



**UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
DEPARTAMENTO DE INFORMÁTICA
MESTRADO EM INFORMÁTICA**

WAGNER DE ANDRADE PERIN

***iMap* – UM MECANISMO DE INFERÊNCIA PARA
MAPAS CONCEITUAIS**

VITÓRIA, AGOSTO DE 2014

WAGNER DE ANDRADE PERIN

***iMap* – UM MECANISMO DE INFERÊNCIA PARA
MAPAS CONCEITUAIS**

Dissertação submetida ao Programa de Pós-Graduação em Informática da Universidade Federal do Espírito Santo como requisito parcial para a obtenção do grau de Mestre em Informática, na área de concentração em Inteligência Computacional.

VITÓRIA, AGOSTO DE 2014

Dados Internacionais de Catalogação-na-publicação (CIP)
(Biblioteca Setorial Tecnológica,
Universidade Federal do Espírito Santo, ES, Brasil)

P455i Perin, Wagner de Andrade, 1985-
iMap: um mecanismo de inferência para mapas conceituais /
Wagner de Andrade Perin. – 2014.
120 f. : il.

Orientador: Davidson Cury.
Coorientador: Crediné Silva de Menezes.
Dissertação (Mestrado em Informática) – Universidade
Federal do Espírito Santo, Centro Tecnológico.

1. Mapas conceituais. 2. Sistemas especialistas
(Computação). 3. Inteligência artificial. 4. PROLOG (Linguagem
de programação de computador). I. Cury, Davidson. II.
Menezes, Crediné Silva de. III. Universidade Federal do Espírito
Santo. Centro Tecnológico. IV. Título.

CDU: 004

WAGNER DE ANDRADE PERIN

***iMap* – UM MECANISMO DE INFERÊNCIA PARA
MAPAS CONCEITUAIS**

Dissertação submetida ao Programa de Pós-Graduação em Informática da Universidade Federal do Espírito Santo como requisito parcial para a obtenção do grau de Mestre em Informática, na área de concentração em Inteligência Computacional.

Aprovada em 29 de Agosto de 2014.

COMISSÃO EXAMINADORA

Prof. Dr. Davidson Cury
Universidade Federal do Espírito Santo (UFES)
(Orientador)

Prof. Dr. Crediné Silva Menezes
Universidade Federal do Espírito Santo (UFES)
(Co-orientador)

Prof. Dr. Orivaldo de Lira Tavares
Universidade Federal do Espírito Santo (UFES)

Prof. Dr. Tania Barbosa Salles Gava

Prof. Dr. Rosane Aragón

VITÓRIA, AGOSTO DE 2014

DEDICATÓRIA

Dedico este trabalho a você, Camila, companheira no amor, na vida, nos sonhos e realizações. Pelo apoio nas horas difíceis e pelos momentos de alegria quando eram extremamente necessários, e principalmente por me ensinar tanto, todos os dias, sobre o funcionamento da vida e do amor.

AGRADECIMENTOS

*“[...]Pois seja o que vier, venha o que vier
Qualquer dia, amigo, eu volto a te encontrar
Qualquer dia, amigo, a gente vai se encontrar[...]*”

(Milton Nascimento)

“Canção da América”

Como esquecer-me daqueles que tornaram um sonho realidade? Como esquecer-me daqueles que viram na minha felicidade sua verdadeira fonte de alegria? Certamente, eu estaria sendo injusto se não fizesse menção dos seguintes importantes “~~meliantes~~” companheiros que compartilharam comigo esse trajeto.

Agradeço, em primeiro lugar, ao meu orientador, professor e amigo, Dr. Davidson Cury (vulgo *Dede*). Mais que pelo incentivo, pela disponibilidade e pela “desorientação”, eu agradeço por ter confiado em mim e acreditado na minha capacidade e no meu trabalho. Estou até agora tentando contar quantas foram as portas que você abriu pra mim! Vai ser difícil retribuir por tudo o que você fez por mim... Poucas vezes na vida senti tamanha confiança e são também poucas as pessoas com o dom de colocar pra cima até mesmo pessoas que desacreditam de si próprias: parabéns por esse dom divino! (Risos).

Agradeço também ao meu co-orientador, o professor Dr. Crediné, por sua capacidade insuperável de levantar questões que “desorientariam” qualquer pesquisador e que, por esse motivo, colaborou tanto com meu aprendizado nessa trajetória.

Aos professores membros da Banca Examinadora, por terem atendido ao convite para desempenharem este papel, dispondo, cada um, de seu tempo, paciência e conhecimentos para analisarem e contribuírem com esta pesquisa.

A minha esposa, Camila Mesquita Chisté Perin, pelo amor, carinho, compreensão e companheirismo até este ponto. Principalmente por ter aberto mão de boa parcela de tempo (que hoje entendo como a coisa mais valiosa que temos a oferecer) que foi necessário para o desenvolvimento desta pesquisa.

Aos meus pais, Eloi Perin e Therezinha D. de Andrade Perin, e também à minha avó, Idalina. Todos vocês que mesmo sem saberem, foram, e continuam sendo, os principais colaboradores no meu processo de aprendizagem.

Aos meus amigos, alguns desenvolvidos e outros reaproximados ao longo deste curso, por compartilharem alegrias, dificuldades e conquistas na realização deste. Arriscarei em mencionar alguns que foram, digamos, marcantes:

- Roberto Moratti: pare com suas ideias proprietárias meu amigo!
- Vinicius Barcellos: nosso artigo do Rede Virtual vai sair quando?
- Maison Melotti: maneira sua banda! :)
- Geraldo Vassoler: vai uma “escudo” aí?
- Daniel de Assis: por favor... não manche minha reputação! =>
- Existem muitos outros que foram importantes, se você não está aqui, saiba que nossas boas recondações não ficam em papéis guardados em gavetas ou prateleiras! ;)

Aos colegas de trabalho, especialmente àqueles que sempre se disponibilizaram a acertos necessários de horário para que eu pudesse cumprir com todas as minhas obrigações profissionais e acadêmicas.

A todos os meus amigos e amigas que, direta ou indiretamente, sempre estiveram presentes me aconselhando e incentivando. Hoje sei como os amigos são fundamentais para nos orientar e ajudar a fazer boas escolhas.

Por ultimo, tenho que agradecer o apoio dado por pessoas que me atenderam tão bem quando tive “problemas técnicos” com soluções adotadas no protótipo do *iMap* e do *CMPaaS*. Um agradecimento especial para:

- Julien Dupont e Frédéric Madiot (*ObeoDesigner*): pelas licenças acadêmicas concedidas e pela paciência em me ajudar a solucionar problemas enfrentados na “operação” desta ferramenta.
- Walter (*GoJS*): Pela grande atenção, prontidão e velocidade em me ajudar a solucionar problemas no editor que está sendo preparado para o *CMPaaS*.

*“[...]Hoje me sinto mais forte, mais feliz quem sabe,
Só levo a certeza de que muito pouco eu sei,
Ou nada sei[...]*”

Almir Sater

RESUMO

Mapas Conceituais são representações gráficas do conhecimento de uma pessoa num dado momento e área de conhecimento. Por sua natureza investigativa, são utilizados como ferramentas de apoio em abordagens pedagógicas que objetivam promover a aprendizagem significativa. No entanto, o processo de avaliação de um mapa tende a ser custoso pois acarreta uma pesada carga de processamento cognitivo por parte do avaliador, já que este precisa mapear os conceitos e relações em busca de nuances de conhecimento ali presentes.

Essa pesquisa tem por objetivo aumentar o nível de abstração nas interações entre o avaliador e os mapas conceituais fornecendo uma camada intermediária de inteligência computacional que favoreça a comunicação por meio de perguntas e respostas em linguagem natural, fornecendo ao avaliador ferramentas que lhe permita examinar o conteúdo do mapa conceitual sem exigir deste o mapeamento visual dos conceitos e relações presentes nos mapas avaliados.

Uma ferramenta é prototipada e uma prova de conceito apresentada. A análise da arquitetura proposta permitiu definir uma arquitetura final com características que permitem potencializar o uso de mapas conceituais e facilitar diversas operações pedagógicas com estes.

Essa pesquisa situa-se na área de investigação de sistemas de perguntas e resposta, aplicando técnicas de processamento de linguagem natural para análise da pergunta e interpretação do mapa conceitual e aplica técnica de inteligência artificial para inferir respostas às perguntas.

Palavras-chave: Mapas Conceituais, Sistemas Especialistas, Sistemas de Perguntas e Respostas, Avaliação da Aprendizagem, Inteligência Artificial, Prolog.

ABSTRACT

Concept maps are graphical representations of a person's knowledge at a given time and area of expertise. For his investigative nature, they are used as tools to support pedagogical approaches that aim to promote meaningful learning. However, the maps evaluation process tends to be costly because it causes a heavy load of cognitive processing on the part of the evaluator, since it needs to map out the concepts and relations in order to find nuances of knowledge present there.

This research aims to increase the level of abstraction in the interactions between the evaluator and the concept maps by providing an intermediate layer of computational intelligence that favors the communication through questions and answers in natural language, producing tools that allow the evaluator to examine the content of the concept map without requiring the visual mapping of concepts and relations present in the evaluated maps.

A tool is prototyped and a proof of concept is presented. The analysis of the proposed architecture has led to the definition of a final architecture with features that allow enhancing the use of concept maps and facilitating various pedagogical applications.

This research is dedicated to the investigation of Question Answering Systems, by applying natural language processing techniques to analyze the question and interpret the concept map, as well as applying techniques of artificial intelligence to infer answers to the questions.

Keywords: Concept Maps, Expert Systems, Question Answering Systems, Assessment of Learning, Artificial Intelligence, Prolog.

SUMÁRIO DE FIGURAS

Figura 1.1 – Planejamento das Fases de Pesquisa	8
Figura 2.1 – Um exemplo de Mapa Conceitual	14
Figura 2.2 – Grafo RDF.....	22
Figura 2.3 – Mapa Conceitual de Um curso de Sistemas e Computação	23
Figura 2.4 – Uma rede semântica e sua correspondente transformação em fatos para o <i>Semantic</i>	25
Figura 2.4 – Respostas obtidas com o <i>Semantic</i>	25
Figura 3.1 – Arquitetura Conceitual dos Mecanismos de Inferência em Representações Visuais de Conhecimento	28
Figura 3.2 – Arquitetura Conceitual de Sistemas de Pergunta-Resposta	32
Figura 3.3 – Etiquetagem realizada pelo VISL	37
Figura 3.4 – Definição de uma <i>frase</i> utilizando notação BNF	38
Figura 3.5 – Arquitetura dos Sistemas Especialistas	40
Figura 4.1 – Visão Geral da Arquitetura Proposta.	45
Figura 5.1 – Atividades de MDA para construção do Editor do iMap.....	53
Figura 5.2 – Processo de transformação de um mapa em fatos (proposições) <i>prolog</i>	54
Figura 5.3 – Etapas do Desenvolvimento do Editor do <i>iMap</i>	55
Figura 5.4 – O metamodelo de Mapas Conceituais do Editor de Mapas.	56
Figura 5.5 – Definição da Sintaxe Concreta Visual.	57
Figura 5.6 – Regras de Transformação de Mapas em Proposições para Prolog.	59
Figura 5.7 – Organização conceitual das consultas na arquitetura (a) original (b) evoluída do iMap.	62
Figura 5.8 – Arquitetura funcional do <i>PLN-iMap</i>	62
Figura 5.9 – Definição BNF da <i>gramática intermediária</i>	64
Figura 5.10 – Saídas geradas pelo VISL para o Processador PLN.....	64
Figura 5.11 – Protótipo do <i>PLN-iMap</i>	66
Figura 5.12 – Um Mapa conceitual construído no Editor do <i>iMap</i>	67
Figura 5.13 – Execução da Transformação do Mapa Conceitual em arquivo de base para <i>PIE</i>	68
Figura 5.14 – Arquivo de saída do processo de transformação.	69
Figura 5.15 – Interação Direta através de Regras do <i>PIE</i>	70
Figura 5.16 – Tela de Inicialização do <i>PIE</i>	71
Figura 5.17 – Perguntas-Respostas através de Regras do <i>PIE</i>	72
Figura 5.18 – Resultado do processo de tradução do <i>PLN-iMap</i>	73
Figura 5.19 – Resultado do processamento da regra selecionada.	73
Figura 6.1 – (a) Visão de Integração do Portal do Conhecimento com a Plataforma. (b) Visão de Integração de Serviços Externos à Plataforma.	81
Figura 6.2 – Arquitetura Funcional da plataforma <i>CMPaaS</i>	82
Figura 6.3 – Arquitetura Conceitual do <i>CMPaaS</i>	83

SUMÁRIO DE TABELAS

Tabela 1.1 – Características da metodologia científica aplicada ao trabalho.....	6
Tabela 2.1 – Características dos editores atuais de mapas conceituais.	18
Tabela 5.1 – As regras de Consulta/Inferência.....	60
Tabela 5.2 – Exemplo de Perguntas que podem ser realizadas ao <i>iMap</i>	63
Tabela 5.3 – Exemplo de Perguntas que podem ser realizadas ao <i>iMap</i>	65
Tabela 6.1 – Funcionalidades do <i>CMaaS</i> em Pesquisa/Desenvolvimento	86
Tabela 6.2 – Ferramentas do Portal do Conhecimento em Desenvolvimento.....	87

SUMÁRIO

CAPÍTULO 1	INTRODUÇÃO	1
1.1	Motivação	4
1.2	Objetivo	5
1.3	Metodologia de Pesquisa	5
1.4	Produção Científica	11
1.5	Estrutura da Dissertação	12
CAPÍTULO 2	MAPAS CONCEITUAIS E A EDUCAÇÃO	13
2.1	Teoria e Origem dos Mapas Conceituais	13
2.2	O Estado da Arte em Mapas Conceituais	15
2.3	Estado da Prática em Mapas Conceituais	17
2.4	Trabalhos Correlatos	21
2.5	Considerações	26
CAPÍTULO 3	ARQUITETURA E TECNOLOGIAS	27
3.1	Arquitetura Conceitual	27
3.2	Técnicas e Tecnologias	30
3.3	Considerações	43
CAPÍTULO 4	A ARQUITETURA PROPOSTA	44
4.1	Visão Geral da Arquitetura	44
4.2	Fluxo de Atividades	47
4.3	Considerações	49
CAPÍTULO 5	O <i>IMAP</i>	50
5.1	O Protótipo	50
5.2	Uma Prova de Conceitos	67
5.3	Considerações	74
CAPÍTULO 6	O PROJETO CMPAAS	76

6.1	Motivação.....	76
6.2	<i>CMaaS</i> : A Plataforma.....	79
6.3	Considerações.....	87
CAPÍTULO 7	CONCLUSÃO E TRABALHOS FUTUROS	89
CAPÍTULO 8	REFERENCIAS.....	92

CAPÍTULO 1 INTRODUÇÃO

"Todo o conhecimento que não leva a novas perguntas rapidamente morre: não consegue manter a temperatura necessária para a manutenção da vida."

*(Wisława Szymborska)
"The Poet and the World"*

Há séculos um questionamento perdura na mente de vários pesquisadores e filósofos. Longas discussões são realizadas a fim de descobrir o que é mais importante: saber responder ou saber perguntar? Afinal, o que nos torna mais sábios? Ter resposta para tudo? Ou, saber questionar aquilo que nos é apresentado?

Concordamos com Shank e Birnbaum (1996) quando afirmam que “um aspecto central da inteligência é a necessidade de gerar questões e responder a elas. Nenhuma entidade pode aprender sem gerar para si mesma a necessidade de conhecer”. Por isso, consideramos a pergunta como peça fundamental do processo de construção de conhecimento, acreditando que os questionamentos são a essência dos seres pensantes e socialmente críticos. Quando questionamos, damos início a um processo cognitivo que nos retira da zona de conforto mental e nos faz ultrapassar os limites daquilo que conhecemos atualmente, abrindo espaço para a inserção de novos conhecimentos. Para aqueles que buscam a resposta para uma pergunta, o caminho para encontrar uma resposta concreta envolve, quase sempre, a passagem por questionamentos intermediários que ampliam exponencialmente os campos de investigação e, conseqüentemente, os vieses de aprendizagens.

Por isso, é na educação que as perguntas exercem um papel fundamental. Costumeiramente, professores lançam mão desses artifícios a fim de averiguar o efetivo aprendizado de conceitos por parte de seus alunos. Para o auxiliarem nesse processo, existem diversas abordagens ou artifícios pedagógicos que eles utilizam e que se baseiam no modelo de perguntas e respostas, tais como: provas, exercícios, trabalhos etc. Todos esses possuem unicamente a finalidade de verificar se o conhecimento externado foi corretamente construído pelo estudante. Nesta pesquisa limitamos nosso foco nas

abordagens pedagógicas que utilizem mapas conceituais como ferramentas de representação do conhecimento.

Mapas conceituais são ferramentas gráficas para a organização e representação do conhecimento. São compostos por conceitos, geralmente dentro de círculos ou retângulos, e relações entre esses conceitos, que são indicadas por linhas que os interligam. Sobre essas linhas encontram-se as frases de ligação, que especificam a relação entre os conceitos conectados. Entendemos o conceito como uma regularidade percebida em eventos ou objetos, designada por um rótulo. Na maioria dos conceitos, o rótulo é uma palavra ou, em alguns casos, mais de uma palavra. A cada tripla (conceito, relação, conceito) damos o nome de proposição. Proposições são afirmações sobre algum objeto ou evento no universo, seja ele natural ou artificial. Elas contêm dois ou mais conceitos conectados por palavras ou frases de ligação para compor uma afirmação com sentido. Indo um pouco além, Piaget (PIAGET e GARCIA, 2000) expande a importância das ligações sugerindo que sua natureza semântica define diferentes níveis de conhecimento de quem as produz. Por vezes, são chamadas de unidades semânticas ou unidades de sentido (NOVAK e CAÑAS, 1998). Consideramos as proposições como a menor unidade de conhecimento e acreditamos que ao formá-las o autor está, na realidade, realizando uma enunciação de uma unidade de seu conhecimento.

Nesta pesquisa consideramos que a formação de proposições é *condicio sine qua non* para concepção de mapas conceituais uma vez que tencionamos desenvolver uma ferramenta inteligente que interpreta o conhecimento presente nos mapas conceituais. Kremer (1994, *tradução própria*) esclarece tal necessidade ao afirmar que “existe uma dicotomia entre a necessidade do usuário humano de trabalhar com um sistema flexível e tolerante (informal) e a necessidade do computador de que haja um rigor semântico em sua concepção (formal)”. Portanto, para esta pesquisa, é essencial que as frases de ligações possuam verbos que estabeleçam claramente a relação semântica existente entre os dois conceitos conectados.

Um aspecto central que motiva essa pesquisa reside no fato de os mapas conceituais serem ferramentas que, de certa forma, incentivam a realização de investigações cada vez mais profundas do conhecimento acerca de um dado assunto. A cada conceito e relação externado, novas questões surgem na mente de quem o constrói. Por causa disto, podemos afirmar que os mapas conceituais não são apenas representações do conhecimento, mas

também compõem enquanto ferramenta de construção do conhecimento. Comumente, quem constrói um mapa conceitual se envolve num processo iterativo baseado em questões, tais como: “Que outro conceito pode ser representado neste mapa? Com quais conceitos previamente representados este novo conceito está relacionado?”. Para auxiliar nesse processo de elencagem de novos conceitos e relações, Ribeiro, Menezes, *et al.* (2011) propõem uma arquitetura computacional capaz de, por meio de uma base de inteligência, enriquecer o mapa com conceitos e relações comumente representados em mapas do assunto abordado.

Mas há também uma investigação profunda realizada por parte dos docentes, ou especialistas, ao avaliarem o conteúdo de um mapa conceitual em sua área de conhecimento. Ao avaliarem, eles buscam identificar nuances de conhecimento por meio dos conceitos e relações externados pelo construtor do mapa conceitual. Para isso, eles costumam lançar, mentalmente, uma série de questões relacionadas ao conteúdo do mapa que consideram importantes de serem externadas a fim de averiguar o efetivo reconhecimento dos conceitos e relações importantes. Nesse momento, algumas das questões levantadas são: “O que o aluno sabe sobre o conceito A? O aluno identificou a relação existente entre o conceito A e o conceito B? Qual o termo de ligação utilizado nas relações existentes entre os conceitos A e B? Quais conceitos o estudante relacionou ao conceito A?”. Essas são apenas algumas das questões mentalizadas por professores e especialistas ao avaliarem um mapa conceitual.

Como é possível observar, no entanto, a busca por resposta às perguntas levantadas pelos docentes e especialistas remete, para cada uma, a uma pesada carga de processamento cognitivo. Muitas vezes, esse processo de avaliação do mapa demanda tempo pois, ao avaliar, diversos conceitos e relações são reprocessados para cada uma das perguntas para as quais se buscam respostas. Essa característica inviabiliza, muitas vezes, a adoção dos mapas conceituais em muitas abordagens pedagógicas já que grande parte dos docentes não dispenderiam do tempo necessário para realizar essas investigações em todos os mapas gerados por seus estudantes.

Outro fator complicador é que, dependendo da abordagem utilizada, os mapas conceituais podem tomar proporções que tendem a dificultar sua navegação. Isso é comum em mapas gerados em grandes comporações, onde o número de conceitos e relações gerados são muito grandes.

Sendo assim, esta pesquisa está interessada em apresentar uma solução computacional que exima os avaliadores de mapas conceituais da exigência de realizarem sucessivas navegações nos mapas em busca de informações.

1.1 MOTIVAÇÃO

Diversas pesquisas têm sido realizadas no sentido de ampliar as funções, o aparato ferramental e, conseqüentemente, as aplicações e a utilização dos Mapas Conceituais nos processos de aprendizagem. Há, por exemplo, pesquisadores que estudam maneiras de: ampliar a coesão e a coerência dos conceitos presentes em mapas conceituais (RIBEIRO, MENEZES, *et al.*, 2011); apoiar a construção automática, ou semiautomática, de mapas conceituais (KOWATA, CURY e BOERES, 2011, 2010, 2009); modelar ambientes para o acompanhamento da aprendizagem baseado em mapas conceituais (SANTOS, MENEZES, *et al.*, 2005); construir mecanismos de apoio automatizado para avaliação da aprendizagem utilizando mapas conceituais (ARAUJO, MENEZES e CURY, 2003); comparar mapas conceituais utilizando correspondência de grafos (LAMAS, BOERES, *et al.*, 2006); construir ontologias a partir de mapas conceituais utilizando a teoria dos grafos (ZOUAQ, GASEVIC e HATALA, 2011); associar sistemas lógicos e lógica natural aos mapas conceituais para acompanhamento dos processos de conceituação (DUTRA, FAGUNDES, *et al.*, 2006); e, utilizar mecanismos de inteligência artificial para favorecer a interação com mapas conceituais (HANDRICK, 2005; PERIN, CURY e MENEZES, 2012).

A motivação para esta pesquisa é, portanto, o desenvolvimento de uma nova ferramenta computacional de apoio à análise de mapas conceituais baseado na arquitetura de sistemas de perguntas e respostas. Por meio dele, tencionamos eximir, ou tornar opcional, a navegação visual dos mapas conceituais por parte dos avaliadores ao buscarmos informações acerca do conhecimento presente neles. Para isso, aplicamos nessa pesquisa técnicas avançadas que incluem o Processamento de Linguagem Natural, Programação Lógica, Arquitetura de Sistemas Orientados a Serviços e Sistemas de Perguntas e Respostas¹.

¹ Também comumente chamado na língua inglesa de sistemas *question answering* – QA

1.2 OBJETIVO

O objetivo geral desta pesquisa é desenvolver uma ferramenta computacional que favoreça a análise do conhecimento presente em mapas conceituais por meio de perguntas e respostas, adotando técnicas distintas e complementares, como: recuperação de informação, processamento de linguagem natural, sistemas inteligentes, sistemas de perguntas-respostas, dentre outras; a fim de compor uma solução completa que vai desde a análise e interpretação de perguntas realizadas em linguagem natural até a construção de respostas baseadas no conhecimento presente em um mapa conceitual.

Para atingir o objetivo geral, os seguintes objetivos específicos foram traçados:

- Investigar, junto a docentes que adotam mapas conceituais como ferramentas de avaliação do conhecimento, suas necessidades e principais questões que norteiam suas análises dos mapas gerados pelos estudantes;
- Investigar, analisar, registrar e tabular as ferramentas existentes à luz das necessidades dos docentes a fim de validar a hipótese de que as soluções atuais não possuem ferramentas que favorecem uma análise dos mapas conceituais sem a necessidade de realizar navegações;
- Definir uma arquitetura conceitual para um sistema inteligente que permita a interação por meio de perguntas e respostas construídas em linguagem natural;
- Projetar, construir, testar e avaliar um protótipo funcional e identificar seus pontos fortes e suas limitações;
- Descrever uma arquitetura geral de um sistema que permita a expansão de ferramentas para a construção, manipulação e avaliação de mapas conceituais.

1.3 METODOLOGIA DE PESQUISA

O trabalho científico é avaliado, segundo Demo (1996), por meio de dois parâmetros: sua qualidade política e sua qualidade formal. A qualidade política diz respeito aos conteúdos, aos fins e à substância do trabalho científico. Já a qualidade formal diz respeito aos meios e formas usados na produção do trabalho, ou seja, as técnicas de coleta e interpretação de dados, manipulação de fontes de informação, conhecimento demonstrado na apresentação

do referencial teórico e apresentação escrita ou oral em conformidade com os ritos acadêmicos.

No entanto, cada pesquisa possui características específicas de modo que inexitem regras fixas e rigorosas que especifiquem todos os caminhos para a elaboração de um projeto. Sua estrutura, no entanto, é determinada pelo tipo de problema pesquisado e, ainda, pelo estilo adotado por seus autores.

1.3.1 Referencial Metodológico

O projeto precisa esclarecer os processos da pesquisa, as etapas que serão desenvolvidas e os insumos que serão alocados para atingir os objetivos (GIL, 2002). O método de investigação científica adotado nesta pesquisa possui as características descritas na Tabela 1.1.

Tabela 1.1 – Características da metodologia científica aplicada ao trabalho.

Quanto à natureza	Pesquisa aplicada
Quanto aos objetivos	Exploratório e descritiva
Quanto às abordagens	Quantitativa e Qualitativa
Quanto aos procedimentos	Experimental
Quanto ao método	Empírico e Indutivo

A natureza deste estudo é classificada como **pesquisa aplicada** que, segundo Barros e Lehfeld (2000, *apud* VILAÇA, 2010), tem como motivação a necessidade de produzir conhecimento para aplicação de seus resultados, com o objetivo de “contribuir para fins práticos, visando à solução mais ou menos imediata do problema encontrado na realidade”. As pesquisas aplicadas dependem de dados que podem ser coletados de formas diferenciadas, o que inclui a pesquisa de campo, utilizada nesta pesquisa.

Do ponto de vista de seus objetivos, esta pesquisa requer designação dupla: **pesquisa descritiva** e **exploratória**. Por **pesquisa descritiva** (GIL, 1999) entendemos que o resultado esperado para o problema definido é traduzido na especificação de uma solução para o problema. Isso inclui, obviamente, o detalhamento das ações empregadas no processo e recursos consumidos por cada ação. É também uma **pesquisa exploratória** (GIL, 1999) já que boa parcela dos esforços desta pesquisa estavam concentrados no estudo bibliográfico em busca de métodos de avaliação de mapas conceituais adotados

por professores e soluções computacionais que se baseiam na interseção entre mapas conceituais e inteligência artificial.

Já no que tange à forma de abordagem do problema, Silva e Menezes (2001) argumentam que uma pesquisa pode abordar um problema de duas maneiras: quantitativa ou qualitativa. Novamente aqui, esta pesquisa adota as duas formas de abordagem já que o a solução proposta tem seus processos e significados descritos e visa aprimorar, de maneira **qualitativa**, o trabalho docente. É também **quantitativa** pois baseia-se na quantificação, análise e classificação de dados coletados em campo.

Os procedimentos técnicos empregados nesta pesquisa incluem os seguintes recursos:

- i. a pesquisa bibliográfica, na fase de concepção;
- ii. a pesquisa experimental, na fase de elaboração;
- iii. o estudo de caso, na fase de consolidação.

Por último, no que diz respeito ao método científico, novamente há uma dupla designação pois, ao mesmo tempo que a pesquisa explora situação e dificuldades vivenciadas pelo próprio autor – caracterizando uma pesquisa **Empírica** – ela se baseia também em análise de dados e ocorrências envolvendo terceiros que permitem ao autor retirar conclusões gerais acerca do fenômeno estudado – caracterizando uma pesquisa **Indutiva**.

Além disso, esta pesquisa considera também a ABNT que orienta que uma dissertação é

[...] o resultado de um trabalho experimental ou exposição de um estudo científico retrospectivo, de tema único e bem delimitado em sua extensão, **com o objetivo de reunir, analisar e interpretar informações. Deve evidenciar o conhecimento de literatura existente** sobre o assunto [...] (ABNT NBR 14724, 2005, grifo nosso).

Assim, uma dissertação pode ser definida como a descrição de um trabalho experimental, em todas as suas etapas (concepção, elaboração e consolidação), à luz do conhecimento disponível na literatura existente. Por isso, na seção seguinte descrevemos o planejamento aplicado no desenvolvimento desta dissertação.

1.3.2 Planejamento

O planejamento desta pesquisa é definido através de três fases distintas e complementares, a saber: Concepção, Elaboração e Consolidação. Os insumos produzidos nas fases antecedentes são consumidos pelas fases precedentes, conforme observado na Figura 1.1.

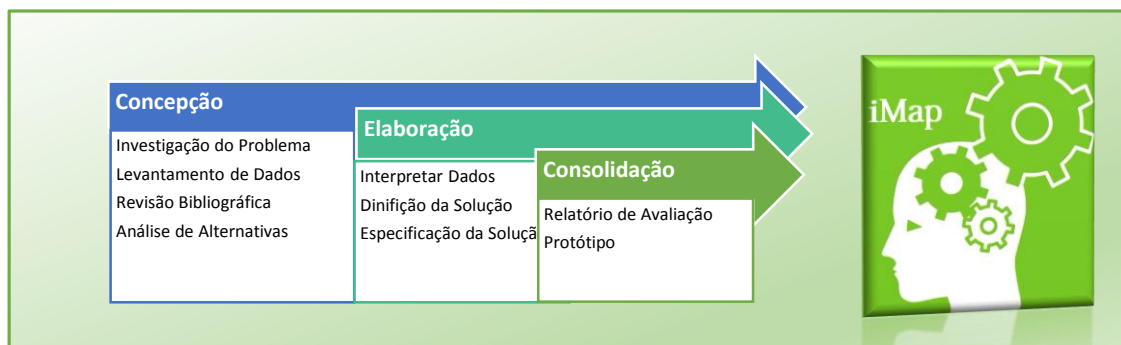


Figura 1.1 – Planejamento das Fases de Pesquisa

É importante frisar que os resultados obtidos em cada fase desta pesquisa estão registrados ao longo desta dissertação.

Na fase de **Concepção** definimos um conjunto de hipóteses que foi verificado por meio de questionários de investigação e da análise e cruzamento de informações presentes na bibliografia científica disponível. Os dados e a revisão bibliográfica nos permitiram, além da validação de hipóteses, uma maior apropriação do problema e a concepção de alternativas para solucioná-lo.

Passando para a fase de **Elaboração** onde, de posse dos dados da pesquisa e das alternativas para solução do problemas, pudemos definir a melhor abordagem para o problema e especificar a arquitetura básica deste sistema. Esta etapa da pesquisa culminou com uma publicação no “*XVII Version - Congreso Internacional de Informática Educativa (TISE)*”.

Por último, na fase de **Consolidação**, detalhes funcionais da arquitetura especificada na fase anterior puderam ser melhor explorados após análise do relatório de avaliação. Diversas funcionalidades puderam ser aprimoradas e uma nova arquitetura foi concebida. As novas funcionalidades e a nova arquitetura renderam duas publicações no “*Sixth International Conference on Concept Maps*”.

