objetivos, alguns a serem realizados na própria organização estudada, para validar os resultados obtidos, e outros para ampliar o escopo de utilização do *framework* desenvolvido.

- Extensão das ontologias para contemplarem outros processos de negócio da organização estudada, o que permitirá uma completa integração entre as diversas áreas usuárias;
- Atualização da ontologia de TI para versão ITIL v3.0;

- Definição de uma ontologia de alto nível que incorpore todas as ontologias de negócio e de TI da organização, através de definições de ANS;
- Integração do *Framework* Ontológico proposto com os demais modelos de qualidade para desenvolvimento de *software*, tais como CMMI, PMBOK, MPS-BR, RUP, entre outros;
- Definição de ontologias de negócio abrangendo todos os negócios e serviços da organização;
- Definição do portfólio de negócios e serviços da organização;
- Desenvolvimento de uma arquitetura integradora entre os serviços de negócio e de TI da organização estudada de acordo com a abordagem SOA. Essa arquitetura garantirá que qualquer aplicação seja implementada somente se estiver devidamente associada a um processo de negócio existente, provendo, dessa forma: agilidade para solucionar as demandas, redução de custos com desenvolvimento, simplicidade, padronização, reaproveitamento e compartilhamento de *software*;
- Implementação de um processador para a linguagem desenvolvida a partir das ontologias construídas;
- Implantação dos princípios da Governança de TI a partir dos termos explicitados no Portfólio Ontológico da Organização e da OAN.

Tanto o *framework* ontológico proposto quanto o modelo internacional COBIT baseiam-se no alinhamento entre o negócio e a TI para capacitarem as organizações a executarem seus processos em conformidade com os princípios de governança corporativa. Eles garantem, assim, que os processos e as atividades desempenhadas pelas respectivas áreas e funções corporativas concorram de forma sistemática para o alcance dos objetivos do negócio e redução dos riscos operacionais.

Todavia, diferentemente do modelo internacional COBIT, que utiliza-se de recursos tais como sumário executivo, controle de objetivos, mapas de auditoria, ferramentas de implementação e guias com técnicas de gerenciamento, o *framework* ontológico proposto parte da premissa da existência da necessidade de uma linguagem comum entre essas camadas para se desenvolver, de fato, o gerenciamento e o controle eficaz das iniciativas de TI nas empresas. Afinal não há como compreender e controlar riscos aos serviços de negócio na utilização de novos recursos de TI, garantir que os serviços de infra-estrutura de TI estejam sendo bem gerenciados, avaliar o nível de segurança e qualidade da gestão de TI, sugerir melhorias de controle na organização - fundamentos COBIT Institute (2007) - se não há um alinhamento no entendimento do significado de termos como riscos, serviços de negócio, recursos de TI, serviços de infra-estrutura de TI, segurança, entre outros nas camadas de negócio e TI da organização.

Apêndice A

Ontologia de TI

```
# Base: http://www.owl-ontologies.com/unnamed.owl#
@prefix p3:      <http://www.owl-ontologies.com/unnamed.owl#Dept°> .
@prefix p6:      <http://www.owl-ontologies.com/unnamed.owl#Suporte_
2°> .
@prefix p1:      <http://www.owl-ontologies.com/unnamed.owl#CUSTO_AS
SOCIADO__$> .
@prefix xsd:     <http://www.w3.org/2001/XMLSchema#> .
@prefix p4:      <http://www.owl-ontologies.com/unnamed.owl#Objetivo_
do_Dept°> .
@prefix p5:      <http://www.owl-ontologies.com/unnamed.owl#Suporte_3°
> .
@prefix p2:      <http://www.owl-ontologies.com/unnamed.owl#Suporte_1°
> .
@prefix default: <http://www.owl-ontologies.com/unnamed.owl#> .
@prefix rdfs:    <http://www.w3.org/2000/01/rdf-schema#> .
@prefix rdf:     <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix owl:   <http://www.w3.org/2002/07/owl#> .

default:Texto__informação_
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Documento)
    ] ;
  rdfs:label "Texto (informação)"^^xsd:string ;
  rdfs:range xsd:string .

default:Escalação_do_Problema
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_
Serviço__ANS_)
    ] ;
```

```

    rdfs:label "Escalação do Problema"^^xsd:string ;
    rdfs:range xsd:string .

default:Gerenciamento_da_Continuidade
    rdf:type owl:Class ;
    rdfs:comment "O objetivo do Gerenciamento da Continuidade dos Ser-
viços em TI é dar suporte a todo o Gerenciamento da Continuidade do
Negócio (GCN) com a garantia de que depois de um desastre a infra-
estrutura de TI e os serviços de TI exigidos, inclusive o suporte
e a central de Serviços, possam ser restaurados dentro de limites
de tempo especificados."^^xsd:string ;
    rdfs:label "Gerenciamento da Continuidade"^^xsd:string ;
    rdfs:subClassOf default:Concept .

default:Documentos_Específicos_Configuração
    rdf:type owl:Class ;
    rdfs:subClassOf default:Documento .

default:Mud_Emerg_autoriz_por
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Mudança)
        ] .

default:Gerenciamento_de_Liberação
    rdf:type owl:Class ;
    rdfs:comment "O Gerenciamento de Liberação administra e distri-
bui versões de software e de hardware usadas para a produção,
que recebem suporte do departamento de TI para oferecer o nível
de serviço exigido."^^xsd:string ;
    rdfs:label "Gerenciamento de Liberação"^^xsd:string ;
    rdfs:subClassOf default:Concept .

default:Disponibilidade
    rdf:type owl:Class ;
    rdfs:comment "Alta disponibilidade significa que o serviço de TI
está continuamente disponível para o cliente, havendo pouco down
time e recuperação rápida. A disponibilidade alcançada é indicada
por métricas."^^xsd:string ;
    rdfs:subClassOf default:Gerenciamento_da_Disponibilidade .

default:Responsável_Requisição_Serviço
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Chamado)
        ] .

```

```

    ] ;
    rdfs:label "Responsável Requisição Serviço"^^xsd:string .

default:Orçamento
    rdf:type owl:Class ;
    rdfs:subClassOf default:Gerenciamento_Financeiro .

default:Identificador_do_Problem
    rdf:type owl:DatatypeProperty ;
    rdfs:comment "Número único identificador do problema"^^xsd:string ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Problema)
        ] ;
    rdfs:label "Identificador do Problem"^^xsd:string ;
    rdfs:range xsd:int .

default:Versão
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Hardware default:Software)
        ] ;
    rdfs:range xsd:string .

default:Métrica_Utilizada
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Disponibilidade)
        ] ;
    rdfs:label "Métrica Utilizada"^^xsd:string ;
    rdfs:range xsd:string .

default:Segmento
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Cliente_Externo__Usuá
            rio_Final__)
        ] ;
    rdfs:range xsd:string .

default:Provedor_Externo__Fornecedor_
    rdf:type owl:Class ;
    rdfs:comment "Representante do provedor externo de Serviço (fornecedor)"^^
    rdfs:label "Provedor Externo (Fornecedor)"^^xsd:string ;

```

```

    rdfs:subClassOf default:Pessoa .

default:O_Padrão_de_Cuidado
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Geral)
    ] ;
  rdfs:label "O Padrão de Cuidado"^^xsd:string ;
  rdfs:range xsd:string .

default:Descrição_do_ANO
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_Operaci
onal__ANO_)
    ] ;
  rdfs:label "Descrição do ANO"^^xsd:string ;
  rdfs:range xsd:string .

default:Adequação_as_políticas_de_segurança_do_cliente
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
viço__ANS_)
    ] ;
  rdfs:label "Adequação as políticas de segurança do cliente"^^xsd
:string ;
  rdfs:range xsd:string .

default:Observações
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Geral)
    ] ;
  rdfs:range xsd:string .

p3:_do_Provedor_Externo
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Contrato_de_Apoio__CA_)
    ] ;
  rdfs:label "Deptº do Provedor Externo"^^xsd:string ;

```

```

    rdfs:range xsd:string .

default:Cronogramas_Auditoria
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Documentos_Específicos_
            Configuração)
        ] .

default:Programa_de_Aperfeiçoamento_de_Serviços__PAS_
    rdf:type owl:Class ;
    rdfs:comment "Trata-se de um projeto nos qual se definem as ati
vidades, fases e marcos associados a melhoria de um serviço de
TI."^^xsd:string ;
    rdfs:label "Programa de Aperfeiçoamento de Serviços (PAS)"^^xsd:
string ;
    rdfs:subClassOf default:Gerenciamento_do_Nível_de_Serviço .

default:GESTOR_NEGÓCIO_____
    rdf:type owl:Class ;
    rdfs:comment "Gestor responsável pela inserção de novos serviços
/negócios na organização."^^xsd:string ;
    rdfs:label "GESTOR_NEGÓCIO ????"^^xsd:string ;
    rdfs:subClassOf default:Pessoa .

default:associado_IC
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Incidente default:Pro
            blema default:Chamado)
        ] .

default:Gerente_de_Liberação
    rdf:type owl:Class ;
    rdfs:comment "É um papel definido no contexto do MOF que é res
ponsável pelo processo de gerenciamento das atividades de versi
onamento em uma organização de TI."^^xsd:string ;
    rdfs:subClassOf default:Gerente .

default:Mudanças_de_Serviços
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
            viço__ANS_)
        ] .

```

```

    ] ;
    rdfs:label "Mudanças de Serviços"^^xsd:string ;
    rdfs:range xsd:string .

default:Gerente
    rdf:type owl:Class ;
    rdfs:comment "Entidade abstrata que tem todas categorias de ge
rentes como subclasses."^^xsd:string ;
    rdfs:subClassOf default:Cliente_Interno__Área_Tecnológica_ .

default:Capacidade_de_Processamento
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Capacidade_de_Recurso)
        ] ;
    rdfs:label "Capacidade de Processamento"^^xsd:string ;
    rdfs:range xsd:string .

default:Ferramenta_utilizada_para_monitoração
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Disponibilidade)
        ] ;
    rdfs:label "Ferramenta utilizada para monitoração"^^xsd:string ;
    rdfs:range xsd:string .

default:Suporte
    rdf:type owl:Class ;
    rdfs:comment "Funcionário relacionado ao suporte na resolução de
incidentes e problemas"^^xsd:string ;
    rdfs:subClassOf default:Cliente_Interno__Área_Tecnológica_ .

default:Oficiosidade_do_serviço
    rdf:type owl:Class ;
    rdfs:comment "Se refere as obrigações contratuais dos fornecedo
res de erviço externos. Os contratos definem o suporte a ser for
necido para os serviços terceirizados"^^xsd:string ;
    rdfs:label "Oficiosidade do serviço"^^xsd:string ;
    rdfs:subClassOf default:Gerenciamento_da_Disponibilidade .

default:Despesas_Reembonsáveis
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser

```

```

        viço__ANS_)
    ] ;
    rdfs:label "Despesas Reembonsáveis"^^xsd:string ;
    rdfs:range xsd:string .

default:Descrição_da_Causa_Raíz
    rdf:type owl:DatatypeProperty ;
    rdfs:comment "Caso o erro seja conhecido, deverá conter uma des
crição da causa raíz."^^xsd:string ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Problema)
        ] ;
    rdfs:label "Descrição da Causa Raíz"^^xsd:string ;
    rdfs:range xsd:string .

default:Confidencialidade
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
viço__ANS_)
        ] ;
    rdfs:range xsd:string .

default:Urgencia
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Incidente)
        ] ;
    rdfs:range xsd:string .

default:Cliente_Externo__Usuário_Final__
    rdf:type owl:Class ;
    rdfs:comment "Cliente externo a organização."^^xsd:string ;
    rdfs:label "Cliente Externo (Usuário_Final )"^^xsd:string ;
    rdfs:subClassOf default:Pessoa .

default:Número_Identificador_do_Chamdo
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Chamado)
        ] ;
    rdfs:label "Número Identificador do Chamdo"^^xsd:string ;
    rdfs:range xsd:string .

```

```

default:Propósitos_e_Objjetivos
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
        viço__ANS_)
    ] ;
  rdfs:label "Propósitos e Objjetivos"^^xsd:string ;
  rdfs:range xsd:string .

default:Pagamento_no_Encerramento
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Encerramento)
    ] ;
  rdfs:label "Pagamento no Encerramento"^^xsd:string ;
  rdfs:range xsd:string .

default:Reuniões_para_Revisão_do_Serviço
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
        viço__ANS_)
    ] ;
  rdfs:label "Reuniões para Revisão do Serviço"^^xsd:string ;
  rdfs:range xsd:string .

default:Mudança
  rdf:type owl:Class ;
  rdfs:comment "A medida de impacto da implantação de uma mudança
    em TI e nos negócios. A complexidade da mudança e recursos neces
    sários, incluindo pessoas, dinheiro e tempo, são avaliados para
    se determinar a categoria. O risco da implantação também é um fa
    tor a ser considerado."^^xsd:string ;
  rdfs:subClassOf default:Gerenciamento_de_Mudança .

default:Normas_de_Preenchimento
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Documentos_Específicos_
        Configuração)
    ] .

