LEANDRO HALLE NAJM

ARQUITETURA ORIENTADA A SERVIÇOS PARA AQUISIÇÃO DE DADOS DE EXPERIMENTOS EM WEBLAB DE ABELHAS

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do título de Mestre em Engenharia.

SÃO PAULO

2011

LEANDRO HALLE NAJM

ARQUITETURA ORIENTADA A SERVIÇOS PARA AQUISIÇÃO DE DADOS DE EXPERIMENTOS EM WEBLAB DE ABELHAS

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do título de Mestre em Engenharia.

Área de Concentração:
Sistemas Digitais

Orientador:
Prof. Dr. Pedro Luiz Pizzigatti Corrêa

São Paulo
2011

Este exemplar foi revisado e alterado em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, 21 de julho de 2011.

Assinatura do autor _____

Assinatura do orientador _____

FICHA CATALOGRÁFICA

Najm, Leandro Halle
Arquitetura orientada a serviços para aquisição de dados de experimentos em weblab de abelhas / L.H. Najm. -- São Paulo, 2011.
116 p.

Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Computação e Sistemas Digitais.

1. Meio ambiente (Experimentos) 2. Metadados I. Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Computação e Sistemas Digitais II. t.

Dedico

Aos meus pais, Said Halle Meileide Najm e Margarida Maria Najm, que me educaram e formaram a base do que sou hoje.

Aos meus irmãos, Alexandre e Marcelo Halle Najm, que, mesmo nos poucos momentos de discórdias, me fizeram aprender e crescer com honra.

À minha amada namorada Bianca Lopes Moraes Elias, que me apoiou, incentivou e esteve comigo desde o início.

AGRADECIMENTOS

Agradeço primeiramente ao G.A.D.U. em toda sua essência, que me ofertou a oportunidade e me deu a disposição e sabedoria para superar os obstáculos que surgiram durante a elaboração deste trabalho.

Ao meu orientador, o professor Dr. Pedro Luiz Pizzigatti Corrêa, pelo apoio constante, pela confiança depositada e pelos indispensáveis esclarecimentos.

A todos os amigos que me ajudaram a entrar no mestrado, em especial Pablo Salvanha por me alertar e Rogério Leis, meu maior motivador e parceiro de estudos.

Ao amigo Fernando Correa, companheiro de viagens, estudos, conselhos e iluminação nos momentos em que me sentia “perdido”.

Aos meus colegas mestrandos Juan Carlos Benavente, Marcelo Queiroz Leite e Ayrton Vollet, pelas contribuições em projetos, experimentos, artigos e viagens.

Aos professores do LAA, Antonio Mauro Saraiva, Carlos Eduardo Cugnasca e André Riyuiti Hirakawa, que muito me aconselharam e auxiliaram na lapidação deste material.

A Antonietta Rosalina da Cunha Losso Pedroso de Mello, José Rosário Losso Netto, Marcelo Batuíra da Cunha Losso Pedroso de Mello, diretores do Jornal de Piracicaba, e Nobumitsu Chinen, que me permitiram conciliar emprego e estudos.

Aos colegas de trabalho, que, em meus momentos de ausência, trabalharam para que ela não fosse sentida.

Aos colegas Daniel Silva, Evandro Silvestre, Cristiano Menezes, Gabriela Ceschini, Nelson Gonzalez, Rafael Cunha, Rafael Domingues, Alex das Chagas, Larissa Carneiro, que participaram comigo de artigos e tarefas.

Aos meus amigos Leandro Galende, Maurício Beraldo, Carlos Verneque, Marcelo Rossy, sempre presentes.

A todos aqueles que, direta ou indiretamente, contribuíram para a realização deste trabalho.

"O conhecimento é o processo de acumular dados; a sabedoria reside na sua simplificação."

Martin H. Fischer

RESUMO

Experimentos ambientais são fundamentais para entender os efeitos das mudanças climáticas, como o decréscimo de polinizadores encontrados na natureza. Esses experimentos devem ser compartilhados com uma metodologia integrada. Desenvolver e aplicar ferramentas de tecnologia da informação em diferentes áreas de pesquisa é primordial para melhorar processos de controle e análise de dados, sem requisitar que pesquisadores de outras áreas tenham conhecimentos avançados em tecnologias da computação. Para isso, é importante a utilização de uma infraestrutura de hardware e software aberta e disponível aos pesquisadores, por meio de portais na *web* conhecidos como Weblabs, para aquisição e compartilhamento de dados obtidos através de sensores. Este trabalho apresenta uma arquitetura de sistemas de informação para a implementação de Weblabs a partir dos conceitos de SOA, para solucionar o problema de heterogeneidade e interoperabilidade de ambientes, visto que os dados são coletados por diferentes tecnologias de redes de sensores em suas bases de dados. Para tanto, fez-se necessária a modelagem de uma base de dados central capaz de armazenar dados oriundos de diferentes sistemas, acessíveis por meio do consumo de serviços disponibilizados pelo Weblab.

Palavras-chaves: Weblab, metadados, bioinformática, experimentos ambientais, redes de sensores.

ABSTRACT

Environmental experiments are fundamental to understand the effects of climate change, such as the decline of pollinators in nature. These experiments should be shared with an integrated methodology. Develop and apply tools of information technology in different areas of research is essential for improving process control and data analysis, without requiring that researchers from other fields have advanced knowledge in computing technologies. For this it is important to use an open infrastructure of hardware and software made available to researchers through web portals, known as Weblab for acquisition and sharing of data obtained by sensors. This paper presents a model of information systems architecture for the implementation of a Weblab based on the concepts of SOA, to solve the problem of heterogeneity and interoperability of environments, since the data is collected by different network technologies of sensors in its databases. It was necessary for the modeling of central database capable of storing data from different systems accessible through the consumption of the service provided by the Weblab.

Key-words: Weblab, metadata, biodiversity informatics, environmental experiments, sensors networks.

LISTA DE ILUSTRAÇÕES

Figura 1: Site do GBIF. 26
Figura 2: Site do BIS. 27
Figura 3: Site da IABIN. 28
Figura 4: Site do sistema SEEK. 29
Figura 5: Arquitetura em camadas do projeto SEEK. 31
Figura 7: Site do projeto ViNCES. 33
Figura 8: Diagrama Entidade-Relacionamento simplificado do BD do Webbee. 34
Figura 9: Ciclo de experimento do myExperiment. 38
Figura 10: Exemplo de arquivo baseado no EML. 44
Figura 11: Exemplo de arquivo baseado no Darwin Core. 45
Figura 12: Exemplo de arquivo baseado no Plinian Core. 46
Figura 13: Exemplo de arquivo baseado no Dublin Core. 47
Figura 14: Exemplo de arquivo baseado no FGDC BDP. 49
Figura 17: Nó sensor da rede LonWorks. 53
Figura 18: Ligação entre nós sensores da rede LonWorks. 53
Figura 19: Sete principais passos da abordagem SOA (ENDREI, 2004). 57
Figura 20: Diagrama de utilização de serviços. 60
Figura 21: Colaboração de serviços web (ENDREI, 2004). 62
Figura 22: Fragmento de um código WSDL. 64
Figura 23: Funcionamento do SOAP. 65
Figura 24: Modelo de Interação no UDDI. 67
Figura 25: Topologia de interligação para experimentos com Weblabs 69
Figura 26: Fases para realização de um experimento. 69
Figura 27: Estrutura para monitoramento interno das colônias de abelhas. 73
Figura 28: Temperatura a cada 10 minutos nos diferentes locais da colmeia 74
Figura 29: Umidade relativa (UR) nos diferentes locais da colmeia 75
Figura 30: Tela inicial do Weblab Bioabelha 77
Figura 31: Estrutura lógica da base de dados de experimentos do Weblab. 78
Figura 32: Estrutura lógica da base de dados do RemoteLon 79
Figura 33: Estrutura dos sensores da rede LonWorks para o experimento. 80
Figura 34: Estrutura da base de dados da RSSF. 81
Figura 35: Esquema lógico da base de dados central do Weblab. 85
Figura 36: Principais passos da abordagem SOA (adaptado de ENDREI, 2004). 86
Figura 37: Caracterização do domínio do sistema. 87
Figura 38: Casos de uso do sistema. 89
Figura 39: Arquitetura do Weblab de abelhas. 91
Figura 40: Mapeamento do padrão Dublin Core e BD Weblab. 93
Figura 41: Diagrama de sequência do Monitor Experiment. 95
Figura 42: Diagrama de sequência dos serviços do Monitor. 96
Figura 43: Relação AccessLevel. 106
Figura 44: Relação City. 106
Figura 45: Relação Country. 107
Figura 46: Relação Data. 107
Figura 47: Relação Experiment. 108
Figura 48: Relação ExperimentSensor. 109
Figura 49: Relação Institution. 109
Figura 50: Relação Keyword. 109

Figura 51: Relação Laboratory. *110*
Figura 52: Relação Language. *110*
Figura 53: Relação Researcher. *111*
Figura 54: Relação ResearcherExperiment. *111*
Figura 55: Relação Sensor. *112*
Figura 56: Relação Specie. *113*
Figura 57: Relação SpecieExperiment. *113*
Figura 58: Relação Specimen. *114*
Figura 59: Relação SpecimenGeo. *114*
Figura 60: Relação State. *115*
Figura 61: Relação User. *115*

LISTA DE TABELAS

Tabela 1: Palavras-chaves utilizadas na revisão de literatura 24
Tabela 2: Vantagens e desvantagens do uso de diferentes tipos de laboratórios. 36
Tabela 3: Temperatura e umidade relativa (UR) média nos locais da colmeia. 74
Tabela 4: Tratamento dos dados coletados da base de dados da rede Lonworks 82
Tabela 5: Casos de uso da caracterização do domínio 90

LISTA DE ABREVIATURAS E SIGLAS

ASCENT – *Adaptive Self-Configuring sEnsor Networks Topologies*

BD – Bancos de Dados

BIS – *Biodiversity Information Standards*

CORBA – *Common Object Request Broker Architecture*

DCMI – *Dublin Core Metadata Initiative*

DP – Desvio Padrão

DwC – *Darwin Core*

EEA – *European Environment Agency*

EMAN – *Ecological Monitoring and Assessment Network*

EML – *Ecological Metadata Language*

ERIN – *Environmental Resources Information Network*

EUA – Estados Unidos da América

FAO – *Food and Agriculture Organization*

FAPESP – Fundação de Amparo à Pesquisa do Estado de São Paulo

FGBC BDP – *Federal Geographic Data Committee Biological Data Profile*

GBIF – *Global Biodiversity Information Facility*

GPS – *Global Positioning System*

IABIN-PTN – *Inter American Biodiversity Information Network-Pollinators Thematic Network*

IBM – *International Business Machines*

INBio – Instituto Nacional de Biodiversidade

IR – *Infra-Red*

JEE – *Java Enterprise Edition*

KNB – *Knowledge Network for Biocomplexity*

LAA – Laboratório de Automação Agrícola

MERMAid – *Metadata Enterprise Resource Management Aid*

MIT – *Massachusetts Institute of Technology*

MULEs – *Mobile Ubiquitous LAN Extensions*

NBII – *National Biological Information Infrastructure*

OASIS – *Organization for the Advancement of Structured Information Standards*

OSI – *Open Systems Interconnection*

PEAS – *Probing Environment and Adaptive Sleeping*

PliC – *Plinian Core*

RSSF – Rede de Sensores Sem Fio

SEEK – *Science Environment for Ecological Knowledge*

SGBD – Sistema Gerenciadores de Bancos de Dados

SGML – *Standard Generalized Markup Language*

SMMS – *Spatial Metadata Management System*

SMS – *Short Message Service*

SOA – *Service Oriented Architecture*

SOAP – *Simple Object Access Protocol*

SOC – *Service Oriented Computing*

SPM – *Species Profile Model*

SQL – *Structured Query Language*

TCP/IP – *Transmission Control Protocol / Internet Protocol*

TIC – Tecnologia da Informação e Comunicação

TDWG – *Taxonomic Database Working Group*

TIC – Tecnologia da Informação e Comunicação

UDDI – *Universal Description, Discovery and Integration*

UR – Umidade Relativa

URI – Uniform Resource Identifiers

URL – Uniform Resource Locator

USP – Universidade de São Paulo

UT – *University of Tenneesse*

ViNCES – *Virtual Network Center of Ecosystem Services*

W3C – *World Wide Web Consortium*

WSDL – *Web Services Description Language*

WSN – *Wireless Sensor Networks*

XML – *eXtended Markup Language*

SUMÁRIO

1. INTRODUÇÃO ... 16

1.1 OBJETIVOS ... 19
1.1.1 Objetivo principal ... 19
1.1.2 Objetivos específicos ... 19
1.2 JUSTIFICATIVA ... 20
1.3 METODOLOGIA ... 21
1.4 ORGANIZAÇÃO DO TRABALHO ... 22

2. REVISÃO DE LITERATURA ... 23

2.1 SELEÇÃO DAS FONTES BIBLIOGRÁFICAS ... 24
2.2 INFORMÁTICA DA BIODIVERSIDADE ... 25
2.2.1 GBIF ... 25
2.2.2 BIS (TWDG) ... 26
2.2.3 IABIN ... 27
2.3 PROJETOS PARA BIODIVERSIDADE ... 29
2.3.1 Projeto SEEK ... 29
2.3.2 Projeto ViNCES ... 31
2.4 WEBLABS ... 35
2.5 PROCESSOS PARA REALIZAÇÃO DE EXPERIMENTOS EM WEBLABS ... 37
2.6 EXPERIMENTOS COM ABELHAS ... 39

3. ELEMENTOS DA ARQUITETURA PARA A AQUISIÇÃO DE DADOS DE EXPERIMENTOS AMBIENTAIS ... 42

3.1 METADADOS PARA APLICAÇÕES DE BIODIVERSIDADE ... 42
3.1.1 EML – Ecological Metadata Language ... 42
3.1.2 Darwin Core ... 44
3.1.3 Plinian Core ... 45
3.1.4 Dublin Core ... 46
3.1.5 FGDC BDP - Federal Geographic Data Committee - Biological Data Profile ... 48
3.2 REDES DE SENSORES ... 50
3.2.1 Redes de sensores sem fio ... 50
3.2.2 Lonworks ... 52
3.3 SOA - ARQUITETURA ORIENTADA A SERVIÇOS ... 54
3.3.1 SOC – *Service Oriented Computing* ... 59
3.3.2 Web Services ... 60
3.3.3 *XML – eXtensible Markup Language* ... 62
3.3.4 *WSDL – Web Service Description Language* ... 63
3.3.5 *SOAP – Simple Object Access Protocol* ... 64
3.3.6 *UDDI – Universal Description Discovery and Integration* ... 65

4. WEBLAB PARA EXPERIMENTOS COM ABELHAS ... 68

4.1 APRESENTAÇÃO ... 68
4.2 INFRAESTRUTURA ... 68
4.3 EXPERIMENTO ... 72
4.4 ARQUITETURA DE SISTEMAS PARA GERENCIAMENTO DE DADOS DO EXPERIMENTO ... 77
4.4.1 BIOABELHA - Sistemas Web ... 77
4.4.2 Sensores ... 79
4.4.2.1 LonWorks ... 79
4.4.2.2 Redes de sensores sem fio ... 80
4.4.3 Análise de requisitos ... 81

5. ARQUITETURA PARA COLETA AUTOMÁTICA DE DADOS ... 83

5.1 APRESENTAÇÃO ... 83
5.2 ESQUEMA LÓGICO DO BANCO DE DADOS DO WEBLAB ... 83
5.3 DEFINIÇÃO DE DOMÍNIO, COMPONENTES E SERVIÇOS ... 86
5.4 DEFINIÇÃO DOS METADADOS ... 92
5.5 DESCRIÇÃO DOS SERVIÇOS ... 94

6. CONCLUSÃO ***97***

6.1 CONSIDERAÇÕES FINAIS 97
6.2 TRABALHOS FUTUROS 98

6. REFERÊNCIAS BIBLIOGRÁFICAS ***99***

1. INTRODUÇÃO

A importância de serviços ambientais, tais como purificação do ar, estabilização do clima e polinização de culturas, está cada vez mais evidente. A necessidade de estudos nessas áreas vem aumentando, gerando experimentos complexos e que envolvem um grande volume de dados, como é o caso do laboratório de abelhas na *web* desenvolvido neste trabalho.

Nas últimas décadas, tem sido crescente o interesse da sociedade no estado atual e futuro do meio ambiente, podendo ser observadas, inclusive, adaptações no modo de vida das pessoas. Empresas devem informar sobre o impacto ambiental de seus produtos e atividades, e governos estão preocupados, mais do que nunca, em estabelecer políticas que controlem o consumo de recursos naturais.

Nesse contexto, foram desenvolvidas tecnologias de computação direcionadas ao meio ambiente, constatando-se uma evolução no que se refere ao processamento e armazenamento de informações, sendo ambos aplicados às mais diversas finalidades. Esse progresso amplia a necessidade de sistemas mais rápidos e eficientes que possibilitem a consulta de grandes quantidades de dados.

Tal fato não é diferente para os dados da biodiversidade, que tem como ponto mais relevante a informação em si, pois é por meio dessa informação que os pesquisadores conseguem agregar valor a seus trabalhos e evoluir em suas pesquisas (SALVANHA, 2009).

A aplicação dessas novas tecnologias, tanto na coleta como no processamento e no armazenamento da informação, torna-se necessária para facilitar o entendimento do efeito das mudanças climáticas no meio ambiente. Uma maneira de auxiliar os pesquisadores na obtenção de resultados mais rapidamente é o desenvolvimento de laboratórios acessíveis via Internet, denominados Weblabs.

A ideia principal é possibilitar a realização, o monitoramento e a análise de experimentos sem a necessidade da presença física do pesquisador no laboratório, o que permitiria, ainda, estender a sua utilização para mais pesquisadores, por meio de computadores conectados à Internet (GARCIA-ZUBIA, 2005).

A realização de experimentos em laboratórios é essencial em muitas atividades. Entretanto, os custos relativos a aquisição e montagem da estrutura

física podem representar um fator inibidor para a criação de laboratórios de pesquisa, nos quais as soluções não avançam nas mesmas proporções que os problemas. Outras desvantagens são a limitação de acesso ao laboratório, devido a restrições de horários de funcionamento, e a necessidade de maior espaço físico (FERREIRA, 2007).

A virtualização dos laboratórios, com interfaces *web* simples e objetivas, permitirá aos pesquisadores realizar experimentos utilizando serviços e equipamentos apropriados, sem a necessidade de manter uma estrutura própria. Nesse tipo de ambiente, será possível controlar os equipamentos, agendar ações, inserir dados e compartilhar resultados e conclusões, interagindo com outros interessados após o término do experimento.

O acesso a dados integrados, porém, pode ser bastante limitado, de modo que um *website* oferece informações em mapas, outro as apresenta em tabelas estáticas, outro em relatórios em texto, etc. Para obter uma completa visão do dado desejado, agrupado, o pesquisador precisa acessar cada um desses fornecedores de dados separadamente, selecionar e obter o dado em um único formato e, em seguida, reconciliar as diferenças nos dados e integrá-los (ALESHEIKH et al., 2004).

O estudo de Weblab aplicado a abelhas é justificado pelo fato de o Brasil ser o país com a maior diversidade de meliponineos, conhecidos popularmente como abelhas sem ferrão. Isso facilita a obtenção de parcerias entre laboratórios e pesquisadores dessas áreas específicas já com demanda por pesquisas e experimentos.

As abelhas são consideradas os principais agentes da polinização, a qual contribui em dois aspectos: para a manutenção da biodiversidade natural e a produção de alimentos e matérias-primas para os seres humanos, por meio de uma agricultura bem sucedida (HEARD, 1999; KLEIN et al., 2007).

Estima-se que das 115 principais culturas no mundo, 87 sejam dependentes de alguma forma da polinização por animais (KLEIN et al., 2007). Em uma escala global, a polinização na agricultura tem um valor estimado em U$ 200 bilhões por ano (SLAA et al., 2006). Na região tropical, as abelhas sem ferrão são visitantes de grande parte das espécies selvagens cultivadas e participam da polinização de aproximadamente 250 espécies vegetais, das mais de 1.000 espécies cultivadas (HEARD, 1999).

Atualmente, alguns autores observam um declínio dos polinizadores, em especial as abelhas, provavelmente resultado da destruição dos *habitats* naturais. O aquecimento global surge como uma ameaça às populações de polinizadores e, consequentemente, ao equilíbrio do ecossistema (DEUTSCH et al., 2008).

Deutsch et al. (2008) concluíram que o aquecimento global nos trópicos pode ser muito mais prejudicial aos insetos tropicais, que apresentam maior sensibilidade a variações de temperatura e estão vivendo constantemente próximos de suas temperaturas ótimas.

As abelhas sem ferrão realizam a termorregulação de seus ninhos, ou seja, controlam, dentro de certos limites, as condições de temperatura dentro do ninho (NOGUEIRA-NETO, 1997; ENGELS, 1995). Porém, esse processo ainda é pouco estudado.

Nesse cenário, faz-se necessária a análise da temperatura e umidade nas colônias de abelhas sem ferrão e da reação destas às suas variações no ambiente com maior precisão. Isso permitirá construir uma base de conhecimentos para que seja possível prever as reações desses insetos às mudanças globais.