1.3.3 Hipóteses

As hipóteses levantadas ao longo da fase de planejamento desta pesquisa foram:

- i. Os professores carecem de ferramentas que permitam a análise e a avaliação de mapas conceituais sem exigir destes a navegação visual do mapa;
- ii. A existência de ferramentas que auxiliem na análise e avaliação de mapas conceituais pode torná-los mais praticáveis no dia-a-dia docente;
- iii. Os conhecimentos presentes em mapas conceituais podem ser utilizados como fonte para alimentar uma base de inteligência de sistemas especialistas;
- iv. Um sistema especialista cuja base de inteligência esteja alimentada com proposições presentes num mapa conceitual pode favorecer uma interação por meio de perguntas e respostas;
- v. Perguntas construídas em linguagem natural podem ser traduzidas para um formato interpretável por sistemas especialistas; e
- vi. Uma arquitetura orientada a serviços pode favorecer a expansão constante de um sistema para gestão completa de mapas conceituais.

1.3.4 Questões Norteadoras

Silva e Menezes (2001, p.19) afirmam que a pesquisa científica visa encontrar respostas para as indagações propostas. De acordo com os objetivos gerais, as indagações que esta pesquisa se propõe a responder são:

- i. Quais as limitações das atuais ferramentas existentes que dão suporte à construção de mapas conceituais no que diz respeito à avaliar o conhecimento neles presente?
- ii. Quais as necessidades principais dos docentes ao acompanhar e avaliar a aprendizagem por meio dos mapas?
- iii. É possível construir uma ferramenta computacional capaz de favorecer a interação com Mapas Conceituais por meio de perguntas e respostas em linguagem natural?
- iv. Como um sistema inteligente pode ser alimentado por proposições presentes em Mapas Conceituais?
- v. Como esse mesmo sistema pode ser capaz de interpretar as perguntas em linguagem natural?

1.3.5 Etapas da Pesquisa

Para atingir os objetivos listados na Seção 1.2, os seguintes passos se fazem necessários:

1. Revisar a bibliografia existente com objetivo de encontrar estudos relacionados a necessidades e métodos de avaliação de aprendizagem utilizando mapas conceituais.
2. Investigar, por meio de questionário aplicado junto a docentes que utilizam mapas conceituais como ferramenta de aprendizagem, as suas principais necessidades e métodos utilizados por eles no acompanhamento e avaliação da aprendizagem. Para isso, serão utilizadas questões tais como:
 - a. Ao avaliar um mapa conceitual, o que você julga importante?
 - b. Quais questionamentos vêm à sua mente nesses momentos?
 - c. Como você conclui se um estudante aprendeu ou não o assunto abordado?
 - d. Você utiliza ferramentas computacionais nesse processo?
 - e. Se sim, quais e como elas te ajudam no processo de avaliação?
 - f. Como você registra os resultados encontrados?
 - g. Você utiliza dados estatísticos nesse processo?
 - h. Se sim, quais são esses dados?
3. Investigar as ferramentas existentes no mercado que auxiliam na utilização de mapas conceituais na educação. Nessa análise serão levadas em consideração as necessidades anteriormente levantadas a fim de relacionar essas necessidades às ferramentas existentes ou identificar limitações ferramentais.
4. Descrever e organizar um ambiente computacional que integre ferramentas existentes a possíveis extensões, ou novas aplicações, a fim de fornecer mecanismos que auxiliem os docentes no processo de acompanhamento e avaliação da aprendizagem.
5. Testar, analisar e registrar os resultados obtidos, descrevendo as limitações identificadas, pontos fortes e registrar as conclusões.

6. Redigir a dissertação.

1.4 PRODUÇÃO CIENTÍFICA

As partes constituintes desta pesquisa resultaram na publicação dos seguintes artigos:

- i. PERIN, Wagner de A. ; CURY, Davidson. . **CONSTRUINDO MAPAS CONCEITUAIS UTILIZANDO A ABORDAGEM *iMap***. In: Congresso Internacional de Informática Educativa, 2012, Santiago. Anais do Evento, 2012. v. 1. p. 108-113.
- ii. PERIN, Wagner de A. ; MORATI Jr., Roberto G. ; MENEZES, Crediné S. ***iMap* - UM MECANISMO DE INFERÊNCIAS BASEADO EM MAPAS CONCEITUAIS**. In: COMPUTER ON THE BEACH 2013, Florianópolis. Anais do Evento, 2013. v. 1. p. 208-218.
- iii. PERIN, Wagner de A.; CURY, Davidson. **NLP – *iMap*: INTEGRATED SOLUTION BASED ON QUESTION-ANSWER MODEL IN NATURAL LANGUAGE FOR AN INFERENCE MECHANISM IN CONCEPT MAPS**. In: Sixth International Conference On Concept Mapping, Santos – SP. Accepted for publication in the Conference Proceedings, 2014.
- iv. CURY, Davidson; PERIN, Wagner de A.; SANTOS JR., Paulo S. ***CMPaaS* – A PLATAFORM OF SERVICES FOR CONSTRUCTION AND HANDLING OF CONCEPT MAPS**. In: Sixth International Conference On Concep Mapping, Santos – SP. Accepted for publication in the Conference Proceedings, 2014.
- v. ASSIS, Daniel V.; PERIN, Wagner de A.; VASSOLER, Geraldo A.; CURY, Davidson. **VMAP – CARACTERIZAÇÃO DE UMA ABORDAGEM PARA VERIFICAÇÃO SINTÁTICA DE MAPAS CONCEITUAIS**. In: Sixth International Conference On Concep Mapping, Santos – SP. Accepted for publication in the Conference Proceedings, 2014.
- vi. VASSOLER, Geraldo A.; PERIN, Wagner de A.; CURY, Davidson. **MERGEMAPS – A COMPUTATIONAL TOOL FOR MERGING OF CONCEPT MAPS**. In: Sixth International Conference On Concep Mapping, Santos – SP. Accepted for publication in the Conference Proceedings, 2014.

1.5 ESTRUTURA DA DISSERTAÇÃO

Esta dissertação está organizada da seguinte forma:

Capítulo 1: Apresenta uma visão geral da dissertação do ponto de vista de seu planejamento e organização. Por meio dele é apresentado o contexto, as motivações, objetivos e método de pesquisa empregado.

Capítulo 2: Explora os Mapas Conceituais do ponto de vista da problemática proposta para a pesquisa e explora algumas ferramentas e métodos de trabalho existentes que se apoiam em mapas como ferramenta de organização e representação de conhecimento. Revisão Bibliográfica, investigações e análises de dados se misturam para compor este capítulo cujo objetivo é comprovar hipóteses fundamentais que justificam os esforços dedicados a esta pesquisa. Além disso, soluções existentes e trabalhos correlatos são explorados neste capítulo.

Capítulo 3: Apresenta a arquitetura conceitual geral dos mecanismos de inferência a partir de representações gráficas de conhecimento bem como seus processos, as técnicas e ferramentas que adotam. Algumas soluções tecnológicas adotadas na arquitetura geral proposta são analisadas e discutidas.

Capítulo 4: Descreve a arquitetura geral do sistema proposto, cerne desta pesquisa. Especificamos aqui cada componente do mecanismo proposto bem como a forma como esses componentes se relacionam.

Capítulo 5: Toda arquitetura necessita de testes que comprovem sua função. Neste capítulo um protótipo é apresentado, uma prova de conceitos é realizada e os resultados obtidos são analisados. Com base nos resultados, uma arquitetura final é discutida e apresentada.

Capítulo 6: A característica evolutiva desta pesquisa resultou na elaboração de um projeto. Neste capítulo são apresentados detalhes da arquitetura de integração de serviços proposta como principal trabalho futuro.

Capítulo 7: Apresenta as considerações finais, direciona o andamento futuro desta pesquisa e apresenta os trabalhos futuros.

CAPÍTULO 2 MAPAS CONCEITUAIS E A EDUCAÇÃO

"Todo o conhecimento exige um conceito, por mais imperfeito ou obscuro que ele possa ser."

*(Emmanuel Kant)
"Crítica da Razão Pura"*

Este capítulo tem por objetivo apresentar as características gerais dos mapas conceituais: sua origem, função, métodos e ferramentas que dão suporte à sua construção, gestão e manipulação. Objetiva também apontar a limitação explorada por esta pesquisa e embasá-la por meio de uma revisão bibliográfica e análise de dados coletados numa pesquisa de campo.

Para isso, iniciamos a discussão, na Seção 2.1, analisando a origem e a teoria da aprendizagem que culminou no surgimento dos mapas conceituais. Já na Seção 2.2 discutimos as aplicações atuais dos mapas conceituais, apontando soluções computacionais que ampliam suas funções em algumas abordagens pedagógicas.

Após, na Seção 2.3, listamos e analisamos algumas ferramentas existentes para criação e gestão de mapas, apontando uma limitação comum que dificulta o trabalho docente ao utilizar os mapas como ferramenta para representação do conhecimento em suas turmas. Uma pesquisa de campo é apresentada e analisada ainda nesta seção. Utilizamos a Seção 2.4 para pontuar trabalhos correlatos encontrados na bibliografia disponível.

Por fim, encerramos este capítulo na Seção 2.5, onde apresentamos nossas considerações sobre os assuntos abordados neste capítulo.

2.1 TEORIA E ORIGEM DOS MAPAS CONCEITUAIS

Os mapas conceituais foram desenvolvidos em 1972 como fruto da pesquisa de Novak² na qual procurava acompanhar e entender as mudanças no conhecimento de crianças ao

² **Joseph Donald Novak:** É um educador americano, professor emérito da Universidade de Cornell (USA) e pesquisador senior no *Florida Institute for Human & Machine Cognition*. Ele é reconhecido por ter desenvolvido os mapas conceituais na década de 70.

aprenderem ciências (NOVAK e MUSONDA, 1991). Ao longo de suas pesquisas, Novak percebeu que era difícil identificar alterações específicas na compreensão de novos conceitos científicos por parte das crianças utilizando apenas provas e questionários. Assim, ele desenvolveu um método baseado na teoria da aprendizagem de David Ausubel (1963; 1968; AUSUBEL, NOVAK e HANESIAN, 1978). Segundo Ausubel, a assimilação de novos conceitos ocorrem através da formação de proposições com conceitos previamente existentes na estrutura cognitiva de um indivíduo. Assim, Novak percebeu a necessidade de uma melhor maneira para representar a compreensão conceitual de uma criança e propôs que isto se desse na forma de um mapa conceitual (NOVAK e GOWIN, 1984). Assim surgiu esta ferramenta que posteriormente seria adotada em muitas outras tarefas além da representação do conhecimento (NOVAK e CAÑAS, 2010).

Mapas Conceituais são ferramentas para representação gráfica de um dado domínio do conhecimento que “possibilitam a ressignificação de conteúdos, apoiando, desta forma, práticas educacionais voltadas para a aprendizagem significativa” (KOWATA, CURY e BOERES, 2010). São constituídos por uma rede de nós, que representam os conceitos ou objetos, conectados por arcos com rótulos descritores das relações entre os pares de nós (NOVAK e CAÑAS, 2006b). A Figura 2.1 apresenta um exemplo de Mapa Conceitual que explica o que é um Mapa Conceitual.

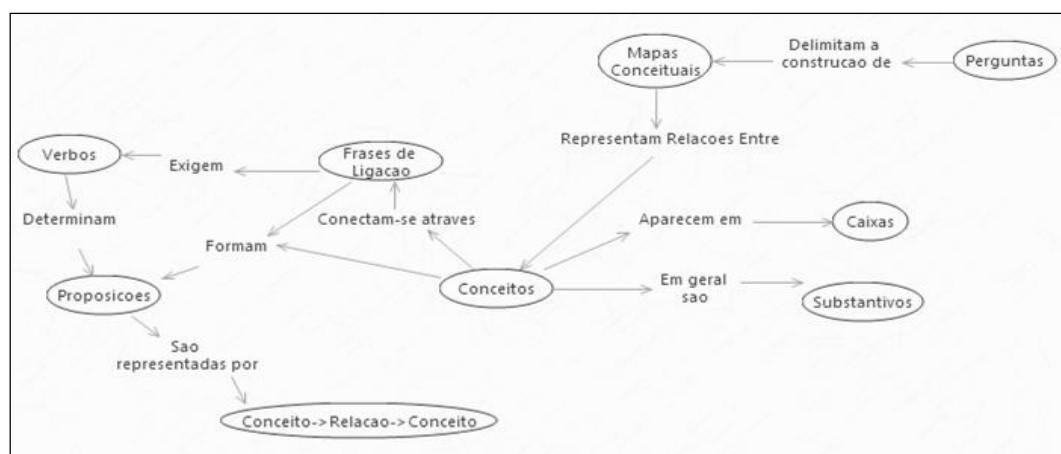


Figura 2.1 – Um exemplo de Mapa Conceitual

Devido à sua versatilidade, é possível utilizá-lo para sintetizar o conhecimento desenvolvido numa única aula, numa unidade de um curso, num capítulo de um livro ou mesmo num programa educacional completo. À medida que os alunos vão construindo

conhecimento em sala de aula, podem utilizar os mapas conceituais para integrar, reconciliar e diferenciar conceitos, servindo este, portanto, como um recurso de aprendizagem (NOVAK e CAÑAS, 2006a). Dessa forma, os mapas representam a estrutura cognitiva do aprendiz e podem ser, obviamente, valiosas fontes de informação para os professores no acompanhamento e na avaliação da aprendizagem.

Por ser considerada uma ferramenta “aberta”, já que não traz informações prévias em excesso aos estudantes além de apenas um pequeno conjunto de regras para sua utilização, os mapas conceituais deixam os professores e estudantes livres para explorar o conhecimento dentro do campo de estudo de seus interesses (SANTOS, MENEZES, *et al.*, 2005).

Como é possível observar, a simplicidade é a principal característica dos mapas conceituais, pois concede aos autores uma total liberdade para representarem o conhecimento com menos formalismos, e é justamente esta característica que os tornam tão atrativos aos docentes preocupados com a aprendizagem significativa na sua prática. Essa característica tem também motivado pesquisadores a aprofundarem ainda mais nas aplicações dos mapas. Na seção seguinte apresentamos algumas das aplicações e soluções desenvolvidos por pesquisadores da área de mapeamento conceitual.

2.2 O ESTADO DA ARTE EM MAPAS CONCEITUAIS

A utilização de recursos computacionais na promoção e no aprimoramento das práticas educacionais é uma realidade existente em muitas universidades. De fato, diversas pesquisas comprovam que o uso bem direcionado dos recursos computacionais podem proporcionar resultados efetivos na aprendizagem, já que as ferramentas computacionais são mais atrativas, modernas e, em muitos casos, acessíveis de qualquer lugar e a qualquer momento (MANSUR, BIAZUS, *et al.*, 2013; VERA, FUENTES e RIVERA, 2013; BARCELOS e BATISTA, 2013; PRIETO, CHROBAK, *et al.*, 2012).

Neste contexto, Carvalho, Nevado e Menezes (2005) pesquisam o conceito de arquiteturas pedagógicas e objetivam descrever métodos de ensino que integrem ferramentas tecnológicas à educação com o objetivo de causar impactos significativos no processo de aprendizagem. Neste sentido, os Mapas Conceituais se destacam como uma das ferramentas que dão suporte a diferentes abordagens pedagógicas.

Diversão são as aplicações dos mapas conceituais nos processos de ensino e aprendizagem (MOREIRA e BUCHWEITZ, 1993). Eles podem ser utilizados, por exemplo, como instrumentos no processo de ensino e aprendizagem (ALVES, PEREIRA e CARDOSO, 2001; CLARIANA e KOUL, 2004; LAU, CHUNG, *et al.*, 2008), como ferramenta para definição de estratégias pedagógicas (BAI e CHEN, 2008; CHANG, TAM, *et al.*, 2008; CHEN, KINSHUK, *et al.*, 2008; LEE, LEE e YUNGHO, 2009; TSENG, SUE, *et al.*, 2007) e como fonte de informação na avaliação de desempenho da aprendizagem teórica por parte dos estudantes (GRAUDINA e GRUNDSPENKIS, 2008; VILLALON e CALVO, 2008).

Por isso, diversas pesquisas têm sido realizadas no sentido de ampliar as funções, o aparato ferramental e, conseqüentemente, a utilização desta ferramenta nos processos de aprendizagem. Há, por exemplo, pesquisadores que estudam maneiras de: ampliar a coesão e a coerência dos conceitos presentes em mapas conceituais (RIBEIRO, MENEZES, *et al.*, 2011); apoiar a construção automática, ou semiautomática, de mapas conceituais (KOWATA, CURY e BOERES, 2011, 2010, 2009); modelar ambientes para o acompanhamento da aprendizagem baseada em mapas conceituais (SANTOS, MENEZES, *et al.*, 2005); construir mecanismos de apoio automatizado para avaliação da aprendizagem utilizando mapas conceituais (ARAÚJO, MENEZES e CURY, 2003); comparar mapas conceituais utilizando correspondência de grafos (LAMAS, BOERES, *et al.*, 2006); construir ontologias a partir de mapas conceituais utilizando a teoria dos grafos (ZOUAQ, GASEVIC e HATALA, 2011); associar sistemas lógicos e lógica natural aos mapas conceituais para acompanhamento dos processos de conceituação (DUTRA, FAGUNDES, *et al.*, 2006); e, utilizar mecanismos de inteligência artificial para favorecer a interação com mapas conceituais (HANDRICK, 2005; PERIN, CURY e MENEZES, 2012).

Observamos, portanto, que diversas soluções computacionais estão sendo prototipadas e apresentadas. Na seção seguinte apresentaremos, ainda, algumas das ferramentas mais conhecidas que permitem a construção, gestão e manipulação de mapas conceituais e exploraremos algumas limitações e potencialidades comuns a eles.

2.3 ESTADO DA PRÁTICA EM MAPAS CONCEITUAIS

Devido à sua simplicidade, não é necessário mais que papel e caneta para construir um mapa conceitual. No entanto, utilizar apenas estes recursos reduz a produtividade pois complica o reposicionamento e a reorganização do mapa. Além disso, armazenar e gerenciar os mapas construídos pode, a longo prazo, se tornar uma tarefa árdua e cansativa. A soma destes fatores ao crescente número de pesquisas e de novas aplicações dos mapas conceituais na educação e na sociedade despertou rapidamente o interesse dos usuários em aplicações computacionais que auxiliem na construção e gestão de mapas conceituais.

Nesta pesquisa interessamo-nos por levantar e analisar as ferramentas computacionais existentes que são anunciadas como “editores de mapas conceituais”. Da análise das ferramentas existentes foi possível agrupar e tabular os prós e contras de cada um dos tipos de editores e registrar as características que são de especial interesse para esta pesquisa. A Tabela 2.1 sintetiza esta análise.

Ao longo do processo de análise percebemos a importância de organizar os *softwares* existentes por uma taxonomia, de acordo com o tipo do mapa o qual ele é originalmente destinado a produzir. Isto porque percebemos que há muita confusão por parte dos usuários, e também dos desenvolvedores, das diferenças entre mapas conceituais, mapas mentais, organogramas e memória digital. Assim, utilizamos a seguinte taxonomia para categorizar os *softwares* de acordo com seu tipo.

- **MC:** Editor cuja finalidade é construir mapas conceituais.
- **OG:** Editores cuja finalidade é a construção de organogramas e, alternativamente, permitem a representação de conhecimento na forma de mapas conceituais.
- **MM:** Editores cuja finalidade é a construção de Mapas Mentais.
- **MD:** Editor cuja classificação é definida pelos próprios desenvolvedores como memória digital, ou seja, permite o mapeamento de qualquer tipo de conhecimento e atividade.

Consideramos nesta pesquisa que os mapas conceituais possuem uma característica que o torna singular em relação às demais formas de representação do conhecimento. Para

nós, mapas conceituais exigem a formação de proposições, ou seja, afirmações diretas de unidades de conhecimento contendo, cada uma, dois conceitos conectados por uma frase curta que possua um verbo capaz de estabelecer uma relação semântica entre os conceitos.

Outro ponto importante no processo de análise das ferramentas foi verificar se apresentavam recursos de inteligência artificial que auxiliem os usuários de alguma forma a construir e/ou analisar os mapas gerados. Assim, constatamos que apenas o *CmapTools* possui recursos que auxiliam na elencagem de conceitos candidatos ao mapa conceitual orientados numa base de conhecimento previamente alimentada (CAÑAS e CARVALHO, 2004). As demais, apesar de possuírem recursos pontuais que auxiliam na navegação e localização de conhecimentos presentes nos mapas, não oferecem recursos que sejam notadamente portadores de Inteligência Artificial.

Tabela 2.1 – Características dos editores atuais de mapas conceituais.

Nome	Tipo	Prós	Contras
CmapTools v5.06	MC	<ul style="list-style-type: none"> - Oferece uma plataforma completa para construção e gestão de mapas. - Oferece serviços de repositório, compartilhamento e construção colaborativa de mapas. - Possui muitos recursos para formatação, multimídia, exportação e importação de mapas. - Possui recursos de IA que auxiliam na construção de mapas conceituais. - É oferecido em diversos idiomas. 	<ul style="list-style-type: none"> - Últimas novas funções oferecidas em 25/07/2010, não apresentando avanços em sentido funcional desde então. - É proprietário e não permite a criação de extensões pela comunidade.
Bizagi Process Modeler 2.7	OG	<ul style="list-style-type: none"> - Possuem recursos que permitem construir, formatar e armazenar mapas conceituais. - Permite a formatação e inserção de elementos multimídia. - Alguns estão disponíveis na Web. 	<ul style="list-style-type: none"> - É direcionado para modelagem de Processos e Organogramas, a representação do conhecimento por mapas conceituais é apenas possível. - Não possui qualquer recurso de inteligência que facilite operações com mapas conceituais. - São soluções proprietárias que não permitem a criação de extensões pela comunidade.
XMind 3.3.1			
yED			
VUE			
ExploraTree	MM	<ul style="list-style-type: none"> - Possuem recursos que permitem representar conhecimento. - Possuem ferramentas de formatação e de multimídia para enriquecimento do mapa. - Alguns estão disponíveis na Web. 	<ul style="list-style-type: none"> - Direcionado para a construção de Mapas Mentais. - Não possuem recursos de inteligência que facilitem operações com Mapas. - Maioria não permite a criação de extensões por terceiros, à exceção
MindMeister			
WiseMapping			
iMindMap 7.0.2			
FreePlane 1.1.3			
Flo 1.1			

Mindomo 5.22		- MindMeister adota arquitetura orientada a serviços que permite disponibilização e criação de extensões por terceiros.	do MindMeister que permite mas mediante utilização de licença apropriada, e do Mind Map Architect e VYM que são livres e de código aberto.
Mind Map Architect		- Alguns permitem a construção coletiva de mapas.	- Software Flo é pobre em recursos para aprimoramento dos mapas.
Coogle		- Coogle e iMindMap estão disponíveis para dispositivos móveis.	
VYM			
TheBrain 8.0.1.1	MD	- Permite a construção de mapas com ligações de diversos tipos de elementos multimídia. - Possui diversas ferramentas para registro de conhecimento, ideias, pessoas e documentos.	- Não é específico para Mapas Conceituais e se apresenta como uma ferramenta de Memória Digital. - Não permite a construção de proposições e sim a conexão de ideias, pessoas, documentos, dentre outros. - Não permitem a criação de extensões por terceiros. - Não possuem ferramentas de inteligência artificial mas alguns recursos de busca avançada por elementos presentes no mapa.

Ao analisar os *softwares* da Tabela 2.1 percebemos que apenas o *CMapTools* dá, de fato, suporte à construção de mapas conceituais. As demais ferramentas anunciadas como editores de mapas conceituais possuem finalidade real de construir organogramas, modelagem de processos, mapas mentais e outros tipos de diagramas, mapas e representações visuais de conhecimento. Assim como estes, muitas outras ferramentas de modelagem de processos e organogramas, bem como editores de texto e imagens podem ser utilizados na construção de mapas conceituais.

No entanto, para as pesquisas direcionadas aos mapas conceituais este é um fato preocupante pois, à exceção do *CMapTools*, as ferramentas existentes não disponibilizam soluções que facilitem operações específicas com mapas conceituais. Mesmo o *CMapTools* apresenta um número limitado de recursos. O fato de a maioria dos editores utilizados não serem destinadas à construção de mapas conceituais nos permitiu definir a seguinte hipótese de trabalho: “Os professores carecem de ferramentas que permitam a análise e a avaliação de mapas conceituais sem exigir destes a navegação visual do mapa”.

Para validar esta e outras hipóteses, a Seção 2.3.1 apresentará uma investigação realizada ao longo desta pesquisa.

2.3.1 Investigando Necessidades e Limitações

A investigação realizada nesta pesquisa teve por objetivo validar duas das hipóteses propostas, quais sejam:

1. Os professores carecem de ferramentas que permitam a análise e a avaliação de mapas conceituais sem exigir destes a navegação visual do mapa.
2. A existência de ferramentas que auxiliem na análise e avaliação de mapas conceituais pode torná-los mais praticáveis no dia-a-dia docente.

A fim de validar estas hipóteses, realizamos uma pesquisa de campo junto à 46 professores dos cursos superiores oferecidos pela faculdade Multivix³. Esta pesquisa foi realizada durante a “Semana de Capacitação Docente” onde, após uma palestra intitulada “Uso de mapas conceituais como ferramenta de apoio à aprendizagem”, aplicamos um questionário de investigação cujo objetivo era identificar as dificuldades, necessidades e ferramentas utilizadas por docentes para analisar mapas conceituais.

Não foram definidos critérios para inclusão na amostragem já que a instituição tencionou uma capacitação que atingisse todo o público. Assim, havia entre os professores aqueles que já tinham experiências com Mapas Conceituais e outros que nunca os haviam utilizado. Ao longo da palestra foram discutidas abordagens para concepção e avaliação dos mapas, de tal forma que todos pudessem perceber os desafios que enfrentariam ao analisarem mapas conceituais. Sabendo da característica heterogênea do público-alvo, elaboramos um único questionário onde todos pudessem colaborar com esta pesquisa, dentro de suas experiências vivenciadas.

O ANEXO I apresenta o questionário aplicado que possui, claramente, os seguintes objetivos:

1. Verificar se os professores sentem necessidade de ferramentas que auxiliem no processo de avaliação / verificação de mapas conceituais, e;

³ **Faculdade Multivix:** É um faculdade privada com unidades distribuídas em 5 municípios do Espírito Santo e oferece diversos cursos (Técnicos e Superiores) nas áreas de Humanas, Saúde e Exatas.

2. Levantar perguntas que eles gostariam de fazer a uma ferramenta inteligente de modo a extrair conhecimento ali presente.

Ao analisar os resultados obtidos, percebemos que 100% dos professores afirmaram que necessitam de ferramentas que auxiliem, de maneira inteligente, no processo de análise e avaliação de mapas. Além disso, as perguntas mais recorrentes sugeridas por eles nos permitiram direcionar o desenvolvimento do protótipo para a solução de problemas que eles apontaram como necessidades.

Assim, o ANEXO II apresenta uma lista de perguntas que foram recorrentes na investigação realizada e que foram desenvolvidas e testadas no protótipo que será apresentado no Capítulo 5 desta dissertação.

Antes de arquitetar e desenvolver o protótipo, no entanto, foi necessário investigar a existência e as características de alguns trabalhos correlatos. A seção seguinte apresenta alguns dos trabalhos que foram explorados nesta pesquisa.

2.4 TRABALHOS CORRELATOS

A criação de sistemas que facilitem o processo de consulta a modelos visuais de representação de conhecimento não é novidade na comunidade científica. Diversas pesquisas já foram realizadas neste sentido de modo que apresentaremos nesta seção alguns trabalhos presentes na bibliografia disponível.

2.4.1 TRIPLE

O TRIPLE (SINTEK e DECKER, 2002) é uma linguagem modelada com o objetivo de realizar consultas, inferências e transformações sobre especificações RDF⁴. As especificações RDF são responsáveis por atribuir valor semântico às trocas de informações na WEB. Essas especificações podem ser descritas na forma de triplas do

⁴ **RDF (Resource Description Framework)** – Uma recomendação da W3C - Órgão responsável por difundir padrões para WEB - para descrição de dados a fim de garantir a interoperação na WEB.

tipo A(O, V), onde O é o objeto que tem um atributo A e um valor V. Dois exemplos de uso deste formato de especificação são apresentados a seguir:

temNome("www.ppgi.inf.ufes.br/estudantes/2012130894", "Wagner").

curso("www.ppgi.inf.ufes.br/estudantes/2012130894", "www.ppgi.inf.ufes.br/cursos/mestrado").

Mas as especificações RDF também podem ser representadas na forma de grafos. A figura 2.2 apresenta um exemplo de um grafo RDF.