```

```

default:Adequação_Legal_e_Resolução_dos_Impasses
  rdf:type owl:Class ;
  rdfs:label "Adequação Legal e Resolução dos Impasses"^^xsd:string ;
  rdfs:subClassOf default:Acordo_de_Nível_de_Serviço__ANS_ .

default:Capacidade_de_Recurso
  rdf:type owl:Class ;
  rdfs:comment "O objetivo é determinar e conhecer o uso da infra-
estrutura e dos componentes de TI. Exemplos: Largura da faixa de
rede, capacidade de processamento, capacidade de disco, etc.
"^^xsd:string ;
  rdfs:label "Capacidade de Recurso"^^xsd:string ;
  rdfs:subClassOf default:Gerenciamento_da_Capacidade .

default:Distinções_Separações_de_Cláusulas
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Geral)
    ] ;
  rdfs:label "Distinções/Separações de Cláusulas"^^xsd:string ;
  rdfs:range xsd:string .

default:Convenções_Etiqueta
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Documentos_Específicos
_Configuração)
    ] .

default:PQS_associado_ao_Serviço
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Serviço_de_TI__especi
ficação_)
    ] ;
  rdfs:label "PQS associado ao Serviço"^^xsd:string .

default:Controle_da_Exportação
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Adequação_Legal_e_Reso
lução_dos_Impasses)
    ] ;

```

```

    rdfs:label "Controle da Exportação"^^xsd:string ;
    rdfs:range xsd:string .

default:Nome_da_Empresa
    rdf:type owl:DatatypeProperty ;
    rdfs:comment "Nome da empresa prestadora de serviço."^^xsd:string ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Provedor_Externo__Fornecedor_)
        ] ;
    rdfs:label "Nome da Empresa"^^xsd:string ;
    rdfs:range xsd:string .

default:Cargo
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Cliente_Interno__Área_Tecnológica_)
        ] ;
    rdfs:range xsd:string .

default:Cliente_Interno__Área_Tecnológica_
    rdf:type owl:Class ;
    rdfs:comment "Cliente interno ao ambiente de TI da Organização."^^xsd:string ;
    rdfs:label "Cliente_Interno (Área Tecnológica)"^^xsd:string ;
    rdfs:subClassOf default:Pessoa .

default:Desciçao_do_Negocio
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Análise_de_Risco)
        ] ;
    rdfs:label "Desciçao do Negocio"^^xsd:string ;
    rdfs:range xsd:string .

default:Soluções_de_Contorno
    rdf:type owl:Class ;
    rdfs:label "Soluções de Contorno"^^xsd:string ;
    rdfs:subClassOf default:Documento .

default:Gerenciamento_Financeiro
    rdf:type owl:Class ;

```

```
    rdfs:comment "O Gerenciamento Financeiro visa ajudar a organiza
ção de TI interna com o gerenciamento a um custo compensador dos
recursos de TI necessários para o fornecimento de serviços de TI.
"^^xsd:string ;
    rdfs:label "Gerenciamento Financeiro"^^xsd:string ;
    rdfs:subClassOf default:Concept .
```

```
default:Arbitragem
```

```
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Adequação_Legal_e_Reso
lução_dos_Impasses)
        ] ;
    rdfs:range xsd:string .
```

```
default:Atribuição
```

```
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Geral)
        ] ;
    rdfs:range xsd:string .
```

```
default:Capacidade_do_Recurso
```

```
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Hardware default:Soft
ware)
        ] ;
    rdfs:label "Capacidade do Recurso"^^xsd:string .
```

```
default:Medidas_preventivas_pertinentes
```

```
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Análise_de_Risco)
        ] ;
    rdfs:label "Medidas preventivas pertinentes"^^xsd:string ;
    rdfs:range xsd:string .
```

```
default:Partes_Envolvidas_no_Acordo
```

```
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
```

```

        viço__ANS_)
    ] ;
    rdfs:label "Partes Envolvidas no Acordo"^^xsd:string ;
    rdfs:range xsd:string .

default:Prioridade_Mudança
    rdf:type owl:DatatypeProperty ;
    rdfs:comment "Prioridade da mudança, varia de 1 a 5 e o valor pa
drão é 3."^^xsd:string ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Mudança)
        ] ;
    rdfs:range xsd:int .

default:Normas_de_estados_ICs
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Documentos_Específicos
_Configuração)
        ] .

default:Escopo_do_Trabalho
    rdf:type owl:Class ;
    rdfs:label "Escopo do Trabalho"^^xsd:string ;
    rdfs:subClassOf default:Acordo_de_Nível_de_Serviço__ANS_ .

default:Detalhamento_da_avaliação_de_impacto
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Avaliação_de_Impacto)
        ] ;
    rdfs:label "Detalhamento da avaliação de impacto"^^xsd:string ;
    rdfs:range xsd:string .

default:Duração_do_Acordo
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
viço__ANS_)
        ] ;
    rdfs:label "Duração do Acordo"^^xsd:string ;
    rdfs:range xsd:string .

```

```

default:ANS_associado_ao_Serviço
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Serviço_de_TI__especi
        ficação_)
    ] ;
  rdfs:label "ANS associado ao Serviço"^^xsd:string .

p4:_envolvido
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_Opera
        cional__ANO_)
    ] ;
  rdfs:label "Objetivo do Deptº envolvido"^^xsd:string ;
  rdfs:range xsd:string .

p6:_Nível
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Incidente default:Chama
        do)
    ] ;
  rdfs:label "Suporte 2º Nível"^^xsd:string .

default:Está_relacionado_ao_incidente
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Problema)
    ] ;
  rdfs:label "Está relacionado ao incidente"^^xsd:string .

default:Mud_padrao_autoriz_por
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Mudança)
    ] .

default:Serviços_Não-Padronizados
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;

```

```

        owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
viço__ANS_)
    ] ;
    rdfs:label "Serviços Não-Padronizados"^^xsd:string ;
    rdfs:range xsd:string .

default:Plano_de_Disponibilidade
    rdf:type owl:DatatypeProperty ;
    rdfs:comment "Plano de longo prazo que diz respeito a disponibi
lidade nos anos seguintes"^^xsd:string ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Disponibilidade)
        ] ;
    rdfs:label "Plano de Disponibilidade"^^xsd:string ;
    rdfs:range xsd:string .

default:CA_associado_ao_Serviço
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Serviço_de_TI__especifi
cação_)
        ] ;
    rdfs:label "CA associado ao Serviço"^^xsd:string .

default:Script_suficiente
    rdf:type owl:DatatypeProperty ;
    rdfs:comment "Indica se a Central de Atendimento tem script sufi
ciente para resolução do chamado"^^xsd:string ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Chamado)
        ] ;
    rdfs:label "Script suficiente"^^xsd:string ;
    rdfs:range xsd:boolean .

default:Equipe__facilidades_e_recursos_clientes
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
viço__ANS_)
        ] ;
    rdfs:label "Equipe, facilidades e recursos clientes"^^xsd:string ;
    rdfs:range xsd:string .

```

```

default:Medidas_de_segurança_da_informação_e_dos_dados
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Serviço__ANS_)
    ] ;
  rdfs:label "Medidas de segurança da informação e dos dados"^^xsd:string ;
  rdfs:range xsd:string .

```

```

<http://www.owl-ontologies.com/unnamed.owl>
  rdf:type owl:Ontology .

```

```

default:Comitê_de_Mudanças
  rdf:type owl:Class ;
  rdfs:comment "Trata-se de um grupo multifuncional de pessoas que avaliam as requisições de mudanças em relação aos aspectos de necessidade, prioridade, custo/benefício e impactos potenciais em outros processos e sistemas."^^xsd:string ;
  rdfs:subClassOf default:Gerenciamento_de_Mudança .

```

```

default:Necessidades_de_disponibilidade_p__Serviço
  rdf:type owl:DatatypeProperty ;
  rdfs:comment "O Gerenciamento de Disponibilidade pode começar assim que o negócio indique claramente quais são as suas necessidades de disponibilidade para o serviço. É um processo contínuo que só acaba quando um serviço é descatado"^^xsd:string ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Disponibilidade)
    ] ;
  rdfs:label "Necessidades de disponibilidade p/ Serviço"^^xsd:string ;
  rdfs:range xsd:string .

```

```

default:Impacto_avaliado_em
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Mudança)
    ] ;
  rdfs:label "Impacto avaliado em"^^xsd:string .

```

```

default:Situação_do_IC
  rdf:type owl:DatatypeProperty ;
  rdfs:comment "A situação do IC poderá ser: Agendada, Em atualiza

```

```

ção e Instalado"^^xsd:string ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Hardware default:Software)
    ] ;
rdfs:range xsd:string .

default:Catálogo_de_Serviços
rdf:type owl:Class ;
rdfs:comment "Oferece uma descrição detalhada dos serviços operaci
onais na linguagem do cliente, juntamente com o resumo dos níveis
de serviço associados que a organização de TI pode fornecer para
seus clientes."^^xsd:string ;
rdfs:label "Catálogo de Serviços"^^xsd:string ;
rdfs:subClassOf default:Gerenciamento_do_Nível_de_Serviço .

default:Gerentes_de_Incidentes
rdf:type owl:Class ;
rdfs:comment "Gerentes de equipes destinadas ao controle de fas
es
individuais do "ReleasePlan"."^^xsd:string ;
rdfs:subClassOf default:Gerente .

default:Impacto
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Incidente default:Proble
ma)
    ] ;
rdfs:range xsd:string .

default:Gerente_de_Problemas
rdf:type owl:Class ;
rdfs:subClassOf default:Gerente .

default:Documento
rdf:type owl:Class ;
rdfs:subClassOf default:IC_s .

default:Requisitos_de_Nível_de_Serviços__RNS_
rdf:type owl:Class ;
rdfs:label "Requisitos de Nível de Serviços (RNS)"^^xsd:string ;
rdfs:subClassOf default:Gerenciamento_do_Nível_de_Serviço .

default:Descrição_do_CA
rdf:type owl:DatatypeProperty ;

```

```

rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Contrato_de_Apoio__CA_)
    ] ;
rdfs:label "Descrição do CA"^^xsd:string ;
rdfs:range xsd:string .

default:ANO_associado_ao_serviço
rdf:type owl:ObjectProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Serviço_de_TI__especificação_)
    ] ;
rdfs:label "ANO associado ao serviço"^^xsd:string .

default:Ordem_da_Comissão
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Serviço__ANS_)
    ] ;
rdfs:label "Ordem da Comissão"^^xsd:string ;
rdfs:range xsd:string .

default:IC_s
rdf:type owl:Class ;
rdfs:label "IC's"^^xsd:string ;
rdfs:subClassOf default:BD_Configuração_CMDB_ .

default:Está_relacionada_ao_Incidente
rdf:type owl:ObjectProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Mudança)
    ] ;
rdfs:label "Está relacionada ao Incidente"^^xsd:string .

default:Impacto_Mudança
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Mudança)
    ] ;
rdfs:range xsd:string .

```

```

default:Limitação_de_responsabilidade
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Adequação_Legal_e_Reso
        lução_dos_Impasses)
    ] ;
  rdfs:label "Limitação de responsabilidade"^^xsd:string ;
  rdfs:range xsd:string .

default:Descrição_do_Suporte
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Suporte)
    ] ;
  rdfs:label "Descrição do Suporte"^^xsd:string ;
  rdfs:range xsd:string .

default:Capacidade_de_Negocio
  rdf:type owl:Class ;
  rdfs:comment "O objetivo é conhecer as necessidades atuais e fu
    turas do negócio. Isso se faz obtendo-se informações do cliente,
    por exemplo fornececidas por planos estratégicos ou planos de
    marketing, ou realizando análises de tendencias (proativo).
    "^^xsd:string ;
  rdfs:label "Capacidade de Negocio"^^xsd:string ;
  rdfs:subClassOf default:Gerenciamento_da_Capacidade .

default:Horário_de_Abertura
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Chamado)
    ] ;
  rdfs:label "Horário de Abertura"^^xsd:string ;
  rdfs:range xsd:string .

default:Mudança_na_Equipe-Chave
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
        viço__ANS_)
    ] ;
  rdfs:label "Mudança na Equipe-Chave"^^xsd:string ;
  rdfs:range xsd:string .

```

```

default:Erro_Conhecido
  rdf:type owl:DatatypeProperty ;
  rdfs:comment "Um problema para o qual a causa raiz é conhecida
e para o qual se identificou uma solução de contorno"^^xsd:string ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Problema)
    ] ;
  rdfs:label "Erro Conhecido"^^xsd:string ;
  rdfs:range xsd:boolean .

default:Prioridade_do_Incidente
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Incidente)
    ] ;
  rdfs:label "Prioridade do Incidente"^^xsd:string ;
  rdfs:range xsd:string .

default:Plano_de_Recuperação
  rdf:type owl:Class ;
  rdfs:label "Plano de Recuperação"^^xsd:string ;
  rdfs:subClassOf default:Gerenciamento_da_Continuidade .

default:Contrato_de_Apoio__CA_
  rdf:type owl:Class ;
  rdfs:comment "É um contrato com um provedor externo."^^xsd:string ;
  rdfs:label "Contrato de Apoio (CA)"^^xsd:string ;
  rdfs:subClassOf default:Gerenciamento_do_Nível_de_Serviço .

default:Coordenador
  rdf:type owl:Class ;
  rdfs:comment "Coordenador de reuniões ou procedimentos"^^xsd:string ;
  rdfs:subClassOf default:Cliente_Interno__Área_Tecnológica_ .

default:Marcas_de_nível__objetivos_e_métricas
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
viço__ANS_)
    ] ;
  rdfs:label "Marcas de nível, objetivos e métricas"^^xsd:string ;
  rdfs:range xsd:string .