A fim de evitar possíveis problemas de aquisição, manipulação, disseminação e análise de dados biológicos e ambientais, aplicam-se as tecnologias de computação, que podem ser ofertadas por meio da adoção em grande escala de métodos padronizados de armazenamento, transferência e recuperação do dado acumulado pela base de dados central do Weblab.

Uma significante melhoria no gerenciamento de dados pode ser provida por uma aplicação computacional especialmente modelada e desenvolvida para facilitar a entrada e transferência dos dados ambientais. Para atender essa demanda, o software deve ser de baixa complexidade para o usuário final, capaz de ser instalado em uma configuração básica de computadores pessoais e estar disponível em mais de um idioma (MEESE et al., 2002).

1.1 OBJETIVOS

1.1.1 Objetivo principal

Este trabalho propõe um modelo de arquitetura de sistemas orientados a serviços – SOA (*Service Oriented Architecture*) – para integração de diferentes bases de dados, por meio da aquisição e gerenciamento de dados de experimentos ambientais obtidos através de redes de sensores, destinadas à aplicação de Weblab de abelhas.

1.1.2 Objetivos específicos

Os objetivos específicos desta pesquisa são:

- Apresentar os procedimentos atuais realizados pelos pesquisadores da área biológica no estudo de abelhas e os sistemas em utilização;
- Apresentar os serviços que se conectam e extraem informações de diferentes bases de dados geradas por sensores já modeladas e configuradas;
- Modelar uma base de dados para o Weblab, definindo relações e atributos com nomenclaturas e tipos de dados para inserção de outras bases de experimentos ambientais envolvendo abelhas;
- Propor a utilização de um padrão de metadados para compartilhamento de dados obtidos de diferentes sensores para que possam ser centralizados numa base de dados de experimentos;

- Definir um conjunto de serviços que permita selecionar, obter e transformar os dados de experimentos ambientais; e

1.2 JUSTIFICATIVA

Este trabalho visa contribuir com um ambiente para experimentos ambientais envolvendo abelhas no apoio à aquisição de dados coletados por sensores, para que pesquisadores da área não precisem dominar tecnologias de redes de sensores, redes de computadores, banco de dados e demais ferramentas computacionais. No entanto, esse conhecimento ainda se faz necessário devido à falta de uma mesma base de dados para diferentes redes de sensores.

Atualmente, as informações coletadas são armazenadas automaticamente em bases de dados modeladas para receber os dados específicos da rede dos sensores. Com isso, ao extrair e consultar os dados para seus experimentos, pesquisadores necessitam de conhecimentos avançados em consultas a bancos de dados (*query*), rotinas de backup, noções de conversão de dados e de binários para decimais, por exemplo, e, como esse não é seu foco, o uso de tecnologias mais avançadas de armazenamento, análise e gerenciamento de dados pode se tornar inviável.

Além de todo o trabalho de consulta manual, fica também a cargo desses pesquisadores a manutenção de redes de sensores e computadores a fim de se manter todo o processo em funcionamento. Nesse caso, corre-se o risco de se perder um experimento em andamento em função de um problema computacional relativamente simples de ser resolvido para um especialista da área.

Um dos componentes críticos em desenvolver Weblabs utilizando sensores é criar uma infraestrutura de informações, um gateway que conecte as estruturas heterogêneas de redes de sensores com ou sem fio (ALESHEIKH et al., 2004), o que ficaria também a cargo desses pesquisadores de abelhas.

A proposta deste trabalho é propor uma arquitetura orientada a serviços para integrar as bases de dados a serem utilizadas, em suas diferentes plataformas, centralizando o acesso dos pesquisadores em uma base única, a partir do Weblab.

1.3 METODOLOGIA

A metodologia adotada neste trabalho iniciou-se pelo levantamento bibliográfico, apresentando as tecnologias envolvidas atualmente pelos pesquisadores da área biológica e o desenvolvimento do Weblab. Experimentos ambientais com abelhas foram realizados pelo mestrando Ayrton Vollet Neto, da Faculdade de Filosofia, Ciências e Letras da USP (Universidade de São Paulo) em Ribeirão Preto, sob coordenação da professora Dra. Vera Lucia Imperatriz Fonseca.

Foram definidos, então, os componentes de SOA baseados no modelo proposto pela IBM (*International Business Machines*), *Patterns: Service-Oriented Architecture and Web Services* (ENDREI et al., 2004) para a inserção dos dados das diferentes bases de dados no Weblab.

Estudou-se a base de dados do Weblab Bioabelha (BIOABELHA, 2004) para entender seu funcionamento e propor melhorias para integração e acesso aos dados por ferramentas de análises de dados.

Foram pesquisados padrões de metadados ambientais para a definição de como a base de dados central deveria ser modelada para ofertar uma maior interoperabilidade aos diferentes metadados ambientais disponíveis.

Observou-se, também, como a coleta das informações e procedimentos é executada pelos pesquisadores, por meio dos experimentos realizados, a fim de se conhecer todo o processo, desde a montagem da estrutura física do experimento até a análise dos dados coletados. Com isso, foi possível identificar em que processos há possibilidade de informatização e automação de rotinas.

1.4 ORGANIZAÇÃO DO TRABALHO

O presente trabalho está organizado em cinco capítulos a contar com esta introdução, na qual são descritos os objetivos, as justificativas e a metodologia.

No Capítulo 2 são apresentados os conceitos e a fundamentação teórica das tecnologias envolvidas, como o Weblab e os processos para realizar experimentos ambientais com abelhas.

No Capítulo 3 apresentam-se os métodos e tecnologias computacionais a serem utilizados: Weblabs, redes de sensores, SOA, padrões de metadados e suas aplicações.

No Capítulo 4 detalha-se a estrutura atual do laboratório de experimentos e o experimento em si, realizado para validar as revisões bibliográficas e tecnologias utilizadas atualmente.

Na sequência (Capítulo 5), são mostradas as principais etapas da arquitetura proposta para aquisição automática dos dados.

E, por fim, no Capítulo 6, são feitas as considerações finais do trabalho.

2. REVISÃO DE LITERATURA

A primeira etapa da revisão de literatura consistiu no estudo das tecnologias utilizadas para aquisição e armazenamento dos dados que poderão ser acessados no Weblab.

Os sensores e redes de sensores, por exemplo, se utilizam de diferentes conceitos, como comunicação entre si, transmissão dos dados coletados e armazenamento dos dados. Posteriormente, estudaram-se os serviços já existentes no Weblab, como softwares de comunicação com os sensores, que iniciam e param os serviços de coleta, além de compreender o armazenamento desses dados.

A próxima etapa envolveu a revisão bibliográfica do conceito dos sistemas gerenciadores de bancos de dados (SGBD), nos quais os dados coletados são armazenados a fim de se conhecer o funcionamento e as peculiaridades de cada um para, então, se propor o modelo a ser utilizado como padrão pelo Weblab.

Definido o SGBD, o próximo passo da pesquisa foi buscar padrões de metadados que contribuam para um armazenamento de qualidade dos dados, a fim de se criar uma base de dados com informações padronizadas e de fácil acesso a buscas e consultas, com disponibilidade de utilização por pesquisadores de diferentes países e instituições. No caso deste trabalho, foram pesquisados padrões de metadados ambientais, visto que o foco são as abelhas sem ferrão.

Depois de compreendidos os conceitos e literatura disponíveis e estabelecidos os padrões, foram definidas as tecnologias a serem utilizadas no desenvolvimento e codificação dos sistemas para modelar o software para os processos de aquisição, conversão e inserção dos dados adquiridos pelos sensores. Estas informações padronizadas no banco de dados central, por sua vez, serão utilizadas para consulta, possibilitando aos pesquisadores o monitoramento e a análise de experimentos pelo Weblab.

2.1 SELEÇÃO DAS FONTES BIBLIOGRÁFICAS

Foram realizadas pesquisas em livros, revistas, artigos, entrevistas e sites, em português e inglês. Os critérios utilizados para seleção de referências foram a relevância nas áreas de interesse, a possibilidade de acesso aos artigos completos, além das indicações de professores e profissionais envolvidos no contexto do trabalho.

O processo de revisão sistemática se apoiou nas recomendações apresentadas por Biolchini et al. (2005). Abaixo, na Tabela 1, é possível observar as palavras-chaves buscadas para cada uma das etapas anteriormente citadas:

Tabela 1: Palavras-chaves utilizadas na revisão de literatura

Pesquisas	
Etapa	**Palavras-Chaves**
Aquisição dos dados	“sensors”, “Wireless Sensor Networs”, “WSN”, “LonWorks”, “Echelon”, “data acquisition”, “aquisição de dados”, “redes de sensores”, “redes de sensores sem fio”, “sensores”
Softwares residentes	“Crossbow”, “RemoteLon”, “Moteview”
Armazenamentos dos dados	"SGBD”, “DBMS”, “Postgresql”, “postgress”, “MySQL”, “comparação bancos de dados”, “diferenças SGBD”
Padrão de Metadados	"EML”, “Darwin Core”, “metadata”, “padrões de metadados”, “metadata patters”, “Dublin Core”, “FGDC”, “Plinian Core”, “EML”
Desenvolvimento de sistemas	"extração de dados”, “transformação de dados”, “Java”, “hibernate”, “extração de diferentes bases de dados”, ”encapsulamento de dados”, “web-services”, “SOA”, “OASIS”
Outros	“experimentos”, “laboratórios virtuais”, “weblabs”, “metadados sensores”, “monitoramento experimentos”, “monitoramento”, “experimentos abelhas”

Outras pesquisas foram feitas com termos mais complexos, formando cadeias de palavras a fim de filtrar resultados ou, até, utilizar termos mais simples para ampliar as possibilidades nos resultados. Dada a relevância dos dados obtidos, novos filtros foram criados abrangendo melhor o assunto que se era desejado.

Manuais e textos técnicos foram encontrados para os sensores sem fio considerados neste trabalho. As pesquisas por metadados e padrões de metadados foram realizadas em diversos meios, principalmente em uma disciplina cursada a distância, intitulada Informática Ambiental (*Environmental Informatics*), em parceria com a *University of Tennessee*, nos Estados Unidos.

2.2 INFORMÁTICA DA BIODIVERSIDADE

A diversidade biológica abrange todos os níveis de variação natural, do molecular e genético ao das espécies, com diferentes formas de interação, seja por meio da apreciação das formas comuns, estranhas e belas da vida, ou através do sofrimento causado pelos efeitos de pragas, parasitas e doenças (HUSTON, 1994).

A atenção dada ao tema levou à adição de uma nova palavra na língua inglesa, "*biodiversity*", uma contração de "*biological diversity*", (WILSON & PETER, 1988), criando-se, então, o termo biodiversidade.

Refere-se, portanto, à variedade de vida no planeta Terra, incluindo a variedade genética dentro das populações e espécies, da flora, fauna, de fungos macroscópicos e de microrganismos, as funções ecológicas desempenhadas pelos organismos nos ecossistemas, além da variedade de comunidades, *habitat* e ecossistemas formados pelos organismos.

Assim como em todas as áreas de estudo e aplicação de novas tecnologias, a complexidade e a abrangência da Ciência da Biodiversidade exigem o suporte da Tecnologia da Informação. O surgimento da Informática da Biodiversidade é uma indicação do papel decisivo da tecnologia no apoio às ações de conservação e uso sustentável da biodiversidade (SARAIVA et al., 2003).

Esta seção busca apresentar algumas iniciativas de projetos cujo objetivo é facilitar o acesso e a utilização de informações, neste caso, pesquisas e trabalhos relacionados à biodiversidade.

2.2.1 GBIF

O GBIF (*Global Biodiversity Information Facility* ou Facilitador de Informações sobre Biodiversidade Global) (GBIF, 2009) é um provedor de dados que permite o acesso livre e gratuito a informações sobre biodiversidade. É uma iniciativa internacional, financiada por vários governos, focada em produzir e disponibilizar dados sobre biodiversidade para todos que trabalham com pesquisas cientificas,

conservação e desenvolvimento sustentável. Na Figura 1, é possível observar o site do projeto GBIF.

A missão do GBIF consiste em permitir que os dados mundiais sobre biodiversidade sejam livres e universalmente disponibilizados pela Internet. Tem como objetivo proporcionar uma infraestrutura computacional global essencial para pesquisas em biodiversidade e aplicações em todo o mundo.

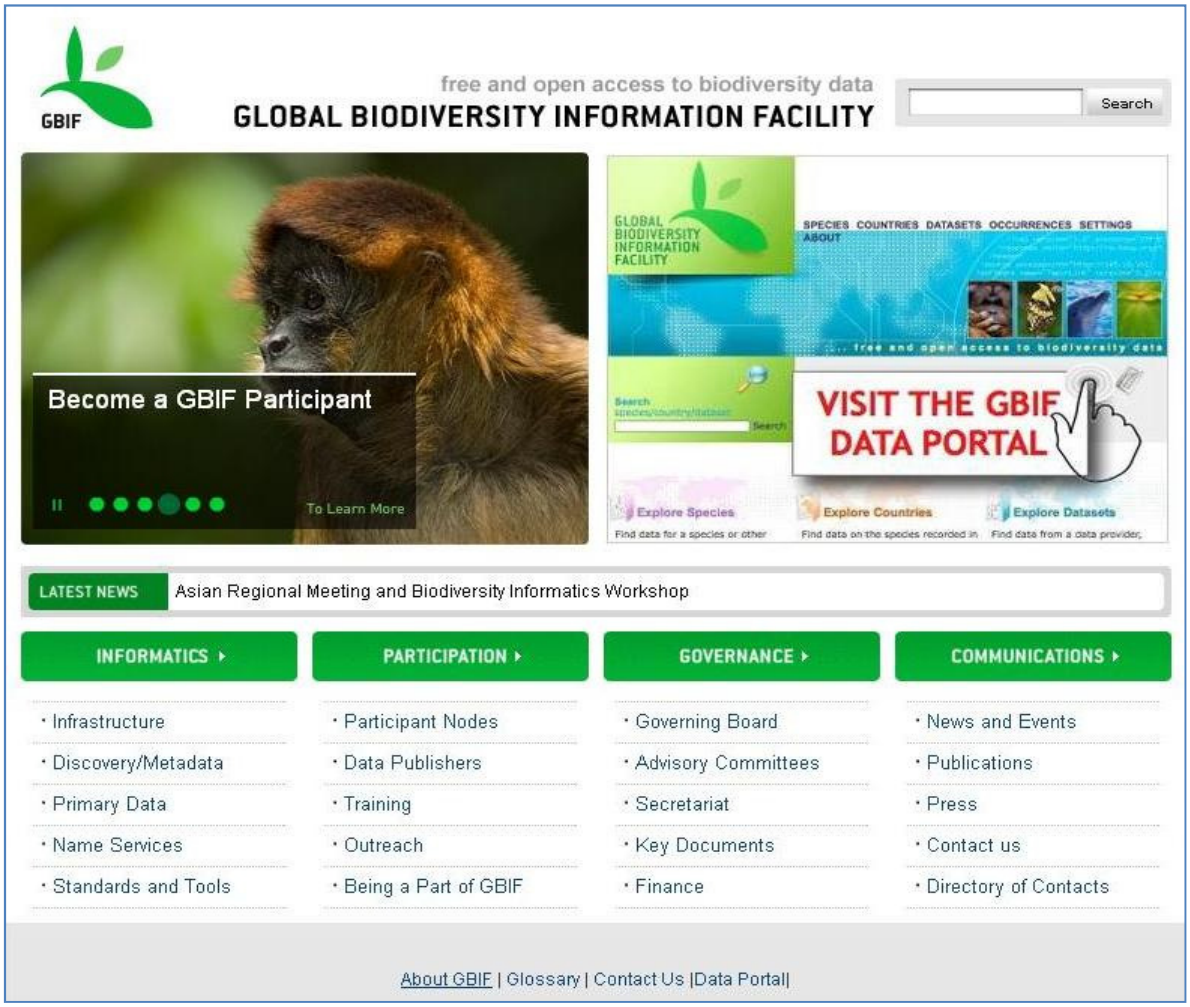

Figura 1: Site do GBIF.

2.2.2 BIS (TWDG)

Antigamente conhecido como TDWG *(Taxonomic Database Working Group* ou Grupo de Trabalho sobre Banco de Dados Taxonômicos), o *Biodiversity Information Standards* ou Padrões de Informação de Biodiversidade (BIS, 2005) são uma

organização sem fins lucrativos e também uma associação científica e educacional afiliada à União Internacional de Ciências Biológicas.

O TDWG foi criado, em sua maioria, por botânicos, para estabelecer uma colaboração internacional entre projetos de banco de dados biológicos. O TDWG promoveu a mais ampla e eficiente divulgação de informações sobre o patrimônio mundial de organismos biológicos. Com a difusão da Internet, o BIS trabalha atualmente no desenvolvimento, adaptação e adoção de padrões que habilitem a troca de informações sobre dados biológicos ou de biodiversidade entre organizações. Na Figura 2, visualiza-se uma tela de acesso ao site do BIS.

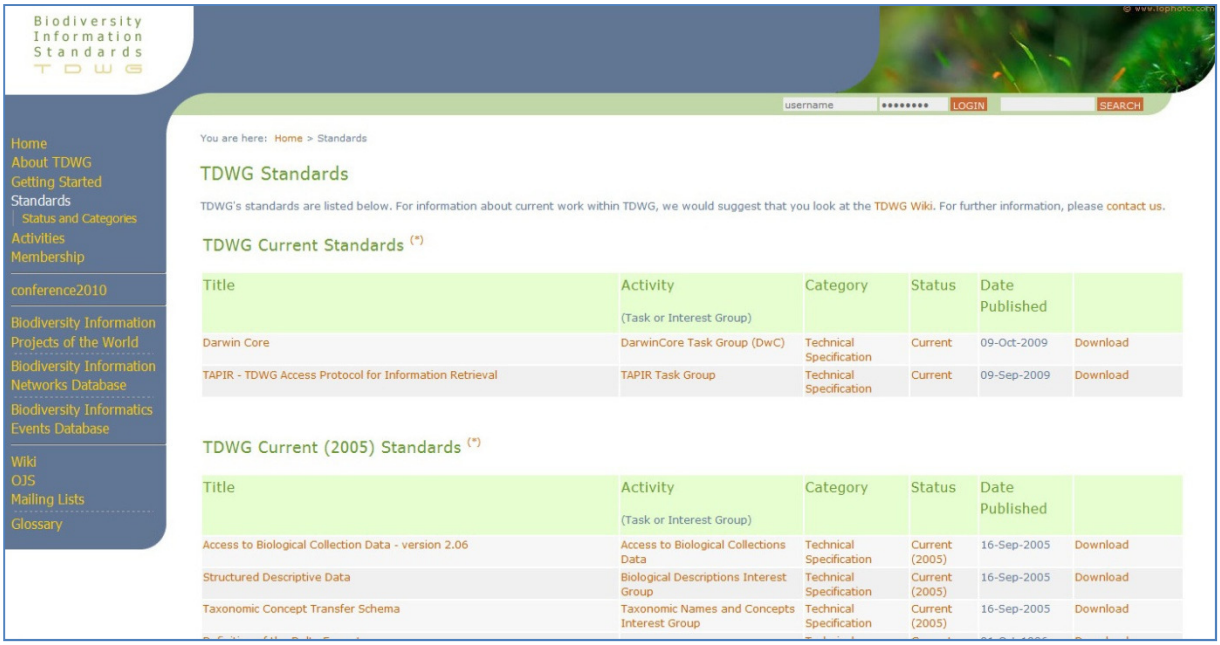

Figura 2: Site do BIS.

2.2.3 IABIN

A IABIN (*Inter American Biodiversity Information Network* ou Rede de Informação Interamericana de Biodiversidade) é um fórum especializado para promover a colaboração técnica e coordenação entre os países das Américas, na coleta, compartilhamento e utilização de informações sobre biodiversidade relevantes para a tomada de decisões sobre a conservação e gestão dos recursos naturais, e também a educação visando ao desenvolvimento sustentável da região (IABIN, 2005). Na Figura 3, pode-se visualizar o portal de acesso ao projeto.

Seus principais objetivos são:

- Construir uma infraestrutura para intercâmbio de informações de biodiversidade;
- Reforçar a capacidade técnica para troca de informações sobre biodiversidade e o conhecimento sobrepondo as fronteiras políticas, linguísticas e institucionais;
- Fornecer acesso a informações úteis aos tomadores de decisão (*decision makers*) para aperfeiçoar a conservação da biodiversidade;
- Reforçar a capacidade de armazenamento, utilização e distribuição de dados científicos, mantendo também constante a atualização dessas informações; e
- Produzir ou adaptar dados aos tomadores de decisão (ferramentas para tomada de decisão), para que possam formular políticas eficientes de gestão ambiental e promover o desenvolvimento sustentável na região.

Figura 3: Site da IABIN.

2.3 PROJETOS PARA BIODIVERSIDADE

Neste item são apresentados dois projetos empregando o uso de tecnologias computacionais no âmbito de pesquisas nas áreas ecológicas e de biodiversidade, um voltado à criação de uma arquitetura utilizada para pesquisas em ecologia (SEEK) e outro voltado ao desenvolvimento de uma rede de laboratórios virtuais com foco em pesquisas ao ecossistema (ViNCES).

2.3.1 Projeto SEEK

O SEEK (*Science Environment for Ecological Knowledge* ou Ambiente Científico para Conhecimento Ecológico) é um sistema projetado não apenas para permitir a aquisição e o armazenamento de dados, mas também para prover a integração, transformação, análise e sintetização de dados de biodiversidade que, até o momento de seu desenvolvimento, eram considerados intratáveis (SEEK, 2009). Na Figura 4, observa-se o portal SEEK.