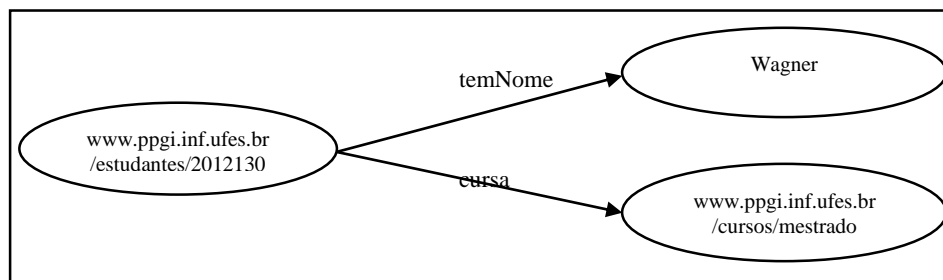


Figura 2.2 – Grafo RDF

Como pode ser observado na Figura 2.2, as triplas formadas pelos atributos, objetos e valores formam sentenças e estas podem, ainda, ser prontamente mapeadas para outras estruturas de dados que permitam integração de sistemas e favoreçam a comunicação, como apresentado no exemplo a seguir onde uma parte do grafo está mapeada para XML:

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:s="http://description.org/schema/">
<rdf:Description about=" www.ppgi.inf.ufes.br/estudantes/2012130894">
<s:temNome>Wagner</s:temNome>
</rdf:Description>
</rdf:RDF>
```

Uma tarefa fundamental na operação do TRIPLE é a transformação de especificações RDF em sentenças num formato (inspirado na sintaxe da linguagem F-Logic⁵) que permite a alimentação da base de conhecimento de um mecanismo de inferências. Esse mecanismo permite a criação de regras para consultas e inferências. O TRIPLE define um conjunto de especificações de operações de baixo nível que podem ser mapeados para consultas utilizando expressões semânticas em diversas linguagens superiores, tais como:

⁵ **F-Logic**: Uma linguagem dedutiva baseada em fatos e regras de consulta, similar ao Prolog.

Topic Maps, UML, DAML+OIL. Assim, o TRIPLE resulta numa linguagem híbrida, que suporta tanto expressões semânticas quanto regras para inferência.

Esta solução apresenta pontos pertinentes a esta pesquisa uma vez que muitos elementos de sua característica arquitetural é similar à arquitetura aqui proposta. Há no TRIPLE a representação de conhecimento por meio de grafos, o mapeamento destes para sentenças que alimentam uma base de IA que permitam definições de regras de inferência e a utilização de regras de IA para definição de regras de consulta a esta base.

2.4.2 ROSA

O ROSA (PORTO, MOURA, *et al.*, 2003) é um repositório de Objetos de Aprendizagem com acesso semântico que permite o gerenciamento de LOs⁶. É utilizado por profissionais da área educacional na preparação e busca de materiais didáticos que forneçam subsídios para preparação de suas aulas e/ou conteúdos instrucionais. Como ROSA, os usuários modelam seus cursos utilizando um Mapa Conceitual que permite ao projetista modelar e visualizar relacionamentos entre os LOs de sua autoria. Um exemplo de mapa de entrada para o ROSA pode ser visto na Figura 2.3, onde há a definição de alguns LOs utilizados num curso de Sistemas e Computação.

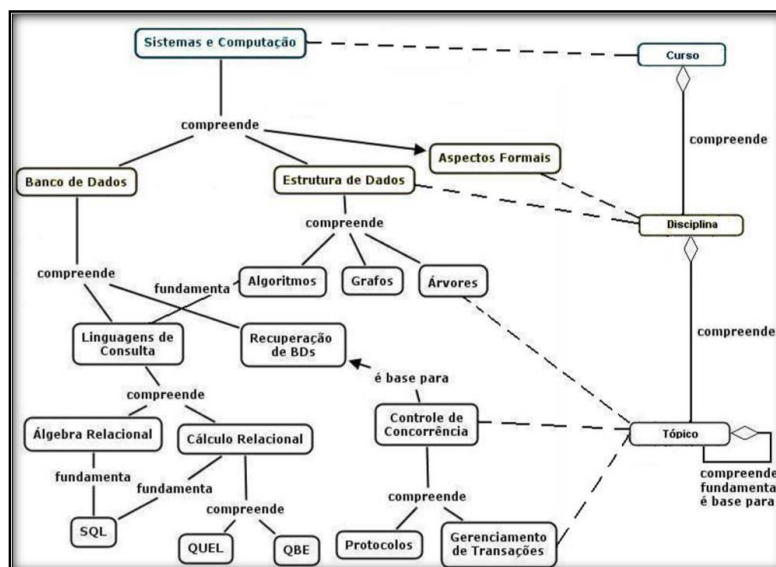


Figura 2.3 – Mapa Conceitual de Um curso de Sistemas e Computação
Fonte: Mattos, Moura *et al.*, 2006.

⁶ **LO (Learning Object):** É uma coleção de materiais reutilizáveis que dão suporte ao aprendizado, caracterizados por um conjunto de metadados definidos por um padrão de metadados.

O ROSA inclui uma álgebra bem definida (COUTINHO e PORTO, 2004) que permite tanto consultar os metadados dos LOs quanto formular consultas com base na semântica das associações, tais como: “*Que tópicos são **relevantes** para o ensino de Banco de Dados?*”, ou “*Que tópicos **compreendem** o capítulo de Otimização de Consultas?*”.

A pesquisa que deu origem ao ROSA abriu espaço para novas pesquisas que, por sua vez, resultaram de extensões de suas funcionalidades originais, tais como:

- ROSA-QL (PORTO, MOURA, *et al.*, 2003): Definição de uma Linguagem de Consulta que permite efetuar consultas do tipo *ad-hoc* ao ROSA.
- ROSAI (HANDRICK, 2005): Uma extensão do ROSA que utiliza a programação lógica para expressar mapas conceituais e regras, implementando herança e propriedades de relacionamentos, permitindo inferência de conhecimento não-explicito sobre sua base de inteligência.
- ROSA+ (MATTOS, MOURA e CAVALCANTI, 2006): Uma extensão do ROSA que visa deduzir conhecimento semântico através de propriedades de relacionamentos e de regras. Baseado na linguagem de ontologia OWL (*Ontology Web Language*) e na linguagem de regras SWRL (*Semantic Web Rule Language*), o ROSA+ realiza inferências sobre uma base de dados OWL, recuperando conhecimento não explicitado em sua representação ontológica.

Consideramos o ROSA e suas extensões os trabalhos correlatos que possuem maior aderência com esta pesquisa. O modelo que será apresentado nas próximas seções deste trabalho possui características que se sobrepõem, em especial, no que diz respeito ao ROSAI, que adota a representação em linguagem lógica para gerar mecanismos de inferências baseados no conhecimento presente nos mapas conceituais.

2.4.3 Semantic

O *Semantic* é uma ferramenta construída como fruto das pesquisas de Bessmertny e Kulagin (BESSMERTNY e KULAGIN, 2009) que investigam a utilização de redes semânticas como bases de conhecimento topologicamente estruturadas. Suas pesquisas objetivam avaliar a aplicação das redes semânticas como sistemas de treinamentos para a construção de bases de conhecimento destinadas a sistemas especialistas.

A Figura 2.4 relaciona os principais componentes do *Semantic*. Como é possível observar, o conhecimento de interesse é estruturado através da construção de uma rede semântica, item a) da figura. A rede então é decomposta em componentes do tipo triplas, seguindo o formato de predicado *Subject* → *Predicate* → *Object*, representado em b). Em seguida, cada predicado é transformado em um fato $f(\text{subject}, \text{predicate}, \text{object})$ para a base de conhecimento, item c) (RIBEIRO, 2012).

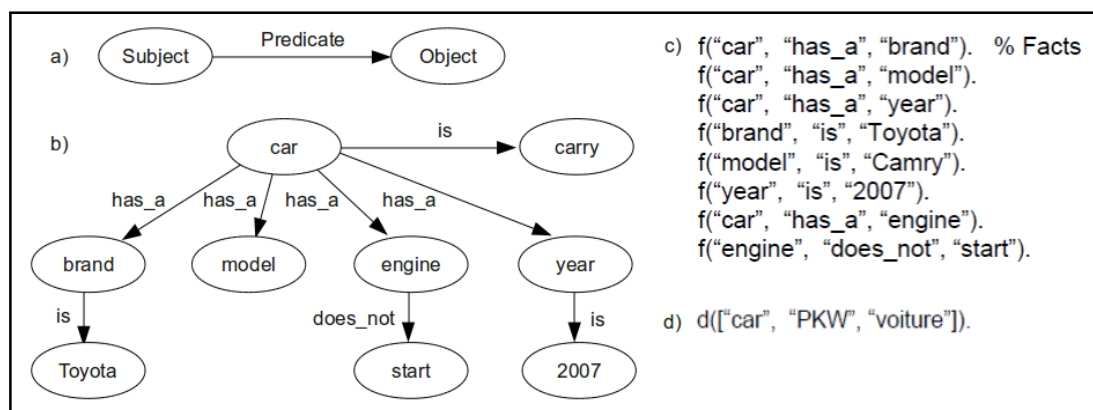


Figura 2.4 – Uma rede semântica e sua correspondente transformação em fatos para o *Semantic*
Fonte: Ribeiro, 2012.

Após esta transformação, o *Semantic* possui uma série de regras e uma gramática definida que permitem realizar consultas a essa base de conhecimento em busca de respostas a perguntas do tipo *WH-Questions*, conjunções e sinônimos definidos na rede semântica. A Figura 2.5 apresenta um exemplo de uso onde o *Semantic* responde corretamente à pergunta: “*What is the seal?*”.

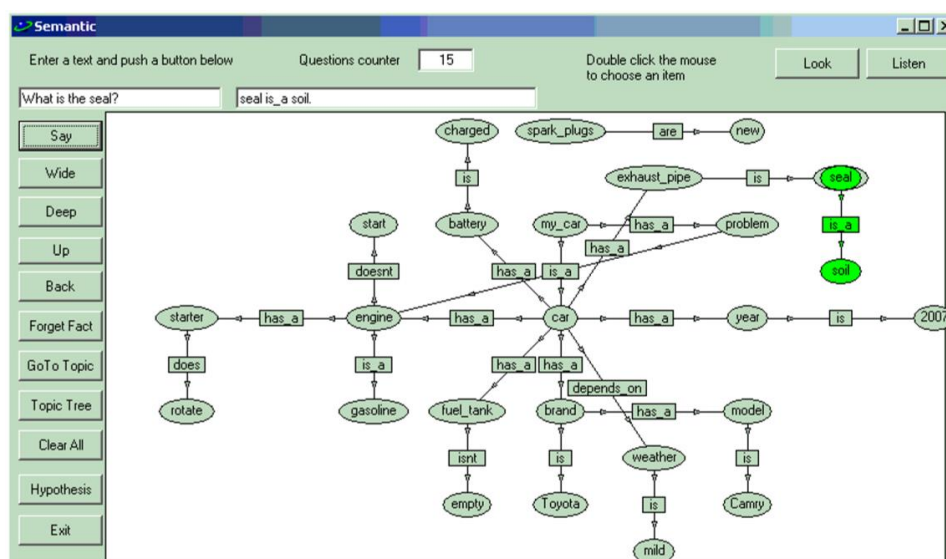


Figura 2.5 – Respostas obtidas com o *Semantic*
Fonte: Bessmertny & Kulagin, 2009.

Novamente, diversas características arquiteturais do *Semantic* são correlatas à arquitetura proposta nesta pesquisa. Vemos, novamente, características que vão desde a representação de conhecimento por meio de uma ferramenta visual e a posterior tradução para fatos numa base de inteligência, até a definição de uma gramática para interpretação de perguntas na realização de consultas à essa base.

2.5 CONSIDERAÇÕES

Como vimos neste capítulo, os Mapas Conceituais são ferramentas gráficas para organização, representação e construção do conhecimento. Assim, podem ser utilizados como ferramenta de apoio em abordagens pedagógicas. No entanto, o processo de análise e avaliação de mapas gerados por estudantes tende a ser cansativo ao docente/avaliador pois remete a uma pesada carga de processamento cognitivo.

É importante destacar ainda que os Mapas Conceituais podem ser definidos como conjuntos de conceitos ligados por uma rede de proposições. A formação de proposições é característica fundamental para composição de mapas computacionalmente interpretáveis. Por meio da análise de trabalhos correlatos, vimos que as proposições presentes em Mapas Conceituais podem ser mapeadas e alimentar uma base de inteligência de mecanismos de inferências.

Conhecendo os trabalhos correlatos e comparando-os à solução proposta foi possível especificar uma arquitetura conceitual geral para mecanismos de inferências a partir de representações visuais do conhecimento. O capítulo seguinte apresentará, com maior rigor, essa arquitetura conceitual, as técnicas e as abordagens que são empregadas em seus processos.

CAPÍTULO 3 ARQUITETURA E TECNOLOGIAS

"O conhecimento cresce exponencialmente. Quanto mais soubermos, maior a nossa capacidade de aprender, e mais rápido expandimos a nossa base de conhecimento."

(Dan Brown)
"O Símbolo Perdido"

Este capítulo apresenta a arquitetura conceitual dos mecanismos de inferências para modelos visuais de representação de conhecimento: como se organizam, quais seus processos e quais são as técnicas e tecnologias adotadas. Cada uma das soluções tecnológicas utilizadas é apresentada e explorada a fim de conhecer suas potencialidades e aplicações para posterior definição da arquitetura proposta nesta pesquisa.

Assim, este capítulo está dividido nas seguintes seções:

- **Seção 3.1:** Apresenta a arquitetura conceitual geral dos mecanismos de inferências baseados em representações visuais de conhecimento, seus atores, componentes, principais atividades e processos.
- **Seção 3.2:** Apresenta as técnicas e tecnologias adotadas na maioria dos mecanismos de inferências em mapas conceituais e pontua aqueles que serão adotados e/ou utilizados na arquitetura proposta por esta pesquisa.
- **Seção 3.3:** Apresenta algumas considerações sobre a arquitetura, técnicas e tecnologias apresentadas neste capítulo.

3.1 ARQUITETURA CONCEITUAL

Por meio das pesquisas realizadas e da análise dos trabalhos correlatos (Capítulo 2) foi possível definir uma arquitetura conceitual desses sistemas. As interações entre os atores, as atividades e os componentes dos sistemas estão sintetizados na Figura 3.1.

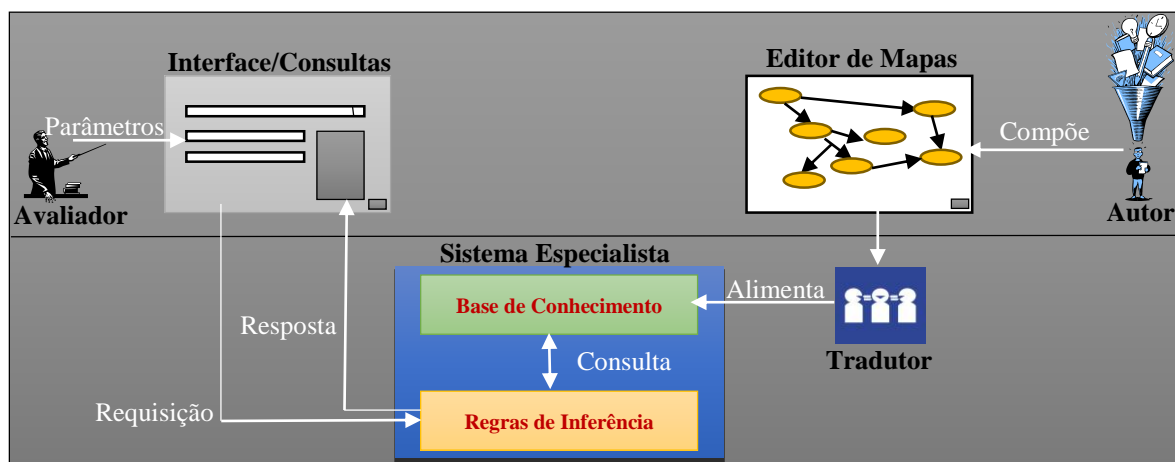


Figura 3.1 – Arquitetura Conceitual dos Mecanismos de Inferência em Representações Visuais de Conhecimento

Como observado na Figura 3.1, os mecanismos de inferências necessitam da atuação de dois atores, quais sejam:

- **Autor:** É o ator responsável por representar seu conhecimento utilizando representações gráficas (geralmente mapas conceituais). Em abordagens pedagógicas estes seriam os alunos.
- **Avaliador:** É o ator responsável por analisar/avaliar o conteúdo do mapa conceitual. Seu desejo, em geral, é extrair das representações gráficas as nuances de conhecimento ali presentes, geralmente expressas pelas proposições formadas pelas relações existentes entre os conceitos ali presentes. Em abordagens pedagógicas, este papel é geralmente desempenhado pelo professor.

Para favorecer a interação destes dois atores, os mecanismos de inferências em mapas geralmente oferecem duas interfaces, a saber:

- **Editor de Mapas:** Interface que permite ao usuário (Autor) representar seu conhecimento por meio de ferramentas visuais de edição. O conhecimento representado é posteriormente traduzido para compor uma base de conhecimento de um sistema especialista.

- **Interface de Consultas:** Interface que favorece a realização de consultas à base de conhecimento do sistema. Permite ao usuário (Avaliador) definir os parâmetros de entrada para realização da consulta. No geral, essa mesma interface recebe, trata e apresenta os resultados obtidos pela execução da consulta.

Na parte inferior da Figura 3.1 percebemos que as bases do mecanismo, ou seja, seus componentes funcionais, são compostos por um mecanismo **tradutor** que alimenta a base de conhecimento de um **sistema especialista** capaz de processar as requisições dos avaliadores utilizando regras de inferência.

O **tradutor** é essencial para o funcionamento dos mecanismos de inferências. Eles atuam com o objetivo de transformar o conhecimento presente nos mapas numa representação formal aceita para compor a **base de conhecimento do sistema especialista**, que podem ser ontológicos ou na forma de predicados de primeira ordem, dependendo do raciocinador utilizado pelo sistema. Na Seção 2.4 foi possível observar a atuação do tradutor para alimentar bases de inteligência tanto em representação ontológica (ocorrendo no *TRIPLE* e no *ROSA+*) quanto na representação em forma de predicados de primeira ordem (ocorrendo no *Semantic* e demais versões do *ROSA*).

Após ter sua base de conhecimento alimentada pelo conhecimento presente nos mapas, o **sistema especialista** é responsável por receber, processar e responder a perguntas construídas pelo usuário (avaliador). Para isso, ele possui um conjunto finito de **regras de inferência** em sua base. Assim, as regras de inferência definidas na base determinam a forma de interação do usuário com o sistema.

As **regras de inferência** possuem, no geral, sintaxes que devem ser rigidamente obedecidas pelo usuário na composição das perguntas. Isso é uma limitação comum às soluções analisadas na Seção 2.4 já que prejudica a experiência de usuário na utilização dessas ferramentas. Das soluções analisadas nesta pesquisa, a única que explora essa limitação fornecendo uma forma mais “humanizada” na composição de consultas à base é o *Semantic* que é capaz de responder a algumas perguntas do tipo *WH-Questions* e de sinônimos. No entanto, a estrutura sintática definida por essa solução ainda é rígida e deve ser rigorosamente respeitada para seu correto funcionamento. Nesta pesquisa definimos

uma abordagem que permite maior expressividade por parte do avaliador na composição de suas perguntas.

A seção seguinte irá apresentar as características tecnológicas e as técnicas que dão suporte à organização de mecanismos de inferências e que serão adotados na arquitetura proposta nesta pesquisa.

3.2 TÉCNICAS E TECNOLOGIAS

Em nossas pesquisas identificamos aspectos arquiteturais necessários para classificar o mecanismo de inferências em mapas conceituais proposto como sistemas de perguntas-respostas já que, segundo Maybury (2004), um sistema pode ser assim classificado se atender aos seguintes aspectos funcionais:

- **Modularidade:** O sistema deve encapsular e separar (definir bem os limites) os módulos (análise da pergunta, recuperação, extração, formulação da resposta, seleção da resposta, entre outras), e os módulos devem permitir integração para apoiar avaliação comparativa, reuso, entre outros.
- **Múltiplo/Flexível fluxo de dados:** O sistema deve apoiar uma variedade de interconexões entre os componentes.
- **Fontes de dados independentes de contexto:** O sistema deve incorporar abstrações apropriadas para apoiar acesso direto a fontes de dados em múltiplos domínios.
- **Raciocínio Inferencial:** O sistema deve produzir respostas que não podem ser encontradas em pesquisas diretas, mas deve utilizar inferências a partir das fontes de dados.

Adotamos também sistemas inteligentes, já que são necessários processamentos inferenciais para produzir respostas às perguntas construídas pelos professores. Além disso, como as perguntas são construídas em linguagem natural, esta pesquisa investigou também como a pergunta pode ser processada, o que envolveu adotar técnicas de Processamento de Linguagem Natural.

Assim, é necessário analisar mais de perto essas técnicas e as tecnologias adotadas por estes sistemas a fim de verificar suas potencialidades e, posteriormente, descrever uma nova arquitetura. Iniciaremos por analisar as características e arquitetura dos sistemas de pergunta-resposta e as técnicas que são adotadas em sua concepção.

3.2.1 Sistemas de pergunta-resposta

Sistemas de pergunta-resposta é uma área de investigação da Ciência da Computação que engloba técnicas de recuperação da informação e processamento de linguagem natural com objetivo de construir sistemas para resposta automática a perguntas compostas por humanos numa linguagem natural. No geral, sua arquitetura inclui um banco de dados usualmente conhecido como base de conhecimento. Esse é o local onde são armazenadas as informações que servem de base para a busca por respostas.

O primeiro sistema de pergunta-resposta que se tem relato é o BASEBALL (GREEN, WOLF, *et al.*, 1961), um programa para responder a perguntas sobre torneios de baseball disputados numa temporada da liga americana. Esse sistema era capaz de responder a questões realizadas em linguagem natural, tais como: “De quem Red Sox perdeu no dia 05 de julho?” ou “Quantos jogos Yankees jogaram em julho?”. O BASEBALL era capaz de analisar a pergunta usando conhecimentos linguísticos, de forma canônica, e gerava uma consulta em uma base de dados estruturada sobre BASEBALL.

Os primeiros sistemas de pergunta-resposta eram compostos, essencialmente, por um *front end* que realizava a análise, interpretação e mapeamento das perguntas redigidas em termos comuns (linguagem natural) para formatos mais específicos a serem processados pelo *back end* que em geral eram bancos de dados relacionais. As primeiras incursões em banco de dados para sistemas de pergunta-resposta foram abandonadas no final de 1980 por razões que incluíam limitações tecnológicas relacionadas à confiabilidade do processamento de linguagem natural (CLARK, FOX e LAPPIN, 2010).

Moldovan, *et al.* (1999) propõem uma taxonomia de classificação dos sistemas de pergunta-resposta dividida em cinco classes, considerando três características principais: (a) o nível de conhecimento, (b) o nível de raciocínio e (c) a indexação e técnicas de PLN utilizadas. Tempos depois, os mesmos autores caracterizaram esses sistemas de acordo

com a complexidade das perguntas e a dificuldade de extração de respostas (MOLDOVAN *et al.*, 2003). Percebemos que a área de pesquisa de um sistema de pergunta-resposta envolve a intersecção de muitos campos científicos que inclui o processamento de linguagem natural, recuperação inteligente da informação, interação humano-computador, representação do conhecimento, raciocínio para interpretação de perguntas e análise das respostas, algoritmos para encontrar respostas preferenciais, extração em fontes de áudio ou vídeo, entre outras (MAYBURY, 2004).

Quanto à arquitetura dos sistemas de pergunta-resposta, Amorim (2012) afirma que são dirigidos para: (a) alguma série de perguntas; (b) processar uma variedade de fontes (documentos, páginas web, bancos de dados etc) a fim de (c) produzir respostas para os usuários. Desta forma, a arquitetura genérica de sistemas pergunta-resposta é modular e integrada conforme apresentado na Figura 3.2.

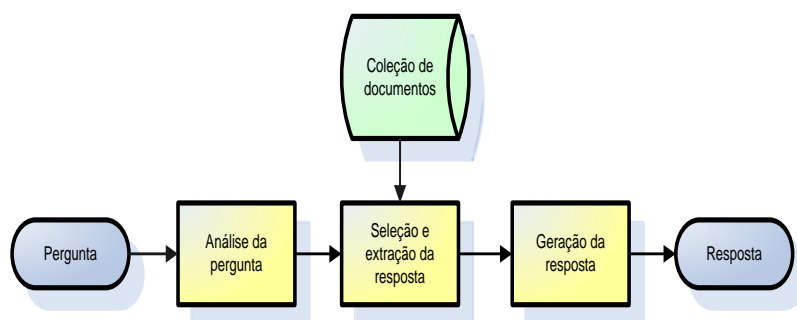


Figura 3.2 – Arquitetura Conceitual de Sistemas de Pergunta-Resposta
Fonte: Amorim (2012)

Como é possível perceber na Figura 3.2, o processamento de um sistema pergunta-resposta consiste em três fases principais: a análise da pergunta, seleção e extração da resposta e a geração da resposta (ATHENIKOS e HYOIL, 2010). Nas próximas seções detalharemos o mecanismo de funcionamento e as tecnologias empregadas em cada um destes três componentes da arquitetura.

3.2.1.1 Análise da Pergunta

A relativa ineficácia dos sistemas de recuperação de informação é, em grande parte, causada pela imprecisão com que uma consulta formada por algumas palavras chaves modela a informação que o usuário necessita (AMORIM, 2012). Para melhorar o estágio de recuperação dos sistemas pergunta-resposta diversas estratégias estão sendo

desenvolvidas e técnicas atuais estão sendo empregadas, tais como: ontologias, *Wordnet*, análise sintática, análise semântica de perguntas baseada em regras, estatística, entre outras (CARPINETO e ROMANO, 2012) (CHIRITA, FIRAN e NEJDAL, 1982).

Muitos sistemas usam na etapa de análise da pergunta módulos para reconhecimento do tipo da pergunta baseado na estrutura sintática e no tipo semântico da resposta esperada (PRAGER, BROWN e CODEN, 2000). Algumas pesquisas interessadas em garantir o processamento de linguagem natural adotam frequentemente ferramentas de apoio à análise sintática textual, tais como o *Wordnet* (MILLER, BECKWITH e FELBAUM, 1993) para enriquecimento da base de dados e da pergunta e o *Visl* (BICK, 2000) para análise sintática e semântica da pergunta. A análise da pergunta possui duplo propósito: extrair palavras-chaves que definam o tipo de resposta esperada e definir os parâmetros necessários para extração da resposta. Muitos sistemas não apenas extraem palavras chaves na análise da pergunta para utilizar como termos de uma consulta, mas além disso modificam a consulta conforme o tipo da pergunta (MONZ, 2003).

Nesta fase geralmente são necessários dois processos para identificar qual o tipo da informação está sendo procurado (classificação da pergunta) e quais os parâmetros necessários para serem encontradas (construção da consulta) (CLARK, FOX e LAPPIN, 2010). A classificação da pergunta visa associar uma etiqueta, indicando o tipo da informação a ser procurado – por exemplo, um conceito, relação ou dado estatístico. As etiquetas atribuídas para pergunta têm sido usadas para apoiar a recuperação de texto por meio de anotação preditiva (PRAGER, BROWN e CODEN, 2000), assim como para apoiar o processo envolvido com a identificação e ordenação das perguntas candidatas. Podem ser construídas regras para classificação da pergunta.

Os mecanismos de inferências analisados nesta pesquisa possuem, em geral, regras bem definidas para realização de consultas e inferências a partir das informações presentes na base de conhecimento. Essas regras possuem em comum a característica de possuírem uma sintaxe rígida e, muitas vezes, pouco amigáveis ao usuário final. Em outras palavras, a utilização dessas regras requer, quase sempre, um treinamento dos usuários devido à sua incapacidade de processar as perguntas construídas em linguagem natural.

Para resolver o problema da interpretação de perguntas construídas em Linguagem Natural, Gong e Chan (2005) propõem uma metodologia que se destina a reduzir o *gap semântico* entre as perguntas construídas em linguagem natural e as perguntas interpretadas por sistemas especialistas. O *gap semântico* é facilmente compreendido por pessoas que trabalham com tradução. Muitas vezes, uma frase construída num dado idioma não possui uma correspondente exata no idioma-alvo. Na maioria das vezes, o que o tradutor faz é construir uma frase que aproxime a correspondência semântica das frases em ambos os idiomas. Assim, essa frase “re”construída por ele reduz o *gap semântico* que distancia esses dois idiomas.

De forma similar, Gong e Chan (2005) afirmam que este objetivo pode ser alcançado por utilizar, no processo de análise da pergunta, passos que incluem a filtragem e identificação de palavras-chaves essenciais da pergunta original, construída em linguagem natural, e que devem ser utilizadas pelo sistema especialista na realização de buscas por respostas. Nesta pesquisa, denominamos o idioma da pergunta “re”construída com termos essenciais do original de *gramática intermediária* cuja finalidade é reduzir o *gap semântico* existente entre a pergunta em linguagem natural e a pergunta correspondente no idioma interpretado pelo sistema especialista. Para isso, utilizamos técnicas de processamento de Linguagem Natural que são analisadas com mais detalhes na Seção 3.2.3.

3.2.1.2 Seleção e Extração da Resposta

Nos mecanismos de inferência para modelos visuais de representação do conhecimento, a seleção e extração da resposta é a fase responsável por recuperar as respostas apropriadas às perguntas realizadas pelos avaliadores em linguagem natural. O maior desafio a ser vencido nesta etapa é identificar a melhor resposta à pergunta. Isso ocorre porque, como vimos, o *gap semântico* faz com que o processo de análise da pergunta uma pergunta em linguagem natural possua, às vezes, mais de uma forma de tradução na linguagem esperada pelo sistema especialista. Denominamos essas ocorrências de *múltiplas correspondências*. Analisamos na literatura duas abordagens para recuperação da resposta: recuperação baseada em relevância e recuperação baseada em padrão.

Dentro do contexto desta pesquisa, na recuperação baseada em relevância as consultas são interpretadas como pedidos de conceitos relevantes para um determinado assunto.

Essa relevância pode ser avaliada por meio de uma combinação booleana dos termos, vetor de termos ponderado ou um modelo de linguagem (CLARK, FOX e LAPPIN, 2010). Outra técnica de recuperação usada é a anotação preditiva, através da qual os conceitos presentes num mapa poderiam ser indexados por meio de cada um dos 20 tipos de entidades, cada qual podendo responder uma pergunta de um tipo determinado (PRAGER, BROWN e CODEN, 2000). Esse tipo de notação pode ser usada como parte da construção da consulta para recuperação da pergunta. Quando a pergunta é mapeada para a consulta, não apenas as palavras-chaves são incluídas na consulta, mas também entidades apropriadas para o tipo de pergunta (CLARK, FOX e LAPPIN, 2010).

Já a recuperação baseada em padrão difere da recuperação baseada em relevância no fato de retornar um pedaço como evidência de uma correspondência (combinação). Consideramos esse tipo de recuperação a forma ideal a ser aplicada na recuperação de informações presentes nas proposições extraídas dos mapas conceituais, já que os padrões geralmente refletem relacionamentos diretos entre a pergunta e sua resposta. Algumas pesquisas apontam, ainda, que esse tipo de recuperação pode ser muito enriquecido com a utilização de ferramentas tais como o *Wordnet* e o *FrameNet* (CLARK *et al.*, 2010).

No Capítulo 4 demonstraremos como adotamos a recuperação baseada em padrões, no componente denominado *solucionador de correspondências*, na seleção e na definição da ordem de prioridade das perguntas candidatas nas ocorrências de *múltiplas correspondências*.

3.2.1.3 Geração da Resposta

O módulo de geração da resposta é responsável por extrair, da base de conhecimento proposicional, informações necessárias para construir uma resposta para a pergunta. Apesar de existirem na literatura sistemas que utilizam diversos formatos de mídias para expor as respostas, como por exemplo: som e imagem (BOSCH e BOUMA, 2011), vídeos (LEI, LI, *et al.*, 2010), serviços (WANG, LIAO e WANG, 2012), múltiplos idiomas (FERRÁNDEZ, SPURK, *et al.*, 2011), diálogos (KIYOTA, KUROHASHI e KIDO, 2002); nesta pesquisa estamos interessados que as respostas obtidas sejam textuais, já que a base de conhecimento do sistema possui apenas proposições extraídas dos mapas. É importante destacar que alguns sistemas utilizam *feedback* (JURCZYK e AGICHTEIN,

2007) para melhorar o processo de recuperação da respostas. Percebemos, portanto, que existem diversas possibilidades para apresentar as respostas e melhorar a interação com o usuário.

Como o processo de interpretação da pergunta é fundamental para o funcionamento adequado de um sistema de pergunta-resposta, a seção seguinte discute algumas técnicas que são fundamentais para a interpretação de perguntas construídas em linguagem natural.

3.2.2 Processamento de Linguagem Natural (PLN)

O Processamento de Linguagem Natural (PLN) consiste no desenvolvimento de técnicas e modelos computacionais para a realização de tarefas que dependem de informações expressas em linguagem natural (COVINGTON, 1994; RUSSEL e NORVIG, 1995). Diversas tarefas atuais dependem funcionalmente do PLN para ocorrerem, tais como: tradução e interpretação de textos; busca de informações em documentos; interface homem-máquina etc.

Para Covington, Nute e Vellino (1997), as pesquisas em PLN estão voltadas, essencialmente, a três aspectos da comunicação em língua natural, quais sejam:

- *Som (fonologia)*: relacionado ao reconhecimento dos sons que compõem as palavras de uma língua;
- *Estrutura*: relacionado ao reconhecimento das unidades primitivas que compõem uma palavra (morfológica) e da forma como as palavras se relacionam na composição de uma frase (sintática);
- *Significado*: relacionado à associação de uma estrutura sintática ao significado das palavras que compõem uma frase (semântica) e a verificar se tal associação é a mais apropriada no contexto (pragmática).

Perceptivelmente, o PLN é uma área de pesquisa que envolve diversas disciplinas do conhecimento humano e, em ambiente computacional, representa um desafio considerável, já que o computador, cujo significado é “aquele que faz computos”, não foi projetado com a capacidade de realizar interpretações linguísticas. Nesta pesquisa

investigamos abordagens para análise sintática de algumas perguntas construídas em linguagem natural no idioma português.

As técnicas de processamento de linguagem natural investigadas e adotadas na definição da arquitetura proposta por essa pesquisa são apresentadas na subseções seguintes.

3.2.2.1 Marcação / Etiquetagem

A técnica de marcação⁷, ou etiquetagem, consiste de um artefato de *software* capaz de receber, como entrada, um texto escrito em linguagem natural e atribuir para cada parte deste texto a função sintática desempenhada por essa parte do texto. No geral, os *softwares* que realizam essa tarefa o fazem por inserir, na frente de cada palavra do texto, símbolos (etiquetas morfossintáticas) que indicam se aquele elemento é um substantivo, adjetivo, verbo e assim por diante (TOUTANOVA, KLEIN, *et al.*, 2003).

São diversos os etiquetadores existentes, estando estes diretamente relacionados ao idioma que é capaz de processar. Em português, por exemplo, identificamos a existência dos seguintes etiquetadores: Aelius (ALENCAR, 2010); VISL (BICK, 1996, 2000); Tree-Tagger e FreeLing (GAMALHO e GARCIA, 2013; ATSERIAS, CASAS, *et al.*, 2006).

Nesta pesquisa adotamos o VISL como mecanismo de etiquetagem por considerar fatores como tempo de maturidade da aplicação, relevância das publicações e resultados documentados na bibliografia existente (BICK, MELLO, *et al.*, 2012), além da facilidade para adequação à arquitetura WEB adotada nesta pesquisa. Um exemplo de resultado produzido pelo VISL pode ser observado na Figura 3.3.

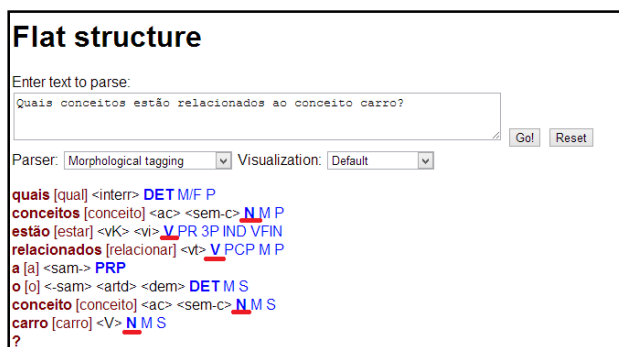


Figura 3.3 – Etiquetagem realizada pelo VISL

⁷ Em inglês o termo adotado é *tagging*.

Em destaque na Figura 3.3 vemos como essa ferramenta foi capaz de identificar os substantivos (assinalados com a letra N) e os verbos (assinalados com a letra V) presentes na pergunta original. Como veremos no Capítulo 4, os resultados produzidos neste processo de marcação são processados a fim de verificar, por meio de uma gramática intermediária, se a pergunta possui características que a tornem válidas para o processamento pelo mecanismo proposto. A subseção seguinte explicará melhor como foi definida a gramática intermediária adotada na pesquisa.

3.2.2.2 Gramática Independente de Contexto

A fim de reduzir o *gap semântico* entre a linguagem utilizada na construção da pergunta (linguagem natural) e sua correspondente na linguagem de baixo nível definida pela arquitetura proposta nesta pesquisa, percebemos a necessidade de definir uma gramática intermediária livre de contexto e uma abordagem computacional capaz de processar um conjunto finito de questões e decidir qual a melhor tradução destas perguntas para o universo de perguntas a cujo mecanismo proposto é capaz de produzir respostas.

Uma gramática é uma especificação formal da estrutura das sentenças permitidas em uma linguagem. O modo mais comum de se especificar uma gramática é por definir um conjunto de símbolos terminais, denotando palavras da linguagem, um conjunto de símbolos não-terminais, denotando os componentes das sentenças, e um conjunto de regras de produção que expandem símbolos não-terminais numa sequência de símbolos terminais e não-terminais. Além disso, a gramática deve ser um símbolo não-terminal inicial (Rich, 1995). Em computação, a técnica de notação mais utilizada para descrever gramáticas livre de contexto é o BNF (*Backus-Naur Form*). Ele utiliza uma sintaxe declarativa que permite a definição dos termos da linguagem via regras de produção. Cada regra contém termos em que cada um deles é expandido até atingir os elementos terminais, que são termos descritos com caracteres literais. Um exemplo de definição em notação BNF pode ser visto na Figura 3.4.