```

```

default:nome
  rdf:type owl:DatatypeProperty ;
  rdfs:comment "Nome"^^xsd:string ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Pessoa)
    ] ;
  rdfs:range xsd:string .

default:Estado_Liberação
  rdf:type owl:DatatypeProperty ;
  rdfs:comment "Estado atual da liberação: produção, concluída, a
  gendada."^^xsd:string ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Liberação)
    ] ;
  rdfs:range xsd:string .

default:Documentos_Específicos_Mudança
  rdf:type owl:Class ;
  rdfs:subClassOf default:Documento .

default:Metas
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Programa_de_Qualidade_
      de_Serviço__PQS_)
    ] ;
  rdfs:range xsd:string .

default:Nível_de_Suporte
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Suporte)
    ] ;
  rdfs:label "Nível de Suporte"^^xsd:string ;
  rdfs:range xsd:string .

default:Capacidade_de_Serviço
  rdf:type owl:Class ;
  rdfs:comment "O objetivo é determinar e conhecer o uso dos servi
  ços de TI (produtos e serviços fornecidos aos clientes). O desem
  penho e as cargas de pico precisam ser conhecidos para garantir-se
  que se possa fazer e entregar acordos de serviço adequados."

```

```

    ^^xsd:string ;
    rdfs:label "Capacidade de Serviço"^^xsd:string ;
    rdfs:subClassOf default:Gerenciamento_da_Capacidade .

default:Local_da_entrega_do_Serviço
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
            viço__ANS_)
        ] ;
    rdfs:label "Local da entrega do Serviço"^^xsd:string ;
    rdfs:range xsd:string .

default:Disponibilidade_de_Serviço
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
            viço__ANS_)
        ] ;
    rdfs:label "Disponibilidade de Serviço"^^xsd:string ;
    rdfs:range xsd:string .

default:Recuperação_de_desastre
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
            viço__ANS_)
        ] ;
    rdfs:label "Recuperação de desastre"^^xsd:string ;
    rdfs:range xsd:string .

default:Exclusões
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
            viço__ANS_)
        ] ;
    rdfs:range xsd:string .

default:Num_Identificador_Incidente
    rdf:type owl:DatatypeProperty ;
    rdfs:comment "Esse atributo corresponde ao número de identifica
    dor único do incidente"^^xsd:string ;

```

```

rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Incidente)
    ] ;
rdfs:range xsd:int .

default:Comitê_de_Mudanças_Emergencias
rdf:type owl:Class ;
rdfs:comment "É um subconjunto do CAB que lida somente com mudan
ças emergenciais. Ele tem a autorização para aprovar e rejeitar
mudanças com prioridade emergencial."^^xsd:string ;
rdfs:subClassOf default:Gerenciamento_de_Mudança .

default:Encerramento
rdf:type owl:Class ;
rdfs:subClassOf default:Acordo_de_Nível_de_Serviço__ANS_ .

default:Geral
rdf:type owl:Class ;
rdfs:subClassOf default:Acordo_de_Nível_de_Serviço__ANS_ .

default:Provedor_externo
rdf:type owl:DatatypeProperty ;
rdfs:comment "Empresa provedora do serviço"^^xsd:string ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Contrato_de_Apoio__CA_)
    ] ;
rdfs:label "Provedor externo"^^xsd:string ;
rdfs:range xsd:string .

default:Processamento_e_autorização_de_faturas
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
viço__ANS_)
    ] ;
rdfs:label "Processamento e autorização de faturas"^^xsd:string ;
rdfs:range xsd:string .

default:Categoria_do_Problema
rdf:type owl:DatatypeProperty ;
rdfs:comment "identificação do domínio pertinente, por exemplo har
dware ou software."^^xsd:string ;
rdfs:domain
    [ rdf:type owl:Class ;

```

```

        owl:unionOf (owl:Thing default:Problema)
    ] ;
    rdfs:label "Categoria do Problema"^^xsd:string ;
    rdfs:range xsd:string .

default:CNPJ_empresa
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Provedor_Externo__Fornecedor_)
        ] ;
    rdfs:label "CNPJ empresa"^^xsd:string ;
    rdfs:range xsd:string .

default:Encerramento_por_Causa
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Encerramento)
        ] ;
    rdfs:label "Encerramento por Causa"^^xsd:string ;
    rdfs:range xsd:string .

default:Políticas_Armazenamento
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Documentos_Específicos_Configuração)
        ] .

default:Categoria_da_Mudança
    rdf:type owl:DatatypeProperty ;
    rdfs:comment "A medida de impacto da implantação de uma mudança em TI e nos negócios. A complexidade da mudança e recursos necessários, incluindo pessoas, dinheiro e tempo, são avaliados para se determinar a categoria. O risco da implantação também é um fator a ser considerado."^^xsd:string ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Mudança)
        ] ;
    rdfs:label "Categoria da Mudança"^^xsd:string ;
    rdfs:range xsd:string .

default:Mudanças_no_acordo

```

```

rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Geral)
    ] ;
rdfs:label "Mudanças no acordo"^^xsd:string ;
rdfs:range xsd:string .

default:Aprovada
rdf:type owl:DatatypeProperty ;
rdfs:comment "Descreve se a mudança foi autorizada ou não para ser
realizada."^^xsd:string ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Mudança)
    ] ;
rdfs:range xsd:boolean .

default:Situação_do_Chamado
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Incidente default:Pro
        blema default:Chamado)
    ] ;
rdfs:label "Situação do Chamado"^^xsd:string ;
rdfs:range xsd:string .

default:membro_comite_mud_emerg
rdf:type owl:ObjectProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Comitê_de_Mudanças_Emer
        gencias)
    ] .

default:Faturas
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
        viço__ANS_)
    ] ;
rdfs:range xsd:string .

default:Data_Inicial_da_Mudança

```

```

rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Mudança)
    ] ;
rdfs:label "Data Inicial da Mudança"^^xsd:string ;
rdfs:range xsd:string .

default:Incidente
rdf:type owl:Class ;
rdfs:comment "Qualquer evento que não faz parte do funcionamento
normal de um serviço e que causa, ou pode causar, a sua interrup
ção ou uma redução da qualidade."^^xsd:string ;
rdfs:subClassOf default:Gerenciamento_de_Incidentes .

default:Limitação_de_Ação
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Adequação_Legal_e_Reso
lução_dos_Impasses)
    ] ;
rdfs:label "Limitação de Ação"^^xsd:string ;
rdfs:range xsd:string .

default:Descrição_itens_necess_const_proj
rdf:type owl:ObjectProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Itens_necessários_constru
ção_projeto_liberação)
    ] ;
rdfs:label "Descrição_itens_necess_const/proj"^^xsd:string .

default:Desastre
rdf:type owl:Class ;
rdfs:comment "Um acontecimento que afeta de tal forma um serviço
ou sistema que a resutaração do seu nível de desempenho original
exige considerável esforço. Mais grave que incidente (INTERRUP
ÇÃO DO NEGÓCIO)"^^xsd:string ;
rdfs:subClassOf default:Gerenciamento_da_Continuidade .

<http://www.owl-ontologies.com/unnamed.owl#Itens_necessários_constr.
_proj.>
rdf:type owl:ObjectProperty ;
rdfs:domain
    [ rdf:type owl:Class ;

```

```

        owl:unionOf (owl:Thing default:Liberação)
    ] ;
    rdfs:label "Itens necessários constr./proj."^^xsd:string .

default:Pessoa
    rdf:type owl:Class ;
    rdfs:comment "É uma entidade que tem 4 subclasses: Usuário Final,
    Cliente Interno, Gestor_Negócio e Fornecedor. Representa uma pes
    soa envolvida nas diversas áreas da organização. Pode ter vários
    papéis (roles) associados."^^xsd:string ;
    rdfs:subClassOf default:Concept .

default:Contratos
    rdf:type owl:Class ;
    rdfs:subClassOf default:Documento .

default:Relatório_do_Nível_de_Serviço
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
          viço__ANS_)
        ] ;
    rdfs:label "Relatório do Nível de Serviço"^^xsd:string ;
    rdfs:range xsd:string .

default:Data_final
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Liberação)
        ] ;
    rdfs:range xsd:string .

default:Exibições
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Geral)
        ] ;
    rdfs:range xsd:string .

default:Acordo_de_Nível_de_Serviço__ANS_
    rdf:type owl:Class ;
    rdfs:comment "É um acordo entre a organização de TI e o cliente,
    que detalha o serviço ou os serviços a serem fornecidos. O ANS
    descreve os serviços em termos não técnicos sintonizados com a

```

```
percepção do cliente, e durante o período do acordo ele serve
como o padrão para medir e ajustar os serviços de TI."^^xsd:string ;
rdfs:label "Acordo de Nível de Serviço (ANS)"^^xsd:string ;
rdfs:subClassOf default:Gerenciamento_do_Nível_de_Serviço .
```

```
p2:_Nível
rdf:type owl:ObjectProperty ;
rdfs:domain
  [ rdf:type owl:Class ;
    owl:unionOf (owl:Thing default:Chamado)
  ] ;
rdfs:label "Suporte 1º Nível"^^xsd:string .
```

```
default:Data_Final_da_Mudança
rdf:type owl:DatatypeProperty ;
rdfs:domain
  [ rdf:type owl:Class ;
    owl:unionOf (owl:Thing default:Mudança)
  ] ;
rdfs:label "Data Final da Mudança"^^xsd:string ;
rdfs:range xsd:string .
```

```
default:Indicadores_de_Desempenho
rdf:type owl:DatatypeProperty ;
rdfs:domain
  [ rdf:type owl:Class ;
    owl:unionOf (owl:Thing default:Programa_de_Qualidade_
de_Serviço__PQS_)
  ] ;
rdfs:label "Indicadores de Desempenho"^^xsd:string ;
rdfs:range xsd:string .
```

```
default:Cobrança
rdf:type owl:Class ;
rdfs:subClassOf default:Gerenciamento_Financeiro .
```

```
default:Código_do_RNS
rdf:type owl:DatatypeProperty ;
rdfs:domain
  [ rdf:type owl:Class ;
    owl:unionOf (owl:Thing default:Requisitos_de_Nível_de_
Serviços__RNS_)
  ] ;
rdfs:label "Código do RNS"^^xsd:string ;
rdfs:range xsd:string .
```

```
default:Número_de_Bem
```

```

rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:IC_s)
    ] ;
rdfs:range xsd:string .

default:Sustentabilidade
rdf:type owl:Class ;
rdfs:comment "A sustentabilidade e a capacidade de recuperação dizem respeito as atividades necessárias para manter o serviço em operação e restaurá-lo em caso de falha. Isso exige manutenção preventiva e inspeções programadas"^^xsd:string ;
rdfs:subClassOf default:Gerenciamento_da_Disponibilidade .

default:Software
rdf:type owl:Class ;
rdfs:subClassOf default:IC_s .

default:Alocado
rdf:type owl:DatatypeProperty ;
rdfs:comment "Localização do IC no ambiente tecnológico (Ex.: Caminho na rede, ambiente de TI, Armário, Biblioteca, etc)"^^xsd:string ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:IC_s)
    ] ;
rdfs:range xsd:string .

default:Categoria_Renda
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Cliente_Externo__Usuário_Final__)
    ] ;
rdfs:range xsd:string .

default:Matrícula_Funcional
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Cliente_Interno__Área_Tecnológica_ default:GESTOR_NEGÓCIO____)
    ] ;
rdfs:label "Matrícula Funcional"^^xsd:string ;
rdfs:range xsd:string .

```

```

default:Disponibilidade_Acordada
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
        viço__ANS_)
    ] ;
  rdfs:label "Disponibilidade Acordada"^^xsd:string .

default:Descrição_do_PQS
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Programa_de_Qualidade_
        de_Serviço__PQS_)
    ] ;
  rdfs:label "Descrição do PQS"^^xsd:string ;
  rdfs:range xsd:string .

default:Concept
  rdf:type owl:Class ;
  rdfs:comment "The common ancestor for all concepts (i.e. types
    of entity such as Person, Address...) in an ontology"^^xsd:string .

default:Riscos_aos_quais_o_negócio_está_exposto
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Análise_de_Risco)
    ] ;
  rdfs:label "Riscos aos quais o negócio está exposto"^^xsd:string ;
  rdfs:range xsd:string .

default:Gerenciamento_do_Nível_de_Serviço
  rdf:type owl:Class ;
  rdfs:comment "O Gerenciamento do Nível de Serviço garante que os serviços
    a monitoração e a comunicação sobre o desenvolvimento da oragani
    zação de TI e também com a criação de uma relação de negócio efi
    caz entre a organização de TI e seus clientes."^^xsd:string ;
  rdfs:label "Gerenciamento do Nível de Serviço"^^xsd:string ;
  rdfs:subClassOf default:Concept .

default:BD_Configuração_CMDB_
  rdf:type owl:Class ;
  rdfs:comment "Banco de Dados contendo todas as informações sobre
    os itens de configuração. Um item de configuração sempre está as