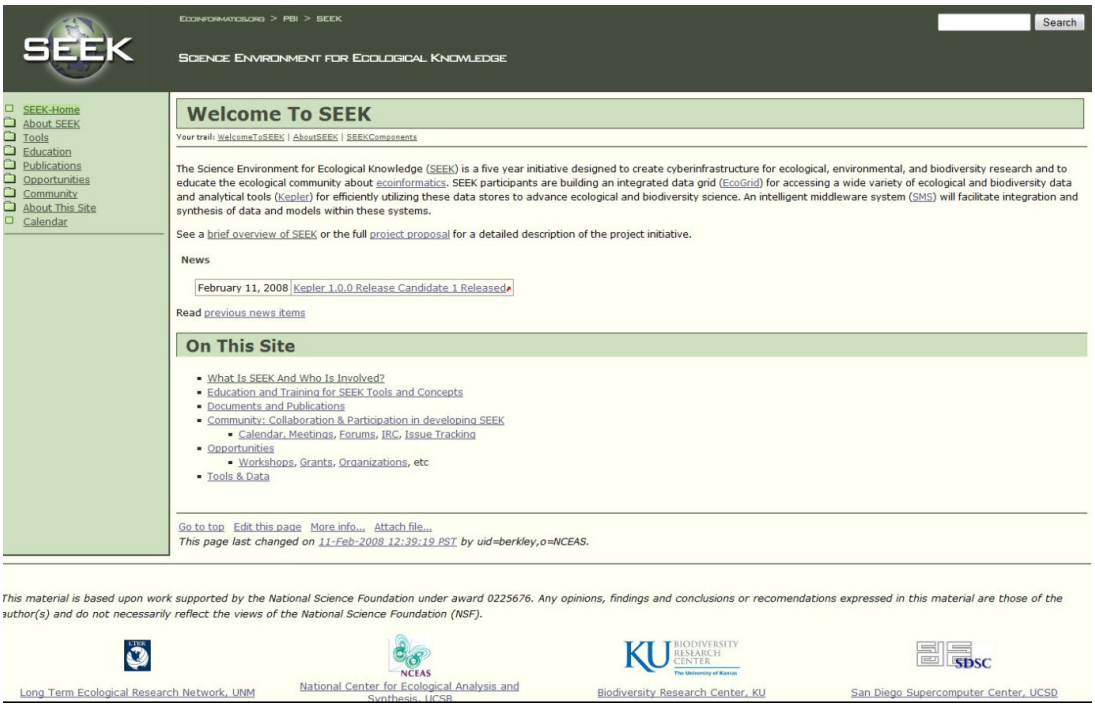

Figura 4: Site do sistema SEEK.

A meta do projeto SEEK é construir uma infraestrutura virtual que proporcione melhorias fundamentais de forma que os pesquisadores possam:

- Ter acesso global a dados e informações ecológicas;
- Localizar e utilizar serviços computacionais distribuídos; e
- Exercitar novos métodos poderosos de captura, reprodução e análise de dados, estendendo capacidades de pesquisa e análises ecológicas e em biodiversidade.

A infraestrutura SEEK abrange três sistemas, de acordo com a Figura 5:

- *Ecogrid* – considerada como a "próxima geração" (SEEK, 2009) de arquitetura Internet para armazenamento de dados, compartilhamento, acesso e análises. Ele combina as características de um *grid computing* de dados para armazenamento de dados ecológicos e outro *grid computing* para serviços de análise e modelagem. O *Ecogrid* permite aos cientistas terem acesso a recursos de biodiversidade ecológicos e ambientais e dados analíticos (tais como: dados, metadados, *workflows* analíticos e processadores) na rede, em diferentes locais e em diferentes organizações por meio da Internet. Camada capaz de permitir a integração de dados provenientes de fontes heterogêneas de informações que serão utilizadas para análises nas camadas superiores.
- *Semantic Mediation System* ou Sistema de Mediação Semântica (SEEK, 2004) – um avançado sistema capaz de determinar se dados relevantes e componentes analíticos podem ser automaticamente transformados para uso com o *workflow* selecionado. Tem a finalidade de permitir uma visão integrada sobre conjuntos de dados ecológicos e de biodiversidade a partir de fontes atualmente incompatíveis. Essa camada faz a mediação entre diferenças conceituais das diversas fontes de dados disponibilizadas pelo *Ecogrid*. Assim, dados provenientes de fontes diversas podem ser disponibilizados e agrupados, facilitando a construção de modelos e posterior análise. Tecnologias de serviços *web* são utilizadas nessa camada para permitir tal acesso uniforme (FERREIRA, 2007).
- *Analysis and Modeling System* ou Sistema de Análise e Modelagem – um ambiente visual automatizado (SEEK, 2009), onde ecologistas podem criar modificar e incorporar análises que compõem os fluxos de trabalho e novos

modelos. Nessa camada, a ferramenta utilizada como apoio para o pesquisador é o software Kepler (ALTINTAS, 2004), uma interface visual para desenvolvimento de análises baseadas em dados ecológicos.

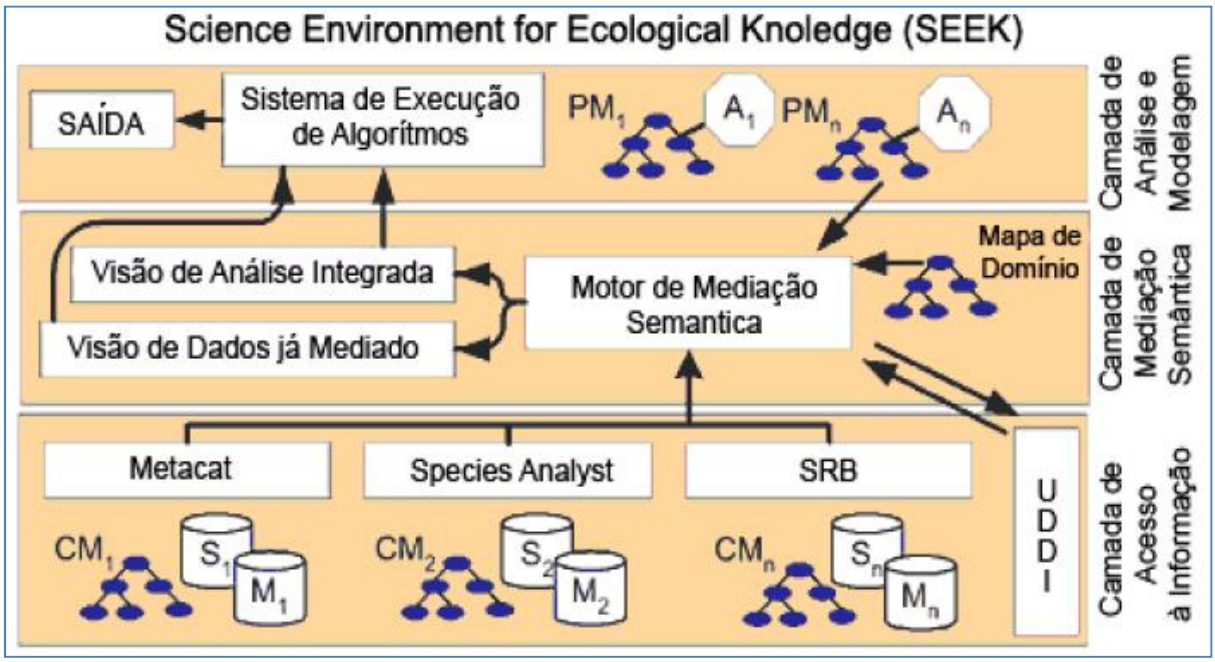

Figura 5: Arquitetura em camadas do projeto SEEK.

2.3.2 Projeto ViNCES

O projeto *Virtual Network Center of Ecosystem Services* ou Centro de Rede Virtual para Serviços de Ecossistema (VINCES, 2009) é um consórcio de laboratórios virtuais (Weblabs) focado nas pesquisas em serviços de ecossistema, tais como fotossíntese e polinização. Seu objetivo é promover um melhor entendimento sobre os serviços de ecossistema, sua importância, magnitude e impacto, a partir de diferentes pontos de vista – científico, econômico e social –, os quais poderão conduzir a uma utilização sustentável dos recursos naturais envolvidos. Coordenado pelo Laboratório de Automação Agrícola da USP, o projeto tem, entre seus parceiros, o Laboratório de Abelhas do Instituto de Biociências da USP, a Seção de Fisiologia e Bioquímica de Plantas do Instituto de Botânica, a Secretaria de Estado do Meio Ambiente, o grupo SFBP-IB & Universidade da

Califórnia, *San Diego*. Na Figura 6, é possível visualizar a arquitetura de referência proposta para o projeto ViNCES (FERREIRA, 2007).

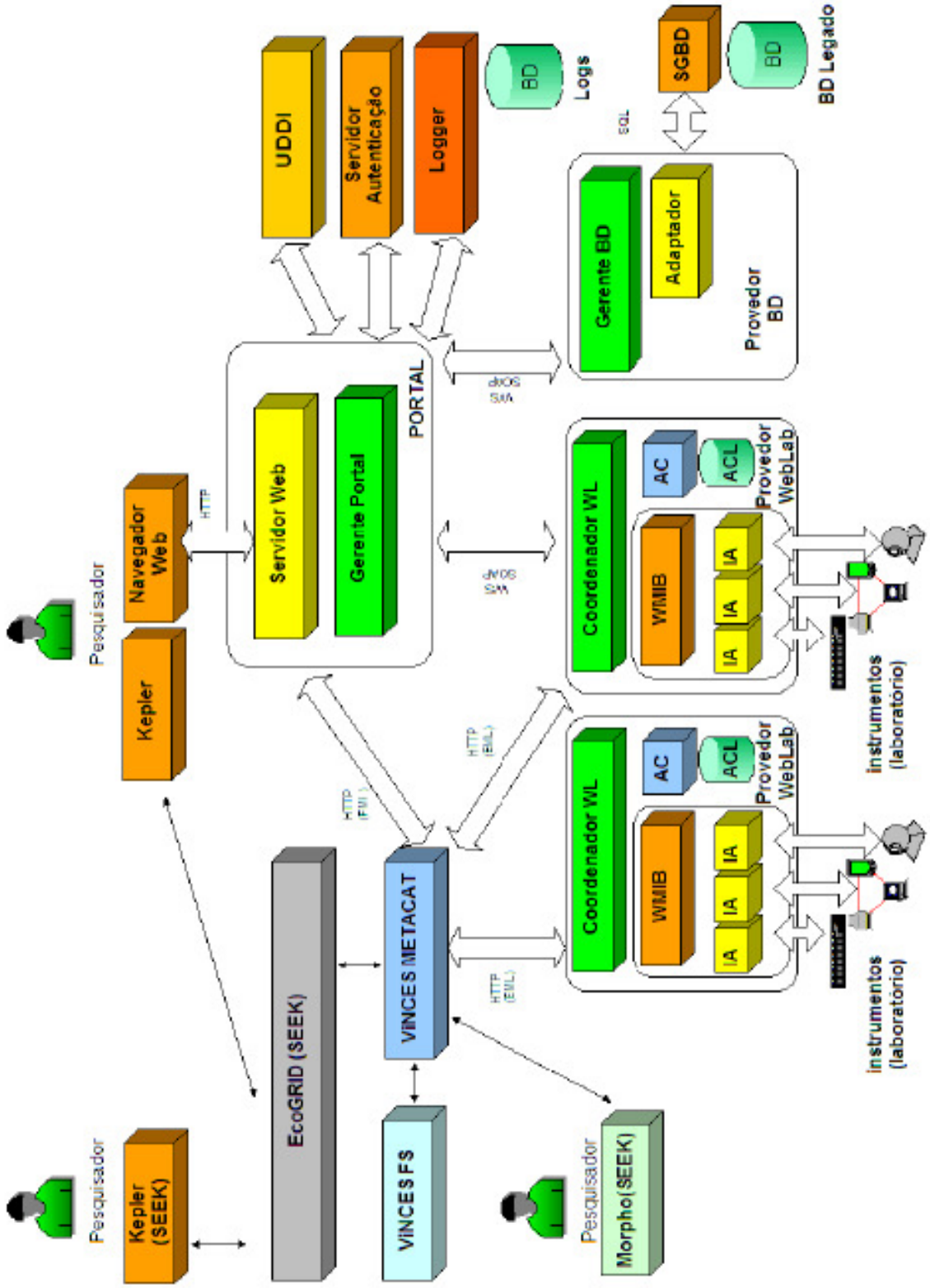

Figura 6: Arquitetura de *Weblab* proposta para o projeto ViNCES.

O projeto ViNCES é um exemplo do trabalho cooperativo de sucesso feito por equipes interdisciplinares, além de ser referência em pesquisa sobre a aplicação de sistemas e instrumentação baseados na Internet para o controle de experimentos científicos (SARAIVA et al., 2003). Na Figura 7, pode-se observar a tela inicial do portal.

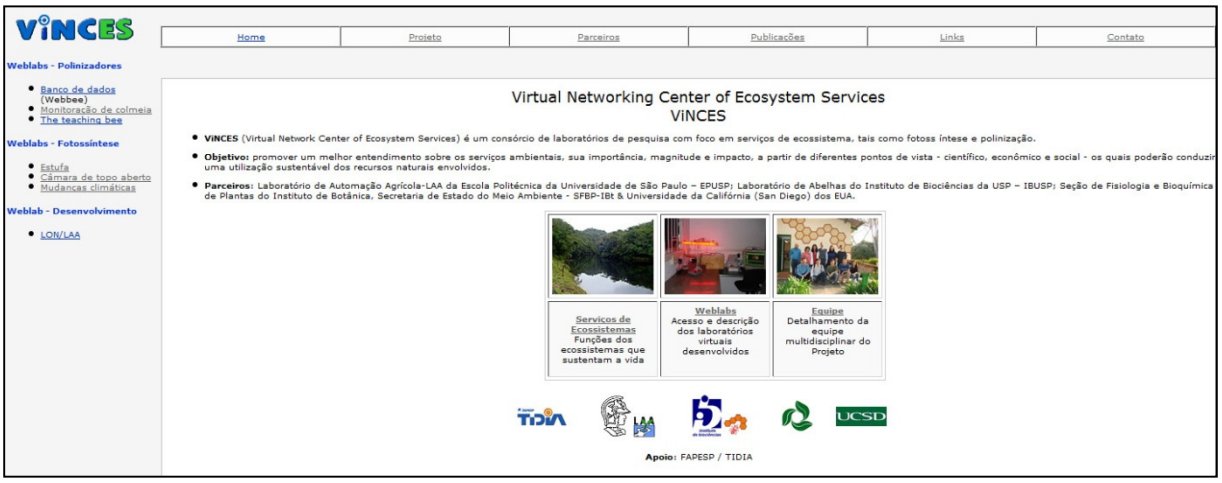

Figura 7: Site do projeto ViNCES.

No caso das abelhas, o Webbee (WEBBEE, 2001) se destaca entre as redes de informação sobre biodiversidade. É um sistema de informação e de instrumentação, ambos baseados na *web*. As colônias do Laboratório de Abelhas do Instituto de Biociências da USP são monitoradas em relação à temperatura do ar interno e à umidade. A atividade de voo é acompanhada por meio de sensores nas entradas das colônias e as condições externas estão ligadas a uma estação meteorológica. Todos os dados obtidos estão disponíveis em tempo real na Internet.

A Figura 8 mostra o diagrama Entidade-Relacionamento com o esquema da base de dados do sistema, extraído de Saraiva et al. (2003).

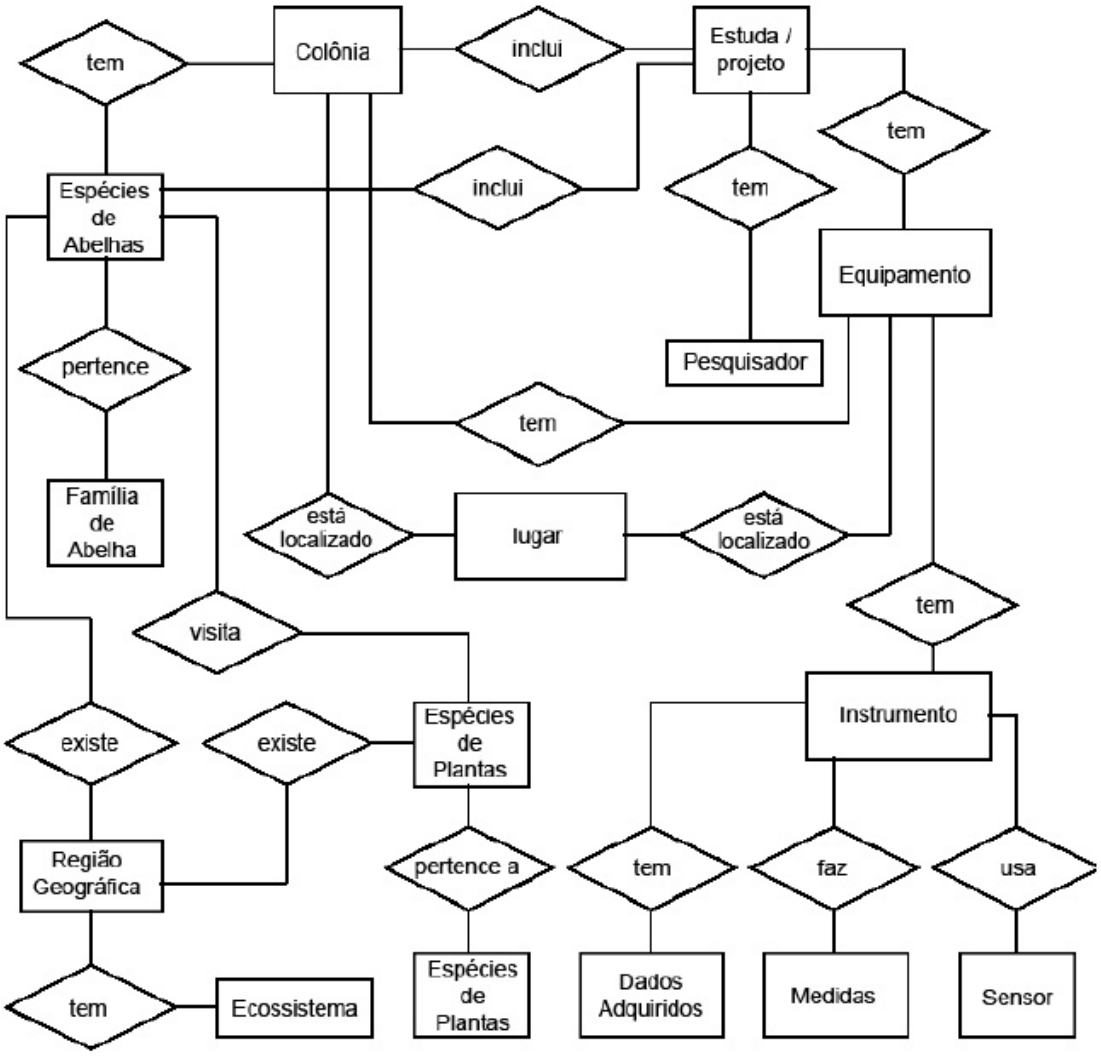

Figura 8: Diagrama Entidade-Relacionamento simplificado do BD do Webbee.

Uma rede de instrumentação inteligente foi desenvolvida para substituir as placas de instrumentação virtuais (baseadas em PC), permitindo, dessa maneira, o controle das condições dentro das colônias. As informações coletadas são armazenadas em um banco de dados, que também dispõe de dados multimídia para cada espécie. O objetivo é fornecer um ponto central de acesso aos dados sobre a biodiversidade de abelhas.

Estão disponíveis textos e imagens sobre dezenas de espécies, mas a qualidade da imagem é limitada pela velocidade de conexão para a maioria dos usuários. Imagens com maior nível de detalhamento e alta resolução são necessárias para a realização de estudos comparativos mais detalhados (AMÂNCIO, 2008), em que a identificação de abelha possa ser realizada remotamente. Vídeos sobre o comportamento das abelhas são recursos de pesquisa

muito importantes que precisam se tornar disponíveis na *web* para experimentos distantes.

2.4 WEBLABS

Weblabs são laboratórios acessíveis remotamente para realização de experimentos (GARCIA-ZUBIA, 2005). Os equipamentos são conectados a uma rede – no caso mais genérico, à Internet – para coleta de dados e alteração das condições do experimento. Dessa forma, tanto o acesso a dados quanto o controle dos equipamentos utilizados nos laboratórios podem ser feitos de maneira remota, por meio de uma rede de computadores.

Entre as vantagens relacionadas ao uso de Weblabs, Garcia-Zubia et al. (2005) citam:

- Melhor uso da infraestrutura do laboratório (disponível o tempo todo, desde que o Weblab esteja operacional e conectado a rede);
- Melhor organização do tempo para a realização dos trabalhos; e
- Possibilidade de cursos à distância utilizando a infraestrutura montada.

No entanto, algumas práticas em laboratório não são totalmente adequadas para a utilização em Weblabs. É o caso da formação de profissionais, em que as atividades presenciais têm melhor resultado, conforme estudo qualitativo mostrado em SOYSAL (2000).

Um quadro comparativo (NEDIC et al., 2003) de vantagens e desvantagens da utilização de diversas classes de laboratórios pode ser observado na Tabela 2.