```

<frase> ::= <sujeito> <predicado>
<sujeito> ::= <artigo> <substantivo>
<predicado> ::= <verbo> <artigo> <predicado>
<artigo> ::= o
<substantivo> ::= gato | rato
<verbo> ::= caçou

```

Figura 3.4 – Definição de uma frase utilizando notação BNF

Na gramática definida na Figura 3.4, os símbolos terminais são, gato, rato e caçou, sendo os demais símbolos os não-terminais. A regra de produção $\langle frase \rangle ::= \langle sujeito \rangle \langle predicado \rangle$ estabelece que uma frase é composta de um sujeito seguido de um predicado; enquanto a regra $\langle substantivo \rangle ::= gato / rato$ estabelece que um substantivo pode ser a palavra “gato” ou “rato”. Além disso, para essa gramática, o símbolo não-terminal inicial será $\langle frase \rangle$. Assim, um exemplo de frase possível para as regras definidas na Figura 3.4 seria: “O gato caçou o rato”.

Nas gramáticas livres de contexto, o lado esquerdo de uma regra será sempre um único símbolo não-terminal, enquanto o lado direito pode conter símbolos terminais ou não-terminais. As gramáticas livres de contexto podem ser usadas para *reconhecer*, ou seja, verificar se uma frase pertence à linguagem definida pela gramática, ou para *gerar*, ou seja, para construir uma frase pertencente à linguagem definida pela gramática.

Nos mecanismos de inferências analisados, e no proposto por esta pesquisa, o responsável por inferir e gerar as respostas às perguntas é um sistema especialista. Assim, a seção seguinte apresentará a organização desse tipo de sistema e como é utilizado no contexto desta pesquisa.

3.2.3 Sistemas Especialistas

Sistemas especialistas são sistemas cujo objetivo é simular o raciocínio/comportamento de um especialista em sua área específica de atuação. Por exemplo: um sistema especialista em “câncer de mama” poderia, com base em alguns dados informados pelos usuários, fornecer um diagnóstico e dar aconselhamentos similares aos que receberia numa consulta convencional com um médico especialista (FALLAHI e JAFARI, 2011). Sua versatilidade faz com que suas aplicações sejam amplas, de forma que existe na bibliografia atual relatos de aplicações nas mais diversas áreas da sociedade, desde em aplicações comerciais, como análise de crédito bancário (MAHMOUD, ALGADI e ALI, 2008), até no monitoramento de colisões aéreas (ATKINSON, JAMES e LAWSON, 1990).

Nos mecanismos de inferências em mapas conceituais, os sistemas especialistas são utilizados para simular o comportamento do autor do mapa analisado. O objetivo é que o

avaliador possa interagir com o sistema especialista que estará reproduzindo o conhecimento do autor do mapa. Assim, é importante conhecer com mais detalhes os componentes da arquitetura dos sistemas especialistas.

3.2.3.1 Arquitetura do Sistemas Especialistas

Os sistemas especialistas são aplicações computacionais que incorporam alguns conhecimentos não-algorítmicos para resolver certos tipos de problemas que comumente dependem da atuação de um especialista na tomada de decisões. Suas aplicações são exploradas em jogos de xadrez, mercado financeiro e outros serviços geralmente desempenhados por humanos (MERRITT, 1989).

Sua arquitetura possui três componentes principais e envolve atuação de quatro indivíduos em diferentes papéis. A Figura 3.3 apresenta a arquitetura básica desses sistemas.

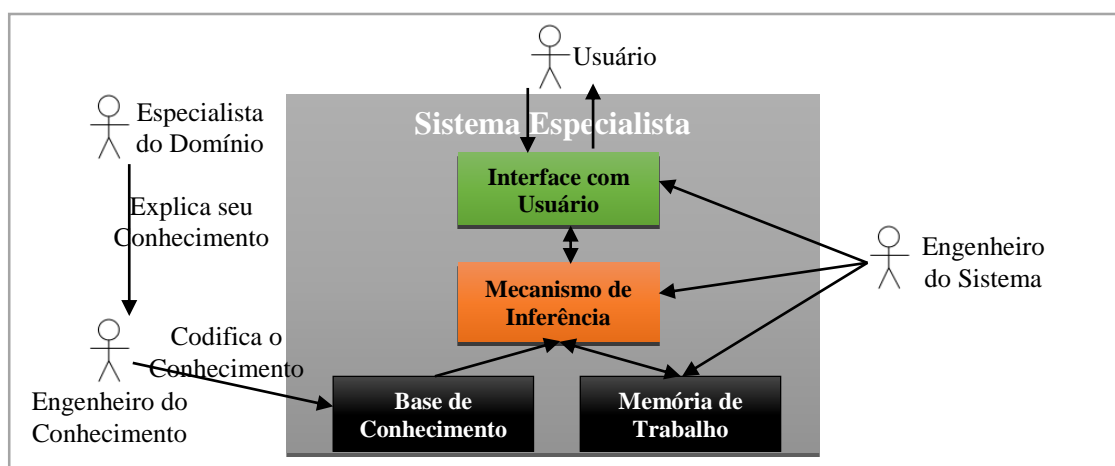


Figura 3.5 – Arquitetura dos Sistemas Especialistas

Como é possível observar na Figura 3.5, os principais componentes dos sistemas especialistas são:

- **Base de Conhecimento:** Representações declarativas de conhecimento, muitas vezes definidas pelo padrão SE... ENTÃO.
- **Memória de Trabalho:** Dados informados pelo usuário que especificam o problema a ser resolvido;

- **Mecanismo de Inferência:** Principal componente do sistema, que processa as recomendações da base de conhecimento para os dados específicos do problema presentes na memória de trabalho;
- **User Interface:** Responsável por controlar o diálogo entre o usuário e o sistema.

Para compreender a arquitetura dos sistemas especialistas, é necessário entender as principais funções desempenhadas pelos indivíduos que interagem com o sistema, também representados na Figura 3.5. São eles:

- **Especialista do Domínio:** A pessoa, ou conjunto de pessoas, que atualmente são os especialistas em resolver o problema que o sistema se destina a resolver;
- **Engenheiro do Conhecimento:** O programador responsável por codificar o conhecimento dos especialistas num formato declarativo o qual o sistema é capaz de processar;
- **Usuário:** Indivíduo que realiza consultas com o sistema especialista a fim de obter respostas que seriam igualmente fornecidas pelos especialistas do domínio.

Muitos sistemas utilizam a arquitetura de sistemas especialistas para resolver problemas específicos. Estes são comumente chamados de *expert system shells*. O *shell* é uma parte do sistema que contém uma interface, uma base para conhecimento declarativo e um motor para inferências. O engenheiro de sistema utiliza o *shell* para resolver problemas particulares em sistemas complexos. Assim, um indivíduo importante para construção de um sistema especialista customizado é o:

- **Engenheiro do Sistema:** Indivíduo que constrói a interface do usuário, projeta o formato declarativo da base de conhecimento e implementa o motor de inferências.

Dependendo do tamanho do projeto, o papel de engenheiro do sistema e de engenheiro do conhecimento podem ser desempenhados pela mesma pessoa já que, em sistemas personalizados, a definição do formato da base de conhecimento e a codificação do conhecimento do domínio estão intimamente relacionados.

Um dos principais gargalos na construção de sistemas especialistas é justamente o processo de codificação do conhecimento, pois sua representação no formato declarativo pode ser uma tarefa difícil e tediosa. Uma vantagem de utilizar *shell* personalizado é que o formato da base de conhecimento pode ser projetado para facilitar o processo de engenharia de conhecimento. O objetivo deste processo é reduzir o *gap semântico*, que pode ser compreendido como a diferença entre a representação natural de um conhecimento e sua representação programática. Por exemplo, compare a diferença semântica existente entre a representação de uma fórmula matemática em linguagem de montagem (assembler) e sua representação em linguagem de alto nível, como o JAVA, por exemplo. Os códigos em JAVA possuem um *gap semântico* menor e, portanto, são mais fáceis de serem programados (MERRITT, 1989).

Uma vez que o principal entrave para o desenvolvimento do sistema especialista é a construção da base de conhecimento, é lógico que a diferença semântica entre a representação do conhecimento do construtor dos mapas conceituais e da representação na base de conhecimento deve ser minimizada. Com um sistema personalizado, o engenheiro de sistema pode implementar uma base de conhecimento cujas estruturas são o mais próximo possível aos utilizados pelo especialista de domínio. Isto é feito em todos os mecanismos de inferências analisados na Seção 2.4 desta pesquisa e também é realizado no mecanismo proposto e que será descrito no Capítulo 4.

Existem diversas linguagens voltadas para construção de sistemas especialistas. Hall & Kandel(1986) realizam um estudo comparativo das linguagens FORTRAN, Modula-2, Ada, Pascal, LISP e Prolog. Laerhoven(1999) compara as ferramentas CLIPS e JESS, linguagens utilizadas na construção de sistemas baseados em regras (uma subcategoria de sistemas especialistas muito utilizadas em aplicações comerciais para definição de regras de produção), que também incluem Drools e MS-BRE (Microsoft Business Rule Engine).

Nesta pesquisa utilizaremos a linguagem *Prolog* para o sistema especialista proposto já que ele possui características estruturais que facilitam o processo de transformação das proposições dos mapas em fatos para a base de conhecimento. Além disso, permite a construção de regras de inferências utilizando fórmulas lógicas com padrões baseados na cláusula de Horn (HORN, 1951).

3.3 CONSIDERAÇÕES

Este capítulo teve por objetivo apresentar um resumo da arquitetura geral dos mecanismos de inferência em representações visuais do conhecimento, tais como os mapas conceituais.

Destacamos que esses sistemas têm por objetivo facilitar o processo de extração do conhecimento presente nessas representações por meio de perguntas e respostas que, em muitos casos, utilizam sistemas especialista para favorecer o processo de inferência para construção de respostas. Assim, este capítulo explorou a organização desses sistemas e as tecnologias empregadas na sua implementação.

Conhecendo essas técnicas e tecnologias, o Capítulo 4 irá apresentar a arquitetura proposta por esta pesquisa. Seus componentes são explorados e as integrações são apresentadas.

CAPÍTULO 4 A ARQUITETURA PROPOSTA

"O projeto é o rascunho do futuro."

(Jules Renard)

Este capítulo apresenta a arquitetura conceitual do sistema proposto, a saber: um mecanismo de inferências para mapas conceituais dotado da capacidade de interpretar e responder a questões independentes de contexto construídas em linguagem natural. Conhecer essa arquitetura é fundamental para compreender o funcionamento do protótipo e dos resultados obtidos e apresentados no Capítulo 5.

Para descrever a arquitetura conceitual deste projeto, este capítulo está organizado da seguinte maneira:

- Seção 4.1: apresenta a visão geral da arquitetura onde é possível identificar os principais módulos e componentes do sistema e onde são descritas suas funções e atividades de usuários.
- Seção 4.2: descreve as atividades e o comportamento sequencial da arquitetura a fim de compreender seu funcionamento interno.
- Seção 4.3: Apresenta algumas considerações o apresentado neste capítulo.

4.1 VISÃO GERAL DA ARQUITETURA

Em engenharia de *software*, para distinguir a especificação funcional da implementação do sistema utilizamos, com frequência, os termos “O que” e “Como”, respectivamente. Por definir “O que” empenhamo-nos em definir as funcionalidades do sistema e as possíveis interações com o usuário. Já o “Como” relaciona-se às funcionalidades que serão implementadas (exemplo: quais técnicas ou tecnologias serão adotadas na codificação de um algoritmo) (FENSEL, 2000). Esta seção objetiva apresentar “O que” o sistema deve fazer. A Figura 4.1 objetiva facilitar essa compreensão.

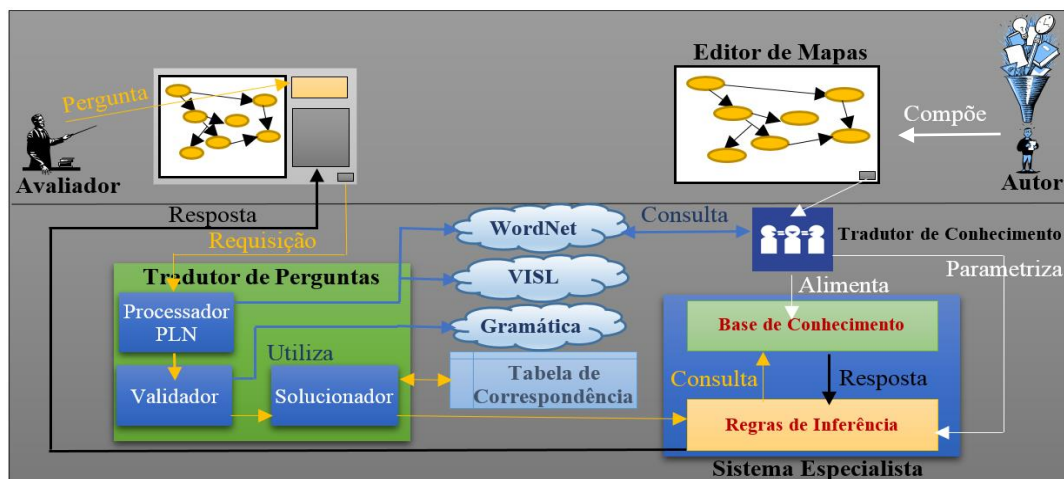


Figura 4.1 – Visão Geral da Arquitetura Proposta.

Na parte superior da Figura 4.1 é possível observar as duas interfaces de interação com o sistema: uma de edição de mapas conceituais e uma para composição de perguntas para consulta à base de inteligência. Observa-se também os atores que interagem por meio destas, quais sejam: autor e avaliador (vide Seção 3.1).

Em termos de finalidades, da Figura 4.1 é possível observar que o sistema proposto se presta a três finalidades principais, a saber:

1. Alimentar uma base de conhecimento de um sistema especialista com proposições presentes num mapa conceitual;
2. Processar e interpretar perguntas em linguagem natural a fim de estabelecer uma relação com a linguagem esperada pelo sistema especialista, e;
3. Construir e apresentar uma resposta satisfatória à pergunta do avaliador.

Na parte inferior central da Figura 4.1 é possível visualizar o posicionamento de três recursos fundamentais para o funcionamento da arquitetura proposta, são eles:

1. **WordNet:** Uma base de relações léxicas, ou seja, de conhecimentos linguísticos, utilizada com o objetivo de enriquecer a base de conhecimentos de modo a permitir a consulta por termos cognatos, ou equivalentes. É utilizado tanto na alimentação da base de conhecimento quanto na extração de respostas por expandir o alcance das perguntas.

2. **VISL:** Serviço de marcação sintática e semântica. Fundamental para o processo de extração de parâmetros que permitam a interpretação computacional da pergunta construída em linguagem natural.
3. **Gramática:** O processo *Validador* necessita validar as perguntas antes de encaminhá-la para o processamento pelo sistema especialista. A gramática intermediária permite verificar se a pergunta construída possui características que a tornem processáveis pelo sistema.
4. **Tabela de Correspondência:** O processo *Solucionador* depende de uma abordagem para estabelecer uma tradução efetiva da pergunta em linguagem natural para sua representação na linguagem esperada pelo sistema especialista. A tabela de correspondência auxilia a verificar quais perguntas podem ser as possíveis soluções por meio de padrões estruturais extraídas pelo solucionador.

Ainda na parte inferior da Figura 4.1 é possível perceber que para atender a essas três finalidades o sistema está organizado em três módulos que se complementam por meio de troca de informações e serviços, quais sejam:

1. **Tradutor de conhecimento:** módulo responsável por receber um mapa conceitual e traduzí-lo para uma representação formal interpretável por um sistema especialista. Nesse processo, ele enriquece a base de conhecimento utilizando a base de relações léxicas do *WordNet*. Esse enriquecimento é fundamental para favorecer a identificação de conceitos cognatos.
2. **Sistema Especialista:** esse módulo possui uma base de conhecimento e uma base de regras de consultas predefinidas para retirar informações desta base de conhecimento. As regras são as principais responsáveis pela geração das respostas, elas utilizam a cláusula de horn para realizar inferências que permitem identificar e extrair o conhecimento presentes na base.
3. **Tradutor de perguntas:** esse módulo é responsável pela interpretação das perguntas construídas em linguagem natural para uma representação correta em forma de regra de consulta à base pelo sistema especialista. Ele possui três processos internos: *Processador PLN* que é responsável por expandir a pergunta para aumentar as chances de identificação do conhecimento utilizando, para isso, o *WordNet*, e realizar marcações (utilizando o *VISL*) a fim de identificar o tipo de pergunta, os conceitos, as relações e algumas palavras chaves que podem

umentar a precisão da tradução. Estas informações são, então, encaminhadas para o *Validador* que utiliza a *Gramática* para verificar se a pergunta possui características que a tornem interpretáveis pelo sistema e, caso afirmativo, encaminha as características para o *Solucionador* que, utilizando uma *Tabela de Correspondências*, identifica, fornece parâmetros e aciona a execução da pergunta correspondente em regras interpretáveis pelo sistema especialista.

Para compreender melhor o fluxo de atividades na arquitetura proposta, a Seção 4.2 apresenta as atividades realizadas na alimentação da base de conhecimento, na interpretação e processamento da pergunta e na geração e apresentação das respostas.

4.2 FLUXO DE ATIVIDADES

Como foi possível perceber na Figura 4.1, são dois os fluxos de atividades-fins do sistema proposto. Nas seções seguintes descreveremos a sequência de cada uma delas.

4.2.1 Alimentação da base de Inteligência

O processo para alimentar a base de inteligência do sistema proposto depende do seguinte fluxo de atividades:

1. O *Autor*, utilizando a *interface de edição de mapas conceituais*, compõe um mapa conceitual;
2. O *tradutor de conhecimento* navega por todo o mapa extraindo todas as proposições formadas;
3. Os conceitos e relações são enriquecidos utilizando a base de relações léxicas do *WordNet*;
4. Os códigos devolvidos pelo *WordNet* são utilizados na construção de proposições para:
 - a. Compor a *Base de Inteligência do Sistema Especialista*, e;
 - b. Fornecer alguns parâmetros de consulta para as *Regras de Inferência do Sistema Especialista*.

4.2.2 Pergunta e Resposta

O processo de elaboração, interpretação e tradução da pergunta em linguagem natural até o momento da construção e apresentação da resposta gerada pelo sistema possui o seguinte fluxo de atividades:

1. O *Avaliador*, por meio da *interface de análise de mapas conceituais*, utiliza o espaço reservado para composição de perguntas utilizando linguagem natural e submete ao sistema.
2. O *Processador PLN* realiza os seguintes processos na pergunta original:
 - I. Marcação sintática utilizando o *VISL*;
 - II. Agrupamento dos conceitos e relações presentes na pergunta, e;
 - III. Expansão da pergunta utilizando o *WordNet*.
3. O *Validador* recebe as informações produzidas na atividade anterior e verifica (utilizando uma gramática intermediária definida) se os resultados das marcações apresentam características que tornem a pergunta processável pelo sistema proposto com base na sua característica estrutural;
4. O *Solucionador* utiliza as marcações para identificar, no universo de regras do sistema especialista, quais são as possíveis traduções para esta pergunta. Para isso, ele utiliza uma tabela de correspondências e aplica filtros sequenciais para remover regras que não atendam a estas características, que são:
 - I. **Tipo da Pergunta:** determinação do tipo de resposta que o usuário espera que seja produzida como resposta, podendo ser: Conceito, Relação, Dado Numérico ou uma informação Lógica;
 - II. **Quantidade de Parâmetros:** utilizando as marcações das atividades anteriores, esse filtro considera a quantidade de parâmetros presentes na pergunta. As regras que esperam mais ou menos parâmetros são descartadas;
 - III. **Palavra-Chave:** um conjunto de palavras chaves estão sendo definidas para determinar com maior precisão, dentre o universo de possíveis traduções, qual a melhor tradução para a pergunta original;
5. O conjunto de regras candidatas a produzir respostas são populadas com os parâmetros extraídos da pergunta original e uma delas é acionada pelo sistema especialista, as demais são encaminhadas para a *interface de análise de mapas*