```

```
sociado a um serviço. A informação gravada no CMDB para cada item
de configuração identificado, que pode incluir o nome do item,
descrição e localização de opções de configuração tecnicamente
detalhadas."^^xsd:string ;
rdfs:label "BD_Configuração(CMDB)"^^xsd:string ;
rdfs:subClassOf default:Gerenciamento_da_Configuração .
```

```
default:Contabilidade
  rdf:type owl:Class ;
  rdfs:subClassOf default:Gerenciamento_Financeiro .
```

```
default:Qualidade_do_serviço
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
viço__ANS_)
    ] ;
  rdfs:label "Qualidade do serviço"^^xsd:string ;
  rdfs:range xsd:string .
```

```
default:A_Lei_Governamental
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Adequação_Legal_e_Reso
lução_dos_Impasses)
    ] ;
  rdfs:label "A Lei Governamental"^^xsd:string ;
  rdfs:range xsd:string .
```

```
default:Liberação
  rdf:type owl:Class ;
  rdfs:subClassOf default:Gerenciamento_de_Liberação .
```

```
default:Direitos_de_Propriedade_Intelectual
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
viço__ANS_)
    ] ;
  rdfs:label "Direitos de Propriedade Intelectual"^^xsd:string ;
  rdfs:range xsd:string .
```

```
default:Código_Identificador_IC
  rdf:type owl:DatatypeProperty ;
```

```

    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:IC_s)
        ] ;
    rdfs:range xsd:string .

default:RNS_associado_ao_serviço
    rdf:type owl:ObjectProperty ;
    rdfs:comment "Cobrem as definições detalhadas das necessidades
do cliente e são usados pra desenvolver, modificar e dar início
aos serviços. Podem servir como estrutura básica para projetar
um serviço e o(s) ANS e ICs a ele associados."^^xsd:string ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Serviço_de_TI__especifica
ção_)
        ] ;
    rdfs:label "RNS associado ao serviço"^^xsd:string .

default:Num_RDM
    rdf:type owl:DatatypeProperty ;
    rdfs:comment "Número da requisição da mudança"^^xsd:string ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Mudança)
        ] ;
    rdfs:range xsd:string .

default:Compensação
    rdf:type owl:Class ;
    rdfs:subClassOf default:Acordo_de_Nível_de_Serviço__ANS_ .

default:Código_do_PQS
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Programa_de_Qualidade_
de_Serviço__PQS_)
        ] ;
    rdfs:label "Código do PQS"^^xsd:string ;
    rdfs:range xsd:string .

default:Treinamentos_equipamentos_e_tarefas_especializadas
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser

```

```

        viço__ANS_)
    ] ;
    rdfs:label "Treinamentos equipamentos e tarefas especializadas"^^xsd:string ;
    rdfs:range xsd:string .

default:Taxas
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Serviço__ANS_)
        ] ;
    rdfs:range xsd:string .

p1:_ rdf:type owl:DatatypeProperty ;
    rdfs:comment "Inclui custo do ativo, custo de manutenção e custo de funcionamento (energia, refrigeração, etc)"^^xsd:string ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:IC_s)
        ] ;
    rdfs:label "CUSTO ASSOCIADO ($)"^^xsd:string ;
    rdfs:range xsd:string .

default:Destruição_de_dados_e_registros
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Serviço__ANS_)
        ] ;
    rdfs:label "Destruição de dados e registros"^^xsd:string ;
    rdfs:range xsd:string .

default:Identificador_Documento
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Documento)
        ] ;
    rdfs:range xsd:int .

default:Serviço_de_TI__especificação_
    rdf:type owl:Class ;
    rdfs:comment "A Especificação de Serviço descreve a relação entre a funcionalidade (que foi acordada com o cliente, sendo portanto

```

```

    focada nele) e a tecnologia (implementada dentro da organização
    de TI, portanto focada na TI) e fornece uma especificação deta
    lhada do serviço."^^xsd:string ;
    rdfs:label "Serviço de TI (especificação)"^^xsd:string ;
    rdfs:subClassOf default:Catálogo_de_Serviços .

```

```

default:Está_relacionado_ao_problema
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Mudança)
        ] ;
    rdfs:label "Está relacionado ao problema"^^xsd:string .

```

```

default:Gerenciamento_da_Capacidade
    rdf:type owl:Class ;
    rdfs:comment "O Gerenciamento da Capacidade visa fornecer regular
    mente e na hora certa os recursos de TI exigidos (quando eles o
    são) e a um preço razoável, ajustando as exigencias atuais e futu
    ras do negócio."^^xsd:string ;
    rdfs:label "Gerenciamento da Capacidade"^^xsd:string ;
    rdfs:subClassOf default:Concept .

```

```

default:Definição_do_Problema
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
            viço__ANS_)
        ] ;
    rdfs:label "Definição do Problema"^^xsd:string ;
    rdfs:range xsd:string .

```

```

default:Descrito
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Hardware default:Software)
        ] .

```

```

default:Data_inicial
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Liberação)
        ] ;
    rdfs:range xsd:string .

```

```

default:Chamado
  rdf:type owl:Class ;
  rdfs:comment "Chamado aberto pela central única de contato Service Desk"^^xsd:string ;
  rdfs:subClassOf default:Central_de_Serviços .

default:Caminho_Documento
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Documento)
    ] ;
  rdfs:range xsd:string .

default:Data_de_Início
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Serviço__ANS_)
    ] ;
  rdfs:label "Data de Início"^^xsd:string ;
  rdfs:range xsd:string .

default:Indenização
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Garantias_e_Soluções)
    ] ;
  rdfs:range xsd:string .

default:Código_do_Plano_de_Recuperação
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Plano_de_Recuperação)
    ] ;
  rdfs:label "Código do Plano de Recuperação"^^xsd:string ;
  rdfs:range xsd:string .

default:Hardware
  rdf:type owl:Class ;
  rdfs:comment "Um banco de dados que contém todos os detalhes relevantes de cada CI e detalhes de importantes relacionamentos entre CIs. O banco de dados pode incluir o código de ID, número

```

serial, categoria, status, modelo, versão, localização, responsabilidade, ou informações históricas sobre o item. O nível de detalhes contidos nesse banco de dados depende dos objetivos ou o grau para o qual a informação deve estar disponível."^^xsd:string ;
rdfs:subClassOf default:IC_s .

```
default:Cadastro__CPF_CNPJ_  
  rdf:type owl:DatatypeProperty ;  
  rdfs:domain  
    [ rdf:type owl:Class ;  
      owl:unionOf (owl:Thing default:Pessoa)  
    ] ;  
  rdfs:label "Cadastro (CPF/CNPJ)"^^xsd:string ;  
  rdfs:range xsd:string .
```

```
default:Não-solicitação  
  rdf:type owl:DatatypeProperty ;  
  rdfs:domain  
    [ rdf:type owl:Class ;  
      owl:unionOf (owl:Thing default:Geral)  
    ] ;  
  rdfs:range xsd:string .
```

```
default:Encerramento_após_termo_do_Acordo_Inicial  
  rdf:type owl:DatatypeProperty ;  
  rdfs:domain  
    [ rdf:type owl:Class ;  
      owl:unionOf (owl:Thing default:Encerramento)  
    ] ;  
  rdfs:label "Encerramento após termo do Acordo Inicial"^^xsd:string ;  
  rdfs:range xsd:string .
```

```
default:Gerenciamento_da_Disponibilidade  
  rdf:type owl:Class ;  
  rdfs:comment "O objetivo do Gerenciamento da Disponibilidade é fornecedor com eficiencia de custo um determinado nível de disponibilidade do serviço de TI que capacite o negócio a atingir seus objetivos."^^xsd:string ;  
  rdfs:label "Gerenciamento da Disponibilidade"^^xsd:string ;  
  rdfs:subClassOf default:Concept .
```

```
default:Itens_necessários_construção_projeto_liberação  
  rdf:type owl:Class ;  
  rdfs:comment "É uma classe abstrata que define um atributo do tipo \"Document\" e três variáveis boolean que serão herdadas por suas subclasses. As subclasses definem itens que podem ser necessários na construção e no projeto de um pacote de Liberação. Elas
```

```

são: "SpecificProcedures" , "AdministrativeTools" , "Automation
Tools" , "NewTechnologiesDevelopment" , "RemotionMechanism" ,
"UpdateConfigurationFiles" , "ScriptCreation" , "TolsAndTechno
logiesAcquisition" , "SpecializedSupport" , "Others".^^xsd:string ;
rdfs:label "Itens_necessários_construção/projeto_liberação"^^xsd
:string ;
rdfs:subClassOf default:Gerenciamento_de_Liberação .

```

```

default:Descrição_do_Serviço_de_TI
rdf:type owl:DatatypeProperty ;
rdfs:comment "Esse atributo traz uma descrição resumida do ser
viço de TI fornecido."^^xsd:string ;
rdfs:domain
[ rdf:type owl:Class ;
owl:unionOf (owl:Thing default:Serviço_de_TI__especi
ficação_)
] ;
rdfs:label "Descrição do Serviço de TI"^^xsd:string ;
rdfs:range xsd:string .

```

```

default:Introdução
rdf:type owl:Class ;
rdfs:comment "Propósitos e objetivos, partes do acordo, duração,
início, etc."^^xsd:string ;
rdfs:subClassOf default:Acordo_de_Nível_de_Serviço__ANS_ .

```

```

default:Gerenciamento_da_Configuração
rdf:type owl:Class ;
rdfs:comment "O Gerenciamento da Configuração visa auxiliar no
controle do valor econômico dos serviços de TI (uma combinação
de exigencias dos clientes, qualidade e custos) mantendo um mode
lo lógico da infra-estrutura de TI e dos serviços de TI e ofere
cendo informações sobre eles para outros processos de negócio.
O Gerenciamento da Configuração implemente isso identificando,
monitorando, controlando e fornecendo inofrmações sobre os itens
da Configuração e suas versões."^^xsd:string ;
rdfs:label "Gerenciamento da Configuração"^^xsd:string ;
rdfs:subClassOf default:Concept .

```

```

default:Prioridade_do_Problema
rdf:type owl:DatatypeProperty ;
rdfs:comment "Combinação de urgencia, impacto, risco e recursos
exigidos."^^xsd:string ;
rdfs:domain
[ rdf:type owl:Class ;
owl:unionOf (owl:Thing default:Problema)
] ;

```

```

    rdfs:label "Prioridade do Problema"^^xsd:string ;
    rdfs:range xsd:string .

default:Central_de_Serviços
    rdf:type owl:Class ;
    rdfs:comment "O objetivo da Central de Serviços é dar suporte
    ao fornecimento dos serviços que foram acordados, ao garantir
    acesso a organização de TI e responsabilizar-se por uma série
    de atividades de suporte (de vários processos)."^^xsd:string ;
    rdfs:label "Central de Serviços"^^xsd:string ;
    rdfs:subClassOf default:Concept .

default:Provedor
    rdf:type owl:Class ;
    rdfs:comment "É o representante de uma organização que está auto
    rizado a em nome dessa organização fazer acordos relacionados com
    o fornecimento de serviços de TI."^^xsd:string ;
    rdfs:subClassOf default:Gerenciamento_do_Nível_de_Serviço .

default:Como_cada_serviço_individual_deve_ser_monitorado
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
            viço__ANS_)
        ] ;
    rdfs:label "Como cada serviço individual deve ser monitorado
    "^^xsd:string ;
    rdfs:range xsd:string .

default:Segurança
    rdf:type owl:Class ;
    rdfs:subClassOf default:Acordo_de_Nível_de_Serviço__ANS_ .

default:Gerenciamento_de_Incidentes
    rdf:type owl:Class ;
    rdfs:comment "O Gerenciamento de Incidentes é uma tarefa reativa,
    ou seja, ele reduz ou elimina os efeitos das perturbações poten
    cias ou reais nos serviços de TI, garantindo que os usuários pos
    sam voltar a trabalhar o mais cedo possível. Por essas razões os
    incidentes são registrados, classificados e alocados para os es
    pecialistas adequados; sua evolução é monitorada; e eles são re
    solvidos e posteriormente encerrados."^^xsd:string ;
    rdfs:label "Gerenciamento de Incidentes"^^xsd:string ;
    rdfs:subClassOf default:Concept .

default:Urgência

```

```

rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Problema)
    ] ;
rdfs:range xsd:string .

default:Serviço_de_Suporte_e_Mesa_de_Serviço
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Serviço__ANS_)
    ] ;
rdfs:label "Serviço de Suporte e Mesa de Serviço"^^xsd:string ;
rdfs:range xsd:string .

default:Criptografia
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Segurança)
    ] ;
rdfs:range xsd:string .

default:Horário_de_Fechamento
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Chamado)
    ] ;
rdfs:label "Horário de Fechamento"^^xsd:string ;
rdfs:range xsd:string .

default:Requerimentos_Específicos_Liberação
rdf:type owl:Class ;
rdfs:comment "Requerimentos específicos para a liberação em questão: treinamentos, planos de suporte, estratégia de comunicação, etc."^^xsd:string ;
rdfs:subClassOf default:Gerenciamento_de_Liberação .

default:Modelo
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Hardware)
    ] ;

```

```

rdfs:range xsd:string .

default:Gerente_de_Mudança
  rdf:type owl:Class ;
  rdfs:comment "É um papel definido no contexto do MOF que tem to
tal responsabilidade sobre o processo de gerenciamento de mudan
ças numa organização de TI."^^xsd:string ;
  rdfs:subClassOf default:Gerente .

default:Identificador_da_Liberação
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Liberação)
    ] ;
  rdfs:range xsd:int .

default:Estaassociadaasliberacoes
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Mudança)
    ] ;
  rdfs:label "Está associada _a(s)_liberação(oes)"^^xsd:string .

default:Status
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Problema)
    ] ;
  rdfs:range xsd:string .

default:Gerente_de_Configuração
  rdf:type owl:Class ;
  rdfs:comment "É o papel no contexto do MOF que é responsável
por gerenciar as atividades do processo de configuração para
uma organização de TI. O papel também seleciona, delega responsa
bilidades e treina a equipe de gerência de configuração."^^xsd:
string ;
  rdfs:subClassOf default:Gerente .

default:Plano_definido_em
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Liberação)