Tabela 2: Vantagens e desvantagens do uso de diferentes tipos de laboratórios.

Tipo	Vantagem	Desvantagem
Real	• Dados reais • Interação com equipamento real • Trabalho colaborativo • Interação com supervisor	• Restrições de lugar e tempo • Caro • Supervisão é exigida
Virtual	• Explicar conceitos • Sem restrição de lugar e local • Meio interativo • Baixo custo	• Dados idealizados • Falta de colaboração • Sem interação com o equipamento real
Remoto	• Interação com o equipamento real • Dados reais • Sem restrição de lugar e local • Custo médio	• Presença real

Um Weblab está diretamente relacionado a um laboratório físico. Para diferentes laboratórios reais, diferentes Weblabs podem ser criados. Em uma arquitetura que ligue a máquina do usuário diretamente à máquina responsável pelo laboratório, considerando que um mesmo usuário utilize diferentes Weblabs, seu cadastro de usuário deverá estar presente em cada um deles, a fim de permitir seu uso em todo o sistema.

Num ambiente que manipula vários Weblabs, pode-se considerar elementos intermediários na arquitetura, entre eles o portal, para fazer a mediação entre as máquinas que controlam o experimento de um laboratório e a máquina de usuário (HARDISON et al., 2005; YAN et al., 2006; LANDI et al., 2006). Nessas arquiteturas, são utilizados *Web Services* para estabelecer a comunicação entre o portal e as máquinas que controlam os laboratórios. Na arquitetura denominada iLab (HARDISON et al., 2005) do MIT (*Massachusetts Institute of Technology* ou Instituto de Tecnologia de Massachusetts), *Web Services* também são utilizados para fazer a comunicação entre o portal e a máquina do usuário.

Porém, como colocado por Yan et al. (2006), a utilização de *Web Services* para a comunicação no lado do usuário exige programas clientes mais complexos, não permitindo que o usuário use um simples programa navegador para acessar um Weblab. A utilização de um portal surge como uma maneira de simplificar o acesso.

2.5 PROCESSOS PARA REALIZAÇÃO DE EXPERIMENTOS EM WEBLABS

A realização de experimentos científicos requer um conjunto de atividades que apresentam operações a serem seguidas para elaboração, especificação, execução e análise de seus dados. Na Figura 9, pode-se visualizar a arquitetura proposta pelo projeto myExperiment (GOBLE et al., 2007). O modelo tem como objetivo apresentar uma referência para definição das etapas a serem executadas para o desenvolvimento de experimentos científicos. A seguir, um resumo sobre as atividades envolvidas em cada camada:

- Criação do experimento: identificam-se os objetivos da pesquisa, a relação da equipe com os pesquisadores envolvidos com o experimento, além da definição das hipóteses a serem validados pelo experimento;
- Personalização: definem-se a infraestrutura a ser utilizada durante a execução do experimento, o período de execução, a especificação da área onde o experimento será executado, testes de infraestrutura, calibração de equipamentos e outros recursos pertinentes ao funcionamento da pesquisa;
- Execução e monitoramento: executa-se o experimento propriamente dito,

realizam-se operações de coleta e o monitoramento das informações obtidas;

- Análise e resultado do experimento: utilizam-se as informações coletadas na etapa de execução do experimento, disponibilizando tais dados como fonte para ferramentas que possam efetuar cálculos a partir de modelos matemáticos, além da execução de outros tipos de análises de acordo com o experimento executado;
- Compartilhamento dos resultados obtidos: ao término da execução do experimento e após a execução das análises dos dados obtidos com a pesquisa, disponibilizam-se essas informações para avaliação, validação ou base de conhecimento a serem utilizadas por outros pesquisadores;
- Reutilização dos dados para novos experimentos: por meio de metadados, ou outras fontes de dados, as informações agora servem como parâmetros para elaboração e execução de outros experimentos científicos.

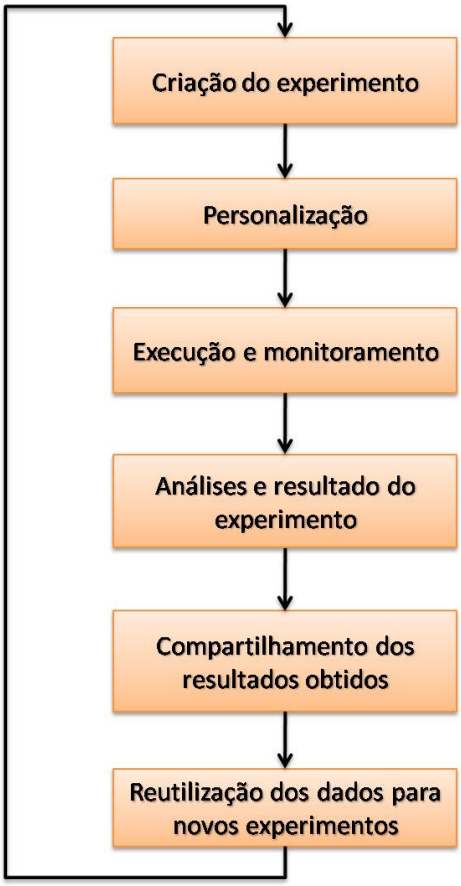

Figura 9: Ciclo de experimento do myExperiment.

Para assegurar uma interpretação segura dos dados obtidos a partir de experimentos de manipulação, o pesquisador deve tentar determinar até que ponto o *habitat* dentro das inclusões difere do *habitat* ao redor ou, ainda, se as diferenças encontradas afetam o fator ou a interação que está sendo analisada (PECKARSKY & PENTON, 1990).

2.6 EXPERIMENTOS COM ABELHAS

Existem diversas evidências de que variações de temperatura que vão além dos limites específicos de espécies podem causar danos ao funcionamento de uma colônia. Para a formiga *Acromyrmex heyeri*, por exemplo, o crescimento do fungo do qual se alimenta pode ser afetado (JONES & OLDROYD, 2007).

No caso das abelhas *Apis*, a incubação das crias em temperaturas fora da faixa ideal eleva a taxa de malformações e mortalidade das crias, além de atrasar a idade de emergência (MARDAN & KEVAN, 2002; KLEINHENZ et al., 2003). Após a emergência, a performance comportamental dos adultos é influenciada pela temperatura com que foram incubados (TAUTZ et al., 2003), e os indivíduos adultos apresentam maior suscetibilidade a alguns parasitas (MCMULLAN & BROWN, 2005).

Pouca literatura é encontrada sobre o efeito da variação de temperatura nas abelhas sem ferrão. Foi observada uma diminuição da construção de células nos períodos frios, em comparação com os mais quentes, em uma colônia de *Melipona bicolor bicolor* (VELTHUIS et al., (1999). Borges e Blochtein (2006) constataram os mesmos resultados para *M. bicolor schencki*.

Em algumas espécies que vivem nas regiões subtropicais, como as do gênero *Plebeia*, ocorre a interrupção total da construção de células de cria pelas operárias e de postura por parte da rainha durante o inverno, além de uma diminuição na atividade e uma mudança na arquitetura do ninho, ficando em um estado denominado “diapausa reprodutiva”. A temperatura, no entanto, pode não ser o único fator atuante nesse processo (RIBEIRO et al., 2003; BENTHEM et al., 1995).

Fica, portanto, evidente a importância da manutenção da temperatura no ninho dos insetos sociais. Porém, percebe-se a necessidade de realizar mais estudos, especialmente em relação aos *meliponíneos* (NETO, 2008).

Percebe-se, claramente, que existem grandes lacunas de conhecimento sobre o processo de termorregulação nos *meliponíneos*. O que há disponível permite concluir que as abelhas sem ferrão são menos capazes de regular a temperatura que as *Apis*, principalmente quando expostas a baixas temperaturas, o que pode ser um dos fatores que explique a distribuição mais limitada dos *meliponíneos* (NETO, 2008). Porém, ainda não é conhecido praticamente nenhum dos efeitos de temperaturas extremas sobre as abelhas sem ferrão. Portanto, elas dependem, em grande parte, das condições ambientais e de um local de nidificação que as auxilie na termorregulação (QUEZADA, 2005).

As técnicas de manejo dos *meliponíneos* ainda são primitivas quando comparadas com o manejo da apicultura, uma vez que conhecimentos sobre a biologia das abelhas sem ferrão são escassos (VENTURIERI et al., 2003) e a variabilidade entre as espécies é muito grande. As técnicas têm aspectos que variam de acordo com o conhecimento tradicional regional; o clima é um fator importante nessa caracterização (CORTOPASSI-LAURINO et al., 2006).

Durante a noite, em regiões subtropicais e até mesmo nas tropicais, a temperatura pode ser um fator bastante influente na prática da criação das abelhas sem ferrão. Várias abelhas nativas dessas áreas têm adaptações que as protegem nesta época do ano, como a diapausa, que é uma parada prolongada ocorrida no desenvolvimento dos insetos. Por meio da diapausa, o animal entra em repouso antes que chegue uma estação desfavorável.

Diversas formas de desenvolvimento de técnicas de manejo voltadas para a manutenção de uma temperatura ideal da colmeia foram testadas (COSTA & VENTURIERI, 2007). O isolamento térmico das colônias é a principal alternativa nas regiões mais frias. Para isso, são utilizadas caixas com madeiras espessas (de 3 a 4 cm) ou, mesmo, proteção externa às caixas, como placas de isopor, serragem ou areia (NOGUEIRA-NETO, 1997).

Foram desenvolvidas caixas térmicas com a finalidade de observação para fins científicos (menor produção de invólucro) e incubação de novas divisões (COSTA & VENTURIERI, 2007). A caixa tem um sistema de aquecimento por resistências instaladas em quadros que se encaixam na parte superior de caixas

padrões. Esses quadros podem ser transferidos de uma colônia para outra. Um sensor ligado a um termostato controla a temperatura de todas as outras caixas por meio de uma caixa controle vazia.

Cada caixa possui duas resistências em série de 20 Watts cada. Em testes com espécies do gênero *Melipona* em regiões frias, observou-se um rápido desenvolvimento das colônias divididas no início do inverno, porém o estudo carece de informações quantitativas ou comparativas.

Valmir Züge, um meliponicultor do estado do Rio Grande do Sul – região com inverno rigoroso, chegando a ficar abaixo de zero grau – desenvolveu sistemas de aquecimento também baseados em resistências elétricas. O sistema funciona por meio de uma placa que é aquecida por resistências, no fundo da caixa. Da mesma maneira que a de Costa e Venturieri, (2007), uma caixa controle vazia ligada a um termostato controla o aquecimento. (KISS, 2006; ZÜGE & AIDAR, 2000).

O sistema de Züge (2006), implantado em outros meliponários, permite criar espécies de diferentes regiões em locais com inverno mais rigoroso. O próprio inventor reconhece que o aparelho é necessário para a criação das espécies oriundas de regiões mais quentes (KISS, 2006).

Na Austrália, alguns meliponicultores montaram caixas complexas, com sistemas arrojados de controle de temperatura e também de ventilação. Ao mesmo tempo, o custo de cada sistema provavelmente é muito superior aos mencionados anteriormente (KLUMPP, 2007).

Portanto, os custos para manter as colônias aquecidas em regiões de inverno rigoroso são altos por conta do elevado consumo energético e dos materiais utilizados. O estudo de técnicas economicamente viáveis e eficientes é importante para impulsionar a meliponicultura em áreas mais frias.

3. ELEMENTOS DA ARQUITETURA PARA A AQUISIÇÃO DE DADOS DE EXPERIMENTOS AMBIENTAIS

3.1 METADADOS PARA APLICAÇÕES DE BIODIVERSIDADE

Com as mais variadas tecnologias disponíveis aos pesquisadores para auxiliá-los nos estudos, é difícil atingir um padrão de gerenciamento desses dados. Muitos trabalham com dados escritos em cadernos, outros em planilhas salvas em discos rígidos ou removíveis e alguns poucos utilizam as redes temáticas já com certos padrões de armazenamento. Na maioria dos casos, quando outro pesquisador precisa verificar ou comparar os dados, é necessário um período de aprendizado referente ao significado dos dados.

Para facilitar esse trabalho, pode-se aplicar metadados, que são dados sobre os dados em si, ou seja, são relatórios de significado para os dados ali contidos.

Mesmo com o entendimento da importância e conceito de metadados, é necessário atender a alguns padrões, visto que sua aplicação é vasta e variada dentro da sociedade. Esses padrões surgem para dar consistência aos termos utilizados, trazendo nomenclaturas e definições em comum para determinados dados. Isso facilita também a leitura dos metadados, informando de maneira simples e rápida.

Dentro dos muitos padrões já criados de metadados, citam-se alguns mais especificamente dentro de bioinformática:

3.1.1 EML – Ecological Metadata Language

Padrão de metadados baseado em *eXtended Markup Language* (XML) (MALER, 1998) e utilizado para catalogar informações de dados ambientais e ecológicos (*Ecological Metadata Language* ou Linguagem de Metadados Ecológicos) (EML, 1997). Tem como base o trabalho realizado pela Sociedade Ecológica dos Estados Unidos e esforços associados (MICHENER et al., 1998). Faz parte da *The*

Knowledge Network for Biocomplexity (KNB) – A Rede de conhecimento para biocomplexidade. EML é elaborado como uma série de tipos de documentos XML que podem ser utilizados de forma modular e extensível ao documento de dados ecológicos – um modelo pode ser visualizado na Figura 10. Cada módulo EML é projetado para descrever uma parte lógica do metadados completo que deve ser incluído no conjunto de dados ecológicos. A arquitetura do EML foi desenvolvida para servir às necessidades de pesquisadores e comunidades sobre ecologia, beneficiando-se de trabalhos anteriores sobre metadados. Algumas das características do EML são:

- Modularidade: desenhado como um conjunto de módulos ao invés de um padrão extenso, com objetivo de facilitar o crescimento da linguagem no futuro em recursos e profundidade. Ao implementar o EML como uma arquitetura extensível, os grupos podem escolher quais módulos são pertinentes para descrever os dados e os recursos do software;
- Estrutura detalhada: se concentra para equilibrar a troca de informações com detalhes suficientes para permitir serviços avançados em termos de processamento de dados por meio da análise de metadados;
- Compatibilidade: adota grande parte de sua sintaxe de outras especificações de metadados, que evoluíram por meio da experiência de grupos em outras áreas. Sempre que possível, adiciona árvores inteiras de informação visando facilitar a conversão de um documento EML em outras linguagens de metadados; e
- Forte tipagem: implementado em linguagem XML, que define as regras de sintaxe do EML. O esquema XML (*schema*) é uma recomendação do W3C, e, assim, um documento de metadados que respeite a sintaxe EML, consequentemente, cumpre os requisitos definidos no esquema XML para EML.

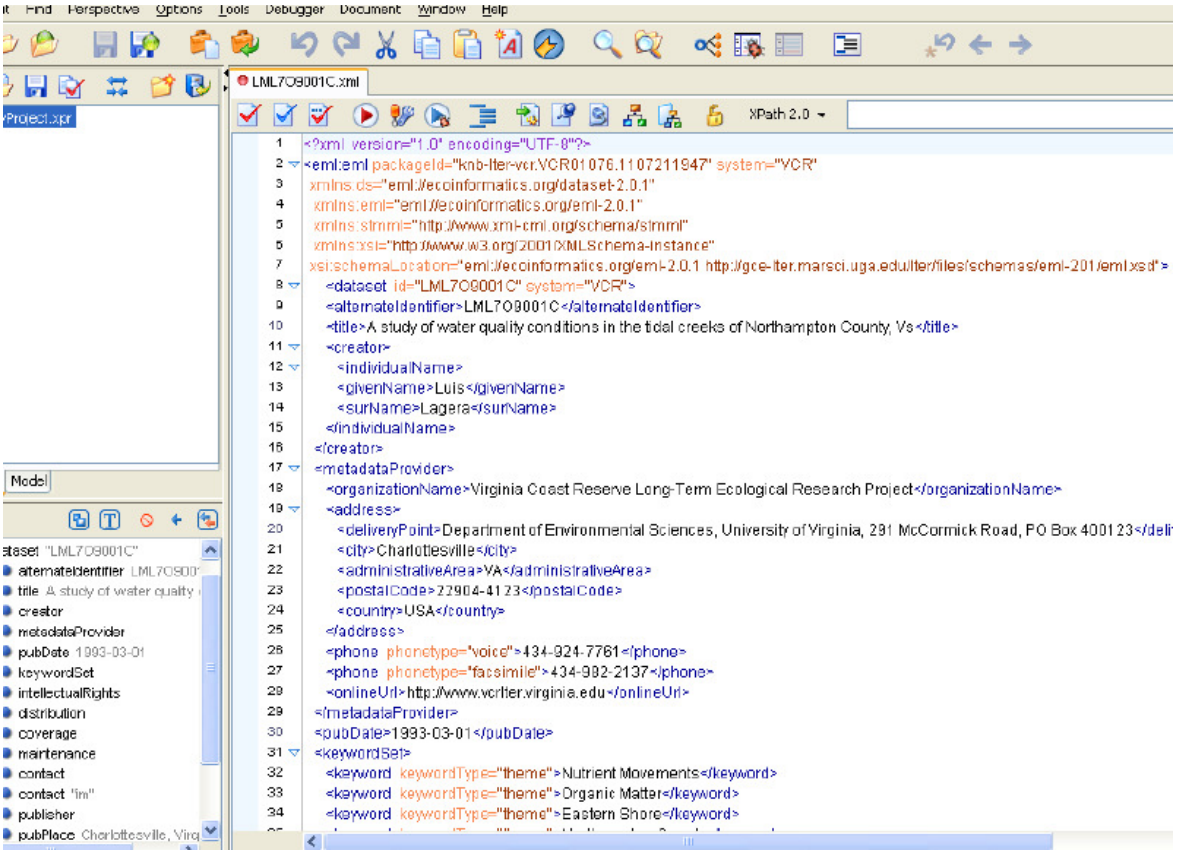

Figura 10: Exemplo de arquivo baseado no EML.

A KNB disponibiliza o Morpho, um software para a criação, gerenciamento e compartilhamento de dados com outros pesquisadores, utilizando o EML como formato padrão de metadados.

3.1.2 Darwin Core

O Darwin Core (BIS, 2007) foi projetado para facilitar a troca de informações sobre a ocorrência geográfica de organismos e a existência física de amostras em coleções bióticas. Extensões para o Darwin Core fornecem um mecanismo para compartilhar informações adicionais. O núcleo de Darwin e suas extensões são minimamente restritivos do conteúdo da informação por projeto, de modo que tornaria inútil a norma para a implementação de ferramentas de qualidade dos dados. Um modelo baseado em Darwin Core pode ser visto na Figura 11.

Segundo Kakodkar et al. (2009), o padrão Darwin Core busca facilitar o intercâmbio de informações entre robustos bancos de dados distribuídos.

Ferramentas de recuperação de dados foram meticulosamente projetadas para compatibilidade com os padrões de dados Darwin Core.

```
<?xml version="1.0" encoding="UTF-8"?>
<archive xmlns="http://rs.tdwg.org/dwc/text/"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:xs="http://www.w3.org/2001/XMLSchema"
        xsi:schemaLocation="http://rs.tdwg.org/dwc/text/ http://rs.tdwg.org/dwc/text/tdwg_dwc_text.xsd">
  <core rowType="http://rs.tdwg.org/dwc/xsd/simpledarwincore/SimpleDarwinRecord" ignoreHeaderLines="1">
    <files>
      <location>http://data.gbif.org/download/specimens.csv</location>
    </files>
    <field index="0" term="http://rs.tdwg.org/dwc/terms/catalogNumber" type="xs:integer"/>
    <field index="1" term="http://rs.tdwg.org/dwc/terms/scientificName" type="xs:string"/>
    <field index="2" term="http://rs.tdwg.org/dwc/terms/individualCount" type="xs:integer"/>
    <!-- A constant value has no index, but applies to all rows -->
    <field term="http://rs.tdwg.org/dwc/terms/datasetID" type="xs:string" default="urn:lsid:tim.lsid.tdwg.org:collections:1"/>
  </core>
</archive>
```

Figura 11: Exemplo de arquivo baseado no Darwin Core.

3.1.3 Plinian Core

O Plinian Core v.2.0 (PliC) é um conjunto de conceitos que definem atributos básicos necessários para integrar e recuperar informações sobre espécies (PLINIANCORE, 2007). É mais extenso que o DwC e carrega dados como: identificação do registro, taxonomia da espécie, ciclo de vida, comportamento, conservação e demografia, distribuição, uso, referências e até folclore (CARTOLANO et al., 2007). O PliC, entre outros esquemas de dados de espécies, é base para a criação de um novo modelo de dados de espécies organizado pelo grupo *Species Profile Model* (SPM) do TDWG. Na Figura 12, pode-se visualizar um fragmento de metadados baseado no Plinian Core.

```
- <xs:element name="Interactions" minOccurs="0">
  + <xs:annotation>
  - <xs:complexType>
    - <xs:sequence>
      + <xs:element name="TextInteractions" type="TextDescriptionType" minOccurs="0">
      - <xs:element name="Interaction" minOccurs="0" maxOccurs="unbounded">
        + <xs:annotation>
        - <xs:complexType>
          - <xs:sequence>
            + <xs:element name="InteractionType" type="NormalString">
            + <xs:element name="InteractionSpecies" type="gisin:GUIDT">
            + <xs:element name="InteractionComments" type="NormalString" minOccurs="0">
            + <xs:element name="References" type="tcs:ReferenceType" minOccurs="0" maxOccurs="unbounded">
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
```

Figura 12: Exemplo de arquivo baseado no Plinian Core.