conceituais e apresentadas ao *avaliador* como outras possíveis interpretações da pergunta;

6. A resposta produzida pelo *sistema especialista* é tratada pela *interface de análise de mapas conceituais* e apresentada;
7. O *Avaliador* pode solicitar a execução direta das outras regras candidatas utilizando recursos facilitadores presentes na *interface de análise de mapas conceituais*.
8. Caso o *Avaliador* acione uma das outras regras candidatas, não será necessário passar por um novo processo de tradução já que as regras candidatas já se encontram no formato esperado pelo mecanismo, essas regras são encaminhadas diretamente para o *sistema especialista* para que busque a nova informação (resposta) na *base de conhecimento*.

4.3 CONSIDERAÇÕES

Como é possível observar, o projeto da arquitetura proposto nesta pesquisa interessou-se por facilitar ao máximo o processo de extração de informações dos mapas conceituais. A presença de elementos capazes de interpretar perguntas em linguagem natural, traduzir essas perguntas para um formato específico e identificar conceitos e relações utilizando termos cognatos é característica importante deste projeto pois o difere das arquiteturas existentes atualmente apresentadas e exploradas na seção 3.1 deste documento.

Neste capítulo foi possível conhecer o fluxo das atividades relacionadas à interação com o sistema proposto, o que é fundamental para a compreensão do protótipo que será apresentado no próximo capítulo.

CAPÍTULO 5 O *iMap*

" [...] toda a avaliação é um produto do que é avaliado pela esfera cognitiva de quem avalia."

*(Arthur Schopenhauer)
"Aforismo sobre a Sabedoria da Vida"*

Todas as propostas de novas arquiteturas computacionais necessitam de testes que validem suas aplicações. Este Capítulo tem por objetivo apresentar o protótipo construído com base na arquitetura apresentada no Capítulo 4 e realizar uma prova de conceitos para análise dos resultados obtidos.

Para facilitar sua compreensão, a Seção 5.1 apresenta a organização do protótipo, onde a segmentação das funções é ideal para uma melhor compreensão do funcionamento interno do protótipo. Já na Seção 5.2 é realizado um estudo de caso a fim de identificar os pontos positivos e negativos da arquitetura proposta que são discutidos e resultam em alterações que são apresentadas na Seção 5.3 por meio de considerações finais.

5.1 O PROTÓTIPO

Em tese, a ciência básica tem como objetivo o puro conhecimento de um determinado assunto, seja ele qual for. A ciência aplicada surge quando aparece a oportunidade de, com os conhecimentos científicos adquiridos, resolver um problema prático sem cogitar das implicações socioeconômicas de sua solução. Quando tais implicações são levadas em conta é que surge a tecnologia, como utilização, e não simples aplicação, de conhecimentos científicos do problema técnico (VARGAS *apud* AMORIM, 2012). Assim, nesta Seção serão descritos todos os componentes tecnológicos utilizados na construção de um protótipo que comprove a praticabilidade da arquitetura conceitual apresentada no Capítulo 4.

O protótipo que será apresentado nesta seção chama-se *iMap* – Um acrônimo à Inferência em Mapas Conceituais. A organização modular da arquitetura proposta por esta pesquisa

permitiu que o desenvolvimento deste protótipo se desse de maneira progressiva e os resultados preliminares foram obtidos ao longo do seu processo de desenvolvimento.

Com efeito, a versão inicial do *iMap* (PERIN, CURY e MENEZES, 2012) já apresentava a capacidade de mapear as proposições presentes nos mapas conceituais como fatos para uma base de conhecimento de um mecanismo inferencial. Além disso, o mecanismo oferecia algumas regras de inferência que permitia realizar alguns tipos de consultas e inferências a partir do conhecimento presente nos mapas. No entanto, a interação entre o avaliador e o sistema ocorria através de uma linguagem de baixo nível oferecida pelo mecanismo de inferência. Assim, a versão mais recente do protótipo (PERIN, CURY e MENEZES, 2014) apresentou a capacidade da ferramenta de receber e processar perguntas construídas em linguagem natural, reduzindo o *gap semântico* e facilitando a interação entre o avaliador e o sistema.

Para facilitar sua total compreensão, essa seção será segmentada em subseções que apresentarão cada processo funcional do protótipo, iniciando por uma subseção que dará uma visão geral do seu desenvolvimento: a organização, técnicas e tecnologias utilizadas.

5.1.1 Visão Geral do Protótipo

No desenvolvimento do *iMap* foi utilizada uma combinação das seguintes abordagens técnicas: Desenvolvimento Orientado a Modelos e Sistemas Especialistas.

O Desenvolvimento Orientado a Modelos⁸ é um estilo de desenvolvimento de *software* em que os principais artefatos são modelos, a partir dos quais são gerados códigos e outros artefatos de acordo com as boas práticas. Um modelo é uma descrição de um sistema a partir de uma determinada perspectiva, omitindo detalhes irrelevantes para que as características de interesse sejam vistas de forma mais clara. No geral, adota-se para construção deste modelo, diagramas expressos em UML (*Unified Modeling Language*). A partir destes modelos, os códigos da aplicação são gerados automaticamente trazendo as seguintes vantagens: maior produtividade, capacidade de manutenção, reutilização, adaptabilidade, consistência, repetibilidade, facilidade de comunicação entre os

⁸ Em inglês: *Model-Driven Development (MD)*

stakeholders, melhor expressividade da ferramenta gráfica, dentre outras; enfim, tudo o que pode despertar o interesse de qualquer analista/desenvolvedor de *software* (BROWN, 2004).

Adotar a abordagem de desenvolvimento orientado a modelos nesta pesquisa foi essencial, visto se tratar de uma abordagem atual de desenvolvimento de linguagens específicas de domínio, seja ela textual ou gráfica, como é o caso dos mapas conceituais. O principal objetivo foi reduzir o tempo gasto para produção de uma aplicação, já que esta abordagem envolve pouca edição de códigos-fonte de aplicações, bastando apenas especificar, visualmente, a sintaxe abstrata do domínio da aplicação e a sintaxe concreta gráfica que se deseja obter das instâncias do metamodelo da aplicação. Essas instâncias podem então ser transformadas em arquivo de texto cujo formato é especificado pelo desenvolvedor da aplicação. Para aproveitar-se de todas estas vantagens, especialmente, da produtividade oferecida por essa técnica, adotamos a ferramenta *ObeoDesigner*⁹ na construção do protótipo.

Já no que diz respeito ao Sistema Especialista adotado para gestão do conhecimento e realização de inferências, esta pesquisa adotou o *PIE*¹⁰ (*Prolog Inference Engine*) para realização desta função. Apesar de existirem muitos outros sistemas especialistas disponíveis, tais como: *JESS*¹¹, *DROOLS*¹², *RuleML*¹³, *MS-BRE*¹⁴, dentre outros; o *Prolog* se mostra mais apropriado, já que esta pesquisa está interessada no tratamento de conhecimento construído em lógica proposicional.

Como apresentado no Capítulo 4, o sistema proposto possui duas telas para interação:

⁹ *Obeo Designer* é um framework proprietário desenvolvido pela empresa Obeo, uma companhia especializada em construção de ferramentas MDA (Model-Driven Architecture) para desenvolvimento Java baseada em Eclipse. Mais informações podem ser obtidas através do site: www.obeodesigner.com

¹⁰ Informações adicionais em: <http://www.visual-prolog.com/>

¹¹ Informações adicionais em: <http://www.jessrules.com/>

¹² Informações adicionais em: <http://www.jboss.org/drools/>

¹³ Informações adicionais em: <http://www.ruleml.org/>

¹⁴ Informações adicionais em: <http://msdn.microsoft.com/en-us/library/aa561216.aspx>

1. Um editor de mapas conceituais que facilita o processo de criação, edição e, de forma indireta, alimentação da base de inteligência, e;
2. Uma tela de avaliação onde o avaliador compõe perguntas para avaliar o conhecimento presente no mapa.

Adotando técnicas derivadas da abordagem MDD (*Model-Driven Development*) a organização das tarefas e atividades para construção de um editor de mapas conceituais pode ser decomposta e relacionada como apresentado na Figura 5.1.

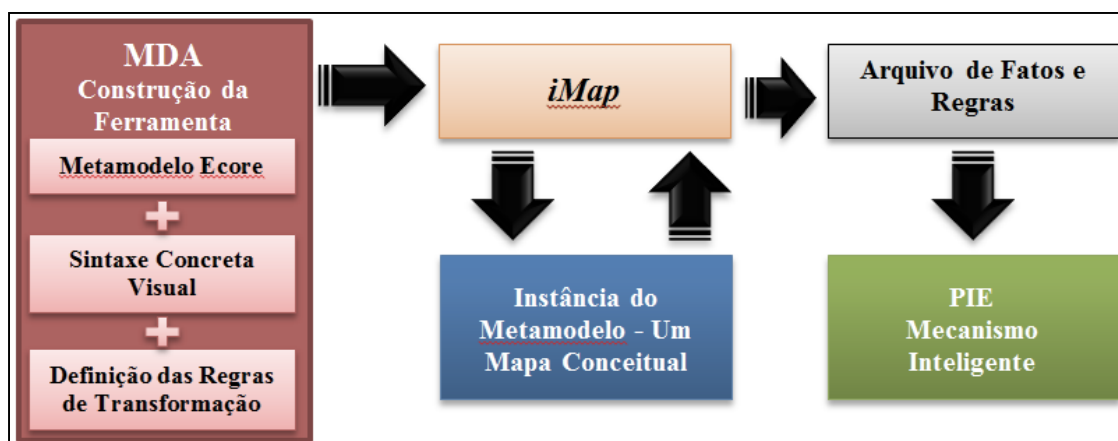


Figura 5.1 – Atividades de MDA para construção do Editor do iMap.

Como pode ser observado na Figura 5.1, a construção do editor de mapas conceituais do *iMap* englobou três subatividades, a saber:

1. Metamodelagem de mapas conceituais;
2. Definição da sintaxe concreta visual, e;
3. Definição de regras de transformação do modelo instanciado para um arquivo de texto.

O editor pode ser utilizado para construção e definição de instâncias de mapas conceituais. Uma vez gerada uma instância do mapa conceitual ela pode ser transformada, a partir do *iMap*, num arquivo contendo os fatos e as regras que servirão de entrada para o mecanismo inteligente.

O arquivo contendo as regras resultantes da transformação da instância do mapa conceitual de origem serve de entrada para um RBS (*Rule-Based System*). RBS é um sistema especialista que utiliza um método combinado de pesquisa e raciocínio (ABRAHAM, 2005). No *iMap*, ele é utilizado como uma ferramenta auxiliar responsável

por fornecer a capacidade de realizar inferências nos mapas conceituais. Esta camada inteligente é útil na criação de regras que permitem aos professores navegar o mapa conceitual e, mais importante ainda, responder a perguntas que permitem aos docentes extrair informações a respeito dos conceitos e das relações presentes nos mapas.

A Figura 5.2 apresenta um mapa conceitual que pode ser construído utilizando o *iMap* e os fatos mapeados que compõem o arquivo gerado pela transformação deste mapa. Ela permite, ainda, entender o processo de transformação do mapa conceitual e a composição dos fatos contidos no arquivo de saída, que deve seguir a sintaxe esperada pelo PIE.

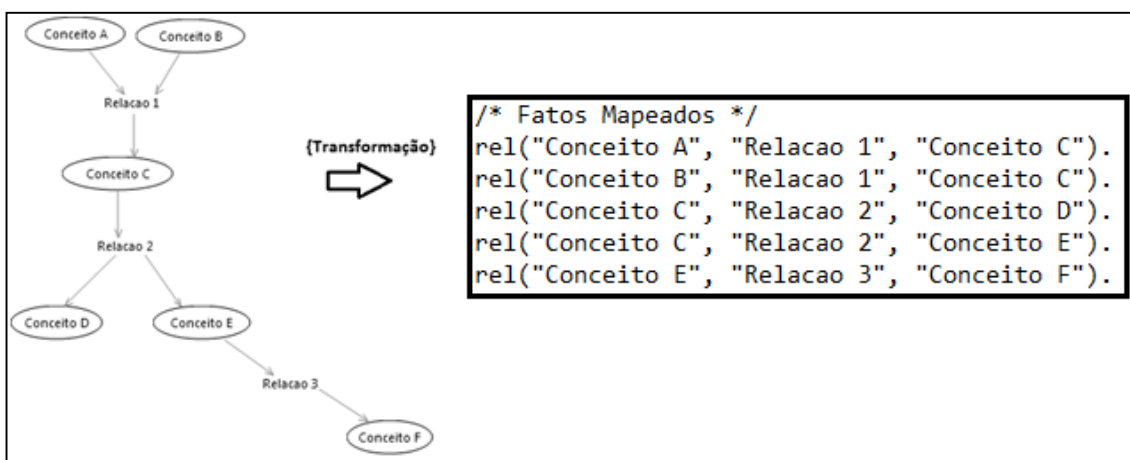


Figura 5.2 – Processo de transformação de um mapa em fatos (proposições) *prolog*.

Os fatos contidos no arquivo gerado pela transformação do mapa são, na realidade, mapeados como representações textuais das relações definidas no mapa conceitual. Como as relações podem possuir mais de um conceito-destino e também muitos conceitos-origem, as relações devem ser decompostas em tuplas simples estruturadas no formato: “conceito-relação-conceito”. Desta forma, sempre que houver relações que possuam mais de um conceito em alguma, ou em ambas, de suas pontas, para realizar a transformação será preciso realizar o produto cartesiano entre o conjunto de conceitos-origem e o conjunto de conceitos-destino para, desta forma, obter todas as possíveis combinações entre os conceitos presentes na relação mapeada. Assim, todas as relações presentes no mapa conceitual estarão presentes no arquivo gerado por este processo de transformação.

O arquivo gerado pelo processo de transformação inclui, ainda, algumas regras de consulta que são utilizadas pelos avaliadores para a realização de perguntas à base de conhecimento. A lista completa e a sintaxe das regras pode ser vista na segunda coluna do ANEXO II desta pesquisa.

Antes de apresentar o processo de extração do conhecimento, no entanto, vamos conhecer um pouco mais a fundo o editor e o processo de alimentação da base de conhecimento. A subseção seguinte apresentará estes detalhes.

5.1.2 O Editor de Mapas: Alimentação da Base de Conhecimento

As três subatividades de desenvolvimento do editor do *iMap*, que podem ser vistas na Figura 5.1, são realizadas com a utilização do *Obeo Designer*. Este *framework* fornece todas as facilidades características da abordagem de desenvolvimento orientado a modelos, sendo, a principal delas, a redução da complexidade no desenvolvimento de ferramentas de modelagem de linguagens específicas de domínio como, neste caso, os mapas conceituais. A Figura 5.3 sintetiza a organização e relação entre essas três subatividades.

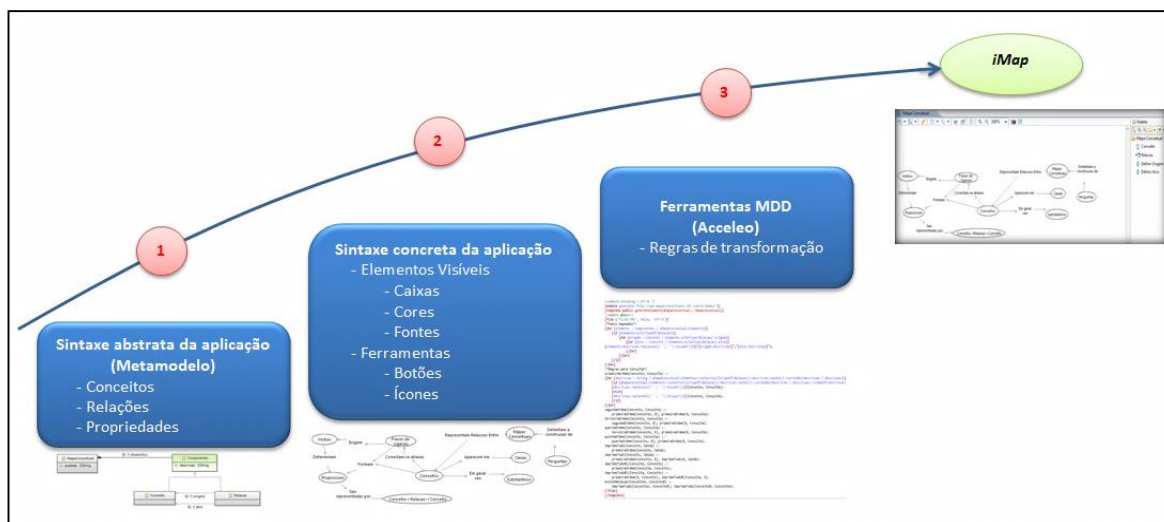


Figura 5.3 – Etapas do Desenvolvimento do Editor do *iMap*.

Na Figura 5.3, na etapa 1 é realizado o processo de especificação de um metamodelo, ou seja, da sintaxe abstrata, que representa a definição do domínio da aplicação com seus conceitos, relações e propriedades. Na etapa 2 é realizada a descrição da sintaxe concreta visual, ou seja, a definição da forma como cada um dos componentes do mapa conceitual será representado graficamente no *iMap*. Por último, na etapa 3, é realizada a especificação de um conjunto de regras de transformação que serão aplicadas ao mapa instanciado no *iMap* a fim de gerar o arquivo de saída. Apesar de parecerem atividades complexas, é preciso destacar que todas estas etapas são feitas com a utilização de ferramentas gráficas utilizando, na maior parte do tempo, apenas o *mouse*, à exceção da

etapa 3, uma vez que as regras de transformação são especificadas utilizando uma linguagem textual.

Analisaremos os resultados obtidos com a realização de cada uma dessas etapas no processo de construção do editor de mapas do *iMap*.

5.1.2.1 O metamodelo da aplicação

A metamodelagem é um processo cujo objetivo é descrever todos os componentes, com suas características e relações, presentes no domínio da aplicação que, neste caso, são aqueles utilizados na construção de um mapa conceitual. A Figura 5.4 apresenta o metamodelo de mapas conceituais construído para o *iMap*.

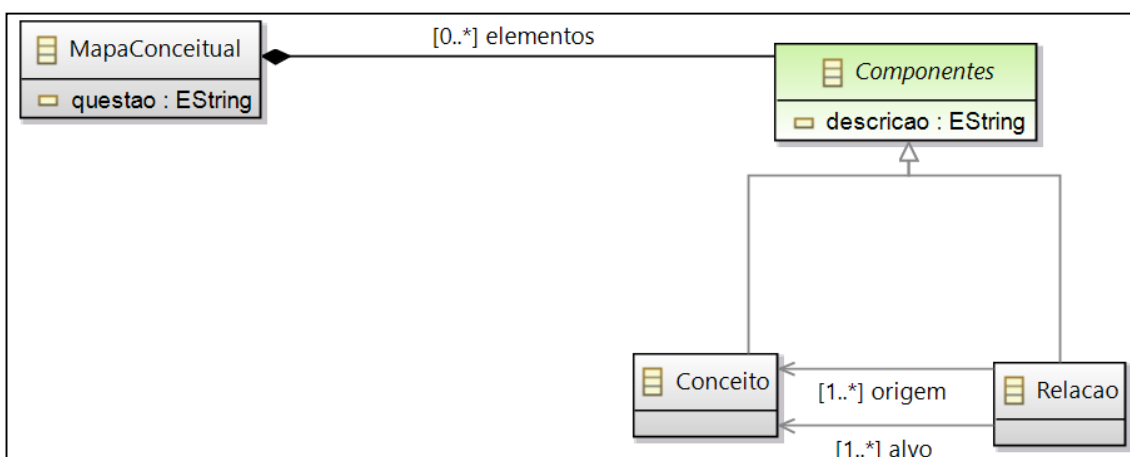


Figura 5.4 – O metamodelo de Mapas Conceituais do Editor de Mapas.

Como é possível observar na Figura 5.1, a metaclassa “MapaConceitual” é considerada um *container* onde são armazenados os componentes (“Conceito” e “Relacao”). Outra característica que pode ser destacada é a possibilidade de especificar a questão de investigação de um mapa conceitual. Nota-se, ainda, que os conceitos e as relações presentes num mapa conceitual possuem, como propriedade, uma descrição.

Utilizando o *Obeo Designer* esta simples especificação é capaz de gerar o código fonte da aplicação e permitir uma posterior especificação da sintaxe concreta dos elementos, ou seja, a forma como os componentes deste metamodelo serão vistos pelo usuário ao construir uma instância de um Mapa Conceitual. A subseção seguinte apresentará como esta especificação é realizada.

5.1.2.2 A sintaxe concreta visual

A segunda etapa para a construção do editor de mapas do *iMap* consiste na definição da sintaxe concreta visual, ou seja, o *layout* da aplicação. Através do *Obeo Designer* é possível especificar, de maneira simples e intuitiva, toda a sintaxe concreta visual, ou seja, a forma como os componentes presentes no metamodelo instanciado no *iMap* serão representados graficamente. Tendo por base as características visuais costumeiramente utilizadas na construção de mapas conceituais, que podem ser observadas na Figura 2.1, nota-se que conceitos presentes num mapa conceitual são representados graficamente por meio de uma elipse contendo, em seu interior, uma breve descrição deste conceito. Por sua vez, as relações são representadas por setas e possuem, similar aos conceitos, uma descrição, que representa o significado da ligação entre os dois, ou mais, conceitos relacionados. A Figura 5.5 apresenta parte do arquivo que especifica a sintaxe concreta visual do editor de mapas do *iMap*.

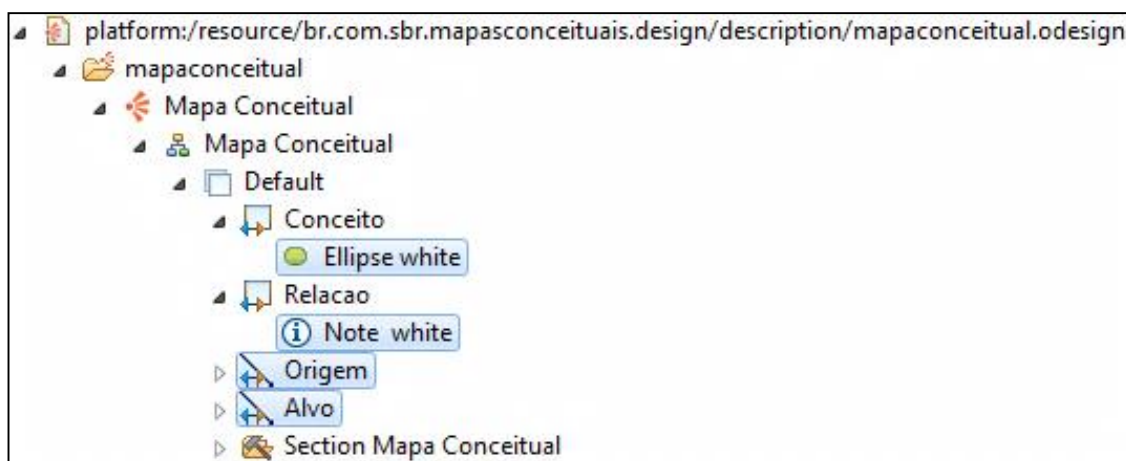


Figura 5.5 – Definição da Sintaxe Concreta Visual.

Os elementos em destaque na Figura 5.5 descrevem a forma como as instâncias de “Conceito” e “Relação” devem ser representadas graficamente. É possível perceber que as instâncias de conceitos serão representadas por elipses brancas enquanto as de relações serão representadas por uma nota com descrição e setas apontando, de um lado, para os conceitos-origem e, do outro lado, para os conceitos-alvo. Todas estas especificações são realizadas visualmente, com raras interferências no código-fonte gerado para a aplicação, característica esta já destacada da abordagem MDD.

Ao fim desta etapa já é possível utilizar o editor de mapas conceituais para criar uma instância de um mapa. No entanto, a etapa a seguir é fundamental pois completa o processo fornecendo a capacidade de transformar o mapa desenhado num arquivo contendo todas as proposições presentes no mapa construído.

5.1.2.3 As regras de transformação

A última etapa para a construção do editor de mapas do *iMap* foi a definição de um conjunto de regras de transformação. A ferramenta gráfica do *iMap* cria e manipula instâncias da metaclassa “MapaConceitual” que é usada como um *container* de instâncias das metaclassas “Conceito” e “Relação”. Para que seja possível manipular os conceitos presentes no mapa conceitual instanciado através do *PIE*, é necessário mapear os conceitos e as relações presentes neste mapa em formato gráfico para o formato de proposições aceito por este mecanismo inteligente. As proposições de entrada para o *PIE* devem respeitar a sintaxe apresentada na Figura 5.2, ou seja, as relações entre os conceitos precisam ser mapeados na forma de tuplas no seguinte formato:

rel (<“Origem”>,<“Descrição”>, <“Alvo”>)

O processo de transformação do metamodelo instanciado para o arquivo de fatos é possível com a utilização de um gerador de código, muito utilizado na abordagem *MDD* e presente no *Obeo Designer*, denominado *Acceleo*¹⁵. Ele permite que sejam definidas as regras de transformação, utilizando sintaxe própria, que são aplicadas às instâncias do *container* do metamodelo. Uma vez que essas regras são executadas no *Acceleo* e é informado a este a instância do metamodelo que deve ser transformado, ele gera, automaticamente, o arquivo de saída contendo as informações que respeitam estas regras.

¹⁵ Mais informações em: <http://www.acceleo.org>