```

```

    ] ;
    rdfs:label "Plano definido em"^^xsd:string .

default:GERENTE_EXECUTIVO_TI
    rdf:type owl:Class ;
    rdfs:comment "Gerente Responsável por assinar acordos de Nível
de Serviço pela TI (representa a TI perante a área negocial)
"^^xsd:string ;
    rdfs:label "GERENTE EXECUTIVO TI"^^xsd:string ;
    rdfs:subClassOf default:Gerente .

default:Acordo_Não-exclusivo
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
viço__ANS_)
        ] ;
    rdfs:label "Acordo Não-exclusivo"^^xsd:string ;
    rdfs:range xsd:string .

default:Se_necessário_RDM
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Problema default:Inci
dente)
        ] .

default:Garantias_e_Soluções
    rdf:type owl:Class ;
    rdfs:label "Garantias e Soluções"^^xsd:string ;
    rdfs:subClassOf default:Acordo_de_Nível_de_Serviço__ANS_ .

default:RDM_de_origem
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Liberação)
        ] ;
    rdfs:label "RDM de origem"^^xsd:string .

default:Descrição_do_Plano_de_Recuperação
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Plano_de_Recuperação)

```

```

    ] ;
    rdfs:label "Descrição do Plano de Recuperação"^^xsd:string ;
    rdfs:range xsd:string .

default:Termos_de_Pagamento
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
            viço__ANS_)
        ] ;
    rdfs:label "Termos de Pagamento"^^xsd:string ;
    rdfs:range xsd:string .

default:Performance__Acompanhamento_e_Relatório
    rdf:type owl:Class ;
    rdfs:label "Performance, Acompanhamento e Relatório"^^xsd:string ;
    rdfs:subClassOf default:Acordo_de_Nível_de_Serviço__ANS_ .

default:Análise_de_Risco
    rdf:type owl:Class ;
    rdfs:comment "Uma análise de risco pode ajudar a identificar os
    riscos a que um negócio está exposto. Ela proporcionará ao geren
    ciamento informações valiosas ao identificar por um lado as amea
    ças e vulnerabilidades e por outro as medidas preventivas perti
    nentes"^^xsd:string ;
    rdfs:label "Análise de Risco"^^xsd:string ;
    rdfs:subClassOf default:Gerenciamento_da_Continuidade .

default:Solução_Definitiva
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Problema)
        ] ;
    rdfs:label "Solução Definitiva"^^xsd:string ;
    rdfs:range xsd:string .

default:Programa_de_Qualidade_de_Serviço__PQS_
    rdf:type owl:Class ;
    rdfs:comment "Define os parametros dos processos do Gerenciamen
    to de Serviços e do gerenciamento operacional. O ANS define o
    que nós produzimos e o PQS está relacionado a maneira como nós
    produzimos, incluindo metas para cada processo, na forma de
    Indicadores de Desempenho."^^xsd:string ;
    rdfs:label "Programa de Qualidade de Serviço (PQS)"^^xsd:string ;
    rdfs:subClassOf default:Gerenciamento_do_Nível_de_Serviço .

```

```

default:Componentes_de_TI_envolvidos
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Análise_de_Risco)
    ] ;
  rdfs:label "Componentes de TI envolvidos"^^xsd:string .

default:Nome_do_Serviço_de_TI
  rdf:type owl:DatatypeProperty ;
  rdfs:comment "Campo com o nome do serviço de TI fornecido. Por
  exemplo: Monitoração de Servidores, Liberação de rede, Suporte
  a Impressoras, etc"^^xsd:string ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Serviço_de_TI__especi
        ficação_)
    ] ;
  rdfs:label "Nome do Serviço de TI"^^xsd:string ;
  rdfs:range xsd:string .

default:Taxas_Profissionais
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
        viço__ANS_)
    ] ;
  rdfs:label "Taxas Profissionais"^^xsd:string ;
  rdfs:range xsd:string .

default:Descrição_da_Ocorrencia
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Chamado)
    ] ;
  rdfs:label "Descrição da Ocorrencia"^^xsd:string ;
  rdfs:range xsd:string .

default:Responsavel
  rdf:type owl:Class ;
  rdfs:comment "Responsável pela execução de determinada ativida
  de (Ex.: Avaliação de Impacto)"^^xsd:string ;
  rdfs:subClassOf default:Cliente_Interno__Área_Tecnológica_ .

```

```

<http://www.owl-ontologies.com/unnamed.owl#Doc.Requerimentos_Libera_em>
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Requerimentos_Específi
        cos_Liberação)
    ] .

default:Requerimentos_Liberação
  rdf:type owl:ObjectProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Liberação)
    ] .

default:Código_do_Acordo_de_Nível_de_Serviço
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
        viço__ANS_)
    ] ;
  rdfs:label "Código do Acordo de Nível de Serviço"^^xsd:string ;
  rdfs:range xsd:string .

default:Descrição
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Problema default:Inci
        dente)
    ] ;
  rdfs:range xsd:string .

default:Gerenciamento_do_Problema
  rdf:type owl:Class ;
  rdfs:label "Gerenciamento do Problema"^^xsd:string ;
  rdfs:subClassOf default:Acordo_de_Nível_de_Serviço__ANS_ .

default:Força_Maior
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
        viço__ANS_)
    ] ;
  rdfs:label "Força Maior"^^xsd:string ;

```

```

    rdfs:range xsd:string .

default:Definições_Gerais
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
        viço__ANS_)
    ] ;
  rdfs:label "Definições Gerais"^^xsd:string ;
  rdfs:range xsd:string .

default:Gerenciamento_de_Mudança
  rdf:type owl:Class ;
  rdfs:comment "O objetivo do Gerenciamento de Mudanças é garantir
    o uso de métodos e procedimentos padronizados, de modo que as
    mudanças possam ser tratadas rapidamente, com o menor impacto
    possível sobre a qualidade do serviço. Todas as mudanças devem
    poder ser rastreadas."^^xsd:string ;
  rdfs:label "Gerenciamento de Mudança"^^xsd:string ;
  rdfs:subClassOf default:Concept .

default:Responsabilidades_e_Deveres_do_Consumidor
  rdf:type owl:Class ;
  rdfs:label "Responsabilidades e Deveres do Consumidor"^^xsd:string ;
  rdfs:subClassOf default:Acordo_de_Nível_de_Serviço__ANS_ .

default:Requisitos_de_Nível_de_Serviços
  rdf:type owl:DatatypeProperty ;
  rdfs:comment "Expecificações externas. Por exemplo: Disponibilida
    de 24 X 7."^^xsd:string ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Serviço_de_TI__especifi
        cação_)
    ] ;
  rdfs:label "Requisitos de Nível de Serviços"^^xsd:string ;
  rdfs:range xsd:string .

default:Soluções_para_rupturas
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
        viço__ANS_)
    ] ;
  rdfs:label "Soluções para rupturas"^^xsd:string ;

```

```

    rdfs:range xsd:string .

default:Descrição_do_RNS
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Requisitos_de_Nível_de_
            Serviços__RNS_)
        ] ;
    rdfs:label "Descrição do RNS"^^xsd:string ;
    rdfs:range xsd:string .

default:Direitos_de_Propriedade_Intelectual_e_Informação_Confidencial
    rdf:type owl:Class ;
    rdfs:label "Direitos de Propriedade Intelectual e Informação Confi
    dencial"^^xsd:string ;
    rdfs:subClassOf default:Acordo_de_Nível_de_Serviço__ANS_ .

default:Tipo_do_Chamado
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Chamado)
        ] ;
    rdfs:label "Tipo do Chamado"^^xsd:string ;
    rdfs:range xsd:string .

default:Serviços_Padronizados
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
            viço__ANS_)
        ] ;
    rdfs:label "Serviços Padronizados"^^xsd:string ;
    rdfs:range xsd:string .

default:Confiabilidade
    rdf:type owl:Class ;
    rdfs:comment "Confiabilidade adequada significa que o serviço esta
    disponível sem interrupções durante um período. Esse conceito inclui
    também a resiliencia. A confiabilidade de um serviço aumenta quando
    o downtime poder ser prevenido. A confiabilidade é calculada com
    o uso da estatística"^^xsd:string ;
    rdfs:subClassOf default:Gerenciamento_da_Disponibilidade .

default:membro_comite_mud

```

```

rdf:type owl:ObjectProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Comitê_de_Mudanças)
    ] .

default:Prioridade_Liberação
rdf:type owl:DatatypeProperty ;
rdfs:comment "alta, média, baixa."^^xsd:string ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Liberação)
    ] ;
rdfs:range xsd:string .

default:Solução_Contorno_aplicada_com_sucesso
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Incidente)
    ] ;
rdfs:label "Solução Contorno aplicada com sucesso"^^xsd:string ;
rdfs:range xsd:boolean .

default:Resolução_informal
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Adequação_Legal_e_Reso
        lução_dos_Impasses)
    ] ;
rdfs:label "Resolução informal"^^xsd:string ;
rdfs:range xsd:string .

default:Probabilidade_Ocorrencia_Desastre
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Análise_de_Risco)
    ] ;
rdfs:label "Probabilidade Ocorrencia Desastre"^^xsd:string ;
rdfs:range xsd:string .

default:Documentos_Específicos_Liberação
rdf:type owl:Class ;
rdfs:subClassOf default:Documento .

```

```

default:Acordo_de_Nível_Operacional__ANO_
  rdf:type owl:Class ;
  rdfs:comment "É um acordo com um departamento interno de TI e tem
  como objetivo detalhar o fornecimento de alguns elementos de um
  serviço."^^xsd:string ;
  rdfs:label "Acordo de Nível Operacional (ANO)"^^xsd:string ;
  rdfs:subClassOf default:Gerenciamento_do_Nível_de_Serviço .

default:Descrição_da_Liberação
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Liberação)
    ] ;
  rdfs:label "Descrição da Liberação"^^xsd:string ;
  rdfs:range xsd:string .

default:Código_do_CA
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Contrato_de_Apoio__CA_)
    ] ;
  rdfs:label "Código do CA"^^xsd:string ;
  rdfs:range xsd:string .

default:Problema
  rdf:type owl:Class ;
  rdfs:comment "Um problema define uma situação indesejável, indi
  cando a causa raiz desconhecida de um ou mais incidentes poten
  ciais ou existentes."^^xsd:string ;
  rdfs:subClassOf default:Gerenciamento_de_Problemas .

default:Acesso_Lógico
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
        viço__ANS_)
    ] ;
  rdfs:label "Acesso Lógico"^^xsd:string ;
  rdfs:range xsd:string .

default:Nome_do_Acordo_de_Nível_de_Serviço
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;

```

```

        owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Serviço__ANS_)
    ] ;
    rdfs:label "Nome do Acordo de Nível de Serviço"^^xsd:string ;
    rdfs:range xsd:string .

default:Capacidade_de_Armazenamento
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Capacidade_de_Recurso)
        ] ;
    rdfs:label "Capacidade de Armazenamento"^^xsd:string ;
    rdfs:range xsd:string .

default:Cliente
    rdf:type owl:Class ;
    rdfs:comment "É o representante de uma organização que está autorizado a em nome dessa organização fazer acordos sobre o uso de serviços de TI. Assim, ele NÃO É O USUÁRIO FINAL DOS SERVIÇOS DE TI."^^xsd:string ;
    rdfs:subClassOf default:Gerenciamento_do_Nível_de_Serviço .

default:Solicitante
    rdf:type owl:ObjectProperty ;
    rdfs:comment "Usuário afinal, Ciente Interno, Gestor do Negócio ou Provedor Externo (quem abre chamado)."^^xsd:string ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Chamado)
        ] .

default:Custos
    rdf:type owl:Class ;
    rdfs:subClassOf default:Gerenciamento_Financeiro .

default:Manuais
    rdf:type owl:Class ;
    rdfs:subClassOf default:Documento .

default:Solução_de_Contorno
    rdf:type owl:ObjectProperty ;
    rdfs:comment "Solução provisória cujo objetivo é a retomada do nível de serviço normal, tal como definido no ANS, o mais breve possível e com o menor impacto possível na atividade da organização e do usuário."^^xsd:string ;
    rdfs:domain

```

```

        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Incidente)
        ] ;
    rdfs:label "Solução de Contorno"^^xsd:string .

p5:_Nível
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Problema default:Chama
            do)
        ] ;
    rdfs:label "Suporte 3º Nível"^^xsd:string .

default:Atraso_de_Clientes
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
            viço__ANS_)
        ] ;
    rdfs:label "Atraso de Clientes"^^xsd:string ;
    rdfs:range xsd:string .

default:Assinaturas
    rdf:type owl:Class ;
    rdfs:subClassOf default:Acordo_de_Nível_de_Serviço__ANS_ .

default:Aprovações_e_Informações
    rdf:type owl:DatatypeProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
            viço__ANS_)
        ] ;
    rdfs:label "Aprovações e Informações"^^xsd:string ;
    rdfs:range xsd:string .

default:Descrição_da_Mudança
    rdf:type owl:ObjectProperty ;
    rdfs:domain
        [ rdf:type owl:Class ;
          owl:unionOf (owl:Thing default:Mudança)
        ] .

default:Reivindicação_de_Terceiros
    rdf:type owl:DatatypeProperty ;

```

```

rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
viço__ANS_)
    ] ;
rdfs:label "Reivindicação de Terceiros"^^xsd:string ;
rdfs:range xsd:string .

default:Encerramento_por_Conveniência
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Encerramento)
    ] ;
rdfs:label "Encerramento por Conveniência"^^xsd:string ;
rdfs:range xsd:string .

default:Acordo_Completo
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Geral)
    ] ;
rdfs:label "Acordo Completo"^^xsd:string ;
rdfs:range xsd:string .

default:Gerenciada_pelo
rdf:type owl:ObjectProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Mudança default:Libera
ção)
    ] ;
rdfs:label "Gerenciada pelo"^^xsd:string .

default:Acesso_Físico
rdf:type owl:DatatypeProperty ;
rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Ser
viço__ANS_)
    ] ;
rdfs:label "Acesso Físico"^^xsd:string ;
rdfs:range xsd:string .

default:Gerenciamento_de_Problemas
rdf:type owl:Class ;