O esquema Plinian Core v.2.0 oferece um suporte natural e estruturado ao relacionamento entre espécies. É preciso verificar se ele atende aos requisitos dos principais serviços a serem oferecidos pela IABIN-PTN (*Inter American Biodiversity Information Network - Pollinators Thematic Network*) e suportados pela Webbee. Entretanto, os esquemas de dados de espécimes, como DwC, não oferecem um suporte adequado, sendo necessário um estudo sobre a adaptação desses esquemas ou a criação de uma nova referência para a troca de dados de relacionamento entre espécimes, encontrados em coleções ou anotados em observações. Seu desenvolvimento é coordenado pelo Instituto Nacional de Biodiversidade (INBio) da Costa Rica.

Algumas categorias de elementos de metadados são necessárias para informações espaciais, como Identificação, Qualidade do dado, Organização do dado espacial, Referência espacial, Informações sobre entidade e atributos, Distribuição e Referência do metadado (KIM, 1998).

3.1.4 Dublin Core

A *Dublin Core Metadata Initiative* (DCMI) ou Iniciativa Dublin Core Metadado é uma organização sem fins lucrativos engajada no desenvolvimento de interoperabilidade de padrões de metadados que suportem uma ampla gama de

propósitos e modelos de negócio, incluindo trabalho em arquiteturas e modelos, discussões e difusão de padrões de metadados (DCMI, 2008).

As principais características e justificativas para aplicação desse padrão de metadados estão no fato de que todos seus elementos são opcionais e podem ser repetidos, reduzindo a restrição de uso para base de dados que não tenham muitas informações. Além disso, é independente de disciplinas mídias, tornando, portando, sua utilização mais apropriada para pesquisadores ou profissionais com menos experiência em metadados.

Apresenta ferramentas de uso livre e gratuito disponíveis para *download*, como Reggie e MetaMaker. As principais organizações que utilizam esse padrão de metadados são *National Biological Information Infrastructure* (NBII), *Environmental Resources Information Network* (ERIN), *Ecological Monitoring and Assessment Network* (EMAN), *European Environment Agency* (EEA*), Inter American Biodiversity Information Network* (IABIN) e *Food and Agriculture Organization* (FAO).

Algumas limitações são apontadas, como a não obrigatoriedade de seus atributos, o que, para uma base de dados complexa, prejudica a qualidade da informação. Também não é recomendado seu uso para tratar de dados biológicos individuais de espécimes, por não possuir a variedade de atributos necessários.

Na Figura 13, é apresentado um registro no padrão Dublin Core.


```xml
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE metadata (View Source for full doctype...)>
- <metadata xmlns:dcmitype="http://purl.org/dc/dcmitype/" xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dcterms="http://purl.org/dc/terms/" xmlns:fo="http://www.w3.org/1999/XSL/Format"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:ags="http://purl.org/agmes/1.1/">
  <!--  ================= Start of metadata =================  -->
  <dc:title xml:lang="en">Using a rain gauge to collect, analyze and create a rainfall monitoring protocol in Brazil</dc:title>
  <dc:creator>Leandro Halle Najm</dc:creator>
  <dc:contributor>Michael Frame</dc:contributor>
  <dc:publisher>Universidade de São Paulo</dc:publisher>
  <dc:date scheme="dcterms:W3CDTF">2009-06-23</dc:date>
  <dc:subject xml:lang="en">monitoring protocol</dc:subject>
  <dc:subject xml:lang="en">rainfall</dc:subject>
  <dc:subject xml:lang="en">rain gauge</dc:subject>
  <dc:subject xml:lang="en">environmental informatics</dc:subject>
  <dc:description xml:lang="en">Design, development and implementation of a rainfall informatics monitoring system, using a rain gauge
    to collect data.</dc:description>
  <dc:type scheme="dcterms:DCMIType">Text.Article</dc:type>
  <dc:format scheme="dcterms:IMT">application/pdf</dc:format>
  <dc:language scheme="ags:ISO639-1">en</dc:language>
  <dc:source>Oakely, Fancy Protocol</dc:source>
  <dc:coverage>Brazil;São Paulo;Piracicaba</dc:coverage>
  <dc:rights>Public Domain</dc:rights>
  <!--  ================= End of metadata =================  -->
</metadata>
<!-- xml generated by http://www.fao.org/aims/tools/metamaker.jsp?lang=en&PAGEWORKMODE=EDITMETAS  -->
```


Figura 13: Exemplo de arquivo baseado no Dublin Core.

3.1.5 FGDC BDP - Federal Geographic Data Committee - Biological Data Profile

O *Federal Geographic Data Committee* (FGDC) é um comitê de diversas agências do governo dos EUA que promove o desenvolvimento coordenado, o uso, o compartilhamento e a disseminação de dados geoespaciais em uma base nacional (FGDC, 2002).

As três principais características desse padrão de metadados são:

- Manter o investimento interno em dados geoespaciais de uma organização;
- Fornecer informações para *clearinghouses* data (câmaras de dados) e catálogos; e
- Fornecer as informações necessárias para processar e interpretar dados transferidos de outras organizações.

Para tratar de dados biológicos, foi criado um grupo de elementos de extensão, denominado *Biological Data Profile* (BDP), adicionando-se atributos de taxonomia, metodologia e ferramentas analíticas, como ArcCatalog, Metavist, SMMS (*Spatial Metadata Management System*) e uma ferramenta online Mermaid (*Metadata Enterprise Resource Management Aid*).

Na Figura 14 tem-se um exemplo de arquivo no padrão FGDC BDP de metadados.

```xml
<?xml version="1.0"?>
<?xml-stylesheet href='NBII_classic.xsl' type='text/xsl'?>
<metadata>
  <idinfo>
    <citation>
      <citeinfo>
        <origin>Leandro Halle Najm</origin>
        <pubdate>200906</pubdate>
        <title>Using a raing gauge to collect, analyse and create a rainfall monitoring protocol in Brazil</title>
        <geoform>document</geoform>
        <pubinfo>
          <pubplace>São Paulo</pubplace>
          <publish>Universidade de São Paulo</publish>
        </pubinfo>
      </citeinfo>
    </citation>
    <descript>
      <abstract>Design, development and implementation of a rainfall informatics monitoring system, using a rain gauge to coll
      <purpose>Class Project for Environmental Informatics for University of Tennessee</purpose>
      <supplinf>Create a FGDC BDP Metadata</supplinf>
    </descript>
    <timeperd>
      <timeinfo>
        <rngdates>
          <begdate>20090608</begdate>
          <begtime>073000</begtime>
          <enddate>20090803</enddate>
          <endtime>000000</endtime>
        </rngdates>
      </timeinfo>
      <current>observed</current>
    </timeperd>
    <status>
      <progress>In work</progress>
      <update>Daily</update>
    </status>
    <spdom>
      <bounding>
        <westbc>179,999999</westbc>
        <eastbc>0</eastbc>
        <northbc>0</northbc>
        <southbc>90,0</southbc>
        <boundalt>
          <altmin>540</altmin>
          <altmax>540</altmax>
          <altunits>meters</altunits>
        </boundalt>
      </bounding>
    </spdom>
    <keywords>
      <theme>
        <themekt>None</themekt>
        <themekey>monitoring protocol</themekey>
        <themekey>environmental informatics</themekey>
        <themekey>rainfall</themekey>
        <themekey>rain gauge</themekey>
      </theme>
      <place>
        <placekt>None</placekt>
        <placekey>Brazil</placekey>
        <placekey>São Paulo</placekey>
        <placekey>Piracicaba</placekey>
      </place>
    </keywords>
  <spref>
  <metainfo>
    <metd>20090623</metd>
    <metc>
      <cntinfo>
        <cntperp>
          <cntper>Leandro Halle Najm</cntper>
          <cntorg>Universidade de São Paulo</cntorg>
        </cntperp>
        <cntpos>Masters Student</cntpos>
        <cntvoice missing="True" />
        <cntemail>hallenajm@gmail.com</cntemail>
        <hours>08:00 - 12:00 / 13:15 - 19:00</hours>
      </cntinfo>
    </metc>
    <metstdn>FGDC Biological Data Profile of the Content Standard for Digital Geospatial Metadata</metstdn>
    <metstdv>FGDC-STD-001.1-1999</metstdv>
    <mettc>local time</mettc>
  </metainfo>
</metadata>
```


Figura 14: Exemplo de arquivo baseado no FGDC BDP.

No contexto de processamento de dados coletados por sensores, em particular nas ciências ambientais, ainda existe um grande vazio entre aquisição de dados e coleta dos metadados, agravado pelo problema de combinar ambos (DAWES et al., 2008).

3.2 REDES DE SENSORES

Uma rede de sensores é composta por uma grande quantidade de nós sensores que são distribuídos, aleatoriamente ou de maneira determinada, em um ambiente que precisa ser monitorado. O objetivo dessa rede é medir (sensoriar) e coletar dados do ambiente em pontos distintos de uma região, além de transmitir essas informações até um ponto de coleta (*gateway*) (CERPA, 2001).

Sensores inteligentes que podem ser implantados em terra, na água, nos corpos, nos veículos e no interior de edifícios. Um sistema de sensores de rede pode detectar e rastrear ameaças (por exemplo, agentes químicos e biológicos). Cada nó sensor tem potencial para ter embutido vários sensores, que operam, por exemplo, na acústica, modos sísmico, infravermelho (IR) e magnético, bem como outros sensores como de temperatura, umidade, pressão.

Também embarcado podem ser feitos processamento, armazenamento, comunicações sem fios para nós vizinhos, localização e posicionamento por meio do Sistema de Posicionamento Global (GPS - *Global Positioning System*) ou algoritmos de posicionamento local (CHONG & KUMAR, 2003).

Redes de sensores podem ser utilizadas em muitas áreas de aplicação (saúde, militar e doméstica, por exemplo). Para cada uma delas, existem diferentes características técnicas que precisam ser consideradas, as quais diversos pesquisadores atualmente trabalham para resolver (AKYILDIZ et al., 2002).

3.2.1 Redes de sensores sem fio

Normalmente, as redes de sensores são constituídas de um número muito grande de nós (chegando a milhares), que são lançados no campo sem posicionamento definido. Essas redes devem ter mecanismos para autoconfiguração e adaptação, visto que os nós são depositados, geralmente, de forma aleatória e não planejada. Além disso, durante a operação da rede, alguns nós podem ser danificados, movidos, desligados, ou perder a capacidade de comunicação (SAVVIDES, 2005).

Essa rede de sensores, envolvida neste trabalho, é a rede de sensores sem fio (RSSF) ou *Wireless Sensor Networks* (WSN), que são redes de comunicação de dados compostas por nós sensores, os quais se comunicam por múltiplos saltos (*multi-hop*), por meio de pequenos rádios, e são equipados com processadores embarcados, memória e sensores (HEIDEMANN & GOVINDAN, 2004). Na Figura 15, é possível visualizar um exemplo da topologia da rede de sensores sem fio.

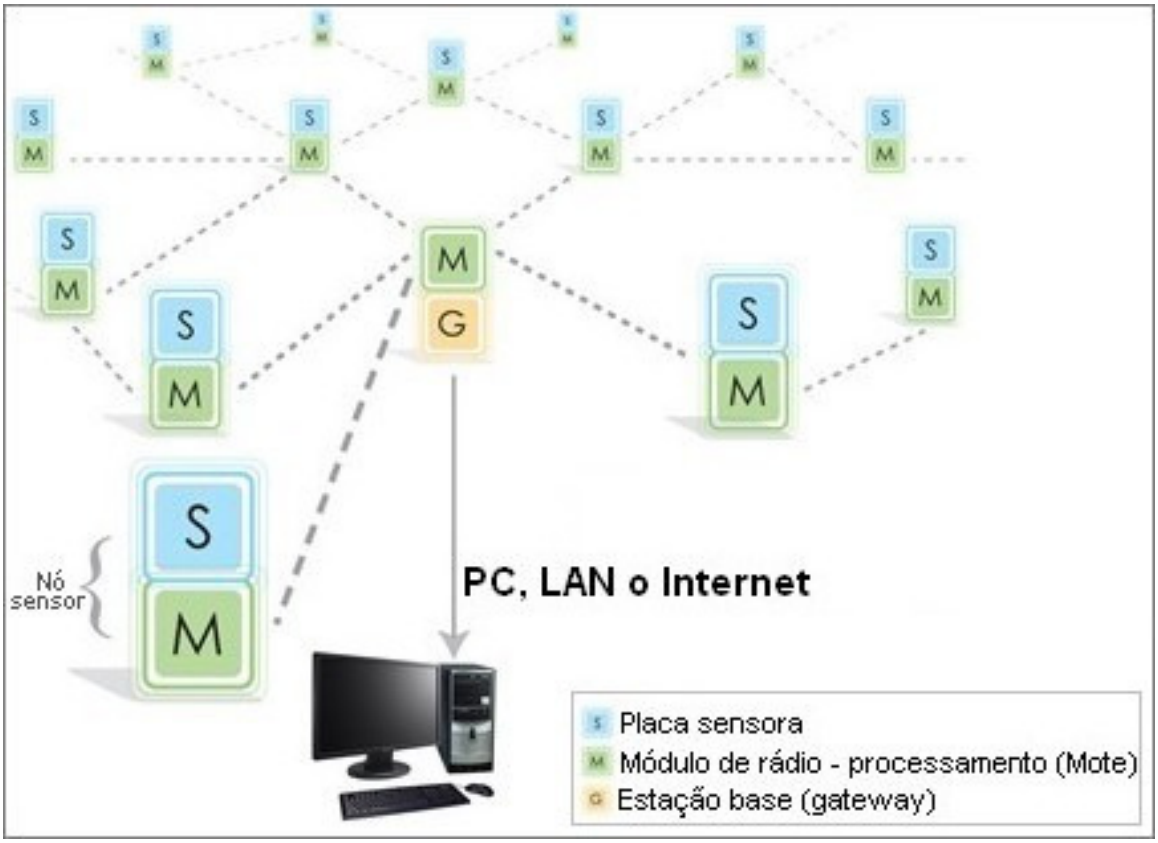

Figura 15: Exemplo de topologia de RSSF.

O maior desafio dentro de uma rede de sensores sem fio é a energia, a bateria dos sensores que é mais utilizada pelo módulo de transmissão dos dados, visto que vários estudos em diversas instituições de pesquisa estão em andamento para determinar diferentes maneiras de economizar energia ao transmitir dados, como PEAS - *Probing Environment and Adaptive Sleeping* (YE, 2003), ASCENT - *Adaptive Self-Configuring sEnsor Networks Topologies* (CERPA, 2002), *FireFly* (WERNERALLEN, 2005), *MULEs - Mobile Ubiquitous LAN Extensions* (SHAH, 2003), que são formas de transmitir os dados conforme a necessidade ou de maneira mais eficaz.

Na Figura 16 visualiza-se um nó sensor da RSSF.

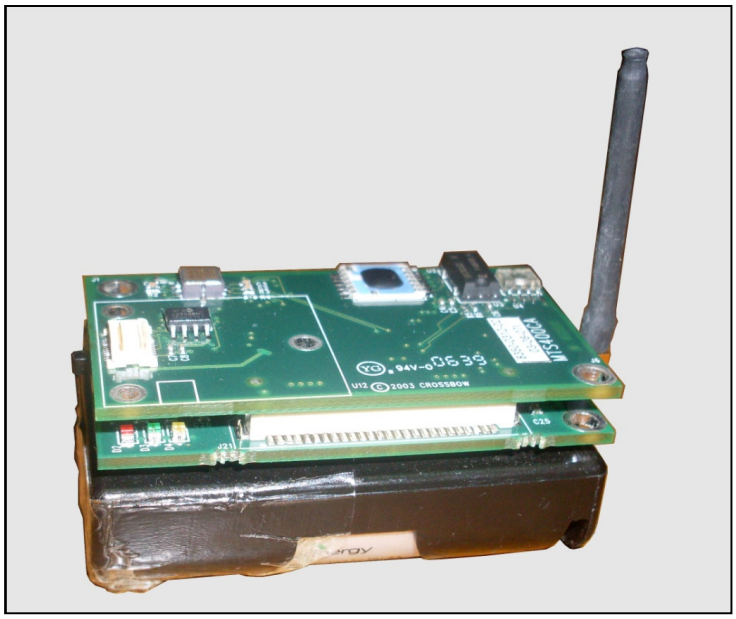

Figura 16: Nó sensor sem fio.

3.2.2 Lonworks

Outro tipo de rede de sensores envolvidos no trabalho é a LonWorks, muito utilizada em sistemas de controle predial. Esse modelo baseia-se em nós interligados em rede, constituídos de sensores de reduzidas dimensões, aos quais se interligam por meio de cabos, transferindo os dados coletados para uma rede TCP/IP (*Transmission Control Protocol / Internet Protocol*) (HUNT, 1998).

LonWorks é uma tecnologia de rede para comunicação entre vários tipos de dispositivos. As redes LonWorks podem ser aplicadas, além de controle predial, em automação industrial, aviação, sistemas de transporte, etc. Seu protocolo de comunicação é o LonTalk, o qual implementa todas as sete camadas do modelo OSI (*Open Systems Interconnection*) (JANSEN & MARIS, 2003).

Sua alimentação, diferentemente da rede de sensores sem fio, ocorre em uma ligação diretamente em uma tomada com energia elétrica.

Na Figura 17 observa-se um nó sensor da rede LonWorks.

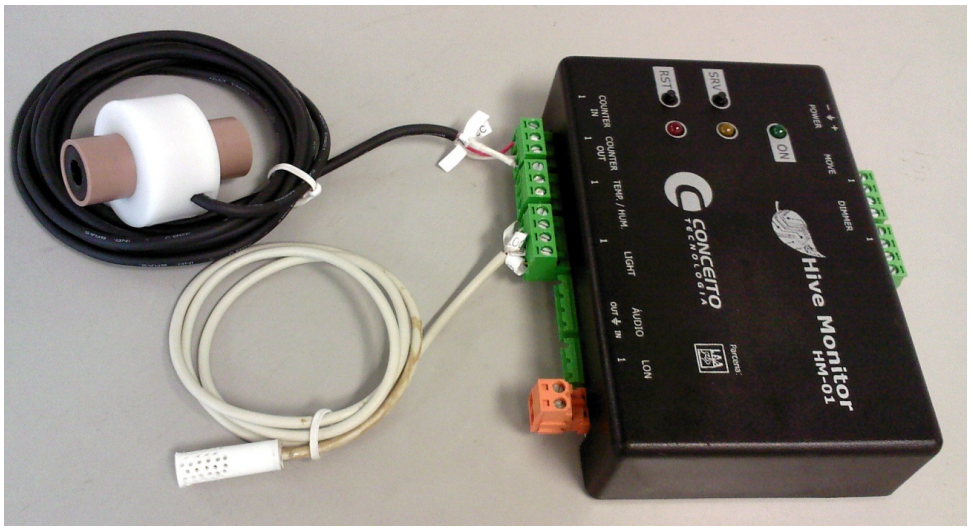

Figura 17: Nó sensor da rede LonWorks.

O que torna uma rede de controle LonWorks diferente de outras redes de controle é a inteligência construída em cada dispositivo. Isso permite que cada dispositivo responda a eventos externos, aja sobre as condições da rede e realize sua própria função localizada – sem precisar de um controlador central. Dispositivos LonWorks também comunicam-se diretamente com todos os outros dispositivos LonWorks na rede. É esse conjunto de atributos que fazem dessa rede de controle uma rede inteligente (ECHELON, 1999). Na Figura 18, pode-se notar a ligação entre os nós sensores da rede LonWorks, utilizando fios como os de *ethernet,* unidos no conector laranja.

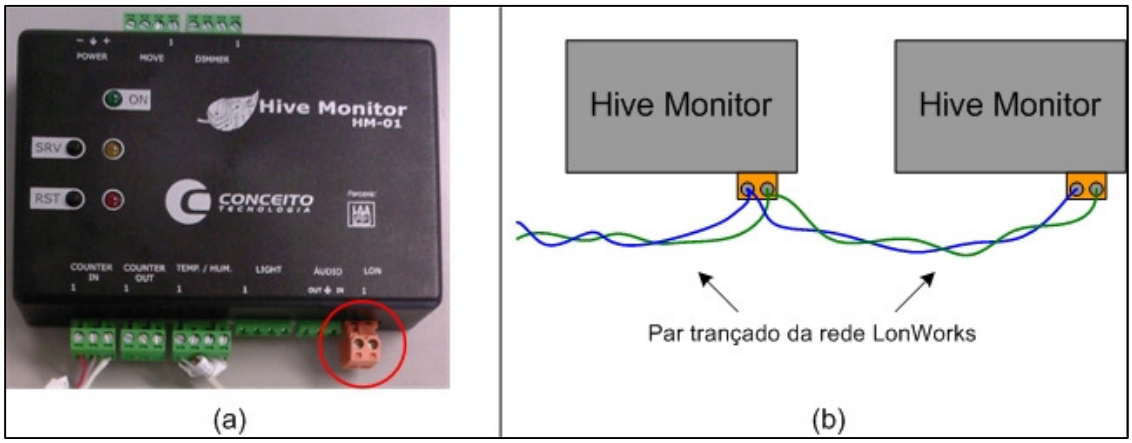

Figura 18: Ligação entre nós sensores da rede LonWorks.