```

*generate.mtl
[comment encoding = UTF-8 /]
[module generate('http://www.cmpaas.ufes.br/imap')]
[template public generateElement(aMapaConceitual : MapaConceitual)]
[comment @main/]
[file ('FILE0.pl', false, 'UTF-8')]
/* FATOS MAPEADOS */
[for (element : Componentes | aMapaConceitual.elementos)]
  [if (element.oclIsTypeOf(Relacao))]
    [for (origin : Conceito | element.oclAsType(Relacao).origem)]
      [for (target : Conceito | element.oclAsType(Relacao).alvo)]
rel(['origin.descricao/'], ['element.descricao/'], ['target.descricao/']).
    [/for]
  [/for]
[/if]
[/for]

/* REGRAS PARA CONSULTA */
primeiraOrdemDireta(ConceitoA, ConceitoB) :-
  rel(ConceitoA,_,ConceitoB).
primeiraOrdemInversa(ConceitoA, ConceitoB) :-
  primeiraOrdemDireta(ConceitoB, ConceitoA).
existeRelacaoDireta(ConceitoA, ConceitoB) :-
  primeiraOrdemDireta(ConceitoA, ConceitoB); primeiraOrdemInversa(ConceitoA, ConceitoB).
todoDestino(Conceito, _, Saida) :-
  primeiraOrdemDireta(Conceito, Saida).
todoDestino(Conceito, V, Saida) :-
  primeiraOrdemDireta(Conceito, X),
  \+ member(X, V),
  todoDestino(X, ['[X|V]'], Saida).
todaOrigem(Saida, _, Conceito) :-
  primeiraOrdemDireta(Saida, Conceito).
todaOrigem(Saida, V, Conceito) :-
  primeiraOrdemDireta(X, Conceito),
  \+ member(X, V),
  todaOrigem(Saida, ['[X|V]'], X).
existeRelacao(ConceitoA, ConceitoB) :-
  todaOrigem(ConceitoA, ['[ConceitoA]'], ConceitoB); todaOrigem(ConceitoB, ['[ConceitoB]'], ConceitoA).
quaisRelacoes(Saida, Conceito) :-
  rel(Conceito, Saida, _),
  rel(_, Saida, Conceito).
quaisRelacoes(Saida, ConceitoA, ConceitoB) :-
  rel(ConceitoA, Saida, ConceitoB),
  rel(ConceitoB, Saida, ConceitoA).
listaRelacao(LR) :-
  findall(R, rel(_,R,_),LR).
buscaRelacao(R) :-
  listaRelacao(LR),
  member(R, LR).
nTotalRelacao(R) :-
  listaRelacao(LR),
  comprimento(R, LR).
nConceitoRelacao(N,R) :-|

```

Figura 5.6 – Regras de Transformação de Mapas em Proposições para Prolog.

Na Figura 5.6 vemos que a única parte que exige processamento é o mapeamento dos fatos que irão compor a base de inteligência do *PIE*. Nesta etapa, todas as relações constantes no Mapa Conceitual de origem são navegados e transformados em uma representação de conhecimento textual usando lógica de primeira ordem a fim de ser processada pelo *Prolog*. Além disso, vemos na parte inferior da Figura que as regras de consulta/inferência são estáticas e padronizadas e objetivam realizar o processamento da base de inteligência. A Tabela 5.1 apresenta as informações que o *iMap* é, atualmente, capaz de obter através de cada uma das regras estabelecidas até agora.

Tabela 5.1 – As regras de Consulta/Inferência

Regra em Prolog	Permite identificar...
primeiraOrdemDireta(A, B)	Se um dado conceito é origem de alguma relação com destino a outro conceito.
	Quais conceitos possuem relação com origem em um dado conceito.
primeiraOrdemInversa(A, B)	Se um dado conceito é alvo de alguma relação com origem a partir de outro conceito.
	Quais conceitos possuem relação com alvos em um dado conceito.
existeRelacaoDireta(A, B)	Se dois conceitos estão relacionados diretamente entre si.
	Quais conceitos estão diretamente relacionados a um dado conceito.
todoDestino(A, _, S).	Todos os conceitos que podem ser alcançados no mapa a partir de um dado conceito.
	Se a partir de um dado conceito é possível atingir outro conceito também informado.
todaOrigem(S, _, A).	Todos os conceitos que são prévios, ou seja, fundamentam um dado conceito.
	Se um determinado conceito é preliminar a outro dado conceito.
existeRelacao(A, B)	Se dois conceitos possuem relação, seja direta ou indiretamente.
	Todos os conceitos que estão relacionados, direta ou indiretamente, a um dado conceito.
quaisRelacoes(S, A)	Em quais relações um dado conceito está presente, seja como origem ou como alvo.
	Quais conceitos estão ligados por uma dada relação.
quaisRelacoes(S, A, B)	Quais relações conectam um dado par de conceitos.
	Quais conceitos são conectados por uma dada relação.
	Qual conceito é origem ou alvo de uma dada relação e um dado conceito.
listaRelacao(L)	A lista de relações existentes.
buscaRelacao(R)	Todas as relações existentes.
	Se dada relação existe.
nTotalRelacao(R).	A quantidade de relações existentes.
nConceitoRelacao(N, R).	Quantos conceitos estão conectados por uma dada relação.
	Quais conceitos estão conectados por uma dada relação.
numeroRelacaoConceito(N, C)	Em quantas relações um dado conceito participa.
	Em quais relações um dado conceito aparece.
nRelacaoConceitos(N, A, B)	Quantas relações distintas conectam diretamente dois conceitos.
	Quais relações conectam diretamente dois conceitos.
listaConceitoA(L)	A lista de conceitos que são origem de alguma relação.
listaConceitoB(L)	A lista de conceitos que são alvos de alguma relação.
listaConceito(L)	A lista de todos os conceitos presentes na base.
buscaConceito(C)	Todos os conceitos presentes na base.
	Se um dado conceito está presente na base.
nTotalConceito(N).	A quantidade total de conceitos presentes na base.
nRelacaoCadaConceito(N, C)	A quantidade de relações das quais um dado conceito participa.
	De quais relações um dado conceito participa.
nRelacaoEntradaConceito(N, C)	A quantidade de relação das quais um dado conceito é origem.

	De quais relações um dado conceito é origem.
listaProposicao(L)	Todas as proposições que compõem a base.
nProposicao(N)	O numero de proposicoes que compõem a base.

Como é possível observar, todas estas regras de consulta possuem uma sintaxe semelhante, pois são definidas seguindo a estrutura padrão esperada pelo *PIE*. É importante destacar, neste ponto, que o cerne desta pesquisa são as regras de consulta à base de inteligência. São elas quem determinam a capacidade e o tipo de informações que poderão ser extraídas da base de inteligência.

Dependendo do nível de conhecimento de *prolog* do usuário, a interação é possível diretamente através das regras de inferência apresentadas na Tabela 5.1. No entanto, esse processo foi facilitado, nesta pesquisa, com a inserção de uma interface para tradução de perguntas realizadas em Linguagem Natural que será apresentado na Seção seguinte. A Seção 5.2 apresentará as duas possíveis interações para realização de consultas à Base de Inteligência.

5.1.3 A Interface de Consulta

A versão inicial do *iMap* (PERIN, CURY e MENEZES, 2012) permitia interação com a base de inteligência apenas através das regras de inferência, ou seja, diretamente através do *PIE*. No entanto, esta seria uma grave limitação para aplicabilidade da arquitetura aqui proposta. Em consequência disto, nossas pesquisas mais recentes (PERIN, CURY e MENEZES, 2014) trabalharam no sentido de propor alterações à arquitetura original do mecanismo de inferências *iMap* a fim de inserir uma nova tarefa que atue como tradutor de perguntas construídas em Linguagem Natural para o formato *Prolog* esperado pelo mecanismo. A Figura 5.7 (a) demonstra como as perguntas são realizadas na arquitetura original do sistema enquanto a Figura 5.7 (b) apresenta a inserção do processo de tradução, ilustrando como este novo processo afeta o fluxo de atividades de uso do mecanismo.

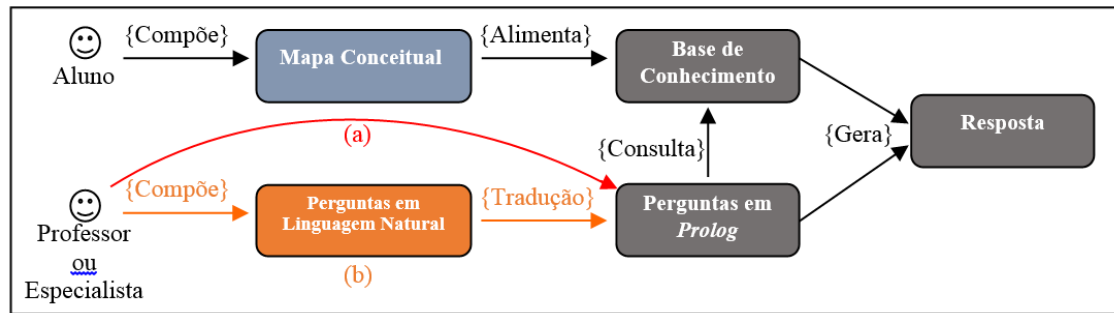


Figura 5.7 – Organização conceitual das consultas na arquitetura (a) original (b) evoluída do iMap.

Denominamos, nesta pesquisa, as evoluções apresentadas no sentido de processar as perguntas em linguagem natural de *PLN-iMap*¹⁶. Para tornar isso possível, foi necessário projetar uma nova arquitetura funcional para a interface de consulta à base de conhecimento. A arquitetura funcional projetada é composta por dois elementos: (1) o *Processador PLN* e (2) o *Solucionador de Correspondência*; e possui dois recursos funcionais: (1) o etiquetador *VISL* e (2) uma tabela de correspondência contendo a gramática intermediária. Essa arquitetura funcional é apresentada na Figura 5.8.

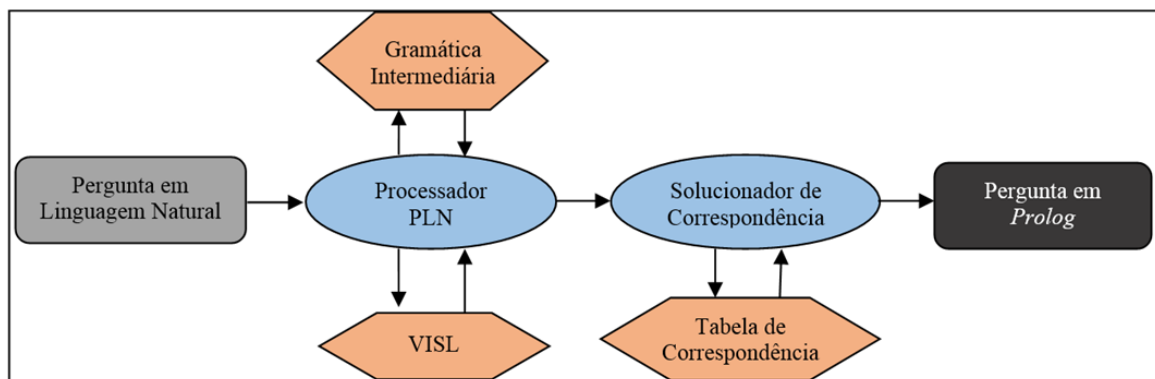


Figura 5.8 – Arquitetura funcional do *PLN-iMap*.

Antes de definir a arquitetura do *PLN-iMap*, analisemos os componentes de sua arquitetura. A primeira atividade realizada foi tabular e relacionar as regras de consulta definidas pelo *iMap* a uma gramática, a qual denominamos de *gramática intermediária*, adotada a fim de reduzir o *gap semântico* entre as perguntas feitas em linguagem natural e a sua correspondente tradução para o padrão utilizado pelo *iMap*.

O *gap semântico* pode ser facilmente compreendido por pessoas que trabalham com tradução. Muitas vezes, uma frase construída num dado idioma não possui uma

¹⁶ Uma referência a “Processamento de Linguagem Natural do *iMap*”.

correspondente exata no idioma-alvo. Na maioria das vezes, o que o tradutor faz é construir uma frase que aproxime a correspondência semântica das frases em ambos os idiomas. Assim, essa frase construída por ele reduz o *gap semântico* que distancia esses dois idiomas.

Como uma mesma pergunta pode ser realizada de várias formas em linguagem natural, notamos que o *gap semântico* seria um problema frequente, tendo em vista que a gramática da linguagem-alvo é limitada, o que significa dizer que todas as perguntas construídas em linguagem natural deveriam ser reescritas para localizar uma correspondente no *iMap*. Assim, ao definirmos uma *gramática intermediária*, reduzimos a distância entre os significados de ambas as gramáticas. A Tabela 5.2 apresenta algumas relações de correspondência entre perguntas em *Prolog*, em gramática intermediária e em linguagem natural. O ANEXO II apresenta uma lista maior de possíveis resultados do processo de tradução.

Tabela 5.2 – Exemplo de Perguntas que podem ser realizadas ao *iMap*

Pergunta em <i>Prolog</i>	Correspondente na gramática intermediária	Exemplos de possíveis correspondentes em Linguagem Natural
primeiraOrdemDireta("Carro", "Roda")	Existir relação direta Carro Roda	Existem conexões diretas entre Carro e Roda? Entre Carro e Roda, existem conexões diretas?
quaisRelacoes(X, "Carro")	Qual relação Carro	Em quais relações está presente o Carro? Qual as relações em que Carro se faz presente?
quaisRelacoes(S, "Carro", "Roda")	Qual relação Carro Roda	Através de quais relações o Carro e a Roda são ligados? Carro e Roda são ligados por quais relações?
quaisRelacoes("possui", X, Y)	Qual conceito possuir	Quais os pares de conceitos conectados por uma relação descrita por "possui"? "Possui" é o termo de ligação que liga quais pares de conceitos?
quantasRelacoes("Carro")	Quantas relação Carro	Quantas relações possuem o conceito Carro? Em quantas relações o Carro está presente? O carro está presente em quantas relações?

Observamos que a gramática intermediária permite extrair, do universo de questões possíveis, os elementos fundamentais que conduzem à seleção de qual pergunta acionar no mecanismo de inferência. Além disso, permite definir uma gramática livre de contexto para a linguagem que facilitará o trabalho de interpretação do mecanismo de PLN. A *gramática intermediária* no formato BNF pode ser vista na Figura 5.9.

<pre> <pergunta> ::= <questão> <tipo> <chave> <questão> ::= qual quanto existir <tipo> ::= relação direto relação conceito <chave> ::= <conceito> <conceito> <conceito> <relação> <relação> <conceito> <relação> <conceito> <conceito> <conceito> ::= <substantivo> <relação> ::= <verbo> <substantivo> ::= carro roda pneu ferro borracha ... <verbo> ::= possuir conter eleger fazer organizar ... </pre>

Figura 5.9 – Definição BNF da *gramática intermediária*.

Uma vez definida a gramática intermediária e construída uma tabela parametrizada com as correspondências no formato esperado pelo mecanismo de inferência, resta-nos especificar a forma como o *processador PLN* extrai os elementos-chave da pergunta em Linguagem Natural para composição da correspondente tradução em gramática intermediária e como o *Solucionador de Correspondência* atua na seleção da pergunta correspondente no *iMap*.

5.1.3.1 O Processador PLN

O *Processador PLN* é o elemento principal do processo de tradução. Ele é responsável pela tradução inicial da pergunta construída em Linguagem Natural para a *gramática intermediária*. Para isso, a primeira etapa do seu processo consiste na realização de uma requisição *web* ao sistema VISL solicitando a marcação do texto contido na pergunta inicial. A Figura 5.10 apresenta uma saída gerada pelo processo de marcação realizada pelo VISL. Nessa figura observamos que mesmo que a pergunta seja construída de forma diferente, os elementos-chave (sublinhados na Figura 5.10) estão presentes tanto na frase submetida em (a) quanto em (b). Esses elementos-chave são fundamentais pois serão utilizados para compor a pergunta correspondente em *gramática intermediária*.

<p>Enter text to parse: de quais relações participa o carro?</p> <p>Parser: Full morphosyntactic parse Visualization: Default</p> <p>de [de] PRP @ADVL <u>quais</u> [qual] <interr> DET F P @>N <u>relações</u> [relação] <ac> <sem-r> N F P @P< <u>participa</u> [participar] <vt> <fmc> V PR 3S IND VFIN @FMV o [o] <artd> DET M S @>N <u>carro</u> [carro] <V> N M S @<ACC ?</p> <p>(a)</p>	<p>Enter text to parse: o carro está presente em quais relações?</p> <p>Parser: Full morphosyntactic parse Visualization: Default</p> <p>o [o] <artd> DET M S @>N <u>carro</u> [carro] <V> N M S @SUB,J> <u>está</u> [estar] <vK> <fmc> V PR 3S IND VFIN @FMV <u>presente</u> [presente] ADJ M S @<SC em [em] PRP @A< <u>quais</u> [qual] <interr> DET F P @>N <u>relações</u> [relação] <ac> <sem-r> N F P @P< ?</p> <p>(b)</p>
--	---

Figura 5.10 – Saídas geradas pelo VISL para o Processador PLN.

Com as marcações do VISL, o Processador PLN inicia o processo de construção da *pergunta* em *gramática intermediária*, respeitando sua notação BNF. Para cada um dos

componentes da *pergunta* (vide Figura 5.9) existe um parâmetro de seleção que leva em consideração as marcações e as posições das palavras na frase. A Tabela 5.3 apresenta alguns parâmetros elaborados para seleção de cada um dos *elementos-chave*.

Tabela 5.3 – Exemplo de Perguntas que podem ser realizadas ao *iMap*

Elemento	Parâmetros de Seleção
Questão	Marcação < <i>interr</i> >
Tipo	Marcação N (noun – substantivo)
	Próximo ao delimitador da Questão (< <i>interr</i> >)
	Limitados a “conceito” e “relação”
Chave (Conceito)	Marcação N
	Não incluir delimitador do <i>tipo</i>
Chave (Relação)	Marcação V (Verbo)
	Não incluir verbos seguidos de adjetivos (ADJ) ou determinantes (DET) pois indicam continuidade de pergunta e não uma relação-foco.

Com base nesses parâmetros, o Processador PLN realiza a construção da pergunta em linguagem intermediária e encaminha para o *solucionador de correspondência*.

5.1.3.2 O Solucionador de Correspondência

Cada pergunta construída em gramática intermediária precisa ser mapeada para uma pergunta correspondente no formato esperado pelo mecanismo de inferência. Para isso, o *Solucionador de Correspondência* possui uma tabela de correspondência parametrizada, similar ao apresentado nas duas primeiras colunas da Tabela 5.2. O solucionador de correspondência analisa a estrutura da pergunta em gramática intermediária e extrai, dessas, os parâmetros necessários para o acionamento da pergunta correspondente na linguagem-alvo.

No entanto, assim como ocorre no processo de tradução de idiomas na vida real, em alguns casos, a pergunta construída em linguagem natural poderá: (a) não possuir tradução correspondente no idioma do *iMap* ou (b) possuir duas ou mais traduções possíveis para o idioma do *iMap*. Caso ocorra (a), o Solucionador de Correspondência notifica, via interface, que não é capaz de responder a este tipo de pergunta e sugere ao usuário maneiras de construir uma pergunta. Já na ocorrência de (b), o solucionador de correspondência aciona a primeira tradução encontrada. Como a resposta gerada pelo mecanismo pode não corresponder ao esperado, o Solucionador de Correspondência indica ao usuário que o sistema identificou outra maneira de responder àquela pergunta.

Assim, o usuário poderá solicitar, a qualquer momento, a execução das outras soluções possíveis.

5.1.3.3 O PLN-iMap Atualmente

Atualmente, o *PLN-iMap* possui um protótipo funcional desenvolvido em linguagem Java empregando técnicas avançadas para desenvolvimento ágil apoiadas no paradigma de desenvolvimento orientado a modelos. Além disso, sua base de inteligência é construída e processada em *Prolog*. A Figura 5.11 apresenta o funcionamento do protótipo do *PLN-iMap*. Nela é possível observar a resposta gerada pelo mecanismo a uma pergunta realizada a um mapa conceitual cuja questão-focal foi: “O que é um carro? ”.

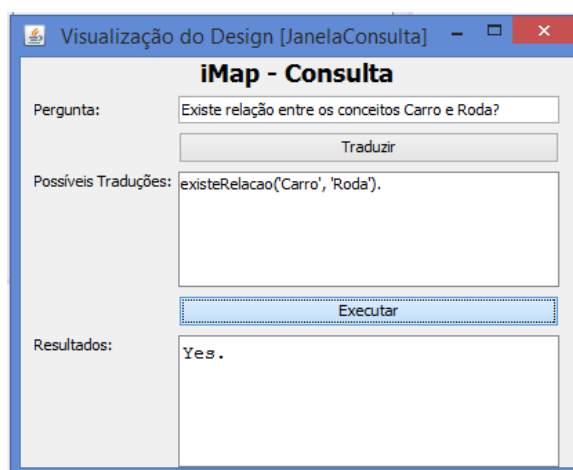


Figura 5.11 – Protótipo do PLN-iMap.

Os esforços atuais, no entanto, estão sendo empregados no sentido de produzir uma ferramenta para utilização final. Assim, apesar de a arquitetura conceitual do *PLN-iMap* e do próprio *iMap* se manterem intactos, a arquitetura funcional sofrerá algumas alterações a fim de se adequar ao padrão de desenvolvimento conhecido como SOA (*Service Oriented Architecture*). A ideia é incorporar todas as funcionalidades aqui apresentadas à uma plataforma de construção, manipulação e gestão de mapas conceituais conhecida como *CMPaaS*¹⁷ (*Concep Map Platform as a Service*) onde os usuários contarão com um conjunto de soluções que tornarão mais completa a experiência de

¹⁷ O *CMPaaS* é uma plataforma de serviços para construção, gestão e manipulação de mapas conceituais que está em fase de desenvolvimento. Diversas soluções que facilitam a utilização de Mapas Conceituais estão sendo desenvolvidas e incorporadas à essa plataforma a fim de compor um conjunto completo de ferramentas para utilização no Portal que também está sendo pensado e desenvolvido pelos mesmos autores.

utilização dos mecanismos aqui apresentados. Nessa nova arquitetura, tanto o *iMap* quando o *PLN-iMap* serão inseridos como serviços prestados pela plataforma *CMPPaaS*.

5.2 UMA PROVA DE CONCEITOS

Existem diversas abordagens possíveis onde o *iMap* poderia ser utilizado como ambiente de apoio no processo de acompanhamento e avaliação de mapas conceituais. Nesta Seção estaremos apresentando uma prova de conceitos direcionada para um cenário de aprendizagem. Um dos cenários possíveis pode ser o modelo descrito no passo-a-passo abaixo. Os passos estão numerados para facilitar o entendimento do fluxo de atividades e a aplicação do *iMap*.

Passo 1: O professor solicita aos alunos que, a partir de um artigo que explica o que são mapas conceituais, construam um mapa, utilizando o *iMap*, que responda à seguinte questão: “O que são mapas conceituais?”.

Um possível resultado obtido por um dos estudantes ao realizar esta etapa é apresentado na Figura 5.12.

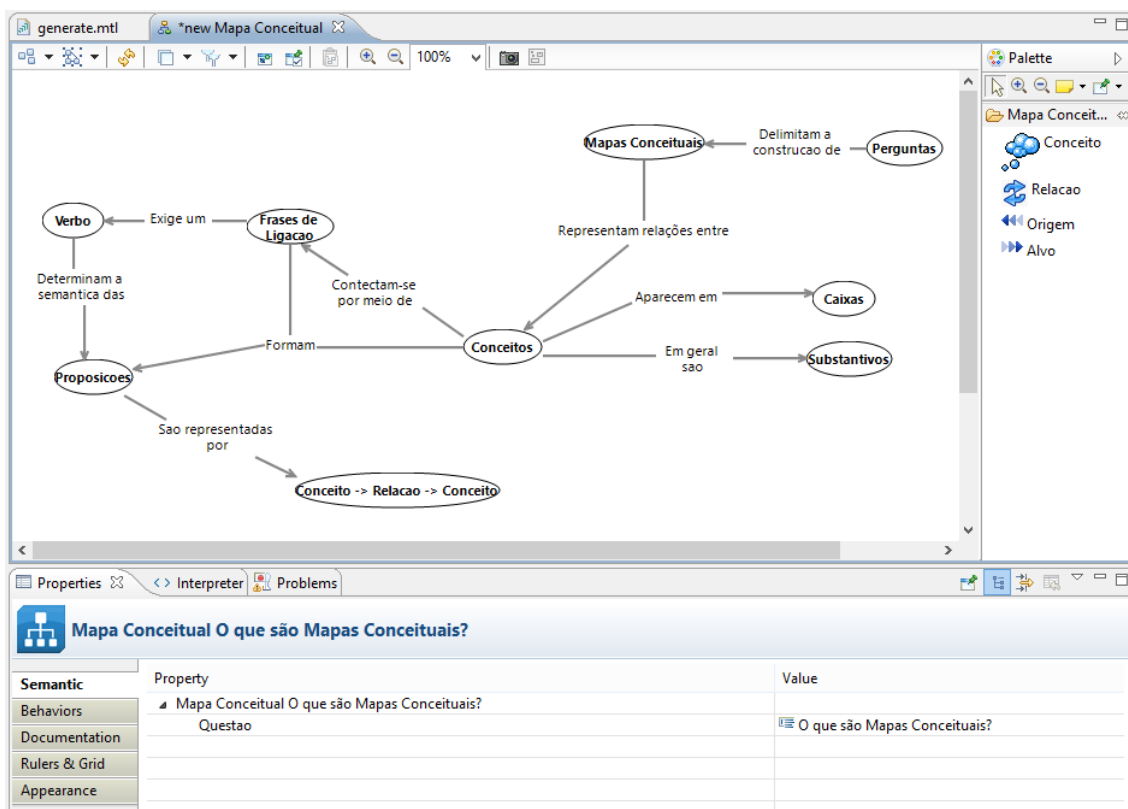


Figura 5.12 – Um Mapa conceitual construído no Editor do *iMap*.

Passo 2: O professor realiza a transformação do mapa em arquivo de texto (em *prolog*) através da execução das regras de transformação do *iMap*.

Esta tarefa é simples, bastando que, utilizando o *mouse*, o professor dê um clique com o botão direito sobre o arquivo que contém as regras de transformação e o execute através da opção “*Run*”. Isto é simples, pois o mecanismo de transformação pode ser previamente configurado com os dois parâmetros os quais necessita, quais sejam: 1) a instância da metaclassa “*MapaConceitual*” que deseja transformar e 2) o local (*path*) de saída do arquivo gerado pela transformação. A Figura 5.13 demonstra como é possível executar o arquivo de transformação, bastando clicar com o botão direito sobre o arquivo “*generate.mtl*” e executar a aplicação em destaque.

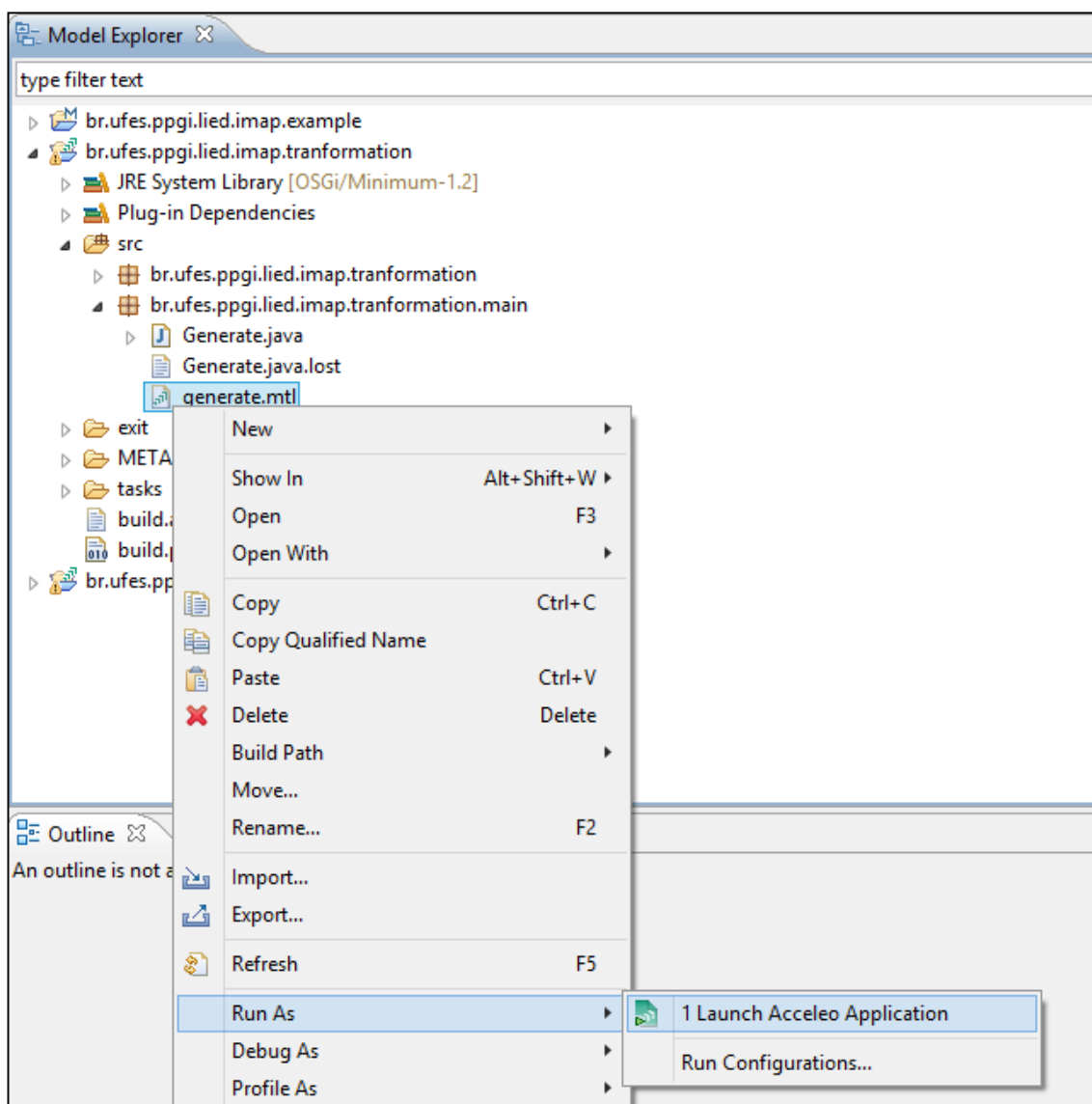


Figura 5.13 – Execução da Transformação do Mapa Conceitual em arquivo de base para *PIE*.

A execução do *Passo 2* resultará na criação do arquivo em “File0.pl”, cuja localização e conteúdo são apresentados na Figura 5.14.

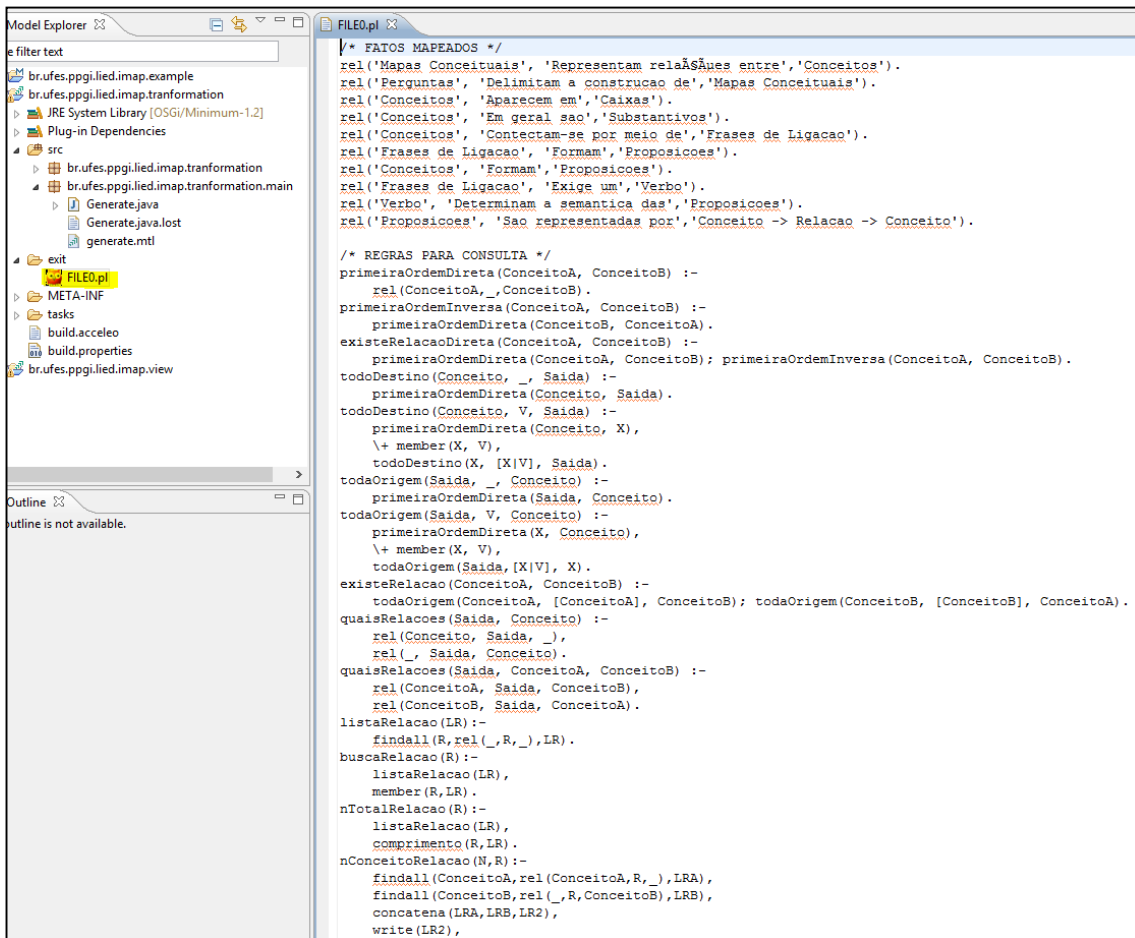


Figura 5.14 – Arquivo de saída do processo de transformação.

Com esses dois passos o sistema já possuirá as informações necessárias para que o professor (avaliador) possa iniciar o processo de avaliação do mapa produzido. Conforme apresentado na Figura 5.7, essa interação entre o professor e a base pode se dar de duas maneiras distintas: Interação direta por meio das Regras em *Prolog* ou Interação com apoio do *PLN-iMap* em Linguagem Natural. As duas Seções seguintes apresentarão como essas interações se dão em cada caso.

5.2.1 Interação para Consultas Diretas utilizando Regras do *PIE*

Uma vez que o editor do *iMap* é composto por duas camadas de *software* onde cada uma possui um propósito específico, a Figura 5.15 sintetiza um possível fluxo de utilização deste protótipo quando da interação direta através das regras oferecidas pelo *PIE*.

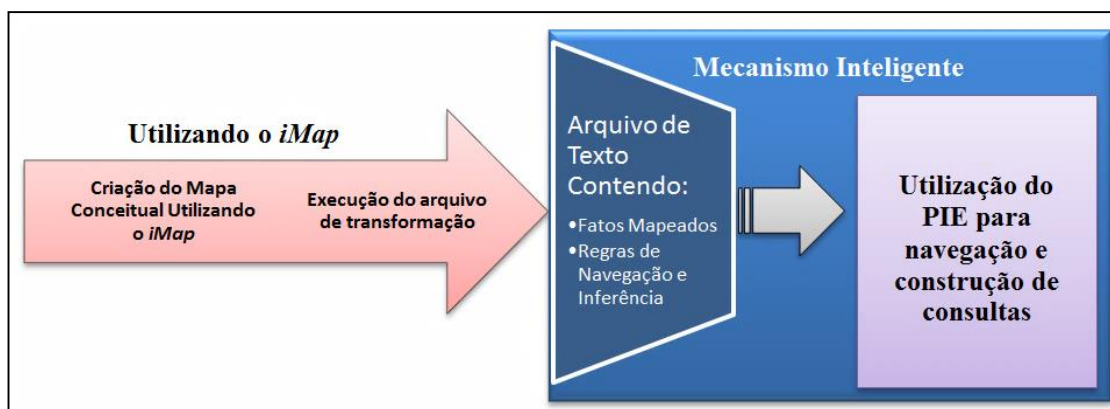


Figura 5.15 – Interação Direta através de Regras do PIE.

Percebe-se, portanto que, o *iMap* é utilizado nas fases iniciais, ou seja, na criação e transformação do mapa conceitual num arquivo que servirá de entrada para o mecanismo inteligente, o *PIE*. Por sua vez, esse mecanismo é “alimentado” pelo conhecimento presente no mapa conceitual e pode ser utilizado pelo avaliador para extrair as informações que deseja. Para isso, basta que os avaliadores utilizem as regras de consulta respeitando a sintaxe apresentada na Tabela 5.1.

Para exemplificar esta forma de interação, daremos prosseguimento ao *passo-a-passo* iniciado na seção anterior com os passos necessários para compreender as atividades de interação por meio de regras do *PIE*.

Passo 3: O professor inicializa o *PIE*¹⁸, que poderá ser previamente configurado para, no momento da inicialização, abrir o arquivo gerado pela transformação no *iMap*. A Figura 5.16 apresenta a tela logo após a inicialização do *PIE*, com parte do arquivo gerado pelo *iMap* posicionado à esquerda e, à direita, a caixa de diálogo que deve ser utilizada por ele para construir perguntas e visualizar as respostas dadas pelo *PIE*.

Como este mecanismo acaba de ser inicializado, o professor deverá atualizar a sua base de fatos e regras. Essa base é conhecida como memória de trabalho, pois armazena todas as informações que necessita para responder às perguntas, que serão feitas em seguida

¹⁸ Neste estudo de caso utilizamos um *Plugin* para a IDE *Eclipse*, denominada PDT (*Prolog Development Tool*), desenvolvido pela *SWI-Prolog*, disponível gratuitamente em: <http://swi-prolog.org>

pelo professor. Para realizar este processo basta clicar com o botão direito do *mouse* sobre o arquivo gerado e, em seguida, na opção “[Re]consult”.

The screenshot shows a Prolog IDE window titled 'prolog-iMapv4.pl'. The main editor contains the following Prolog code:

```

1 /*Fatos Mapeados*/
2 rel('Mapas Conceituais', 'Representam relações entre','Conceitos').
3 rel('Perguntas', 'Delimitam a construção de','Mapas Conceituais').
4 rel('Conceitos', 'Aparecem em','Caixas').
5 rel('Conceitos', 'Em geral são','Substantivos').
6 rel('Conceitos', 'Contectam-se por meio de','Frases de Ligacao').
7 rel('Frases de Ligacao', 'Formam','Proposicoes').
8 rel('Conceitos', 'Formam','Proposicoes').
9 rel('Frases de Ligacao', 'Exige um','Verbo').
10 rel('Verbo', 'Determinam a semantica das','Proposicoes').
11 rel('Proposicoes', 'São representadas por','Conceito -> Relacao -> Conceito').
12
13
14
15 /*Regras para Consulta*/
16 primeiraOrdemDireta(ConceitoA, ConceitoB) :-
17     rel(ConceitoA,_,ConceitoB).
18 primeiraOrdemInversa(ConceitoA, ConceitoB) :-
19     primeiraOrdemDireta(ConceitoB, ConceitoA).
20
21 existeRelacaoDireta(ConceitoA, ConceitoB) :-
22     primeiraOrdemDireta(ConceitoA, ConceitoB); primeiraOrdemInversa(ConceitoA,
23
24 todoDestino(Conceito, _, Saida) :-
25     primeiraOrdemDireta(Conceito, Saida).
26 todoDestino(Conceito, V, Saida) :-
27     primeiraOrdemDireta(Conceito, X),
28     \+ member(X, V),
29     todoDestino(X, [X|V], Saida).
30
31 todaOrigem(Saida, , Conceito) :-

```

The right-hand pane shows the execution output for the SWI Prolog: Default Process (Prolog) environment, displaying 'true.' and a prompt '2 ?- |'.

Figura 5.16 – Tela de Inicialização do PIE.

Passo 4: O professor utiliza a caixa de diálogo (lado direito da Figura 5.16) para construir as perguntas que deseja fazer ao mapa a fim de extrair as informações que deseja. Essas perguntas devem ser feitas utilizando a sintaxe definida no arquivo de entrada, que se encontram descritas na Tabela 5.1. O mecanismo inteligente é responsável por verificar se as perguntas construídas pelo professor estão dentro dos padrões definidos na sintaxe e, através de consultas feitas à sua memória de trabalho, construir uma resposta a essas perguntas. Algumas das respostas apresentam valores lógicos (*true* ou *false*) e outras apresentam um conjunto de saídas possíveis que atendem aos requisitos dados pelo professor (listas ou dados estatísticos).

Para exemplificar, a Figura 5.17 apresenta as respostas dadas pelo mecanismo inteligente, o PIE, às seguintes perguntas: (a) Existe relação direta entre os conceitos “Verbos” e “Proposições”? (b) Existe alguma relação entre os conceitos “Caixas” e “Substantivos”? (c) Quais conceitos o aluno identificou a partir do conceito “Verbos”? (d) Para o aluno, quais conceitos levam ao completo entendimento do conceito “Frases de Ligacao”? (e)

De quais relações o conceito “Conceitos” participa? (f) Em qual(is) relação(ões) os conceitos “Conceitos” e “Proposições” estão relacionados?

<pre>SWI Prolog: Default Process (Prolog) 8 ?- existeRelacaoDireta('Verbo', 'Proposicoes'). true (A)</pre>	<pre>SWI Prolog: Default Process (Prolog) 9 ?- existeRelacao('Caixas', 'Substantivos'). false. (B)</pre>
<pre>11 ?- todoDestino('Verbo', ['Verbo'], X). X = 'Proposicoes' ; X = 'Conceito -> Relacao -> Conceito' ; false. (C)</pre>	<pre>12 ?- todaOrigem(X, ['Frases de Ligacao'], 'Frases de Ligacao'). X = 'Conceitos' ; X = 'Mapas Conceituais' ; X = 'Perguntas' ; false. (D)</pre>
<pre>30 ?- quaisRelacoes(X, 'Conceitos'). X = ['Aparecem em', 'Em geral sao', 'Contectam-se por meio de', 'Formam', 'Representam relacoes entre']. (E)</pre>	<pre>36 ?- quaisRelacoes(X, 'Conceitos', 'Proposicoes'). X = ['Formam']. (F)</pre>

Figura 5.17 – Perguntas-Respostas através de Regras do *PIE*.

Como pode ser observado, o mecanismo inteligente permite que, de maneira rápida, o professor extraia informações do mapa, ou mesmo navegue em seus conceitos. Desta forma, o professor pode acompanhar e avaliar a aprendizagem do estudante certificando-se que estes foram capazes de estabelecer relações entre conceitos importantes presentes no artigo proposto sem depender de um dispendioso tempo para analisar e mapear os conceitos presentes nestes mapas, reduzindo, assim, o tempo gasto neste processo.

No entanto, esta forma de interação exige que o professor/avaliador seja conhecedor da estrutura das regras interpretáveis ao mecanismo de inferência. A Seção seguinte apresenta outra forma de interação que facilita esse processo por favorecer uma comunicação através de perguntas em Linguagem Natural.

5.2.2 Interação através do *PLN-iMap*

Como visto na Figura 5.7, esta pesquisa interessou por facilitar o processo de interação do professor/avaliador na consulta à base de inteligência do sistema. Criamos, para isto, uma ferramenta capaz de realizar tradução de perguntas construídas em Linguagem Natural para as regras interpretadas pelo *PIE*. Esta processo denominamos de *PLN-iMap* (vida Seção 5.1.3). A fim de verificar sua aplicabilidade, construímos um protótipo utilizando a linguagem Java e uma API de Prolog. Por meio dele o professor poderá interagir por meio dos seguintes passos:

Passo 3: O professor inicializa a aplicação do *PLN-iMap* que estará previamente configurada para interagir com o a base de inteligência gerada no *Passo 2*. Na tela inicial

do *PLN-iMap* o professor poderá compor a pergunta em linguagem natural e, em seguida, solicitar que o sistema realize a tradução através do botão “Traduzir”. O sistema analisará a pergunta e preencherá uma lista, logo abaixo do botão “Traduzir” com a(s) possível(eis) tradução(ões) da pergunta original em regras esperadas pelo *PIE*. A Figura 5.18 apresenta a tela inicial já com uma pergunta em linguagem natural e o resultado do processo de tradução.

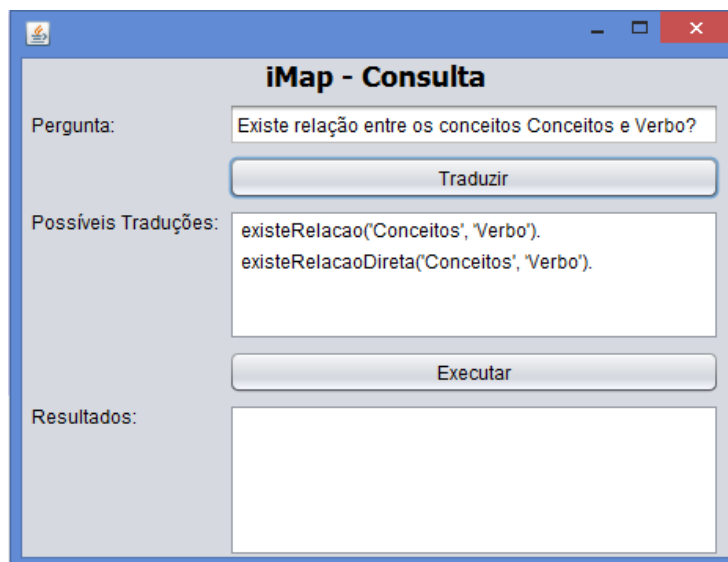


Figura 5.18 – Resultado do processo de tradução do *PLN-iMap*.

Passo 4: O professor pode selecionar uma das possíveis traduções para ver o resultado produzido. Caso ele não selecione nenhuma das opções, a primeira regra será executada. O resultado produzido pelo *PIE* é apresentado abaixo do botão utilizado para “Executar” a regra selecionada. A Figura 5.19 apresenta o resultado produzido pelo processamento da regra selecionada.

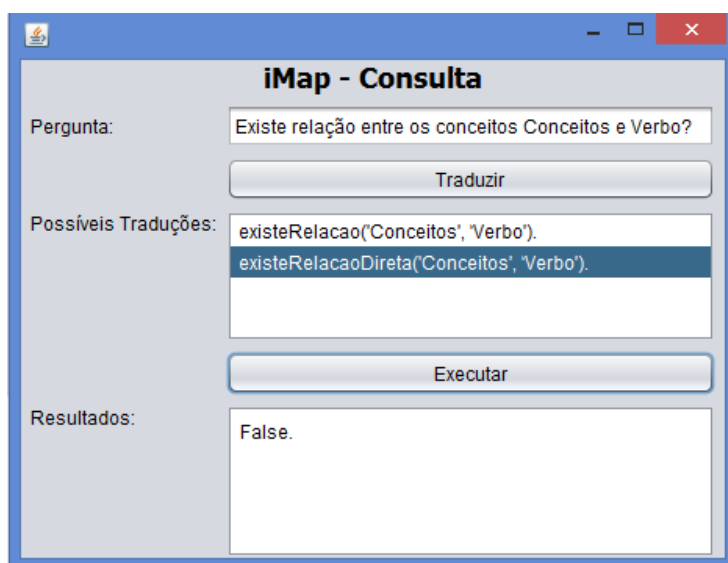


Figura 5.19 – Resultado do processamento da regra selecionada.

Como é possível observar, utilizando o *PLN-iMap* o professor/avaliador fica isento de conhecer a estrutura das regras para consulta à base de inteligência pois estas são produzidas automaticamente a partir dos parâmetros informados por ele na composição da pergunta em linguagem natural.

5.3 CONSIDERAÇÕES

Este capítulo teve por objetivo apresentar o protótipo da arquitetura proposta. Detalhamos alguns aspectos do desenvolvimento e do funcionamento interno e executamos uma prova de conceitos para validar a aplicação da ferramenta desenvolvida. Por meio desta prova de conceito percebemos grandes vantagens na utilização do mecanismo proposto, principalmente quando da avaliação de mapas conceituais com número elevado de conceitos e relações.

No entanto, alguns pontos necessitam de aprimoramentos para produção de uma ferramenta final. Os trabalhos futuros para aprimoramento do protótipo construído incluem:

1. Adotar o *WordNet* a fim de permitir identificação de termos cognatos nas consultas;
2. Analisar e tabular as perguntas sugeridas pelos docentes na investigação apresentada na Seção 2.3 deste trabalho a fim de identificar e produzir regras que ainda não foram implementadas no *PIE*;
3. Implementar serviço de *feedback* para que os professores auxiliem no aprimoramento do processo de tradução informando se os resultados apresentados foram ou não satisfatórios;
4. Produzir uma gramática independente de contexto que facilite a construção de perguntas em linguagem natural, por parte do avaliador, para garantir que sejam processáveis pelo *PLN-iMap*;
5. Realizar testes de qualidade da precisão no processo de tradução do *PLN-iMap*, dentre outros.

No entanto, refletimos também no que diz respeito à aplicação e impacto que esta ferramenta pode trazer à academia e ao mercado e concluímos que apesar de facilitar uma

operação com mapas conceituais (acompanhamento e avaliação de mapas), o *iMap* tornar-se-ia apenas mais uma nova solução isolada no mercado. Ou seja, assim como ocorre com diversas soluções apresentadas na Seção 2.2 deste trabalho, o mecanismo aqui proposto dificilmente será utilizado amplamente se não agregar novas soluções e ferramentas para compor um conjunto maior de operações com mapas conceituais.

Como o processo de reflexão resulta, quase sempre, em mudanças na forma de pensar e organizar os pensamentos, o Capítulo 6 tem por objetivo apresentar uma arquitetura computacional, considerada nesta pesquisa como arquitetura final, que explora justamente a aplicabilidade e expansão funcional do protótipo apresentado neste capítulo.

CAPÍTULO 6 O PROJETO CMPaaS

"Tudo evolui; não há realidades eternas: tal como não há verdades absolutas."

*(Friedrich Nietzsche)
"Humano, demasiado Humano"*

Este Capítulo tem por objetivo apresentar o projeto *CMPaaS*, no qual a arquitetura apresentada nesta pesquisa se enquadra como um serviço embrionário desta nova plataforma. Consideramos este projeto o principal resultado de nossas pesquisas com a união de soluções tecnológicas a mapas conceituais e, como tal, necessita de uma adequada apresentação.

Para apresentar este projeto, seguimentamos este capítulo nas seguintes Seções:

- Seção 6.1: Apresenta uma introdução que explicita a motivação para este projeto;
- Seção 6.2: Apresenta a nova arquitetura proposta e a forma como o *iMap* se encaixa dentro desta arquitetura, e;
- Seção 6.3: Apresenta algumas considerações e trabalhos futuros e em andamento relacionados a esta nova arquitetura.

6.1 MOTIVAÇÃO

Como vimos, dentre um enorme conjunto de aplicações, os mapas conceituais auxiliam estudantes a aprender de forma mais significativa, ajudam professores a indicar visualmente conceitos-chave e resumir suas inter-relações, além de também auxiliar grupos em tarefas colaborativas. Neste último caso, auxiliam na comunicação entre os membros e no gerenciamento do desenvolvimento do projeto, além de facilitar a captura e utilização do entendimento do assunto por cada aprendiz.

Mas a academia não é o único setor da sociedade interessado nas aplicações e ferramentas voltadas pra edição, manipulação e análise de mapas conceituais. Há ainda grande interesse empresarial na criação e utilização dessas ferramentas. A exemplo disso, houve

um aumento considerável na produção de ferramentas computacionais para a construção automática de mapas, diretamente de fontes de dados, tais como textos não estruturados. De fato, 10 de 15 abordagens para construir mapas conceituais automaticamente foram publicadas nos últimos três anos, ou seja, a maioria delas (46,67%) (KOWATA, CURY e BOERES, 2009).

No entanto, já que os mapas conceituais constituem assunto que motiva tanto a indústria quanto a academia, o que impede um desenvolvimento mais acelerado das pesquisas e soluções tecnológicas nesse domínio? Percebemos que existem dois possíveis problemas que afetam tanto a academia quanto a empresa e que carecem de esforços a fim de minimizá-los: (i) isolamento das pesquisas e (ii) problemas para a evolução e reutilização delas.

Falamos em fragmentação das pesquisas no sentido de haver real separação entre as soluções desenvolvidas, às vezes até mesmo pelo mesmo laboratório de pesquisa. Tomemos como exemplo as seguintes soluções: um gerador automático de mapas conceituais (KOWATA, CURY e BOERES, 2010), um ambiente para construção e versionamento de mapas conceituais (SANTOS, MENEZES, *et al.*, 2005), um ambiente para avaliação automática de mapas conceituais (ARAUJO, MENEZES e CURY, 2003), uma ferramenta para comparação de mapas conceituais (LAMAS, BOERES, *et al.*, 2006) e um solução para incremento de coesão e coerência em mapas conceituais (RIBEIRO, 2012). Todas essas pesquisas foram desenvolvidas pelo mesmo laboratório e culminaram na criação de ferramentas tecnológicas que auxiliam, cada uma em sua especificidade, na utilização de mapas conceituais. No entanto, apesar de apresentarem pontos em comum, as ferramentas desenvolvidas por esses pesquisadores não são capazes de se comunicar, ou seja, não compõem um conjunto de ferramentas capaz de definir uma solução completa de apoio à utilização de mapas conceituais. Dessa forma, não integradas, tornam-se de difícil acesso a usuários em geral que executam diferentes tarefas com os mapas conceituais.

Podemos incluir nesse mesmo contexto de fragmentação das pesquisas, as diversas soluções (ferramentas) que poderiam ser mais facilmente desenvolvidas caso não fosse necessário preocupar-se com toda a infraestrutura de criação e gestão de mapas conceituais. Por exemplo, em Lourenço (2008) é apresentado uma abordagem para

análise e avaliação de mapas conceituais. É possível vislumbrar uma série de ferramentas computacionais que auxiliariam o autor no processo de coleta dos dados e na composição das estatísticas apresentadas. No entanto, o desenvolvimento dessas soluções tornam-se inviáveis pois isto envolveria o desenvolvimento de um ambiente completo, com autenticação de usuários, sistema de gestão e edição de mapas conceituais etc.

Como é possível observar, portanto, a fragmentação das pesquisas nos faz pensar que tanto a evolução quanto a reutilização de tais pesquisas devem ser questionadas. O resultado prático dessa fragmentação é observado quando se colocam em prática abordagens pedagógicas baseadas em mapas conceituais. Em alguns casos, inexistem soluções computacionais que auxiliem os professores em suas análises e avaliações, sejam elas estatísticas ou cognitivas, acarretando-lhes uma pesada carga de processamento cognitivo e manual, como já apresentado nos capítulos anteriores.

Outro problema igualmente sério reside no fato de que a comunidade em geral e a academia parecem não estar falando a mesma língua. Afinal de contas, por que a comunidade em geral tem tanta dificuldade em acessar, de maneira fácil e rápida, as pesquisas e seus resultados? Consideramos importante a criação de um mecanismo de acesso eficiente aos resultados das pesquisas científicas, para que a comunidade possa contribuir para a evolução delas. O que propomos, portanto, é o lançamento de bases para uma convivência mais estreita entre o mundo acadêmico e a sociedade em geral.

Portanto, damos ênfase nesta pesquisa à necessidade da adoção de técnicas de desenvolvimento que facilitem o reaproveitamento dos resultados atuais num curto tempo futuro. Para isso, propomos aqui a criação de uma plataforma orientada a serviços cujos objetivos são: (i) transformar os trabalhos acadêmicos em serviços computacionais acessíveis à comunidade em geral e (ii) disponibilizar mecanismos que permitam que a comunidade possa criar, estender e integrar novos serviços, ou funcionalidades, à plataforma.

Nesta plataforma, que estamos denominando *CMPaaS*¹⁹, a arquitetura discutida e apresentada ao longo desta dissertação se enquadra como um dos serviços a serem

¹⁹ Um acrônimo de *Concept Map as a Service* (Mapas Conceituais como Serviços).

prestados pela plataforma. Como serviço ela pode ser consumida diretamente por usuários finais ou por outros serviços que necessitem de algumas de suas funções específicas para outros serviços.

A Seção seguinte apresentará, como mais detalhes, a organização da arquitetura proposta para o CMPaaS e tornará possível identificar como o *iMap* se enquadra dentro desta nova arquitetura.

6.2 *CMPaaS*: A PLATAFORMA

Em se tratando de desenvolvimento de soluções computacionais, o assunto em voga nos últimos anos é, sem dúvida, a computação em nuvem (*Cloud Computing*). Trata-se de um modelo de computação no qual o processamento, o armazenamento e as soluções computacionais (*softwares*) são oferecidos por um provedor de serviços e não da maneira habitual “*in loco*”, sendo acessados remotamente via internet. Essa tecnologia permite o uso de aplicações e a obtenção de informações de qualquer lugar, a partir de qualquer plataforma e a qualquer hora, utilizando a *Web* ao invés de aplicativos instalados localmente em um computador.

A principal vantagem apresentada pelas aplicações disponibilizadas com base na computação em nuvem é sua capacidade de ser facilmente estendida e incorporada a outras aplicações, aumentando a produtividade na criação de novas aplicações. É também por isso que as grandes empresas de tecnologia e redes sociais (e.g, *Facebook*, *Apple*, *Google*, *Twitter* etc) possuem seus serviços disponíveis nesse modelo de computação. Apenas para citar um exemplo, uma aplicação disponível na nuvem amplamente utilizada é o *Google Maps*²⁰. Hoje, são incontáveis as aplicações que estendem as funcionalidades dessa aplicação oferecendo serviços complementares, tais como os aplicativos de geolocalização que controlam a rota, ritmo e calorias consumidas por um atleta ao longo de uma atividade física, disponíveis na grande maioria dos *smartphones* atuais.

²⁰ *Google Maps*: API(Application Programming Interface) desenvolvida pela empresa Google que permite a criação de mapas com locais definidos, controle de zoom, tipos de mapa, geração de rotas, pesquisa por estabelecimentos, entre outras funcionalidades. Disponível em: <https://developers.google.com/maps/>

É exatamente essa capacidade de expansão e produtividade que estamos explorando neste projeto. Estamos criando os serviços básicos de edição, gestão e manipulação de mapas conceituais que estarão disponíveis a qualquer pessoa no mundo por meio de nossa plataforma de serviços.

À plataforma orientada a serviços proposta chamamos *CMPaaS*. A arquitetura que está sendo utilizada neste projeto, conhecida como *SOA*²¹, tem como uma de suas características fundamentais a sua capacidade de promover integração. O que, na prática, significa dizer que novos serviços, que estendem as funcionalidades dos serviços oferecidos pelo *CMPaaS*, podem ser desenvolvidos e disponibilizados por qualquer pessoa, em qualquer parte do mundo. E é justamente a promoção desse cenário de colaboração e de integração entre academia e empresas um dos objetivos centrais desse projeto.

Destacamos ainda que no contexto deste trabalho os serviços da plataforma estão sendo implementados na forma de *Web Services*²². Por se tratar de um mecanismo de funcionamento interno, os *Web Services* precisam ser consumidos por uma aplicação que forneça uma interface de utilização ao usuário final. Sendo assim, o *CMPaaS* está associado a um portal que serve de interface final para uso das ferramentas oferecidas, consumindo os serviços especificados em sua base. A esse portal estamos dando o nome de “Portal do Conhecimento”. Na Figura 6.1(a) é possível observar como as ferramentas do portal interagem com os serviços disponibilizados pela plataforma enquanto a Figura 6.1(b) apresenta como a comunidade poderá integrar novos serviços à plataforma *CMPaaS*.

²¹ **SOA (Service Oriented Architecture):** É um estilo de arquitetura de software cujo princípio fundamental prega que as funcionalidades implementadas pelas aplicações devem ser disponibilizadas na forma de serviços.

²² **Web Service:** É uma solução computacional utilizada na integração de sistemas e na comunicação entre diferentes aplicações.

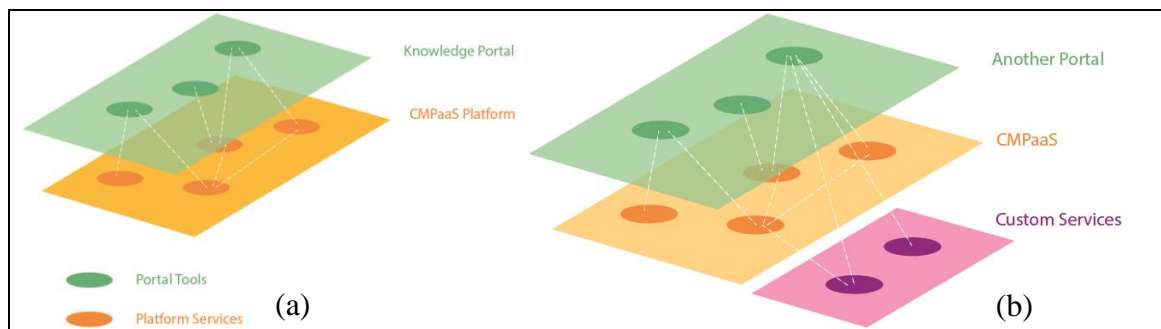


Figura 6.1 – (a) Visão de Integração do Portal do Conhecimento com a Plataforma. (b) Visão de Integração de Serviços Externos à Plataforma.

Como é possível observar, as ferramentas disponíveis no portal estão consumindo os serviços da plataforma. Destaca-se que uma única ferramenta pode estar consumindo mais de um serviço simultaneamente. A título de exemplo, tomemos a ferramenta de edição de mapas do portal. Enquanto um usuário edita um mapa, o serviço de edição está trabalhando para fornecer as características visuais do mapa para o editor. Ao mesmo tempo, o serviço de autenticação está ativo para informar a quaisquer outros serviços qual usuário está realizando alterações no mapa. E, mais ainda, o serviço de versionamento precisa persistir cada uma das versões do mapa. Temos, portanto, uma única ferramenta consumindo três serviços da plataforma.

Também ocorre de um único serviço da plataforma ser consumido por diversas ferramentas do portal. Como exemplo disso podemos mencionar o serviço de listagem de mapas. Ele pode ser utilizado pela ferramenta de consulta de mapas conceituais bem como pela ferramenta de mesclagem de mapas. Ambos cenários demonstram como essa arquitetura permite a integração de fato de ferramentas.

Apresentaremos agora mais detalhes da arquitetura interna do *CMPaaS* e descreveremos seus principais componentes, serviços e ferramentas bem como as etapas do desenvolvimento em que se encontra o portal.

6.2.1 Arquitetura Geral do *CMPaaS* e integração do *iMap*

Internamente, a plataforma é formada por cinco camadas (Figura 6.2):

- **Serviços externos:** responsável por fornecer os serviços que serão consumidos pela comunidade (e.g., entidades governamentais, entidades privadas e pessoas

comuns). Em outras palavras, a camada Serviços Externos é responsável por realizar a interação entre o mundo externo e o mundo interno da plataforma.

- **Processos de negócio:** responsável por gerenciar as regras de negócio da plataforma, ou seja, é nesta camada que estão implementados os processos que fazem a gestão de todos os serviços disponibilizados pela plataforma.
- **Serviços de aplicações internas:** responsável por gerenciar os serviços utilizados pelos processos de negócio. Além disso, é nesta camada que os serviços dos componentes internos são disponibilizados para os processos da plataforma.
- **Componentes internos:** a responsável por gerenciar os componentes que proveem todos os serviços da plataforma, por exemplo, é nesta camada que estão os componentes para a gestão de usuário e de mapas conceituais.
- **Serviços de aplicações externas:** serviços oferecidos externamente que são consumidas por processos internos.

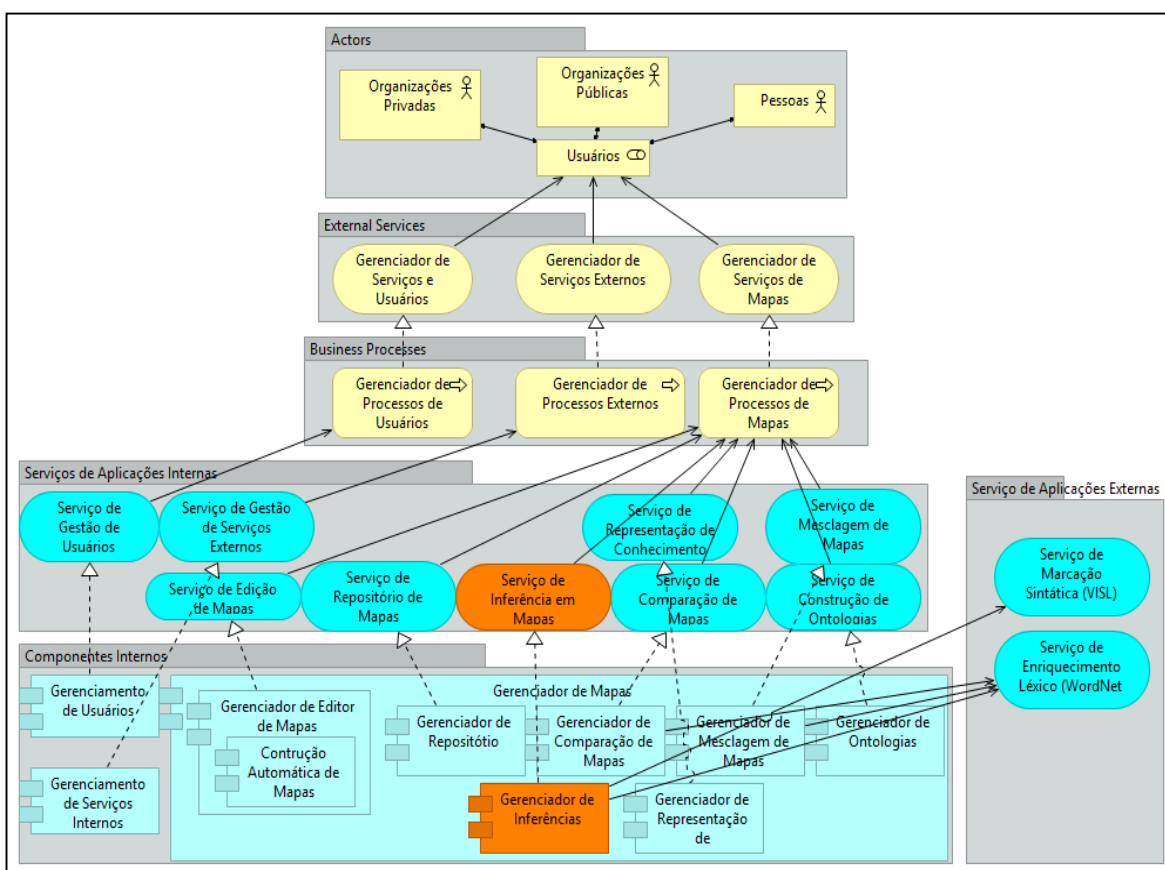


Figura 6.2 – Arquitetura Funcional da plataforma *CMPaaS*.

Da Figura 6.2 é possível perceber como o *iMap* (Serviço de Inferência em Mapas) estará integrada à plataforma *CMPaaS*. A perspectiva é que ele se torne um dos serviços oferecidos e consumidos pelo “Portal do Conhecimento”.

A Figura 6.3, a seguir, apresenta uma visão simplificada dos principais componentes da arquitetura conceitual geral do *CMPaaS* bem como de suas possíveis formas de interações com o mundo exterior.

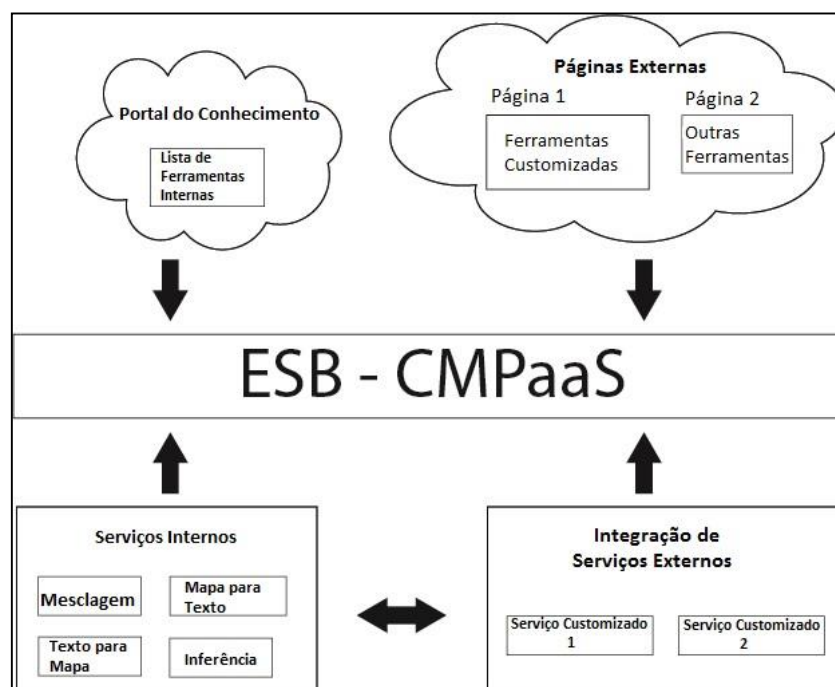


Figura 6.3 – Arquitetura Conceitual do *CMPaaS*.

Como é possível observar, existem três componentes na arquitetura do *CMPaaS* que serão descritos nas subseções seguintes.

6.2.1.1 ESB (*Enterprise Service Bus*)

É o componente da arquitetura que fornece uma base de serviços para arquiteturas mais complexas via *driver* de evento e padrões baseados em mensagens. Ele é responsável por fornecer uma abstração de camadas na implementação do sistema de mensagens que permite a integração da arquitetura para explorar o valor das mensagens, sem a necessidade de escrita de códigos complexos. Esse é o componente central do projeto pois objetiva remover o acoplamento entre os serviços e o meio de transporte.

6.2.1.2 Serviços da Plataforma

Os serviços são as funcionalidades que serão disponibilizadas pela plataforma para a camada de apresentação (o portal). Essas funcionalidades incluem: serviços de autenticação de usuários, persistência e gestão de mapas conceituais, serviços de análise e manipulação de mapas conceituais, dentre outros. Cabe destacar que nem todos os serviços serão de nossa autoria, uma vez que a comunidade em geral poderá se apropriar dos serviços básicos por nós disponibilizadas e gerar novos serviços que poderão ser fornecidos por eles, ou submetidos ao *CMPaaS* para tornarem-se parte integrante de nossos serviços.

Aos serviços disponibilizados diretamente pelo *CMPaaS* estamos denominando de serviços internos. Já os fornecidos por terceiros, chamamos de serviços externos. Cabe destacar que todos os serviços podem se comunicar, integrando-se aos demais serviços, através do *ESB*.

Os serviços internos são funcionalidades necessárias para a gestão dos usuários e seus mapas conceituais. Esses serviços formam o *kernel* da plataforma, ou seja, eles compõem a infraestrutura básica da plataforma. Já os serviços externos são os serviços adicionados à plataforma pela comunidade em geral, seja acadêmica ou comunidade em geral. É por meio desses novos serviços externos que a plataforma torna-se extensível e dinâmica. Dessa forma, a comunidade pode estender a plataforma existente adicionando e ofertando novos serviços aos usuários da plataforma. É importante notar que o processamento dos serviços externos não é, inicialmente, de responsabilidade da plataforma. Tais serviços serão processados fora da infraestrutura da plataforma. A plataforma tem apenas a responsabilidade de gerenciar o uso dos serviços por ela ofertados.

Ainda, em relação aos serviços internos, esses são classificados em: 1) serviços de gestão da plataforma e 2) serviços básicos de construção e manipulação de mapas conceituais. Os serviços internos de gestão da plataforma são responsáveis pela gestão da plataforma em geral, ou seja, esses serviços são responsáveis por gerenciar usuários, os serviços externos (e.g., adicionar usuário e adicionar novos serviços externos) e a utilização de serviços de manipulação de mapas. Os serviços internos de manipulação de mapas conceituais são responsáveis pelas funcionalidades básicas de criação e manipulação de mapas

conceituais. Assim, no contexto desta plataforma, os serviços básicos, frutos de trabalhos acadêmicos já definidos e em desenvolvimento até o momento, são:

- Serviço de edição de mapas conceituais (SANTOS, MENEZES, *et al.*, 2005);
- Serviço de repositório de mapas conceituais (SANTOS, MENEZES, *et al.*, 2005);
- Serviço de comparação de mapas conceituais (LAMAS, BOERES, *et al.*, 2006);
- Serviço de mesclagem de mapas conceituais (VASSOLER, PERIN e CURY, 2014);
- Serviço de geração de mapas conceituais de forma automatizada (KOWATA, CURY e BOERES, 2009; 2010; 2011);
- Serviço de ontologias a partir de mapas conceituais (PERIN, CURY e MENEZES, 2012);
- Serviço de inferência em mapas conceituais (PERIN, CURY e MENEZES, 2014);
- Serviço de representação automática das classes de conhecimento definidas por Piaget (CURY, MENEZES e ARAGON, 2013).

Algumas dessas pesquisas já apresentam resultados que serão apresentados em outros artigos, outras, que findaram antes do início do desenvolvimento desse projeto, apresentam protótipos funcionais que estão sendo, aos poucos, incorporados à plataforma do *CMPaaS*.

6.2.1.3 Serviços Externos

Outro objetivo desta plataforma é a disponibilização dos serviços para a comunidade, ou seja, a preocupação com a apresentação visual de suas ferramentas. Isto é feito de duas formas: (i) portal do conhecimento e (ii) páginas externas. Por meio do portal do conhecimento a comunidade pode acessar, de uma forma simples e rápida, os serviços internos presentes na plataforma. Em outras palavras, o portal é um meio pelo qual a comunidade (professores, alunos e outros interessados) pode utilizar os serviços da plataforma. O portal é a interface de todos os serviços internos da plataforma.

Por outro lado, a plataforma será um meio pelo qual a comunidade desenvolvedora irá consumir os serviços internos e externos bem como divulgar novos serviços para o portal. Vale enfatizar que um serviço externo pode consumir outros serviços externos e não

apenas serviços internos. No entanto, cabe aos responsáveis pelo desenvolvimento dos serviços externos o fornecimento de uma interface de uso de seus serviços. Eles podem fazer isso por meio de páginas externas ou por meio de aplicações para outros dispositivos.

No desenvolvimento deste projeto, adotamos técnicas intimamente relacionadas ao conceito de SOA que, por sua característica, não restringe os desenvolvedores a adotarem linguagens ou tecnologias de desenvolvimento específicas. Sendo assim, há desenvolvedores trabalhando em diferentes linguagens e adotando diferentes técnicas de desenvolvimento que, no entanto, serão integradas com a utilização de uma camada de serviços (*Enterprise Service Bus*).

A fim de garantir a disponibilidade dos serviços da plataforma de maneira a não comprometer outros serviços que deles dependem, serão adotadas técnicas de versionamento que permitirão um controle de versões e possíveis reaplicações de versões anteriores dos serviços.

6.2.2 O *CMPaaS* hoje

Atualmente, o projeto *CMPaaS* encontra-se em desenvolvimento. Como sua arquitetura é baseada no padrão SOA, o desenvolvimento dos vários serviços que o compõem transcorre paralelamente por diferentes equipes compostas por estudantes de graduação sob supervisão de pesquisadores. A Tabela 6.1 apresenta os serviços que se encontram em fase de projeto, de desenvolvimento, de conclusão ou de testes.

Tabela 6.1 – Funcionalidades do *CMPaaS* em Pesquisa/Desenvolvimento

Serviços do <i>CMPaaS</i>				
Serviço	Projeto	Desenvolvimento	Conclusão	Testes
Autenticação de Usuários	X	X	X	X
Persistência de Mapas	X	X	X	X
Geração de Imagens	X	X	X	X
Transformação (Mapa para JSON)	X	X	X	X
Listagem de Mapas Armazenados	X	X	X	X
Mesclagem de Mapas	X	X	X	X
Extração de Proposições	X	X	X	X
Integração com ESB	X	X		
Geração automática de Mapas	X	X		