```

```
rdfs:comment "O Gerenciamento de Problemas investiga a infra-estrutura e todas as informações disponíveis, inclusive o Banco de Dados de Incidentes, para identificar a causa subjacente de falhas reais e potenciais na oferta de serviço."^^xsd:string ;
rdfs:label "Gerenciamento de Problemas"^^xsd:string ;
rdfs:subClassOf default:Concept .
```

```
default:Código_do_ANO
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_Operacional__ANO_)
    ] ;
  rdfs:label "Código do ANO"^^xsd:string ;
  rdfs:range xsd:string .
```

```
default:Serviço_Afetado
  rdf:type owl:ObjectProperty ;
  rdfs:comment "O incidente irá afetar determinado serviço de TI"^^xsd:string ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Incidente default:Problema)
    ] ;
  rdfs:label "Serviço Afetado"^^xsd:string .
```

```
default:Avaliação_de_Impacto
  rdf:type owl:Class ;
  rdfs:comment "A medida de impacto da implantação de uma mudança em TI e nos negócios. A complexidade da mudança e recursos necessários, incluindo pessoas, dinheiro e tempo, são avaliados para se determinar a categoria. O risco da implantação também é um fator a ser considerado."^^xsd:string ;
  rdfs:label "Avaliação de Impacto"^^xsd:string ;
  rdfs:subClassOf default:Gerenciamento_de_Mudança .
```

```
default:Pagamentos_em_atraso
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nível_de_Serviço__ANS_)
    ] ;
  rdfs:label "Pagamentos em atraso"^^xsd:string ;
  rdfs:range xsd:string .
```

```
default:Departamento_de_TI_envolvido
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:Acordo_de_Nivel_Operaci
        onal__ANO_)
    ] ;
  rdfs:label "Departamento de TI envolvido"^^xsd:string ;
  rdfs:range xsd:string .
```

```
default:Categoria
  rdf:type owl:DatatypeProperty ;
  rdfs:domain
    [ rdf:type owl:Class ;
      owl:unionOf (owl:Thing default:IC_s)
    ] ;
  rdfs:range xsd:string .
```

Apêndice B

Ontologia de Negócio

```
; Thu Jan 10 22:58:18 GMT 2008
;
;+ (version "3.3")
;+ (build "Build 399")

(defclass %3ACLIPS_TOP_LEVEL_SLOT_CLASS "Fake class to save top-level
  slot information"
  (is-a USER)
  (role abstract)
  (single-slot Descrição+do+GEFIN
  ;+ (comment "É um canal de auto-atendimento eletrônico que oferece
prestação de serviços e disponibiliza transações bancárias diretamente
do microcomputador de sua empresa, aliando comodidade a máxima seguran
ça.")
  (type STRING)
  ;+ (cardinality 1 1)
  (create-accessor read-write))
  (single-slot GEFIN+formalizado+no+ANS
  (type INSTANCE)
  ;+ (allowed-classes ANS+GEFIN)
  ;+ (cardinality 0 1)
  (create-accessor read-write))
  (single-slot Limitações+TÉcnicas
  ;+ (comment "Exemplo: Máximo de 999 consultas/dia.\nNúmero de Aces
sos, etc.")
  (type STRING)
  ;+ (cardinality 0 1)
  (create-accessor read-write))
  (single-slot Número+de+recomendação
  ;+ (comment "Código identificador da recomendação da auditoria inter
na.")
  (type STRING)
  ;+ (cardinality 0 1)
```

```

(create-accessor read-write))
(single-slot Descrição+das+EstratÉgias
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Receita+por+serviço
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 03.+Auditabilidade+das+Informações
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Normas+Internas+que+regem+o+ANS
(type INSTANCE)
;+ (allowed-classes Livro+de+Normas+e+Regulamentos)
;+ (cardinality 0 1)
(create-accessor read-write))
(multislot Assinantes+do+ANS
(type INSTANCE)
;+ (allowed-classes GERENTE+EXECUTIVO+NEGÓCIO)
(create-accessor read-write))
(single-slot 11.+Plano+de+Contingência
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 10.+Treinamento
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 01.+Gerencia+de+Problemas+e+incidentes
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Indicadores+ANS
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 16.+Prazo
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Descrição+do+Serviço+de+Negócio
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Descrição+do+Produto+fornecido

```

```

(type STRING)
;+ (cardinality 1 1)
(create-accessor read-write))
(single-slot ANS+associado+ao+serviço
(type INSTANCE)
;+ (allowed-classes GERENCIADOR+FINANCEIRO)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Instruções+de+Utilização
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Categoria+%28renda%29
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Detalhes+da+Consulta
;+ (comment "Uma vez selecionado o tipo da consulta, pode se querer
uma consulta mais detalhada. Ex: Tipo: Consulta Sub tipo: consulta de
Poupança.")
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Número+de+Matrícula
(type STRING)
;+ (cardinality 1 1)
(create-accessor read-write))
(single-slot Tipo+de+Pessoa+Jurídica
(type STRING)
(default "Pequena" "Média" "Corporate")
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Código+do+Serviço
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Descrição+da+Norma
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 02.+Controle+de+Acesso
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Tipo+de+Consulta
;+ (comment "As consultas podem ser do tipo saldos, extratos e extra
tos de investimentos")

```

```

(type STRING)
(default "Saldos" "Extratos" "Extratos Investimento")
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 08.+Segurança+da+Informação
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Horário+de+Disponibilidade
;+ (comment "Cada serviço do Gerenciador Financeiro tem um horário
de Funcionamento")
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Normas+externas+que+regem+ANS
(type INSTANCE)
;+ (allowed-classes Normas+Externas)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Nome+do+Serviço+de+Negócio
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Código+da+Norma
;+ (comment "Número identificador da Norma")
(type STRING)
;+ (cardinality 1 1)
(create-accessor read-write))
(single-slot Descrição+do+Canal
;+ (comment "Descrição do canal pelo qual será oferecido o serviço.
Ex:Auto-Atendimento, agência,")
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot visão+de+negócio_Slot_69
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Número+do+ANS
(type STRING)
;+ (cardinality 1 1)
(create-accessor read-write))
(single-slot 12.+Manutenção+Corretiva+do+Sistema
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 14.+Condições+de+Garantia+de+Desempenho

```

```

(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Principais+Erros
;+ (comment "Ex.: ERRO HTTP: Servido inacessível")
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 05.+Monitoração+do+Ambiente
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot FAC+do+serviço
;+ (comment "Descrição das principais dúvidas, levantadas pelo usuá
rio a cerca do serviço.")
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 07.+Gerenciamento+da+Capacidade+de+Processamento
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Instrução+para+Erro
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Código+identificador+%28GEFIN%29
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 04.+Gerenciamento+de+Mudanças
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Descrição+do+Serviço
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Código+da+norma+interna
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Nome
(type STRING)
;+ (cardinality 1 1)
(create-accessor read-write))
(single-slot CPF%2FCNPJ

```

```

(type STRING)
;+ (cardinality 1 1)
(create-accessor read-write))
(single-slot Código+do+Serviço+de+Negócio
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 13.+Disponibilidade
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 15.++Alterações+no+Acordo
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 09.+Gerenciamento+dos+Dados
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Forma+Medição+ANS
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Grau+de+Satisfação
(type STRING)
(default "Ruim" "Bom" "Muito Bom" "Excelente")
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 06.+Gerenciamento+das+Operações
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Descrição+das+Políticas
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write)))

(defclass Pessoa
(is-a USER)
(role abstract)
(single-slot CPF%2FCNPJ
(type STRING)
;+ (cardinality 1 1)
(create-accessor read-write))
(single-slot Nome
(type STRING)
;+ (cardinality 1 1)

```

```

(create-accessor read-write)))

(defclass Funcionário "Pessoa que trabalha na organização mediante concurso. Te
(is-a Pessoa)
(role concrete)
(single-slot Número+de+Matrícula
(type STRING)
;+ (cardinality 1 1)
(create-accessor read-write)))

(defclass GERENTE+EXECUTIVO+NEGÓCIO "Gerente que assina juntamente com
a TI os ANS dos serviços prestados."
(is-a Funcionário)
(role concrete))

(defclass Analista+de+Negocio
(is-a Funcionário)
(role concrete))

(defclass Clientes
(is-a Pessoa)
(role concrete)
(single-slot Categoria+renda%29
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write)))

(defclass Pessoa+Física
(is-a Clientes)
(role concrete))

(defclass Pessoa+Jurídica
(is-a Clientes)
(role concrete))

(defclass Pequenas
(is-a Pessoa+Jurídica)
(role concrete))

(defclass Empresarial
(is-a Pessoa+Jurídica)
(role concrete))

(defclass Corporate
(is-a Pessoa+Jurídica)
(role concrete))

```

```
(defclass Governo
(is-a Clientes)
(role concrete))
```

```
(defclass Fornecedores
(is-a Pessoa)
(role concrete)
(single-slot Descrição+do+Produto+fornecido
(type STRING)
;+ (cardinality 1 1)
(create-accessor read-write)))
```

(defclass ANS+com+fornecedores "O ANS - Acordo de Nível de Serviços, também conhecido pela sigla SLA (do inglês Service Level Agreement), é um documento formal, celebrado entre o Contratante e o Fornecedor, geralmente anexo ao contrato. Sua função é definir o nível de qualidade dos serviços prestados pelo fornecedor. \nA fim de estabelecer uma relação de respeito com seus fornecedores, o Banco do Brasil adotou novos métodos para garantir a qualidade dos serviços. Serão estabelecidos Indicadores de Desempenho, como cumprimento de prazos e agilidade na solução de problemas, e formas de avaliação - que podem variar de acordo com o tipo do serviço prestado - para verificar se o fornecedor está ou não atingindo o nível esperado pelo contratante. \nAs medidas têm como objetivo o comprometimento mútuo em relação às obrigações definidas nos contratos. A manutenção das condições técnicas de habilitação, o aumento da qualidade nos serviços, o cumprimento da legislação vigente (nos âmbitos municipal, estadual e federal), a utilização de princípios éticos de concorrência, a segurança na utilização de produtos e serviços, a obediência às normas de Segurança no Trabalho, o compromisso com as questões sociais dentro e fora da empresa, a conscientização do impacto ambiental das ações do homem relacionadas ou não às atividades fins, incentivo à cultura, ao esporte e ao lazer, são alguns dos compromissos assumidos por ambas as partes. \nEntre os benefícios que a assinatura do ANS pode trazer, tanto para o contratante quanto para o contratado, podemos citar: \nA manutenção da qualidade dos serviços prestados e incentivo a constante busca pela Excelência; \nPossibilidade de controle da situação econômica, operacional e legal da empresa; \nConstrução da imagem de empresa que adota postura ética e socialmente responsável; \nO fato de todos os indicadores a serem observados estarem previamente acordados, com anuência de ambas as partes, minimiza o surgimento de discussões e impasses quanto aos serviços prestados, o que ajuda a manter um bom relacionamento. \nRentabilidade: O Banco do Brasil, como empresa de postura socialmente responsável, procura garantir a seus fornecedores um nível de rentabilidade que não comprometa sua situação econômico-financeira. \nApesar de ser mais comum entre as empresas de Tecnologia da Informação e Telecomunicações, o ANS pode ser aplicado a qualquer serviço, desde que os indicadores e formas de medição aplicados sejam customizados de acordo com o tipo de serviço

```

prestado."
(is-a Fornecedores)
(role concrete)
(single-slot Indicadores+ANS
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Forma+Medição+ANS
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write)))

(defclass Licitações "Licitações seguem a lei federal: 8666/93 (Licitações
e Contratos)"
(is-a Fornecedores)
(role concrete))

(defclass Conselho+Deliberativo+%28DIRETORIA%29
(is-a Pessoa)
(role concrete))

(defclass EstratÉgias
(is-a Conselho+Deliberativo+%28DIRETORIA%29)
(role concrete)
(single-slot Descrição+das+EstratÉgias
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write)))

(defclass Políticas
(is-a Conselho+Deliberativo+%28DIRETORIA%29)
(role concrete)
(single-slot Descrição+das+Políticas
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write)))

(defclass Serviço+de+Negócio
(is-a USER)
(role abstract)
(single-slot Código+do+Serviço+de+Negócio
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Nome+do+Serviço+de+Negócio
(type STRING)
;+ (cardinality 0 1)

```

```

(create-accessor read-write)))

(defclass Serviço+PJ
  (is-a Serviço+de+Negócio)
  (role concrete)
  (single-slot Tipo+de+Pessoa+Jurídica
    (type STRING)
    (default "Pequena" "Média" "Corporate"))
  ;+ (cardinality 0 1)
  (create-accessor read-write)))

(defclass Auto+Atendimento
  (is-a Serviço+PJ)
  (role concrete)
  (single-slot Descrição+do+Canal
    ;+ (comment "Descrição do canal pelo qual será oferecido o serviço.
    Ex:Auto-Atendimento, agência,")
    (type STRING)
    ;+ (cardinality 0 1)
    (create-accessor read-write)))

(defclass GERENCIADOR+FINANCEIRO "É um canal de auto-atendimento eletrô
nico que oferece prestação de serviços e disponibiliza transações bancá
rias diretamente do microcomputador de sua empresa, aliando comodidade
a máxima segurança."
  (is-a Auto+Atendimento)
  (role concrete)
  (single-slot Descrição+do+Serviço+de+Negócio
    (type STRING)
    ;+ (cardinality 0 1)
    (create-accessor read-write))
  (single-slot GEFIN+formalizado+no+ANS
    (type INSTANCE)
    ;+ (allowed-classes ANS+GEFIN)
    ;+ (cardinality 0 1)
    (create-accessor read-write)))

(defclass Consultas
  (is-a GERENCIADOR+FINANCEIRO)
  (role concrete)
  (single-slot Código+do+Serviço
    (type STRING)
    ;+ (cardinality 0 1)
    (create-accessor read-write))
  (single-slot Instrução+para+Erro
    (type STRING)
    ;+ (cardinality 0 1)