O *habitat* e monitoramento ambiental representam uma classe de aplicações das redes de sensores, com enormes potenciais de benefícios para a comunidade científica e a sociedade como um todo. A instrumentação de espaços naturais, com inúmeros sensores em rede e em longo período de coleta de dados, seria a forma mais viável de obter dados em grande escala (MAINWARING et al., 2002).

É possível que a perturbação crônica humana distorça os resultados, alterando os padrões de comportamento, enquanto, na pior das hipóteses, essa perturbação humana pode reduzir seriamente ou mesmo destruir as populações mais vulneráveis, aumentando o estresse, reduzindo o sucesso reprodutivo, elevando a predação ou causando uma mudança de *habitats* inadequados (MAINWARING et al., 2002).

Conclui-se, portanto, que o uso das redes de sensores é de suma importância para o tipo de Weblab envolvido neste trabalho, de monitoramento de abelhas, em que a intrusão humana pode alterar comportamentos desses insetos, modificando, então, os dados coletados para análise e estudos.

Vale salientar que essas diferentes tecnologias de redes de sensores geram dados de formas distintas. Cada qual segue o padrão desenvolvido por seu fabricante, não sendo levada em consideração a aplicação final desses sensores, se serão utilizados para fins militares, ambientais ou na área de saúde, por exemplo.

3.3 SOA - ARQUITETURA ORIENTADA A SERVIÇOS

A Arquitetura Orientada a Serviços (SOA) é um conceito e não uma tecnologia ou ferramenta. “SOA é um paradigma para organização e utilização de competências distribuídas que estão sob controle de diferentes domínios proprietários” (OASIS, 2006).

Muitas empresas hoje contêm uma variedade de diferentes sistemas, aplicações e arquiteturas de idades e tecnologias distintas. Integração de produtos de vários fornecedores e em diferentes plataformas costuma ser custoso, tanto em tempo quanto em dinheiro.

Sistemas com implementação baseada em padrões abertos são recomendados, ao invés de soluções monolíticas e proprietárias. Além de outras razões técnicas, as práticas de gerenciamento de configuração permitem a continuidade e a evolução do sistema, mantendo as características de extensibilidade, reusabilidade e portabilidade, que devem ser contempladas na sua arquitetura, a fim de fornecer uma solução técnica e economicamente viável (SANTANA, 2009).

A visibilidade, interação e efeitos são os conceitos chave para descrever o paradigma SOA. A visibilidade refere-se à capacidade daqueles com necessidades e daqueles com competências estarem aptos a se visualizar mutuamente. Isso é tipicamente feito pelo oferecimento de descrições acerca desses aspectos como as funções e requisitos técnicos, restrições e políticas relacionadas e mecanismos para acesso e resposta. As descrições precisam estar em um formulário (ou podem ser transformadas em um formulário), no qual sua sintaxe e semânticas são amplamente acessíveis e compreensíveis (BUENO et al., 2006).

Nesse panorama, SOA se revela como uma alternativa para a construção de sistemas capazes de proporcionar a flexibilidade necessária para a realização deste trabalho, uma vez que “diferentemente do paradigma de Programação Orientada a Objeto, cujo foco está no empacotamento de dados com operações, o foco central da Arquitetura Orientada a Serviços é a tarefa ou função de negócio” (OASIS, 2006).

Os serviços são entidades autônomas, independentes de plataformas, que podem ser descritos, publicados e descobertos pela rede e são fracamente acoplados entre si (PAPAZOGLOU, 2007).

Dessa forma, o conjunto de funcionalidades providas por um serviço pode ser facilmente consumido por qualquer outra aplicação. O conceito de “orientação a serviço” baseia-se na abordagem de que os sistemas são construídos por meio da composição do consumo de serviços já disponibilizados. Os conjuntos de funcionalidades são, então, agrupados em serviços que realizam suas atividades independentes dos demais.

À medida que um serviço necessita da competência exercida por outro, este consome a funcionalidade do serviço que a provê, não replicando sua implementação. SOA visa facilitar o gerenciamento do crescimento dos sistemas corporativos em larga escala, com foco na integração dos sistemas e no reaproveitamento de funcionalidades já desenvolvidas.

Tal forma de organização extingue a barreira criada pelos sistemas e, como consequência, as inúmeras aplicações necessárias para prover a integração. Nesse cenário, qualquer serviço pode desempenhar o papel de consumidor ou o provedor de uma funcionalidade.

O grande desafio dessa arquitetura é projetar a granularidade adequada para cada serviço. Caso tenham uma granularidade muito pequena, poderá ser gerada uma quantidade excessiva de serviços, o que dificultaria a administração dos serviços disponibilizados.

Por outro lado, caso os serviços englobem uma quantidade excessiva de funcionalidades, será observada a mesma dificuldade de manutenção encontrada na arquitetura tradicional. Para que a arquitetura de uma aplicação seja classificada como SOA, esta deve estar em conformidade com suas principais.

O modelo de referência para SOA disponibilizado pela OASIS (*Organization for the Advancement of Structured Information Standards*) tem como objetivo identificar e fornecer o entendimento claro sobre as características e funcionalidades que cada um dos elementos que compõem a arquitetura em questão deve ter para ser classificada como tal. A seguir, são apresentados os critérios de conformidade para SOA e suas definições:

- Identificar entidades como serviços: um serviço é uma caixa preta, visto que sua implementação é oculta do consumidor (quem utiliza os recursos disponibilizados pelo serviço), exceto pelas informações e comportamentos expostos por meio da interface de acesso ao serviço, além das informações necessárias para verificação se o serviço atende às necessidades do consumidor;
- Identificar como a visibilidade dos serviços é estabelecida: identificar como é realizado o relacionamento entre os consumidores e provedores de serviço, considerando o momento em que ambos estão preparados para interagir entre si (visíveis);
- Identificar como a interação será mediada: identificar como são executadas ações na direção do serviço, ou seja, ações realizadas por meio do envio/recebimento de mensagens ou alteração do estado de um recurso compartilhado;
- Identificar como os efeitos da utilização dos serviços são compreendidos: identificar como os resultados das ações realizadas pelo serviço impactam

no ambiente (mundo real) ou no cenário de abrangência do consumidor;

- Possuir descrições associadas aos serviços: deve ser possível identificar as informações necessárias sobre a forma como o serviço deve ser consumido por meio das descrições;
- Identificar o contexto de execução requerido para suportar interações: identificar o conjunto de elementos necessários para que um serviço possa ser consumido e acessado corretamente, considerando questões de infraestrutura, acordos e políticas;
- Identificar como as políticas são tratadas e os contratos são formados e modelados: as políticas estão associadas a aspectos como segurança, privacidade, gerenciamento, qualidade de serviço, entre outros, e visam formalizar as regras para interação entre o consumidor e provedor de serviços, além de identificar como os contratos entre as partes são estabelecidos e administrados.

Endrei et al. (2004), apresenta uma relação de etapas a serem seguidas para a aplicação de SOA, que podem ser observadas na Figura 19.

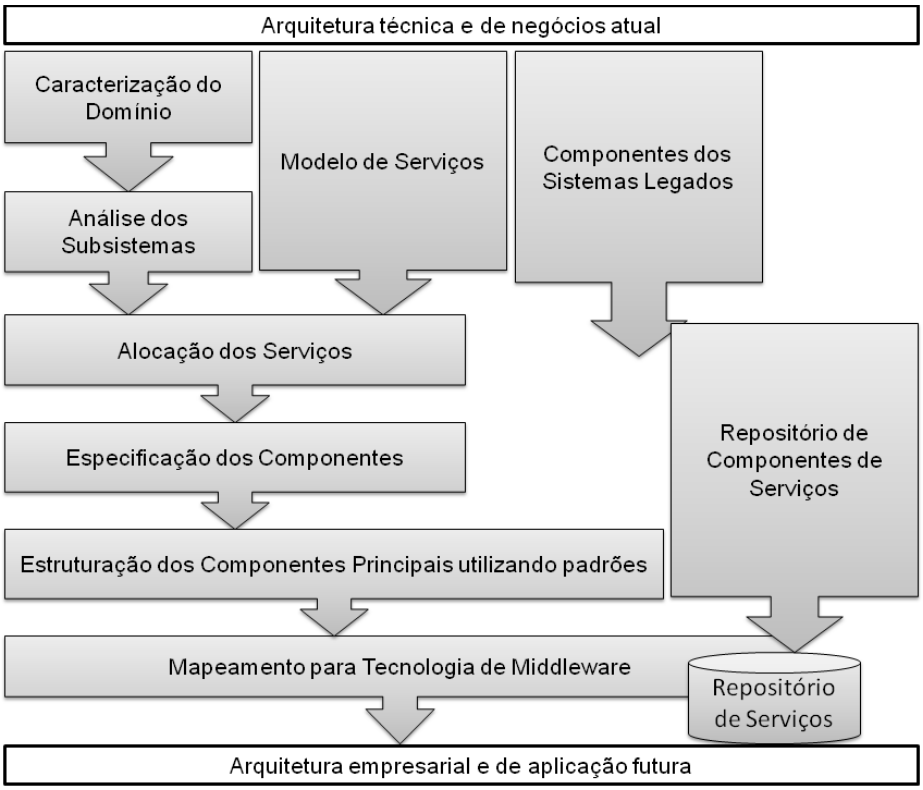

Figura 19: Sete principais passos da abordagem SOA (ENDREI, 2004).

Na etapa de decomposição do domínio, ela é feita em sua arquitetura de negócios consistindo de cadeia de valor, processos de negócios, sub processos e casos de uso. De uma perspectiva comercial, o domínio consiste de um conjunto de áreas funcionais.

Na etapa definição de modelo de serviços, a identificação de serviço implica a descoberta de serviços empresariais alinhados para toda a organização. O nível de negócios usa casos identificados na decomposição de domínio e são bons candidatos para os serviços. Nesta etapa, é criado um modelo de serviço de metas para testar a integridade dos serviços de candidatos identificados.

Depois de completar a decomposição de domínio, obtém-se uma ideia das áreas funcionais no domínio do negócio e como eles interagem na cadeia de valor para o domínio do negócio. Tem-se uma visão fractal do domínio do negócio: uma empresa estendida com fornecedores e clientes, com linhas individuais de negócios dentro de uma organização e casos de uso dentro das linhas de negócios.

A próxima etapa consiste em determinar o *design* e as decisões de arquitetura orientada. Análise de subsistemas refina os casos de uso do sistema que suporte um determinado processo de negócio. Subsistemas são compostos de componentes de negócio (como Cliente, Pedido e Produto) e os componentes técnicos (tais como mensagens, segurança e *logs*).

Atribuição de serviço identifica quem (qual componente) proporcionará a execução e gestão de cada serviço. A resposta vai depender se ele é um prestador do serviço (fornecedor) ou um consumidor.

Na especificação de componentes é desenvolvida especificações para cada um dos componentes. Propriedades precisam ser capturadas para cada componente comercial ou técnico que irá participar de uma liberação no âmbito do projeto. Na etapa seguinte é feita a padronização da estrutura de componentes e serviços.

Na última etapa, uma vez que a funcionalidade dos serviços e componentes foi especificada detalhadamente, o seu mecanismo de execução deve ser resolvido. A escolha de como realizar a implementação da especificação é um passo fundamental na arquitetura.

3.3.1 SOC – *Service Oriented Computing*

Computação orientada a serviço é o paradigma que utiliza serviços como elementos fundamentais para o desenvolvimento de aplicações/soluções (PAPAZOGLOU, 2003). Tal paradigma utiliza serviços para suportar o desenvolvimento rápido, de baixo custo, que permita interoperabilidade e com aplicações massivamente distribuídas.

Serviços podem realizar desde tarefas simples, como requisições de dados, até processos mais complexos, como processos de negócios. Dessa forma, podem prover uma distribuição de informações para plataformas independentes. Serviços são oferecidos por provedores de serviços (THOMAS, 2005), fornecendo também a descrição do serviço. Fornecem uma infraestrutura computacional distribuída que permite integração de aplicações dentro e fora do domínio, e os clientes destes serviços também podem ser aplicações que se encontram ou não no domínio.

Segundo Hunhs e Singh (2005) para poder satisfazer esses requerimentos, o serviço deve:

- Ser neutro em relação à tecnologia, podendo ser chamado por tecnologias diversas, ou seja, os meios de acesso (protocolos, descritores e mecanismos de descoberta) devem utilizar padrões abertos;
- Ter fraco acoplamento, podendo ser em diferentes níveis:
  a) Apenas as interfaces apresentadas pelo serviço são usadas nos mecanismo de composição de serviços;
  b) Por meio de mecanismo de registro e descoberta, desacopla-se o uso da localização do serviço;
  c) Utilizando um protocolo de transporte independente da plataforma; e
  d) Acoplamento pelo estilo da invocação, que pode ser controlado pelo uso de mensagens assíncronas.

- Localização transparente: definições de localização e do próprio serviço devem estar em um repositório, sendo acessíveis a uma variedade de clientes independentes de sua localização;
- Capacidade de composição: certos serviços podem ser compostos para formação de outro serviço; e

- Ubiquidade: serviços podem ser acessados por meio da Internet em qualquer lugar e momento.

A Figura 20 representa um diagrama da utilização de serviços.

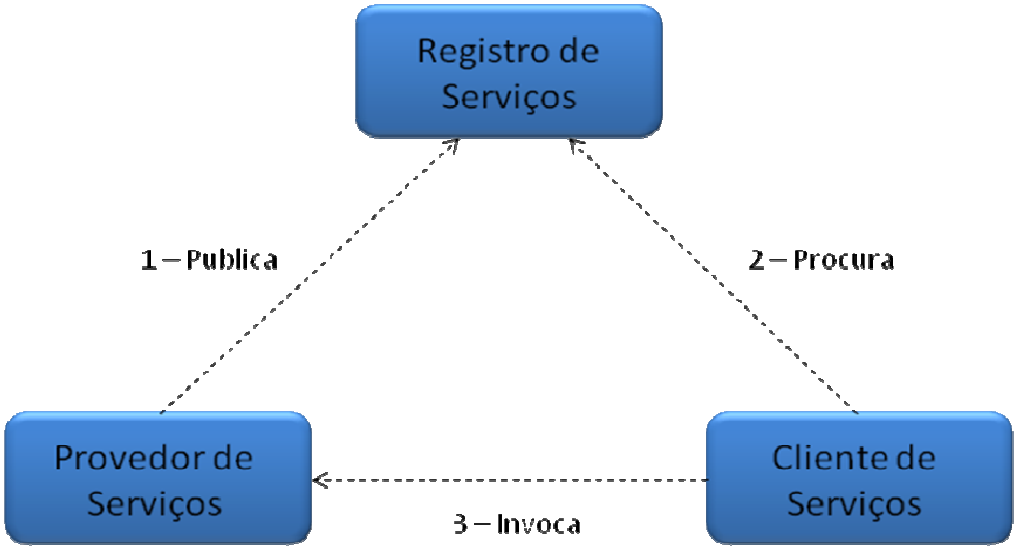

Figura 20: Diagrama de utilização de serviços.

3.3.2 Web Services

Os serviços *web* são uma tecnologia relativamente nova que tem recebido grande aceitação como uma aplicação importante da arquitetura orientada a serviços (ENDREI et al., 2004). Isso ocorre porque os serviços disponibilizados na *web* fornecem uma abordagem de computação distribuída para a integração de aplicações extremamente heterogêneas por meio da Internet. As especificações de serviços *web* são completamente independentes de linguagem de programação, sistema operacional e hardware para promover o acoplamento entre o consumidor e o prestador de serviço. A tecnologia é baseada em tecnologias abertas, como:

- XML (*eXtensible Markup Language*);
- SOAP (*Simple Object Access Protocol*);
- UDDI (*Universal Description, Discovery And Integration*);
- WSDL (*Web Services Description Language*).

O grupo de trabalho de arquitetura de serviços *web* do W3C (*World Wide Web Consortium*) determinou a seguinte definição operacional de um serviço da *web*:

"Um serviço *web* é uma aplicação de software identificado por um URI (Uniform Resource Identifiers – Identificador Uniforme de Recursos), cujas interfaces e ligações são capazes de serem definidas, descritas e descobertas como artefatos XML. Um serviço *web* suporta interações diretas com outros agentes de software baseada em XML usando mensagens trocadas por meio de protocolos baseados na Internet."

Web Services = XML + Protocolo de Transporte (como HTTP)

Algumas das principais características dos serviços *web* são:

- Autossuficientes: na camada cliente, nenhum software adicional é necessário. Uma linguagem de programação com suporte a XML e um cliente HTTP (*HyperText Transfer Protocol*) já são os requisitos mínimos para começar. A camada do servidor deve disponibilizar uma aplicação que seja um servidor *web* e um mecanismo que possibilite a execução de servlets;
- Autodescritivos: o cliente e o servidor têm apenas como preocupação o formato e o conteúdo das mensagens de solicitação e resposta, permitindo, assim, uma flexibilidade e integração de aplicações. A definição do formato da mensagem trafega junto da própria mensagem;
- Modularidade: os serviços *web* são tecnologias para desenvolver, permitir e dar acesso a funcionalidades de negócios por meio da *web*, como JEE (*Java Enterprise Edition*) e CORBA (*Common Object Request Broker Architecture)*;
- Interoperabilidade e independência de linguagem: a interação entre um prestador de serviços e um solicitante do serviço é projetada para ser completamente independente de plataforma e de linguagem. Essa interação requer um documento WSDL (*Web Service Description Language* ou Linguagem de Descrição de Serviços *Web*) para definir a interface e descrever o serviço, além de um protocolo de rede (geralmente HTTP). A interoperabilidade ocorre pelo fato de o prestador do serviço e o solicitante

do serviço não terem ideia de que plataformas ou linguagens a outra está usando;

- Abertos e baseados em padrões: as tecnologias XML e HTTP são a base técnica para o provimento de serviços *web*. Grande parte da tecnologia de serviços *web* foi construída por meio de iniciativas de projetos open source, de modo que a independência de fornecedor e a interoperabilidade são metas alcançadas;
- Dinâmicos: por meio das tecnologias UDDI e WSDL, o processo de descoberta de serviços pode se tornar um recurso automatizado; e
- Combináveis: serviços *web* simples podem ser agregados aos mais complexos, utilizando técnicas de *workflow* ou por "listas de serviços" da camada inferior de uma aplicação de serviço *web*.

A Figura 21 mostra um típico serviço de colaboração *web* baseado no modelo SOA:

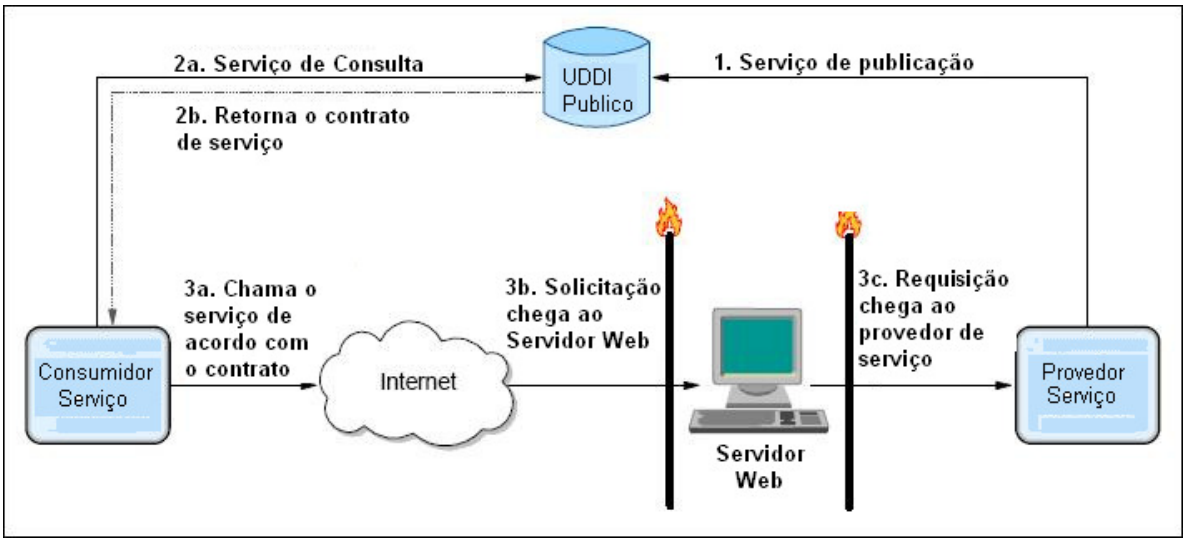

Figura 21: Colaboração de serviços *web* (ENDREI, 2004).

3.3.3 *XML – eXtensible Markup Language*

Expõe uma classe de objetos de dados chamados documentos XML (MALER, 1998), descrevendo o comportamento dos softwares que podem processá-los. XML é considerado um subconjunto de SGML, o *Standard Generalized Markup Language*

[ISO 8879]. Os documentos XML estão em conformidade com o padrão SGML. A SGML surgiu no final da década de 60 para permitir o intercâmbio e a manipulação por computadores de uma grande quantidade de documentos. É um sistema para criar linguagens de marcações, sendo que marcações (*tags*) descrevem a estrutura de um documento e outros atributos do mesmo, em vez de especificar como deve ser feito o seu processamento.