<i>Ferramenta de Inferência em Mapas</i>	X	X		
<i>Perguntas para Mapas</i>	X	X		
Busca por Mapas	X			
Geração automática de Layouts	X			
Validação de Mapas	X			
Versionamento de Mapas	X			
Comparação de Mapas	X			

É importante destacar que a Tabela 6.1 apresenta apenas os serviços que são resultados de trabalhos e pesquisas que estão, atualmente, em andamento. Vislumbramos ainda a integração de diversas soluções computacionais já desenvolvidas como frutos de pesquisas anteriores. Além disso, por sua natureza extensível, acreditamos que diversos novos serviços poderão ser pensados, desenvolvidos e incorporados ao CMPaaS num tempo futuro.

Como os serviços internos da plataforma necessitam de ferramentas visuais para serem explorados pelos usuários finais, o portal do conhecimento e suas ferramentas estão sendo desenvolvidos em paralelo ao desenvolvimento do *CMPaaS*. A Tabela 6.2 apresenta as ferramentas e as atividades que já se encontram em desenvolvimento para o portal. Novamente, à medida que as pesquisas em andamento avançam e novos serviços sejam agregados ao *CMPaaS*, o portal do conhecimento passará a contar com novas ferramentas que serão disponibilizadas para sua comunidade de usuários finais.

Tabela 6.2 – Ferramentas do Portal do Conhecimento em Desenvolvimento

Ferramentas e Atividades do Portal do Conhecimento				
Ferramenta / Atividade	Projeto	Desenvolvimento	Conclusão	Teste
Editor de Mapas Conceituais	X	X	X	X
Ferramenta para Mesclagem de Mapas	X	X	X	X
Página de Gerenciamento de Mapas	X	X	X	
Editor de texto para geração de Mapas	X	X		
Identidade Visual	X			
Layout de páginas	X			

6.3 CONSIDERAÇÕES

Mapas conceituais possuem aplicações interessantes em vários segmentos da sociedade. No entanto, muitas de suas possíveis abordagens de aplicação são evitadas devido à ausência de aparato ferramental que auxilie no processo de elicitação, organização,

análise e avaliação dos mapas. Percebemos que muitas abordagens de uso envolvem uma pesada carga de processamento cognitivo por parte do professor, ou especialista, ao analisar um ou vários mapas gerados por um grupo. Percebemos também que a dificuldade em desenvolver soluções específicas para análise e processamento automático de mapas conceituais reside, na maioria das vezes, no fato de não existir uma plataforma de serviços extensível, por meio da qual os desenvolvedores possam criar suas soluções sem preocupar-se com os serviços básicos necessários para que sua aplicação seja completa, como a gestão de contas de usuários, a manipulação e a edição de mapas conceituais etc.

Assim, nossas pesquisas culminaram por propor o *CMPaaS*, uma plataforma orientada a serviços cujo objetivo é fornecer à comunidade em geral, além dos serviços básicos de gestão e manipulação de mapas conceituais, serviços mais avançados para análise, processamento estatístico e avaliação de mapas conceituais. Assim, o *iMap* caracteriza-se como uma das soluções disponíveis no *CMPaaS* que permite que os serviços sejam estendidos e novos serviços possam ser gerados e integrados, expandindo de maneira significativa as aplicações para as diversas abordagens de uso dos mapas conceituais.

CAPÍTULO 7 CONCLUSÃO E TRABALHOS FUTUROS

"Onde quer que você esteja, esse é o seu ponto de partida."

(Kabir)

Esta dissertação objetivou apresentar um sistema de pergunta-resposta dotado da capacidade de realizar inferências em mapas conceituais a fim de facilitar o processo de análise e acompanhamento da aprendizagem conceitual.

Diversas técnicas e soluções tecnológicas de diversas áreas do conhecimento foram investigadas e integradas ao longo do desenvolvimento desta pesquisa e do protótipo da ferramenta proposta, tais como: Mapas Conceituais como ferramenta para representação de conhecimento, Sistemas de Pergunta-Resposta, Sistemas Inteligentes, Processamento de Linguagem Natural, Arquitetura de Sistemas Orientado a Serviços, Desenvolvimento Orientado a Modelos, dentre outros.

Nesta arquitetura aplicamos técnicas atuais que são importantes no que diz respeito ao desenvolvimento de pesquisas em processamento de linguagem natural, tais como: análise e processamento automática de sintática e semântica de perguntas e identificação de relações léxicas entre palavras para identificação de termos cognatos.

Diversos sistemas realizam parte das técnicas, mas a arquitetura proposta é inovadora no que diz respeito ao agrupamento de todas elas. Os experimentos realizados demonstraram que a arquitetura proposta é viável e importante para facilitar o processo de acompanhamento e avaliação de mapas.

Destacamos como características principais da arquitetura proposta a capacidade de processar perguntas em linguagem natural e utilizá-las para realização de inferências em bases de conhecimento de um sistema inteligente previamente alimentada com o conhecimento presente num mapa conceitual.

Além disso, a análise da aplicabilidade do sistema proposto culminou num importante avanço no que diz respeito à sua evolução e aos possíveis trabalhos futuros. Assim, esta

pesquisa evolui para uma proposta de alcance maior, o *CMPaaS*, cujo objetivo é permitir que a comunidade participe ativamente na utilização, proposta e criação de novos serviços para a arquitetura proposta. Dentro desta arquitetura, o *iMap* tornar-se-há apenas um dos serviços oferecidos pela plataforma que poderá ser expandido por outros serviços da plataforma.

Vislumbramos avanços e trabalhos futuros em ambos os sentidos investigados aqui – no mecanismo de inferências do *iMap* e nos serviços oferecidos para o *CMPaaS*.

Assim, para o *iMap* podemos citar como trabalhos futuros:

1. Definição da gramática intermediária livre de contexto para validação de perguntas em Linguagem Natural;
2. Aprimoramento do mecanismo de tradução de perguntas para obtenção de melhores resultados do processo de tradução;
3. Investigação por meio de testes de revocação e precisão para avaliação da capacidade e evolução do processo de tradução;
4. Criação e avaliação de mecanismo de avaliação do processo de tradução, permitindo aos docentes avaliarem a precisão do sistema por fornecerem *feedback* quanto aos resultados apresentados;
5. Análise e investigação junto a docentes para criação de novas regras para inferência em mapas conceituais;
6. Implementação e utilização do *WordNet* no processo de tradução e de alimentação da base de conhecimento do sistema inteligente;
7. Evolução da arquitetura para permitir utilização em outros idiomas;
8. Modularização e integração dos serviços do *iMap* no *CMPaaS*, e;
9. Investigação de resultados práticos da aplicação da ferramenta proposta.

Quanto ao *CMPaaS*, destacamos que sua capacidade de gerar trabalhos futuros é virtualmente infinita, ou seja, sua concepção é permitir a criação de quantas soluções computacionais para uso de mapas conceituais forem mentalizadas por qualquer pesquisador ao redor do mundo. No entanto, alguns serviços que já possuem investigação documentadas estão sendo desenvolvidas e, aos poucos, integradas ao portal do

conhecimento do *CMPaaS*. Além do *iMap*, alguns serviços que já estão sendo integrados à arquitetura são:

1. Serviço de edição de mapas conceituais (SANTOS, MENEZES, *et al.*, 2005);
2. Serviço de repositório de mapas conceituais (SANTOS, MENEZES, *et al.*, 2005);
3. Serviço de comparação de mapas conceituais (LAMAS, BOERES, *et al.*, 2006);
4. Serviço de mesclagem de mapas conceituais (VASSOLER, PERIN e CURY, 2014);
5. Serviço de geração de mapas conceituais de forma automatizada (KOWATA, CURY e BOERES, 2009; 2010; 2011);
6. Serviço de ontologias a partir de mapas conceituais (PERIN, CURY e MENEZES, 2012);
7. Serviço de representação automática das classes de conhecimento definidas por Piaget (CURY, MENEZES e ARAGON, 2013).

CAPÍTULO 8 REFERENCIAS

ABRAHAM, A. 130: Rule-Based Expert Systems. **One Problem Solving Using Heuristics.**, Oklahoma, 2005. 909-919.

ALENCAR, L. F. D. **Aelius**: uma ferramenta para anotação automática de corpora usando o NLTK. Encontro de Linguística de Corpus. Porto Alegre, RS, Brasil: Anais do evento. 2010.

ALVES, A. O.; PEREIRA, F. C.; CARDOSO, A. **Automatic Reading and Learning from Text**. International Symposium on Artificial Intelligence. Fort Panhala (Kolhpur), India: ISAI'2001. 2001. p. 302-310.

AMORIM, M. T. C. D. D. **Um Sistema Inteligente Baseado em Ontologia para Apoio ao Esclarecimento de Dúvida**. UFES - Universidade Federal do Espírito Santo. Vitória, ES, Brasil. 2012. Dissertação de Mestrado.

ARAUJO, A. M. T.; MENEZES, C. S. D.; CURY, D. **Apoio Automatizado à Avaliação da Aprendizagem Utilizando Mapas Conceituais**. Simpósio Brasileiro de Informática na Educação. Rio de Janeiro, RJ - Brasil: Anais do SBIE. 2003. p. 183-191.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 14724: informação e documentação - projeto de pesquisa - apresentação**. ABNT. Rio de Janeiro. 2005.

ATHENIKOS, S. J.; HYOIL, H. Biomedical question answering: A survey. **Computer Methods and Programs in Biomedicine**, New York, USA, v. 99, n. 1, 2010.

ATKINSON, D. J.; JAMES, M. L.; LAWSON, D. **Automated Spacecraft Monitoring**. International Conference on Systems, Man and Cybernetics. [S.l.]: IEEE. 1990. p. 756-761.

ATSERIAS, J. J. et al. **FreeLing 1.3**: Syntactic and semantic services in an open-source NLP library. 5th International Conference on Language Resources and Evaluation. Elra,

Genoa, Italy: Proceedings of the 5th International Conference on Language Resources and Evaluation. 2006.

AUSUBEL, D. P. **The psychology of meaningful verbal learning**. New York: Grune and Stratton, 1963.

AUSUBEL, D. P. **Educational psychology: A cognitive view**. New York: Holt, Rinehart and Winston., 1968.

AUSUBEL, D. P.; NOVAK, J. D.; HANESIAN, H. **Educational psychology: A cognitive view**. 2nd ed. ed. New York: Holt, Rinehart and Winston, 1978.

BAI, S.-M.; CHEN, S.-M. Automatically constructing concept maps based on fuzzy rules for adapting learning system. **Expert Systems with Applications**, v. 1, p. 41-49, 2008. ISSN ISSN: 0957-4174.

BARCELOS, G. T.; BATISTA, S. F. **Uso de aplicativos em tablets no estudo de sistemas lineares: percepção de licenciandos em Matemática**. Conferência Internacional sobre Informática na Educação. Porto Alegre: TISE'13. 2013.

BESSMERTNY, I.; KULAGIN, V. Semantic Network as a Knowledge Base in Training System., 2009. Disponível em: <<https://smartech.gatech.edu/bitstream/1853/24417/1/154.pdf>>. Acesso em: 11 Julho 2014.

BICK, E. **Automatic Parsing of Portuguese**. II Encontro para o Processamento Computacional de Português Escrito e Falado. Curitiba, PF: Anais do II Encontro para o Processamento Computacional de Português Escrito e Falado. 1996.

BICK, E. The Parsing System "PALAVRAS": Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework. **Syddansk Universitet**, 2000. Disponível em: <<beta.visl.sdu.dk/~eckhard/pdf/PLP20-amilo.ps.pdf>>. Acesso em: 17 Julho 2014.

BICK, E. et al. **The Annotation of the C-ORAL-Brasil through the Implementation of the Palavras Parser**. 8th International Conference on Language Resources and Evaluation. Istanbul: Proceedings of LREC2012. 2012. p. 3382-3386.

BOSCH, A. V.; BOUMA, G. **Interactive multi-modal question answering**. [S.l.]: Springer-Verlag, 2011.

BROWN, A. An introduction to Model Driven Architecture. **IBM - DeveloperWorks**, 17 February 2004. Disponível em: <<http://www.ibm.com/developerworks/rational/library/3100.html>>. Acesso em: 12 August 2014.

CAÑAS, A. J.; CARVALHO, M. **Concept Maps and AI: an Unlikely Marriage?** Simpósio Brasileiro de Informática na Educação. Manaus - AM: Anais do SBIE. 2004. p. 1-10.

CARPINETO, C.; ROMANO, G. A survey of automatic query expansion in information retrieval. **ACM Computing Survey**, New York, USA, v. 44, n. 1, 2012.

CARVALHO, M. J. S.; NEVADO, R. A. D.; MENEZES, C. S. D. **Arquiteturas Pedagógicas para Educação a Distância: Concepções e Suporte Telemático**. XVI Simpósio Brasileiro de Interformática na Educação. Juiz de Fora - MG - Brasil: Anais do XVI SBIE. 2005.

CHANG, T. H. et al. **Automatic Concept Map Constructing using top-specific training corpus**. Asia-Pacific Educational Research Association Board Meeting. Singapore: Proceedings of APERA'2008. 2008.

CHEN, N.-S. et al. Mining e-Learning domain concept map from academic articles. **Computers & Education**, v. 1, n. 3, p. 1009-2021, April 2008. ISSN ISSN: 0360-1315.

CHIRITA, A.-P.; FIRAN, S. C.; NEJDAL, W. **Personalized query expansion for the web**. 30th Annual International Conference on Research and Development in Information Retrieval. New York, USA: Proceedings of SIGIR'07. 1982. p. 139-149.

CLARIANA, R. R.; KOUL, R. **A Computer-Based Approach for Translating Text into Concept Map-Like Representations.** International Conference on Concept Mapping. Pamplona, Spain: CMC'04. 2004. p. 125-133.

CLARK, A.; FOX, C.; LAPPIN, S. Question and Answering. In: _____ **Handbook of computacional linguistics and natural language processing.** [S.l.]: Wiley-Blackwell, 2010. p. 630-654.

COUTINHO, F.; PORTO, F. **Query Processing in ROSA Data Model.** XIX Simpósio Brasileiro de Bancos de Dados. Brasília, DF, Brasil: Proceedings of XIX SBIE. 2004.

COVINGTON, M. **NLP for Prolog Programmers.** [S.l.]: Prentice-Hall, 1994.

COVINGTON, M.; NUTE, D.; VELLINO, A. **Prolog Programming in Depth.** [S.l.]: Prentice-Hall, 1997.

CURY, D.; MENEZES, C. S. D.; ARAGON, R. **Concept Maps: An Automated Support for Monitoring the Learning Process.** IEEE Transactions on Education. [S.l.]: [s.n.]. 2013.

DEMO, P. **Pesquisa e construção do conhecimento.** Rio de Janeiro: Tempo Brasileiro, 1996.

DUTRA, Í. M. et al. **Logical systems and natural logic: Concept Mapping to follow up the conceptualization processes.** Second International Conference on Concept Mapping. San José - Costa Rica: Proceedings of II CMC. 2006.

FALLAHI, A.; JAFARI, S. An Expert System for Detection of Breast Concer Using Data Preprocessing and Bayesian Network. **International Journal of Advanced Science and Technology**, v. 34, p. 65-70, Seteptember 2011.

FERRÁNDEZ, O. et al. Question Answering via Web Services Compositions. In: _____ **The QUALL-ME Framework: A Specifiable-Domain multilingual question answering architecture.** Berlin: Spring Berlin Heidelberg, 2011. p. 29-37.

GAMALHO, P.; GARCIA, M. **FreeLing e TreeTagger: um estudo comparativo no âmbito do Português**. ProLNat. Santiago de Compostela, Galiza, Spain. 2013.

GIL, A. C. **Como elaborar projetos de pesquisa**. São Paulo: Atlas, 1999.

GIL, A. C. Como Encaminhar uma Pesquisa. In: _____ **Como Elaborar Projetos de Pesquisa**. [S.l.]: Editora Atlas, 2002. p. 17-22.

GONG, Z.; CHAN, M. P. iQA: An Intelligent Question Answering System. **Digital Libraries: Implementing Strategies and Sharing Experiences**, v. 3815, p. 332-341, 2005. ISSN ISBN 978-3-540-32291-7.

GRAUDINA, V.; GRUNDSPENKIS, J. **Concept Map Generation from OWL Ontologies**. Third International Conference on Concept Mapping. Tallin, Estonia & Helsinki, Finland: Proceedings of CMC'08. 2008.

GREEN, B. F. et al. **BASEBALL: An automatic question answerer**. 19th Western Joint Computer Conference. New York, USA: Proceeding of WJCC. 1961. p. 219-224.

HALL, L. O.; KANDEL, A. **Languages for expert system building: a comparison**. ACM SIGSMALL/PC symposium on Small systems. New York, NY, USA: Proceedings of the 1986 ACM SIGSMALL. 1986. p. 157-160.

HANDRICK, F. **ROSAI: Uma Proposta de Representação do Modelo ROSA em Linguagem Lógica**. Instituto Militar de Engenharia (IME). [S.l.]. 2005. Dissertação de Mestrado.

HANDRICK, F. **ROSAI: Uma Proposta de Representação do Modelo ROSA em Linguagem Lógica**. Rio de Janeiro, RJ: IME - Instituto Militar de Engenharia, 2005. Dissertação de Mestrado.

HORN, A. On sentences which are true of direct unions of algebras. **Journal of Symbolic Logic**, v. 16, n. 1, p. 14-21, 1951.

JURCZYK, P.; AGICHTEN, E. **Discovering authorities in question answer communities by using link analysis**. 6th Conference on Information and Knowledge Management. New York, USA: Proceedings. 2007. p. 912-922.

KIYOTA, Y.; KUROHASHI, S.; KIDO, F. **"Dialog Novigator" a question answering system based on large text knowledge**. International Conference on Computational Linguistics. Pennsylvania, USA: Proceeding of COLING'02. 2002. p. 1-7.

KOWATA, J. H.; CURY, D.; BOERES, M. C. S. **Caracterização das Abordagens para Construção (Semi) Automática de Mapas Conceituais**. Simpósio Brasileiro de Informática na Educação. Florianópolis, SC - Brasil: Anais do XX SBIE. 2009.

KOWATA, J. H.; CURY, D.; BOERES, M. C. S. **Uma abordagem computacional para construção de Mapas Conceituais**. XXI Simpósio Brasileiro de Informática na Educação. João Pessoa, PB: Anais do XXI SBIE. 2010.

KOWATA, J. H.; CURY, D.; BOERES, M. C. S. **Construindo Mapas Conceituais a partir de Textos: Uma abordagem computacional aplicada à Língua Portuguesa do Brasil**. Simpósio Brasileiro de Informática na Educação. Aracaju, Sergipe - Brasil: Anais do XXI SBIE. 2011.

KREMER, R. **Concept Mapping: Informal to Formal**. **University of Calgary - Publications**, 1994. Disponível em: <<http://kremer.cpsc.ucalgary.ca/papers/ICCS94.html>>. Acesso em: 03 Julho 2014.

LAERHOVEN, K. **Comparison of the CLIPS and JESS expert system shells**. [S.l.]. 1999.

LAMAS, F. S. L. S. et al. **Um ambiente para comparação de Mapas Conceituais utilizando correspondência de grafos**. VIII Ciclo de Palestras Sobre Novas Tecnologias na Educação: Saber criar, saber usar. RS - Brasil: Anais do Evento. 2006. p. 1-10.

LAU, R. Y. et al. **Towards Fuzzy Domain Ontology Based Concept Map Generation for E-Learning**. Advances in Web Based Learning. Berlin: Springer Berlin/Heidelberg. 2008. p. 90-101. ICWL 2007.

LEE, C.-H.; LEE, G.-G.; YUNGHO, L. Application of automatically constructed concept map of learning to conceptual diagnosis of e-learning. **Expert Systems with Applications**, v. 36, n. 2, p. 1675-1684, March 2009. ISSN ISSN: 0957-4174.

LEI, G. et al. **Video reference**: a video question answering engine. Internation Conference on Advances in Multimedia Modeling. Berlin, GER: Sping-Verlag. 2010. p. 1-7.

LOURENÇO, A. B. **Análise de Mapas Conceituais elaborados por alunos da oitava série do ensino fundamental a partir de aulas pautadas na teoria da Aprendizagem Significativa**: a argila como tema de estudo. São Carlos: UFSCar, 2008. Dissertação de Mestrado.

MAHMOUD, M.; ALGADI, N.; ALI, A. **Expert System for Banking Credit Decision**. Computer Science and Information Technology. [S.l.]: Proceedings of ICCSIT'08. 2008.

MANSUR, A. U. et al. **Use of social networks and complexity for enhancement of learning academic in supervised internships**: An internalization by doing. XVIII Conferência Internacional sobre Informática na Educação. Porto Alegre: TISE'13. 2013.

MATTOS, D.; MOURA, A. M. D. C.; CAVALCANTI, M. C. **ROSA+**: Um Repositório de Objetos de Aprendizagem com Suporte a Inferência e Regras. XXI Simpósio Brasileiro de Banco de Dados. Florianópolis, SC, Brasil: Proceedings of XXI SBIE. 2006.

MAYBURY, M. T. **New directions in question answering**. New York, USA: MIT Press, 2004.

MERRITT, D. **Building Expert Systems in Prolog**. Lebanon, USA: Amzi! inc., 1989.

MILLER, G. A.; BECKWITH, R.; FELBAUM, C. Introduction to WordNet: An On-line Lexical Database. **Princeton University - WordNet - A lexical database for English**, 1993. Disponível em: <<http://wordnetcode.princeton.edu/5papers.pdf>>. Acesso em: 17 Julho 2014.

MOLDOVAN, D. et al. **LASSO: A Tool for Surfing the answer Net**. 8th Text Retrieval Conference. [S.l.]: Proceeding of TREC-8. 1999.

MONZ, C. **Information retrieval to question answering**. University of Amsterdam. Amsterdam, Nederland. 2003. PhD Thesis.

MOREIRA, M. A.; BUCHWEITZ, B. **Novas estratégias de ensino e aprendizagem: os mapas conceituais e o vê epistemológico**. Lisboa: Plátano Edições Técnicas. 1993.

NOVAK, J. D.; CAÑAS, A. J. The theory underlying concept maps and how to construct and use them. **IHMC**, 1998. Disponível em: <<http://cmap.ihmc.us/Publications/ResearchPapers/TheoryUnderlyingConceptMaps.pdf>>. Acesso em: 26 June 2014.

NOVAK, J. D.; CAÑAS, A. J. The Origins of the Concept Mapping Tool and The Continuing Evolution of the Tool. **Information Visualization Journal**, p. 175-184, 2006a.

NOVAK, J. D.; CAÑAS, A. J. **The Theory Underlying Concept Maps and How to Construct and Use Them**. Institute for Human and Machine Cognition. Florida. 2006b. Technical Report IHMC CmapTools 2006-01 Rev 01-2008.

NOVAK, J. D.; CAÑAS, A. J. The Theory Underlying Concept Maps and How to Construct and Use Them. **Institute for Human and Machine Cognition**, 2010. Disponível em: <<http://cmap.ihmc.us/publications/researchpapers/theorycmaps/theoryunderlyingconceptmaps.htm>>. Acesso em: 02 Julho 2014.

NOVAK, J. D.; GOWIN, B. D. **Learning How to Learn**. 1ed. ed. [S.l.]: Cambridge University Press, 1984.

NOVAK, J. D.; MUSONDA, D. A twelve-year longitudinal study of science concept learning. **American Educational Research Journal**, v. 28, n. 1, p. 117-153, 1991.

PERIN, W. D. A.; CURY, D.; MENEZES, C. S. D. **Construindo Mapas Conceituais Utilizando a Abordagem iMap**. XVII Version Congreso Internacional de Informática Educativa. Santiago - Chile: Anais do TISE'2012. 2012.

PERIN, W. D. A.; CURY, D.; MENEZES, C. S. D. **NLP - iMap: INTEGRATED SOLUTION BASED ON QUESTION-ANSWER MODEL IN NATURAL LANGUAGE FOR AN INFERENCE MECHANISM IN CONCEPT MAPS**. Sixth International Conference On Concept Mapping. Santos - SP, Brazil: Proceedings of the Sixth International Conference On Concept Mapping. 2014.

PIAGET, J.; GARCIA, R. **Hacia una logica de significaciones**. [S.l.]: [s.n.], 2000.

PORTO, F. et al. **ROSA: A Data Model and Query Language for e-Learning Objects**. I PGL Database Research Conference. Rio de Janeiro, RJ, Brazil: Proceeding of PGLDB'2003. 2003.

PRAGER, J.; BROWN, E.; CODEN, A. **Question-Answering by Predictive Annotation**. Annual International Conference on Research and Development in Information Retrieval. New York, USA: Proceedings of SIGIR'00. 2000. p. 184-191.

PRIETO, A. B. et al. **Webquest and concept maps to learn about waves**. Congreso Internacional de Informática Educativa. Santiago de Chile: TISE'12. 2012.

RIBEIRO, E. F. et al. **Um estudo sobre o incremento da Coesão e Coerência (Expressividade) em Mapas Conceituais**. Simpósio Brasileiro de Informática na Educação. Juiz de Fora, MG - Brasil: Anais do XVI SBIE. 2011. p. 233-242.

RIBEIRO, E. L. R. F. **O aumento da força dos conceitos em Mapas Conceituais**. Vitória, ES, Brasil: UFES - Universidade Federal do Espírito Santo, 2012. Dissertação de Mestrado.

RUSSEL, S.; NORVIG, P. **Artificial Intelligence - a modern approach**. [S.l.]: Prentice-Hall, 1995.

SANTOS, P. S. J. et al. **Um Ambiente para Acompanhamento da Aprendizagem baseado em Mapas Conceituais**. XVI Simpósio Brasileiro de Informática na Educação. Juiz de Fora, MG, Brasil: Anais do XVI SBIE. 2005.

SCHANK, R.; BIRNBAUM, L. **Aumentando a Inteligência**. KHALFA J. (ed.). São Paulo: UNESP. 1996.

SILVA, E. L. D.; MENEZES, E. M. Metodologia da pesquisa e elaboração de dissertação. **Laboratório de Ensino a Distância da UFSC**, 2001. Disponível em: <<http://projetos.inf.ufsc.br/arquivos/Metodologia%20da%20Pesquisa%203a%20educacao.pdf>>. Acesso em: 10 out 2014. Florianópolis.

SINTEK, M.; DECKER, S. **TRIPLE - A Query, Inference, and Transformation Language for the Semantic Web**. First International Semantic Web Conference. Londres - UK: Proceedings of ISWM '02. 2002. p. 364-378.

TOUTANOVA, K. et al. **Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network**. [S.l.]: Proceedings of HLT-NAACL 2003. 2003. p. 252-259.

TSENG, S.-S. et al. A new approach for constructing the concept map. **Computers & Education**, v. 49, n. 3, p. 691-707, November 2007. ISSN ISSN: 0360-1315.

VARGAS, M. Técnica, Tecnologia e Ciência. **Revista Educação & Tecnologia**, 2011.

VASSOLER, G. A.; PERIN, W. D. A.; CURY, D. **MergeMaps - A computational tool for merging of concept maps**. Sixth International Conference on Concep Mapping. Santos - SP, Brasil: Proceedings of the 6th CMC. 2014.

VERA, F.; FUENTES, R.; RIVERA, R. **Learning physics with video nalysis**. Conferência Internacional sobre Informática na Educação. Porto Alegre: TISE'13. 2013.

VILAÇA, M. L. C. **Pesquis e Ensino: Considerações e Reflexões**. 2nd. ed. [S.l.]: Revista Escrita, v. I, 2010.

VILLALON, J. J.; CALVO, R. A. **Concept Map Mining**: A definition and a framework for its evaluation. IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology. [S.l.]: Proceeding of ICWIIAT. 2008. p. 357-360.

WANG, L.; LIAO, L.; WANG, X. Question answering via semantic web service composition. **Advances in Control and Communication**, p. 29-37, 2012.

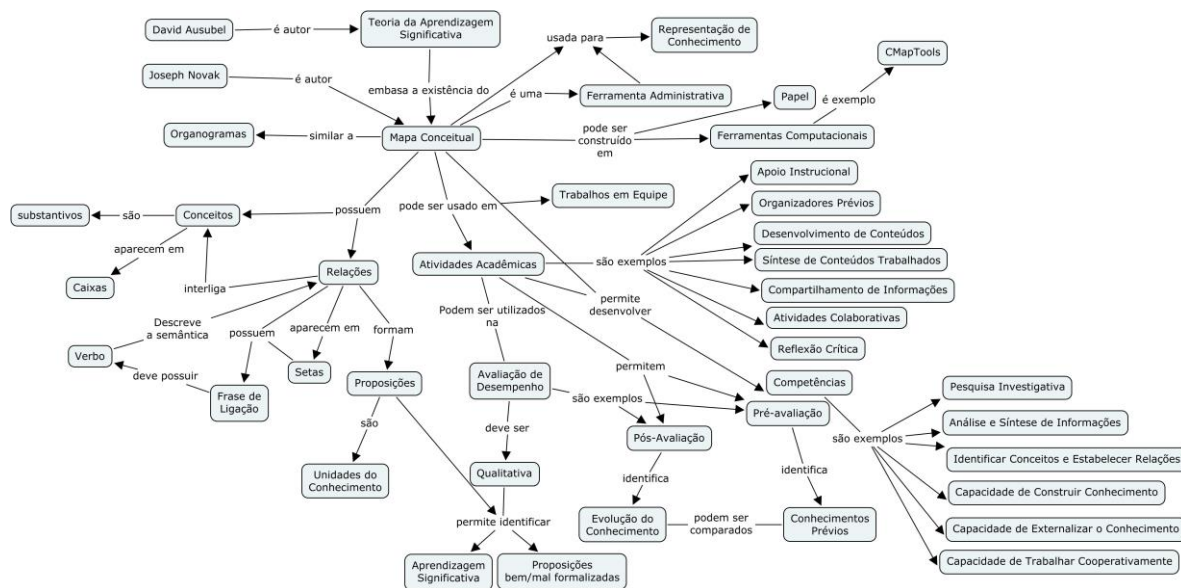
ZOUAQ, A.; GASEVIC, D.; HATALA, M. **Unresolved Issues in Ontology Learning**. Canadian Semantic Web Symposium. Vancouver, Canada: Proceedings of CSWS. 2011. Position Paper.

ANEXO I

QUESTIONÁRIO DE INVESTIGAÇÃO EXECUTADO

Preparação: Imagine que você está avaliando um mapa conceitual de sua área de conhecimento. Nesse momento você provavelmente mentalizará uma série de “perguntas” das quais deseja obter respostas através do mapa. Exemplo: “O aluno identificou a relação X entre os conceitos A e B?”, “O aluno foi capaz de identificar o conceito A?” ou “O que o aluno relacionou ao conceito A?”. Ao fazer isso, você provavelmente realizará uma série de navegações visuais no mapa para identificar as respostas a todas essas perguntas.

Imagine-se analisando um mapa como o seguinte:



1. Acredita que um sistema inteligente capaz de responder automaticamente suas perguntas poderia auxiliá-lo no processo de análise e avaliação de Mapas Conceituais?

() Sim.

() Não.

2. Que perguntas você faria a esse sistema a fim de extrair conhecimentos presentes nos mapas conceituais que julga importantes?

ANEXO II

TABELA DE POSSÍVEIS RESULTADOS DO PROCESSO DE TRADUÇÃO (PLN-iMap)

Perguntas em Linguagem Natural	Mesma pergunta em PL	Tipo de Saída
Quais conceitos estão conectados diretamente ao conceito A?	<code>primeiraOrdemDireta(A, S).</code>	Lista
Quais conceitos são destinados diretamente ao conceito A?	<code>primeiraOrdemInversa(A, S).</code>	Lista
O conceito A é entrada para o conceito B?	<code>primeiraOrdemDireta(A, B).</code>	true ou false
O conceito A é destino do conceito B?	<code>primeiraOrdemInversa(A, B).</code>	true ou false
Existe relação direta entre os conceitos A e B?	<code>existeRelacaoDireta(A, B).</code>	true ou false
Quais conceitos estão diretamente conectados ao conceito A?	<code>existeRelacaoDireta(A, S). / existeRelacaoDireta(S, A).</code>	Lista
Quais conceitos foram representados a partir do conceito A?	<code>todoDestino(A, [A], S).</code>	Lista
A partir do conceito A, é possível chegar ao conceito B?	<code>todoDestino(A, [A], B).</code>	true ou false
Quais conceitos dão origem ao conceito A?	<code>todaOrigem(S, [A], A).</code>	Lista
O conceito B é necessário para conhecer o conceito A?	<code>todaOrigem(B, [A], A).</code>	true ou false
Existe relação entre os conceitos A e B?	<code>existeRelacao(A, B).</code>	true ou false
Quais conceitos estão ligados ao conceito A?	<code>existeRelacao(A, S).</code>	Lista
Quais relações possuem o conceito A?	<code>quaisRelacoes(S, A).</code>	Lista
O conceito A está presente em alguma relação R?	<code>quaisRelacoes(R, A).</code>	true ou false
Quais conceitos estão conectados através de uma relação R).	<code>quaisRelacoes(R, S).</code>	Lista
Quais as relações existentes entre os conceitos A e B?	<code>quaisRelacoes(S, A, B).</code>	Lista
Quais os pares de conceitos conectados por uma relação R?	<code>quaisRelacoes(R, S1, S2).</code>	Lista
Os conceitos A e B estão conectados por uma relação R?	<code>quaisRelacoes(R, A, B).</code>	true ou false
Quantos conceitos e Quais estão conectados por uma Relação R?	<code>nConceitoRelacao(N, R)</code>	Número Inteiro e Lista
Quantos conceitos este mapa possui?	<code>nTotalConceito(C)</code>	Número Inteiro
Quantas relações este mapa possui?	<code>nTotalRelacao(R)</code>	Número Inteiro
Quantas e Quais relações o conceito A possui?	<code>nRelacaoCadaConceito(N, C)</code>	Número Inteiro e Lista

Quantas e Quais relações são entradas para o conceito A?	nRelacaoEntradaConceito(N, C)	Número Inteiro e Lista
Quantas relações existem entre os conceitos A e B?	nRelacaoConceitos(N, CA, CB)	Número Inteiro e Lista
Quantos níveis existem acima do conceito A ?	nivelConceito(N, C)	Número Inteiro
Partindo do conceito A, quantos conceitos foram identificados?	nivelConceitoFinal(N, C)	Número Inteiro
Quantos níveis existem a partir do conceito A?	nivelConceitoFinal(N, C)	Número Inteiro
Quantos níveis existem a partir de uma Relação R ?	nivelRelacaoFinal(N, R)	Número Inteiro
Quantos níveis existem acima de uma Relação R?	nivelRelacao(N, R)	Número Inteiro
Quantas relações possuem cada conceito?	nRelacaoCadaConceito(N, C)	Número Inteiro e Lista
Quais as proposições presentes no mapa?	listaPreposicao(LP)	Lista
Quantas são as proposições?	nPreposicao(N)	Número Inteiro
O Conceitos A existe no Mapa ?	buscaConceito(C)	true ou false
A Relação R existe no Mapa ?	buscaRelacao (R)	true ou false
Quais as Relações do Mapa ?	listaRelacao(LR)	Lista
Quais os Conceitos do Mapa ?	listaConcatenadaConceito(LC)	Lista