```

```

(create-accessor read-write))
(single-slot Horário+de+Disponibilidade
;+ (comment "Cada serviço do Gerenciador Financeiro tem um horário
de Funcionamento")
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Limitações+Técnicas
;+ (comment "Exemplo: Máximo de 999 consultas/dia.\nNúmero de Aces
sos, etc.")
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Descrição+do+Serviço
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Instruções+de+Utilização
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Grau+de+Satisfação
(type STRING)
(default "Ruim" "Bom" "Muito Bom" "Excelente")
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Tipo+de+Consulta
;+ (comment "As consultas podem ser do tipo saldos, extratos e extra
tos de investimentos")
(type STRING)
(default "Saldos" "Extratos" "Extratos Investimento")
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Detalhes+da+Consulta
;+ (comment "Uma vez selecionado o tipo da consulta, pode se querer
uma consulta mais detalhada. Ex: Tipo: Consulta Sub tipo: consulta de
Poupança.")
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Principais+Erros
;+ (comment "Ex.: ERRO HTTP: Servido inacessível")
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot FAC+do+serviço
;+ (comment "Descrição das principais dúvidas, levantadas pelo usuá

```

```

rio a cerca do serviço.")
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))

(defclass Transferencias
(is-a GERENCIADOR+FINANCEIRO)
(role concrete))

(defclass Pagamentos
(is-a GERENCIADOR+FINANCEIRO)
(role concrete))

(defclass Recebimentos
(is-a GERENCIADOR+FINANCEIRO)
(role concrete))

(defclass Investimentos
(is-a GERENCIADOR+FINANCEIRO)
(role concrete))

(defclass Empréstimos
(is-a GERENCIADOR+FINANCEIRO)
(role concrete))

(defclass Internacional
(is-a GERENCIADOR+FINANCEIRO)
(role concrete))

(defclass Outras+Opções
(is-a GERENCIADOR+FINANCEIRO)
(role concrete))

(defclass Pagamento+de+Salários
(is-a GERENCIADOR+FINANCEIRO)
(role concrete))

(defclass Transações+Pendentes
(is-a GERENCIADOR+FINANCEIRO)
(role concrete))

(defclass Gerenciamento+de+Usuário
(is-a GERENCIADOR+FINANCEIRO)
(role concrete))

(defclass Cartão
(is-a GERENCIADOR+FINANCEIRO)

```

```
(role concrete))

(defclass Transferencia+de+Arquivos
(is-a GERENCIADOR+FINANCEIRO)
(role concrete))

(defclass BB+Fone+FAX
(is-a Auto+Atendimento)
(role concrete))

(defclass Mercado+de+Capitais
(is-a Serviço+PJ)
(role concrete))

(defclass Soluções+em+Pagamento
(is-a Serviço+PJ)
(role concrete))

(defclass Pagamento+a+Fornecedores
(is-a Soluções+em+Pagamento)
(role concrete))

(defclass Pagamento+Salários
(is-a Soluções+em+Pagamento)
(role concrete))

(defclass Pagamento+Eletronico+de+Títulos
(is-a Soluções+em+Pagamento)
(role concrete))

(defclass Tributos
(is-a Soluções+em+Pagamento)
(role concrete))

(defclass Soluções+em+Recebimento
(is-a Serviço+PJ)
(role concrete))

(defclass Débito+Programado
(is-a Soluções+em+Recebimento)
(role concrete))

(defclass Depósito+Identificado
(is-a Soluções+em+Recebimento)
(role concrete))

(defclass Cobrança+Integrada
```

```

(is-a Soluções+em+Recebimento)
(role concrete))

(defclass Arrecadação
(is-a Soluções+em+Recebimento)
(role concrete))

(defclass Bloqueto+de+Cobrança
(is-a Serviço+PJ)
(role concrete))

(defclass Serviço+PF
(is-a Serviço+de+Negócio)
(role concrete))

(defclass Serviço+Governo
(is-a Serviço+de+Negócio)
(role concrete))

(defclass Serviços+Sociais%2FComunitários
(is-a Serviço+de+Negócio)
(role concrete))

(defclass Inclusão+Digital
(is-a Serviços+Sociais%2FComunitários)
(role concrete))

(defclass Desenvolvimento+Regional+Sustentável
(is-a Serviços+Sociais%2FComunitários)
(role concrete))

(defclass ACORDO+DE+N%C3%8DVVEL+DE+SERVI%C3%870 "Acordo de nível de servi
ço acordado realizado entre a TI e a Área de Negócio."
(is-a Serviço+de+Negócio)
(role concrete))

(defclass ANS+GEFIN
(is-a ACORDO+DE+N%C3%8DVVEL+DE+SERVI%C3%870)
(role concrete)
(single-slot 07.+Gerenciamento+da+Capacidade+de+Processamento
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 08.+Segurança+da+Informação
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))

```

```

(single-slot Normas+externas+que+regem+ANS
(type INSTANCE)
;+ (allowed-classes Normas+Externas)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 04.+Gerenciamento+de+Mudanças
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 03.+Auditabilidade+das+Informações
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot Normas+Internas+que+regem+o+ANS
(type INSTANCE)
;+ (allowed-classes Livro+de+Normas+e+Regulamentos)
;+ (cardinality 0 1)
(create-accessor read-write))
(multislot Assinantes+do+ANS
(type INSTANCE)
;+ (allowed-classes GERENTE+EXECUTIVO+NEGÓCIO)
(create-accessor read-write))
(single-slot 11.+Plano+de+Contingência
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 13.+Disponibilidade
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 15.++Alterações+no+Acordo
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 10.+Treinamento
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 09.+Gerenciamento+dos+Dados
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 12.+Manutenção+Corretiva+do+Sistema
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 01.+Gerencia+de+Problemas+e+incidentes

```

```

(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 14.+Condições+de+Garantia+de+Desempenho
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 16.+Prazo
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 02.+Controle+de+Acesso
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 06.+Gerenciamento+das+Operações
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
(single-slot 05.+Monitoração+do+Ambiente
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write)))

```

```

(defclass Orçamento "O orçamento envolve a previsão de custos e o contro
le deos gastos. Frequentemente o seu ponto de partida é a preparação de
um plano com a demanda prevista do cliente para os serviços e os custos
a eles associados.\nO objetivo do orçamento é planejar e controlar as
atividades de uma organização. O planejamento corporativo e estratégico
diz respeito aos objetivos a longo prazo de um negócio. Os orçamentos
definem os planos financeiros para os objetivos durante o período cober
to pelo orçamento. Esse período vaira de 1 a 5 anos."
(is-a USER)
(role concrete))

```

```

(defclass Receitas "Receitas obtidas com os serviços oferecidos."
(is-a Orçamento)
(role concrete))

```

```

(defclass Despesas "Despesas provenientes dos serviços fornecidos."
(is-a Orçamento)
(role concrete))

```

```

(defclass Retorno+Operacional+de+Investimento+%28ROI%29 "Retorno operaci
onal de investimentos realizados na área de TI (por exemplo)"
(is-a Orçamento)
(role concrete))

```

```

(defclass Lucros
(is-a Orçamento)
(role concrete))

(defclass Regras++Internas
(is-a USER)
(role concrete))

(defclass Livro+de+Normas+e+Regulamentos
(is-a Regras++Internas)
(role concrete)
(single-slot Código+da+norma+interna
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write)))

(defclass AUDITORIA "Realização de Auditoria Interna. Essa classe elabo
ra recomendações com relação a formalização dos serviços. Ex: recomenda
ção 2004/XXXXX para a formalização do Acordo de Nível de Serviço do GEFIN."
(is-a Regras++Internas)
(role concrete)
(single-slot Número+de+recomendação
;+ (comment "Código identificador da recomendação da auditoria inter
na.")
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write)))

(defclass Órgãos+Reguladores+%28Regras+Externas%29 "Por se tratar de
uma sociedade de economia mista está sujeito a acionistas e ao governo."
(is-a USER)
(role concrete))

(defclass BACEN
(is-a Órgãos+Reguladores+%28Regras+Externas%29)
(role concrete))

(defclass Normas+Externas "Normas expedidas pelo BACEN. Ex: conforme
define circular número XXXX de XX.XX.2001 será de 99,8%"
(is-a BACEN)
(role concrete)
(single-slot Código+da+Norma
;+ (comment "Número identificador da Norma")
(type STRING)
;+ (cardinality 1 1)
(create-accessor read-write))

```

```
(single-slot Descrição+da+Norma
(type STRING)
;+ (cardinality 0 1)
(create-accessor read-write))
```

```
(defclass AUDITORIA+EXTERNA
(is-a Órgãos+Reguladores+%28Regras+Externas%29)
(role concrete))
```

Referências

- Abdala, M. A. D. and Sant'Anna, N. (2003). Modelagem do processo de gerenciamento da configuração de software para um ambiente integrado. In *V Simpósio Internacional de Melhoria de Processo de Software (SIMPROS'03)*, pages 50–60, Recife, Brazil. Citeseer. Disponível em: http://hermes2.dpi.inpe.br:1905/col/lac.inpe.br/worcap/2003/10.31.14.52/doc/Abdala_Workcap3.pdf. Acessado em: Dezembro 2007.
- Almeida, M. B. (2003). Uma visão geral sobre ontologias: pesquisa sobre definições, tipos, aplicações, métodos de avaliação e de construção. *Ciência da Informação*, 32(3):7–20.
- Araújo, M. (2006). *Educação a distância e a Web Semântica: modelagem ontológica de materiais e objetos de aprendizagem para a plataforma COL.2006*. São Paulo, Escola Politécnica da Universidade de São Paulo. Disponível em: www.teses.usp.br. Acessado em: Maio 2006.
- Arpírez, J. C., O. Corcho, and Fernández, L. M. (2001). Webode: a scalable workbench for ontological engineering. In *Proceedings of the international conference on Knowledge*, pages 6–13. ACM.
- Ashpole, B., Ehrig, M., Euzenat, J., and Stuckenschmidt, H. (2005). FOAM - Framework for Ontology Alignment and Mapping. In *Proceedings of the Workshop on Integrating Ontologies*, pages 72–76.
- Bass, L., Clements, P., and Kazman, R. (2003). *Software Architecture in Practice*. Addison-Wesley Longman Publishing Company, Boston, USA.
- Borst, W. N. (1997). Construction of engineering ontologies. University of Twente, England. Disponível em: <http://doc.utwente.nl/fid/1392>. Acessado em: Abril 2006.
- Brockmans, S., Colomb, R., Kendall, E., E. Wallace, Chris, C., and Tong, G. (2006). A Model Driven Approach for Building OWL DL and OWL Full Ontologies. In *In: The 5th International Semantic Web Conference. (LNCS)*, Athens/GA, USA. Springer. Disponível em: <http://www.aifb.uni-karlsruhe.de/WBS/sbr/publications/omgodm.pdf>. Acessado em: Janeiro 2007.
- Campos, M. L. A. (2004). Modelização de domínios de conhecimento: uma investigação de princípios fundamentais. *Ciência da Informação*, 33(1):22–32.

- Castro, J. (2007). Governança de TI na Gestão Pública. *Revista Webinsider*, 5(1).
- Chalupsky, H. (2000). Ontomorph: A Translation System for Symbolic Knowledge. In *Proceedings of the Seventh International Conference (KR2000)*, San Francisco, CA. IOS Press.
- Clements, P. C. (1996). From Subroutines to Subsystems: Component-Based Software Development. In Brown, A. W., editor, *Component-Based Software Engineering: Selected Papers from the Software Engineering Institute*, pages 3–6. IEEE Computer Society Press. Disponível em: citeseer.ist.psu.edu/clements95from.html. Acessado em: Dezembro 2007.
- Corporation, M. (2000). Microsoft Operational Framework MOF. Disponível em: <http://www.microsoft.com/>. Acessado em: Setembro 2006.
- Corporation, P. (2006). The Business of SOA: evolving an Agile Enterprise with service oriented architecture. White Paper. Disponível em: http://www.inst-informatica.pt/servicos/informacao-e-documentacao/biblioteca-digital/soa/The_Business_Case_for_SOA.pdf/view. Acessado em: Novembro 2006.
- Coutinho, M. J. (2004). Um estudo sobre o desenvolvimento orientado a services. Undergraduate Work at the PUC-RIO.
- Croteau, A.-M. and Bergeron, F. (2006). An information technology trilogy: business, strategy, technological deployment and organizational performance. *Journal of Strategic Information Systems*, 10(2):77–99.
- DeLone, W. and McLean, E. (1992). Information and Management. *Journal of Strategic Information Systems*, 3(1):60–95.
- Doan, A., Madhavan, J., Domingos, P., and Halevy, A. (2002). Learning to map between ontologies on the semantic web. In *The 11th International WWW Conference*, pages 49–64, Hawaii, US. Citeseer.
- Domingues, J., Motta, E., Shum, S., Vargas-Vera, M., Kalfoglou, Y., and Farnes, N. (2001). Supporting Ontology Driven Document Enrichment within Communities of Practice. In *Proceedings of the international conference on Knowledge Capture*, pages 30–37, Victoria, British Columbia, Canada. ACM.
- Dou, D., McDermott, D., and Qi, P. (2003). Ontology translation on the semantic web. In *International Conference on Ontologies, Databases and Applications of Semantics*, pages 35–38. Springer.
- Ehrig, M. and Staab, S. (2004). QOM - Quick Ontology Mapping. In *Proceedings of the 3rd International Semantic Web Conference*, pages 683–697, Hiroshima, Japan. Springer. Disponível em: <http://www.aifb.uni-karlsruhe.de/WBS/meh/publications/ehrig04QOMTR.pdf>. Acessado em: Dezembro 2007.