O XML é um conjunto de regras (W3C, 2007) para criação de arquivos texto que permitem a estruturação de dados (em blocos de endereços, parâmetros de configuração, transações financeiras, desenhos técnicos, entre outros) de forma não ambígua, extensível e insensível aos problemas de internacionalização/localização e de dependência de plataformas computacionais.

Documentos XML são formados por unidades de armazenamento chamadas entidades, que contêm dados analisados ou não. Dados analisados são compostos de caracteres, alguns dos quais formam dados, enquanto outros formam marcações de formulário. Essas marcações codificam uma descrição do layout do documento de armazenamento e a estrutura lógica e, dessa maneira, o XML fornece um mecanismo para impor restrições sobre esses esquemas.

3.3.4 *WSDL – Web Service Description Language*

A WSDL (WSDL, 2001) é um formato XML para descrever serviços de rede como um conjunto de parâmetros operando em mensagens que contenham qualquer documento orientado ou procedimento orientado a informação. É projetado de acordo com as especificações do W3C, que descreve como deve funcionar um serviço *web* (PULIER & TAYLOR, 2006). Softwares desenvolvidos para serviços *web* podem processar documentos WSDL e gerar mensagens SOAP automaticamente por meio da chamada de serviços específicos.

A WSDL é extensível para permitir a descrição dos parâmetros e suas mensagens, independentemente de quais sejam os formatos de mensagens ou protocolos de rede utilizados para a comunicação. Na Figura 22, tem-se o fragmento de um arquivo WSDL.

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions
   targetNamespace="http://www.webservicex.net"
   xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
   xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
   xmlns:s="http://www.w3.org/2001/XMLSchema"
   xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
   xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
   xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
   xmlns:tns="http://www.webservicex.net"
   xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">

   <wsdl:types>

      <s:schema elementFormDefault="qualified"
         targetNamespace="http://www.webservicex.net">

         <s:element name="GetGeoIPContext">

            <s:complexType/>
         </s:element>
         <s:element name="GetGeoIPContextResponse">