- Euzenat, J. (2004). D2.2.3: State of the art on ontology alignment. Disponível em: <http://www.starlab.vub.ac.be/research/projects/knowledgeweb/kweb-223.pdf> . Acessado em: Dezembro 2007.
- Farquhar, A., Fikes, R., and Rice, J. (1996). The Ontolingua server: A tool for collaborative ontology construction. Technical report KSL-92-96, Stanford. Disponível em: citeseer.ist.psu.edu/farquhar96ontolingua.html. Acessado em: Novembro 2007.
- Gasevic, D., Djuric, D., and Devedzic, V. (2006). MDA-based Automatic OWL Ontology Development. *International Journal on Software Tools for Technology Transfer (STTT)*, 9(2).
- Gruber, T. (2005). What is an ontology? Disponível em: www-ksl.stanford.edu/kst/what-is-an-ontology.html. Acessado em: Agosto 2007.
- Grüninger, M. and Fox, M. (1995). Methodology for the Design and Evaluation of Ontologies. In *Proceedings of the IJCAI'95, Workshop on Basic Ontological Issues in Knowledge Sharing*, Montreal. Disponível em: <http://citeseer.ist.psu.edu/grninger95methodology.html>. Acessado em: Novembro 2007.
- Guarino, N. (1998). Formal Ontology and Information Systems. In *Proceedings of the International Conference On Formal Ontology In Information Systems (FOIS'98)*, pages 3–15, Trento, Italy. IOS Press.
- Guarino, N. and Welty, C. (2002). Evaluating ontological decisions with Ontoclean. *Communications ACM*, 45(2):61–65.
- Heijst, G. V., Schreiber, A. T., and Wielinga, B. T. (1997). Using Explicit Ontologies in KBS Development. *International Journal of Human-Computer Studies*, 46(1):183–192.
- Herborn, T. and Wimmer, M. A. (2006). Process Ontologies Facilitating Interoperability in e-Government - A Methodological Framework. In *Proceedings of the 3rd European Semantic Web Conference, Workshop on Semantics for Business Process Management (SBPM'06)*.
- Hovy, E. (2000). Combining and standardizing largescale, practical ontologies for machine translation and other uses. In *First International Conference on Language Resources and Evaluation (LREC)*, pages 535–542, Granada, Spain. Springer. Disponível em: <http://www.isi.edu/natural-language/people/hovy/papers/98LREC-ontol-align.pdf>. Acessado em: Dezembro 2007.
- Institute, I. G. (2007). *COBIT Control Practices: Guidance to Achieve Control Objectives for Successful IT Governance*. Information Systems Audit and Control Association (ISACA), Irvine, CA, USA.
- itSMF (2006). *Introdução ao ITIL*. São Paulo, The IT Service Management Forum.

- Jacobson, I., Booch, G., and Rumbaugh, J. (1999). *The unified software development process*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- Kaifoglou, Y. (2001). *Deploying Ontologies in Software Design*. PhD thesis, University of Amsterdam. Disponível em: <http://www.ecs.sotom.ac.uk/~yk1/research.html>. Acessado em: Dezembro 2006.
- Kielmann, T. (1996). Designing a Coordination Model for Open Systems. In Ciancarini, P. and Hankin, C., editors, *Proceedings of the 1st International Conference on Coordination Models and Languages*, volume 1061, pages 267–284, Cesena, Italy. Springer-Verlag, Berlin. Disponível em citeseer.ist.psu.edu/kielmann96designing.html. Acessado em: Outubro 2006.
- López, F. (1999). Overview of the methodologies for building ontologies. In *Proceedings of the IJCAI-99, Workshop on Ontologies and Problem-Solving Methods (KRR5)*, Stockholm, Sweden. Disponível em: citeseer.ist.psu.edu/lpez99overview.html. Acessado em: Julho 2007.
- Luftman, J. (2001). Assessing Business-IT Alignment Maturity. In *Strategic Information Tecnology: Opportunities for Competitive Advantage*, pages 105–133. Idea Group Publishing.
- Maedche, A., Schnur, H. P., Staab, S., and Stder, R. (1997). Representation language-neutral modeling of ontologies. In *Proceedings of the 2nd International Conference on Cooperative Information Systems (IFCIS'97)*. Disponível em: <http://citeseer.ist.psu.edu/272805.html>. Acessado em: Dezembro 2007.
- Mahalingam, K. and Huhns, M. N. (1997). An ontology tool for query formulation in an agent-based context. In *COOPIS '97: Proceedings of the Second IFCIS International Conference on Cooperative Information Systems*, pages 170–178, Washington, DC, USA. IEEE Computer Society.
- Maier, A., Schnurr, H., and Sure, Y. (2002). Ontoly-based Process Support. In *Proceedings of the 2nd International Semantic Web*, Stanford University, Dept. Computer Science. Disponível em: <http://citeseer.ist.psu.edu/198830.html>. Acessado em: Setembro 2007.
- McGuinness, D. L., Fikes, R., Rice, J., and Wilder, S. (2000). An environment for merging and testing large ontologies. In *Proceedings of the 7th International Conference (KR2000)*, pages 483–493, Colorado, USA. citeulike. Disponível em: <http://dit.unitn.it/~accord/RelatedWork/Matching/McGuinnessKR.pdf>. Acessado em: Novembro 2007.
- McGuinness, D. L. and van Harmelen, F. (2004). OWL Web Ontology Language Overview. Editors' Draft W3C. Disponível em: <http://www.w3.org/tr/2004/REC-owl-features-20040210/>. Acessado em: Dezembro 2007.

- Meijler, T. D. and Nierstrasz, O. (1997). Beyond Objects: Components. In Papazoglou, M. P. and Schlageter, G., editors, *Cooperative Information Systems: Current Trends and Directions*, pages 49–78. Academic Press. Disponível em citeseer.ist.psu.edu/meijler98beyond.html. Acessado em: Dezembro 2007.
- Mena, E. (1996). An Approach for Query Processing in Global Information Systems based on Interoperation across Pre-existing Ontologies. In *Proceedings of the 1st International Cooperative Information Systems (IFCIS'96)*, pages 14–25, Brussels, Belgium. IOS Press. Disponível em: <http://citeseer.ist.psu.edu/311175.html>. Acessado em: Dezembro 2007.
- Miller, E. (1998). An Introduction to the Resource Description Framework. *D-Lib Magazine*, 5(1):47–57. Disponível em: <http://www.dlib.org/dlib/may98/miller/05miller.html>. Acessado em: Dezembro 2007.
- Miller, E. (2005). Semantic web. Disponível em: <http://www.w3.org/2001/sw/>. Acessado em: Dezembro 2005.
- Mitra, Wiederhold, G., and Kersten, M. L. (2000). A graph-oriented model for articulation of ontology interdependencies. In *Proceedings of the 7th International Conference on Extending Database Technology*, pages 86–100, London, UK. Springer-Verlag.
- Mizoguchi, R., Vanwelkernhuysenm, J., and Ikeda, M. (1995). Task ontology for reuse of problem knowledge. In *Proceedings of the 1st European Conference on Artificial Intelligence (ECAI'94) Towards Very Large Knowledge Bases*, pages 46–59, Amsterdam, Netherland. IOS Press.
- Niemann, B., Morris, R., Riofrio, H. J., and Carnes, E. (2005). Introducing Semantic Technologies and the vision of the Semantic Web. Semantic Interoperability (XML Web Services) Community of Practice (SICoP). Disponível em: <http://web-services.gov/>. Acessado em: Março 2007.
- Niessink, F. and van Vilet, H. (1999). Towards mature IT services. *Software Process: Improvement and Practice*, 4(2):55–71.
- Novello, T. C. (2003). Ontologias, Sistemas Baseados em Conhecimento e Modelos de Banco de Dados. In *Seminário apresentado na Disciplina de Tópicos Avançados em Banco de Dados*, UFRGS, Brasil. Editora Moderna.
- Noy, N., Fergerson, R., and Musen, M. (2000a). The knowledge model of Protege-2000: Combining interoperability and flexibility. In *Proceedings of the 12th European Workshop on Knowledge*, Stanford, CA. Springer.
- Noy, N., Fergerson, R., and Musen, M. (2000b). PROMPT: Algorithm and tool for automated ontology merging and alignment. In *Seventeenth National Conference on Artificial Intelligence (AAAI-2000)*, pages 450–455, Austin, TX. Disponível em: citeseer.ist.psu.edu/noy00prompt.html. Acessado em: Dezembro 2007.

- Noy, N. and McGuinness, D. L. (2001). *Ontology Development 101: A Guide to Creating Your First Ontology*. Working Paper KSL-01-05, Stanford Knowledge Systems Laboratory.
- Noy, N. and Musen, M. (2001). Anchor-PROMPT: Using non-local context for semantic matching. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'01) workshop on ontology and information sharing*, pages 63–70, Seattle, WA. Disponível em: <http://dit.unitn.it/~accord/RelatedWork/Matching/noy.pdf>. Acessado em: Dezembro 2007.
- Noy, N. and Musen, M. A. (2003). The PROMPT Suite: Interactive Tools For Ontology Merging and Mapping. In *International Journal of Human-Computer Studies*, pages 983–1024. Disponível em: citeseer.ist.psu.edu/noy03prompt.html. Acessado em: Dezembro 2007.
- Paley, S. and Karp, J. D. (1997). A Generic Knowledge-Base Browser and Editor. In *Proceedings of the 9th Innovative Applications of Artificial Intelligence Conference (AAAI'97)*. Disponível em: <ftp://ftp.ksl.stanford.edu/pub/KSL.Reports/KSL-97-11.pdf.gz>. Acessado em: Dezembro 2007.
- Porto, R. M. and Bruggemann, E. S. (2006). Uma Proposta Ontológica para um Sistema de Gestão de Versionamento do Modelo ITIL. Undergraduate Work at the University of Brasília.
- Rus, I. and Lindvall, M. (2003). Knowledge Management in Software Engineering. *Journal of Knowledge Management*, 19(3):26–38.
- Sample, N., Beringer, D., Melloul, L., and Wiederhold, G. (1999). CLAM: Composition Language for Autonomous Megamodules. In *Proceedings of the 3rd International Conference Coordination Languages and Models (COORDINATION'99)*, Amsterdam, The Netherlands.
- Sawy, O. A. E. (2002). *Redesigning Enterprise Process for e-business*. McGraw-Hill.
- Schneider, J., Nierstrasz, O., Achermann, F., Lumpe, M., and McIlroy, M. (1999). Components, Scripts and Glue. In *Software Architecture Advances and Applications*, pages 13–25. Springer Verlag.
- Shum, A. and Dhillon, A. (2005). Achieve IT Agility by Integrating SOA with ITIL based BSM. Business Service, Butler, 2005. Disponível em: <http://whitepapers.zdnet.com/whitepaper.aspx?docid=166935>. Acessado em: Dezembro 2007.
- Softex (2006). MPS.BR - Melhoria de Processo do Software Brasileiro. Guia Geral - Versão 1.1, Disponível em: http://cps.softex.br/noticia_interna.php?id=615. Acessado em: Outubro 2007.
- Sordi, J. O. D. (2003). *Tecnologia da Informação Aplicada aos Negócios*. Editora Atlas, São Paulo.

- Stevens, R., Aranguren, M., Wolstencroft, K., Sattler, U., Drummont, N., Horridge, M., and Rector, A. (2002). Using Explicit Ontologies in KBS Development. *International Journal of Human and Computer Studies*, 65(7):583–594.
- Tetlow, P., Pan, J., Oberle, D., Wallace, E., Uschold, M., and Kendall, E. (2005). Ontology driven architectures and potential uses of the semantic web in software engineering. Editors’ Draft W3C. Disponível em: <http://www.w3.org/2001/sw/BestPractices/SE/ODA/>. Acesso em: junho de 2005.
- Uschold, M. and Grüninger, M. (1996). Ontologies: principles, methods, and applications. *Knowledge Engineering Review*, 11(2):93–155. Disponível em: citeseer.ist.psu.edu/uschold96ontology.html. Acessado em: Dezembro 2007.
- van der Raadt, B., Hoorn, J. F., and van Vliet, H. (2005). Alignment and Maturity Are Siblings in Architecture Assessment. In *Proceedings of the 17th Conference on Advanced Information Systems Engineering (CAiSE’05)*, pages 357–371.
- van Rijsbergen, C. J. (1975). *Information Retrieval*. Butterworths, MA, USA.
- van Vliet, H., Roeller, R., and Lago, P. (2006). Recovering Architectural Assumption Method. *Journal of Systems and Software*, 79(4):552–573. Disponível em: <http://portal.acm.org/citation.cfm?id=1142766>. Acessado em: Dezembro 2007.