            <s:complexType>
```

Figura 22: Fragmento de um código WSDL.

3.3.5 *SOAP – Simple Object Access Protocol*

Segundo Pulier e Taylor (2006), SOAP pode ser considerada a língua nativa de serviços *web*. É a estrutura XML com a qual todas as mensagens de serviços *web* são construídas. Quando se diz que serviços *web* são baseados em XML, na verdade, significa que os serviços *web* são baseados em mensagens SOAP, estas sim, escritas em XML. O que torna as mensagens SOAP especiais em relação a outras estruturas XML é que cada mensagem SOAP segue um padrão que foi especificado pelo W3C. A Figura 23 mostra uma mensagem SOAP, completa com "envelope" e de seu corpo, viajando por meio de uma rede de um computador consumidor de serviços da *web* para um computador provedor de serviços, neste caso, um *mainframe*.

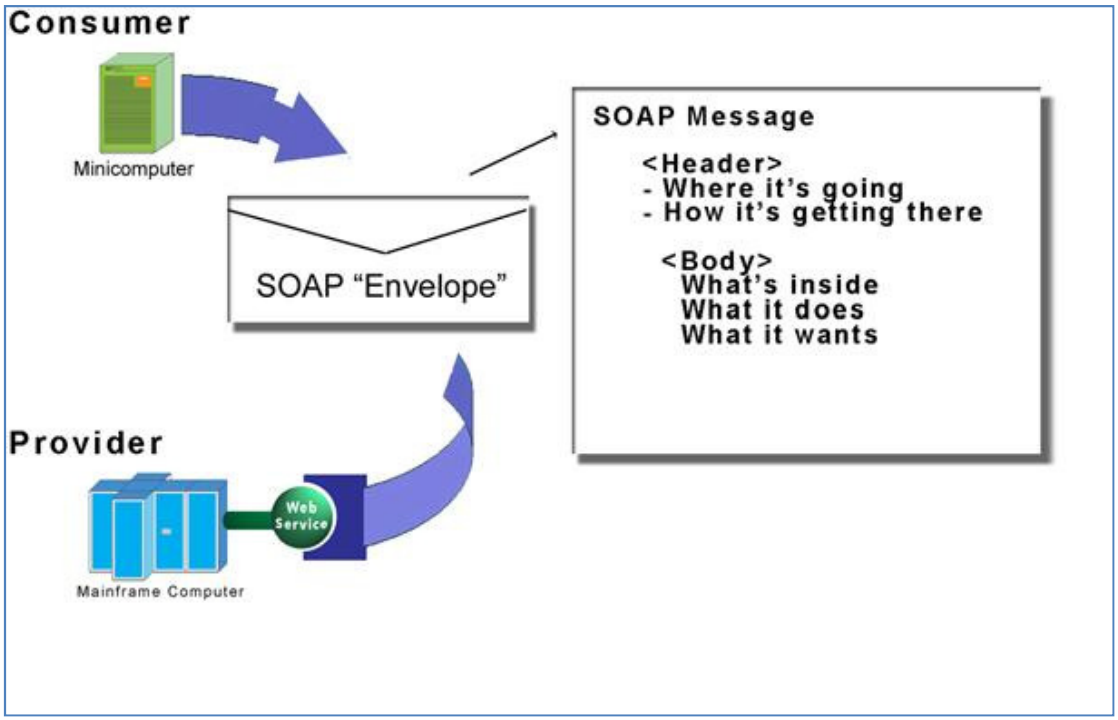

Figura 23: Funcionamento do SOAP.

SOAP, muitas vezes, pode ser comparado com um invólucro de dados ou "envelope" de dados. Cada mensagem SOAP começa com uma *tag* que se lê <*SOAP-ENV:envelope*>. Essa etiqueta inserida no envelope indica o destinatário da mensagem. O que segue é um cabeçalho, que contém as informações essenciais sobre o local de destino da mensagem e de origem. Depois, há o corpo da mensagem SOAP, que estabelece os dados reais ou instruções de operação necessárias para a máquina consumidora.

3.3.6 *UDDI – Universal Description Discovery and Integration*

Segundo Gunzer (2002), o UDDI (UDDI, 2001) funciona como um padrão desenvolvido para fornecer um diretório de busca para negócios e serviços. Tem como objetivo ser um mediador de serviços, permitindo que os clientes requisitantes encontrem um fornecedor de serviço apropriado. Em termos de funcionalidade, a finalidade de um registro UDDI é a representação de dados e seus metadados sobre os serviços disponibilizados por um site.

Um registro que pode ser para uso em uma rede pública ou dentro de uma infraestrutura interna da organização oferece um mecanismo baseado em padrões para classificar, catalogar e gerenciar serviços *web*, de modo que possam ser descobertos e consumidos por outros aplicativos. Assim, a norma especifica protocolos para o acesso, métodos para controlar o acesso e um mecanismo para delegar a distribuição de registros para outros registros.

Em outras palavras, a norma fornece um meio para localizar um serviço de software, para invocar o serviço e para gerir os metadados sobre esse serviço. Para esses fins, os principais conceitos e funcionalidades para trabalhar com UDDI incluem o modelo de dados UDDI. A especificação UDDI define tipos de dados fundamentais, que incluem uma descrição da função do serviço de negócio, informações e detalhes técnicos do serviço e da API, além de outros metadados. Esses tipos de dados são definidos em vários esquemas XML, que, juntos, formam uma base de informações e modelos de interação e estrutura de registros UDDI. Eles incluem:

- Uma descrição da função de um serviço de negócio (*businessService*): por meio da associação com uma entidade de negócio disponibiliza-se uma lista de *business services*. Cada entrada dessa lista apresenta uma descrição do serviço e categorias que descrevem o serviço;
- Informações sobre a organização que publicou o serviço (*businessEntity*): uma entidade de negócio representa informações sobre uma companhia. Cada entidade contém um único identificador, nome da companhia, descrição resumida, dados básicos de contato, lista de categorias, identificadores que descrevem a companhia e, ainda, uma URL (Uniform Resource Locator – Localizador Uniforme de Recursos) com mais informações sobre a mesma;
- Detalhes técnicos do serviço (*bindingTemplate*), incluindo uma referencia à interface de programação de aplicação, ou API, e outros atributos ou metadados como taxonomia, transportes, assinaturas digitais, entre outros (*tModels*); e
- Relações entre as entidades no registro (*PublisherAssertion*).

A Figura 24 demonstra o funcionamento dessa interação no UDDI.

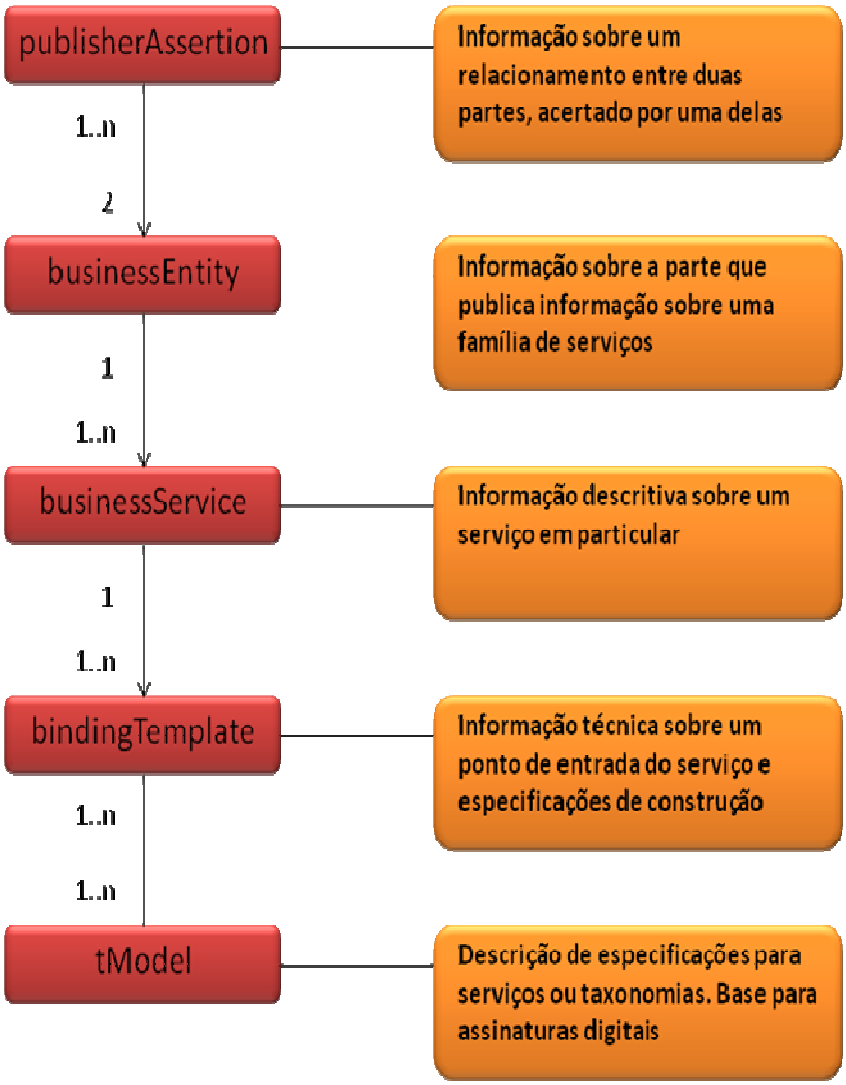

Figura 24: Modelo de Interação no UDDI.

Neste capítulo foram apresentados conceitos envolvidos nos processos de coleta e manipulação de dados dos sensores, e tecnologias utilizadas para sua disponibilização aos usuários.

A seguir é apresentado em detalhes os experimentos realizados, bem como adaptações na infraestrutura disponível, necessárias para sua execução.

4. WEBLAB PARA EXPERIMENTOS COM ABELHAS

4.1 APRESENTAÇÃO

Este capítulo apresenta a infraestrutura utilizada no Weblab de abelhas e discute os requisitos para aquisição de dados por meio da realização de um experimento com o Weblab de abelhas.

4.2 INFRAESTRUTURA

A infraestrutura computacional e de sensores implementada para o Weblab de abelhas, foi adquirida no projeto ViNCES (TIDIA, 2005), mas foi necessária sua adaptação, por se tratar de um projeto novo, mas atende a demanda de gerenciamento de informações com experimentos relacionados a serviços ambientais de polinizadores , de acordo com a Figura 25.

Foram envolvidos no projeto ViNCES um consórcio de laboratórios de pesquisa com foco em serviços ambientais, coordenado pelo Laboratório de Automação Agrícola do Departamento de Engenharia de Computação e Sistemas Digitais da Escola Politécnica da Universidade de São Paulo. Esse projeto fez parte do programa TIDIA-KyaTera da FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo) (TIDIA, 2005), o qual visou a criação de uma rede de alta velocidade entre diversas instituições acadêmicas com o objetivo de permitir o desenvolvimento e a demonstração de tecnologias para aplicação de Internet Avançada.

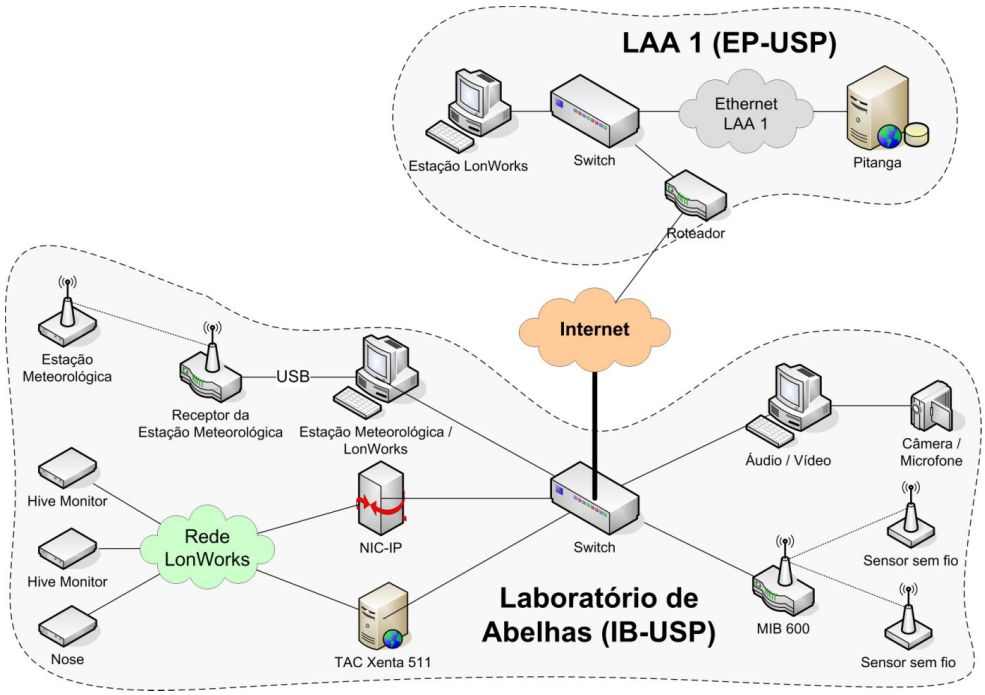

Figura 25: Topologia de interligação para experimentos com *Weblabs*

Para realizar os experimentos ambientais com abelhas, foco deste trabalho, foi preciso adaptar o modelo proposto por Goble (2007) para o contexto, o que pode ser observado na Figura 26. O experimento ocorre também em seis fases, porém com um nível de detalhamento maior e específico para o uso do experimento.

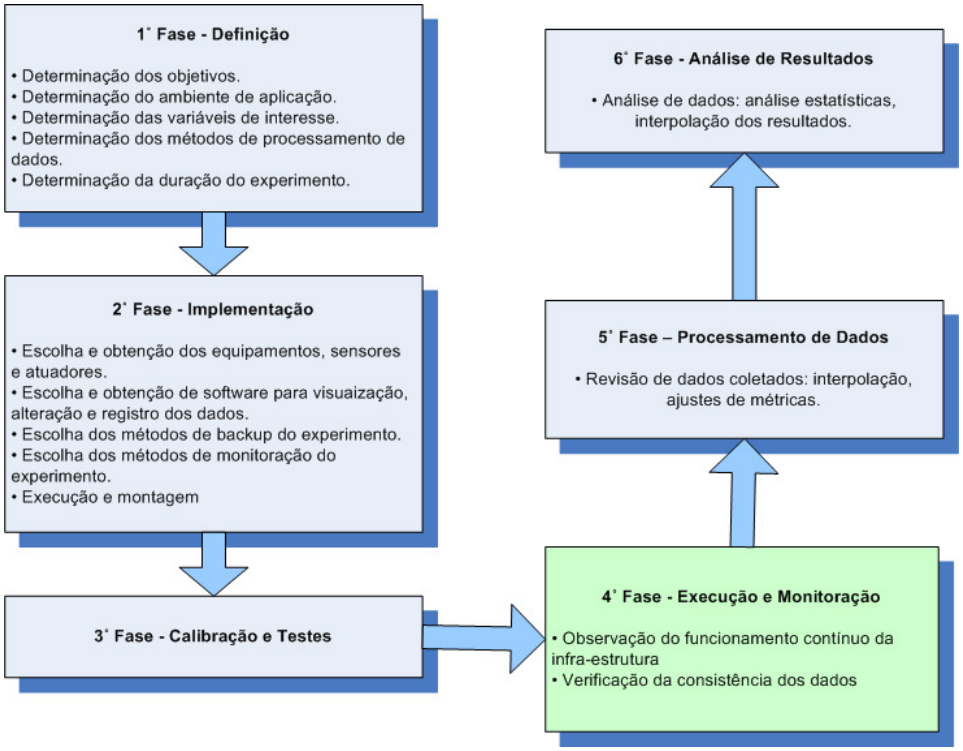

Figura 26: Fases para realização de um experimento.

A primeira fase é a definição do experimento, em que há a determinação dos objetivos e seu propósito é verificar quais são os objetivos do experimento, que devem ser enunciados de maneira mensurável (ou seja, para que se possa determinar se foram atingidos ou não). Destaca-se que tais objetivos devem ser usados posteriormente para guiar a análise dos dados (ao fim do experimento).

A determinação do ambiente de aplicação tem o propósito de decidir em que ambiente vai ser instalado o experimento - por exemplo, local de posicionamento de sensores e atuadores (no caso de experimentos com infraestrutura fixa, esta etapa pode ser desconsiderada).

Segue então a determinação das variáveis de interesse que tem como propósito enunciar quais variáveis devem ser monitoradas/modificadas no experimento (por meio de sensores), sendo estas todas aquelas que ajudem a revelar o estado do sistema e a alcançar os objetivos previamente traçados. Deve-se destacar a importância, nesta etapa, da escolha do intervalo, resolução e da taxa de amostragem das variáveis (que irá influenciar, mais adiante, a escolha de sensores a serem usados no experimento).

A determinação dos métodos de processamento dos dados estipula os passos a serem seguidos durante o processamento dos dados do experimento, de forma a condensá-los, torná-los compreensíveis e permitir a eliminação de dados suspeitos (incluindo que tipos de análises estatísticas devem ser feitas e qual a forma de representação dos dados). Esses passos podem ser executados de forma manual ou automática.

A determinação da duração total decide o tempo de duração do experimento, assim como sua data e hora de início.

A segunda fase do experimento consiste na implementação do experimento que abrange a escolha e obtenção dos equipamentos e sensores e tem o propósito de determinar e obter os equipamentos e sensores que deverão compor o experimento. Esta etapa deve estar de acordo com aquilo estipulado na determinação das variáveis de interesse, quanto a intervalo, resolução e da taxa de amostragem das mesmas.

A escolha e obtenção dos softwares para visualização, alteração e registro dos dados tem o propósito de determinar e obter os softwares que deverão compor o experimento (como bancos de dados e programas para fornecimento de acesso aos dados do experimento por meio de interface *web*), o que deve ser coerente com os

equipamentos escolhidos na etapa anterior. Tais softwares devem permitir a monitoração e o controle do experimento, assim como o registro dos dados (como aqueles provenientes dos sensores), se isto for aplicável ao experimento. A partir desse ponto, deve-se especificar, então, a configuração mínima dos computadores que deverão executar os softwares escolhidos.

A escolha dos métodos de monitoração do experimento tem de estar de acordo com os equipamentos e softwares escolhidos, determinar o regime de monitoração do experimento durante a sua execução (por exemplo, sua periodicidade, se vai ser realizada de forma automática ou manual, local ou remotamente, etc.).

É na execução da montagem que deve ser feita a montagem física da infraestrutura necessária para o experimento, bem como a configuração dos softwares (eventualmente incluindo a automatização de monitoração, a habilitação de acesso remoto ao experimento, etc.). Adicionalmente, deve-se garantir que os equipamentos tenham relógios sincronizados, quando for o caso, para que a captura e estoque de dados tenham coerência.

A terceira etapa é de calibração e teste do experimento, e é nesta etapa, que pode ser executada paralelamente à montagem do experimento, que compreende a verificação de todos os componentes do sistema (sensores, equipamentos e softwares, bem como as conexões físicas e lógicas entre eles), e a verificação final do funcionamento da infraestrutura completa do experimento. Outros aspectos importantes são verificar o sincronismo entre os relógios de todos os equipamentos, e verificar se a monitoração do sistema está funcionando adequadamente.

A quarta etapa é de monitoração do experimento, que compreende a observação do funcionamento contínuo da infraestrutura cujo propósito é verificar se todos os componentes do sistema (sensores, equipamentos e softwares, bem como as conexões físicas e lógicas entre eles), assim como o sistema como um todo, estão funcionando adequadamente durante a execução do experimento.

Essa verificação pode ser manual ou automática, local ou remota, e deve ser executada na periodicidade previamente estipulada na definição dos métodos de monitoração do experimento. Sugere-se que essa verificação seja feita do mais alto ao mais baixo nível de abstração – ou seja, partindo da captura e visualização de dados pelos softwares, até se chegar aos sensores – caso alguma falha seja verificada. Além disso, se ocorrerem falhas, a equipe de suporte ao projeto deve ser

contatada (sendo estes muito provavelmente, pessoas que participaram da implementação do sistema), e os danos ao experimento devem ser avaliados (para se decidir se o experimento deve ser reiniciado ou não).

Na verificação da consistência dos dados é preciso notar se os dados do experimento estão consistentes com o esperado (por meio de métodos de detecção de dados suspeitos, estipulados na etapa de determinação dos métodos de processamento dos dados do experimento). Mais uma vez, se ocorrerem falhas, a equipe de suporte ao projeto deve ser contatada e os danos ao experimento devem ser avaliados.

No processamento dos dados do experimento o propósito é executar os passos estabelecidos na determinação dos métodos de processamento dos dados, de forma manual ou automática (com a ajuda de softwares auxiliares).

Na análise dos dados do experimento é que é preciso avaliar o que os dados processados permitem concluir em relação aos objetivos (ou seja, contrastá-los com os resultados esperados e/ou inferir informações a partir deles). Soma-se a isso a verificação da necessidade de se refazer o experimento corrente ou de se fazer um novo experimento face às conclusões obtidas.

A seguir é apresentado um experimento com o Weblab de abelhas, detalhando os sistemas e sensores utilizados.

4.3 EXPERIMENTO

Para entender os processos e o funcionamento de um experimento realizado por pesquisadores em Weblab, mais especificamente no gerenciamento de dados de abelhas, fez-se necessário definir uma arquitetura que permitisse a aquisição automática dos dados. Durante o desenvolvimento do trabalho realizou-se um experimento prático de aquisição e armazenamento dos dados coordenado pela Profª. Drª. Vera Lúcia Imperatriz Fonseca e pelo biólogo Ayrton Vollet, do Laboratório de Abelhas da Faculdade de Ciências e Letras da USP em Ribeirão Preto – SP.

O objetivo geral do experimento foi determinar o comportamento dos dados de temperatura e umidade relativa nas diversas regiões do ninho de abelhas sem ferrão

da espécie *Scaptotrigona depilis*, em função da temperatura externa às caixas, e levantar os requisitos da arquitetura de sistemas para o monitoramento das variáveis ambientais.

Especificamente foi realizada uma avaliação da infraestrutura computacional e de sensores entre as tecnologias de instrumentos coletores de dados, visando oferecer soluções práticas aos pesquisadores que trabalham com abelhas; avaliando a variação da temperatura e umidade de maneira precisa, comparando os resultados com a literatura existente, relacionado a regulação das condições climáticas dos ninhos de *meliponíneos*.

No experimento realizado foi utilizada uma colônia de abelhas sem ferrão da espécie *Scaptotrigona depilis* (*Hymenoptera, Apidae, Meliponini*), todas em caixas de madeira, padronizadas no formato de 50 x 30 x 14 cm (medidas externas), e 2 cm de espessura, cobertas por placas de vidro de 5 mm de espessura, (ver Figura 27). As caixas foram colocadas em uma sala no Laboratório de Abelhas, as quais se conectavam com o ambiente externo através de furos nas paredes, por onde mangueiras plásticas permitiam o fluxo das operárias forrageiras.

Através de furos nos vidros, foram colocados dois sensores da rede LonWorks (região de cria e entrada/saída) e um nó sensor da RSSF (pote de alimento). Três sensores foram espalhados pela sala (dois sem fio e um da rede Lonworks) a fim de coleta dos dados do ambiente.

Todos os sensores foram programados para coletar os dados a cada 10 minutos, e ficaram funcionando durante cinco dias ininterruptos, entre 29 de abril de 2010 e 03 de maio de 2010.

Figura 27: Estrutura para monitoramento interno das colônias de abelhas

Analisando a localização das colmeias, dois fatores influenciavam as condições internas: a temperatura interna da sala e a do ambiente externo, sendo

esta última de importância significativa, devido ao ambiente não apresentar nenhuma barreira fazendo fronteira, permitindo a passagem de ar, facilitando a troca de calor e umidade. Portanto, a comparação dos dados fica comprometida. Conforme apresentado na Figura 28, a temperatura na entrada fica sempre menor que a da sala, esta não representando a temperatura ambiente.

Pode-se observar também que a temperatura da região de cria está sempre maior que a da sala, da entrada, e dos potes de alimento, sugerindo que *Scaptotrigona depilis* realiza a termorregulação dessa região (ENGELS, 1995).

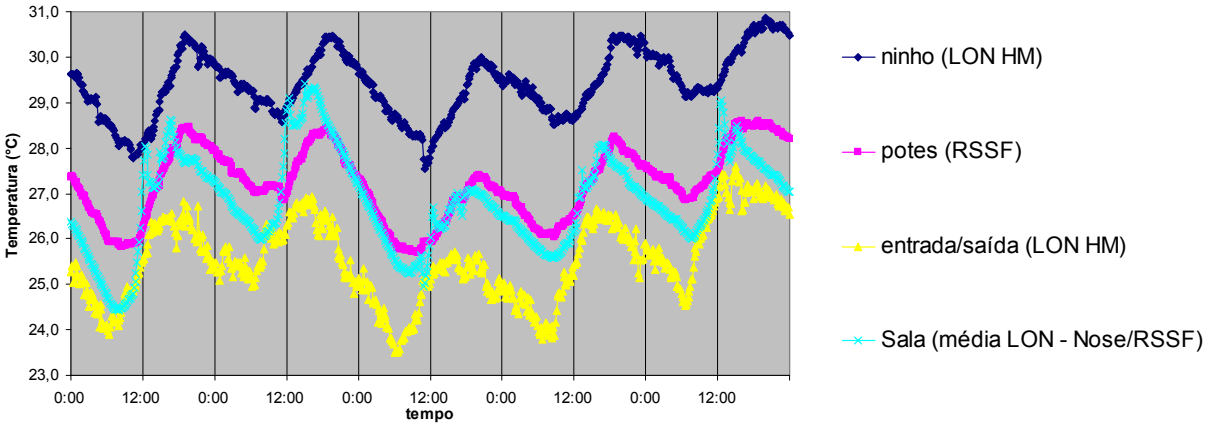

Figura 28: Temperatura a cada 10 minutos nos diferentes locais da colmeia

Os dados da entrada da colmeia representam uma contribuição nos estudos sobre o comportamento das abelhas no controle da temperatura. Porém, devido à localização da colmeia dentro da sala, fica impossibilitada a comparação destes dados com a temperatura da sala. Pode-se inferir que a temperatura do ambiente externo estava mais fria que a da sala por meio destes dados.

Como nos demais trabalhos com termorregulação nas abelhas sem ferrão, as maiores temperaturas encontradas foram na região das células de cria, assim como a maior média (29,4 ºC e desvio padrão (DP) de 0,73) (ver Tabela 3), ligeiramente inferior à temperatura encontrada por Engels et al. (1995) trabalhando com uma espécie do mesmo gênero, *Scaptotrigona postica*, que encontrou uma média de 32 ± 3 ºC nesta região.

Tabela 3: Temperatura e umidade relativa (UR) média nos locais da colmeia.

	Ninho	**Potes**	**Entrada/Saída**	**Sala**
Temperaturas médias (°C)	29,4 (DP 0,73)	27,2 (DP 0,8)	25,6 (DP 0,91)	26,8 (DP 1,06)
UR médias (%)	80 (DP 1,5)	72 (DP 2,2)	64 (DP 5,3)	54 (DP 5,6)

O acompanhamento constante da umidade relativa (UR) em várias partes de uma colônia representa uma contribuição nas pesquisas com *meliponíneos* (NICOLSON, 2009), uma vez que não foram localizados na literatura trabalhos semelhantes. Estima-se que as abelhas não coletam água diretamente, mas que a umidade presente no ninho provenha da desidratação do néctar e do metabolismo das abelhas (NOGUEIRA-NETO, 1997).

Como se pode observar na Figura 29, a UR na região da cria é mantida maior e mais constante que nas outras partes estudadas. Isto pode ser o resultado de uma manutenção da UR ativa das operárias e das estruturas do ninho, como o invólucro, ou então apenas um subproduto da presença deste invólucro para outros fins.

As operárias do *meliponineos* realizam a ventilação do ninho por meio do batimento das asas na entrada, para controlar as condições climáticas internas (NICOLSON, 2009; NOGUEIRA-NETO, 1997). Ainda não existem estudos relacionando isto com a umidade, mas observando a Figura 29, supõe-se que a alta variação da UR na entrada da colônia poderia indicar a ventilação, podendo-se também fazer inferências sobre este comportamento em trabalhos futuros.

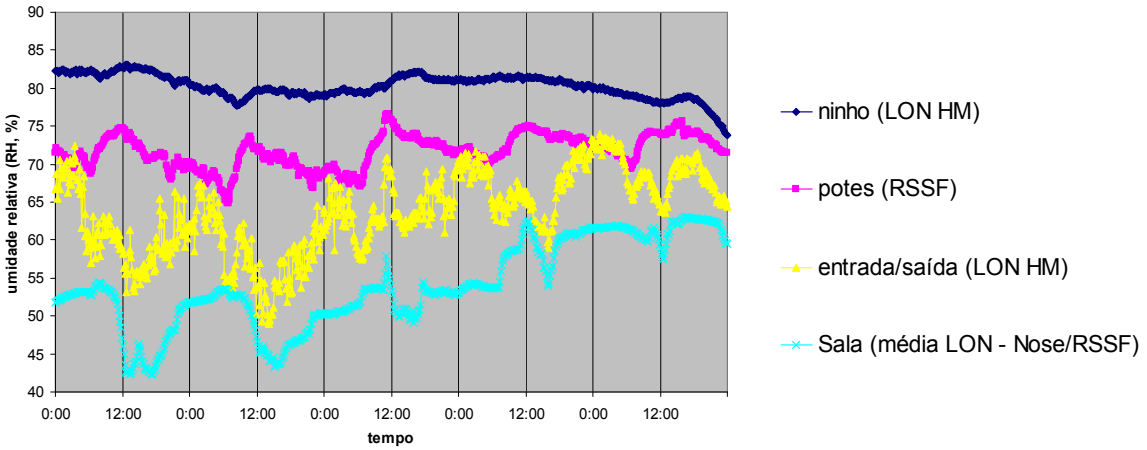

Figura 29: Umidade relativa (UR) nos diferentes locais da colmeia

O experimento realizado permitiu comparar as duas tecnologias de redes de sensores citadas anteriormente, e entre as diferenças destaca-se a dimensão dos sensores da rede LonWorks, permitindo que por seu tamanho menor sejam facilmente introduzidos nas colônias. Em contrapartida uma limitação dessa tecnologia é a mobilidade dos dispositivos devido ao cabeamento, tornando-o mais apropriado para monitoramento interno. Já as RSSF com maior facilidade na

implantação, possuem características de maior mobilidade, o que possibilita seu uso também em ambientes externos.

Dentre as dificuldades percebidas, considera-se de maior relevância o tempo de vida dos nós sensores, com uma duração média de suas baterias menor que três dias. Uma solução adotada foi ligando-os em um transformador conectado diretamente na rede elétrica.

Outro problema encontrado foi o das abelhas cobrirem os sensores com cerume (cera misturada com resina), o que poderia danificá-los. Isto foi resolvido cobrindo-os com um fino material metálico aramado, que mantém seu perfeito funcionamento.

Dentro do contexto desta proposta fica destacado que todo o trabalho de extração e análise de dados foi feito manualmente, executando consultas no banco de dados e exportação manual dos dados para planilhas eletrônicas para efetuar análises mais relevantes.

4.4 ARQUITETURA DE SISTEMAS PARA GERENCIAMENTO DE DADOS DO EXPERIMENTO

A seguir é apresentada a infraestrutura de gerenciamento dos dados utilizada para realização do experimento, onde não havia acesso direto a código-fonte de sistemas ou possibilidade de alteração nas bases de dados. No capítulo 4 é apresentada a proposta de arquitetura de sistemas para aquisição automática de dados.

4.4.1 BIOABELHA - Sistemas Web

Neste trabalho foi desenvolvido um sistema *web* para efetuar os cadastros necessários, como laboratório, experimentos, pesquisador responsável, sensores disponíveis para o experimento, objetivos, informações de contato, entre outras. Na Figura 30 pode-se visualizar a tela inicial de acesso ao Weblab Bioabelha.

Figura 30: Tela inicial do Weblab Bioabelha

O esquema lógico do banco de dados do sistema *web* Bioabelha foi feito exclusivamente para atender as demandas do experimento em questão, sem alterações no processo habitual dos pesquisadores responsáveis. Ela pode ser visualizada abaixo na Figura 31.

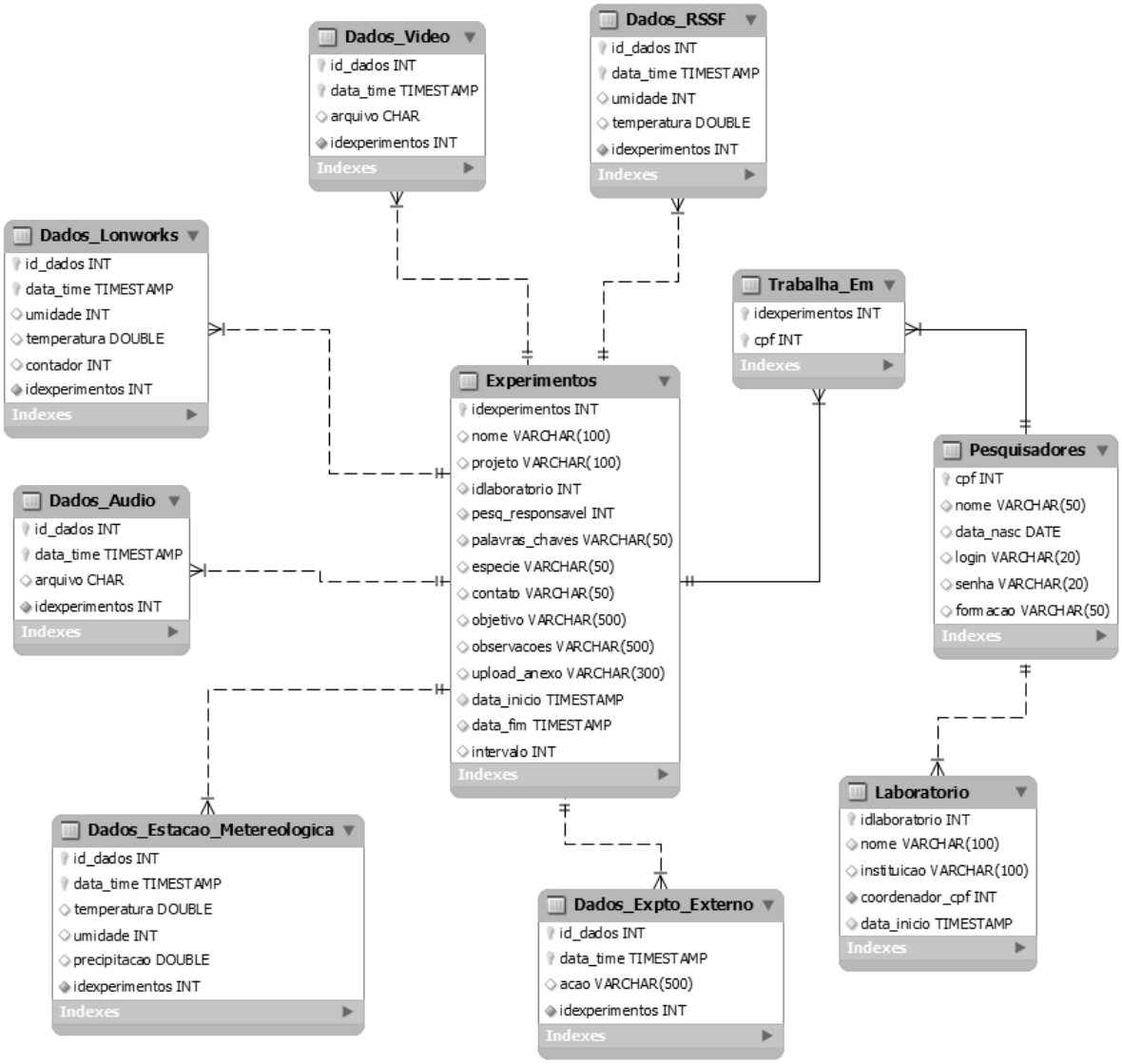

Figura 31: Estrutura lógica da base de dados de experimentos do Weblab.

O banco de dados foi modelado com visão específica do laboratório em que foram feitos os experimentos, com relações e atributos preparados para receber dados dos sensores já instalados e sistemas já em funcionamento, portanto com uma restrição de futuras alterações e novas tecnologias, perdendo a funcionalidade em caso de troca dos modelos e sistemas que insiram os dados no Weblab.

4.4.2 Sensores

Para realizar a coleta de dados do interior das colmeias e a realização do experimento foram utilizadas duas tecnologias diferentes de sensores, uma delas é a de sensores ligados por uma rede LonWorks e outros sensores em uma rede de sensores sem fio.

4.4.2.1 LonWorks

Na rede de sensores LonWorks disponível no laboratório para coleta de dados, há componentes de software, chamado de RemoteLon, com a finalidade de supervisionar um sistema de controle e automação da rede e inserir o dado coletado em uma base de dados. A base de dados desse sistema é gerenciada pelo SGBD *MySQL* e sua estrutura lógica pode ser vista abaixo, na Figura 32.

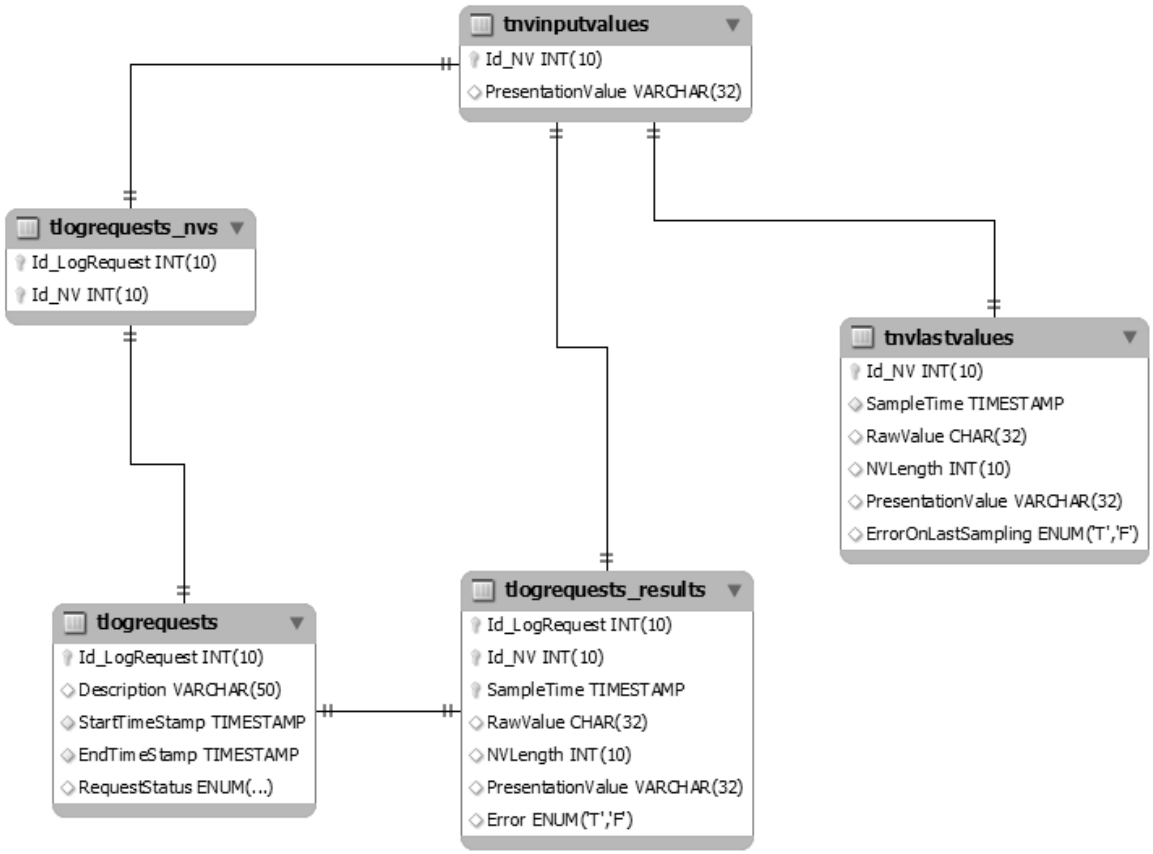

Figura 32: Estrutura lógica da base de dados do RemoteLon

Essa base de dados é estritamente direcionada para receber os dados coletados pelos sensores da rede LonWorks, não podendo ser alterada ou reconfigurada para trabalhar com diferentes dados de outras fontes coletoras.

Para a realização do experimento, a rede de sensores LonWorks foi montada de acordo com a Figura 33, com 3 módulos de sensores, contendo um sensor de temperatura e outro de umidade em cada módulo, um NIC-IP para a conexão com a rede TCP/IP e o computador para monitoramento e atuação no sistema.

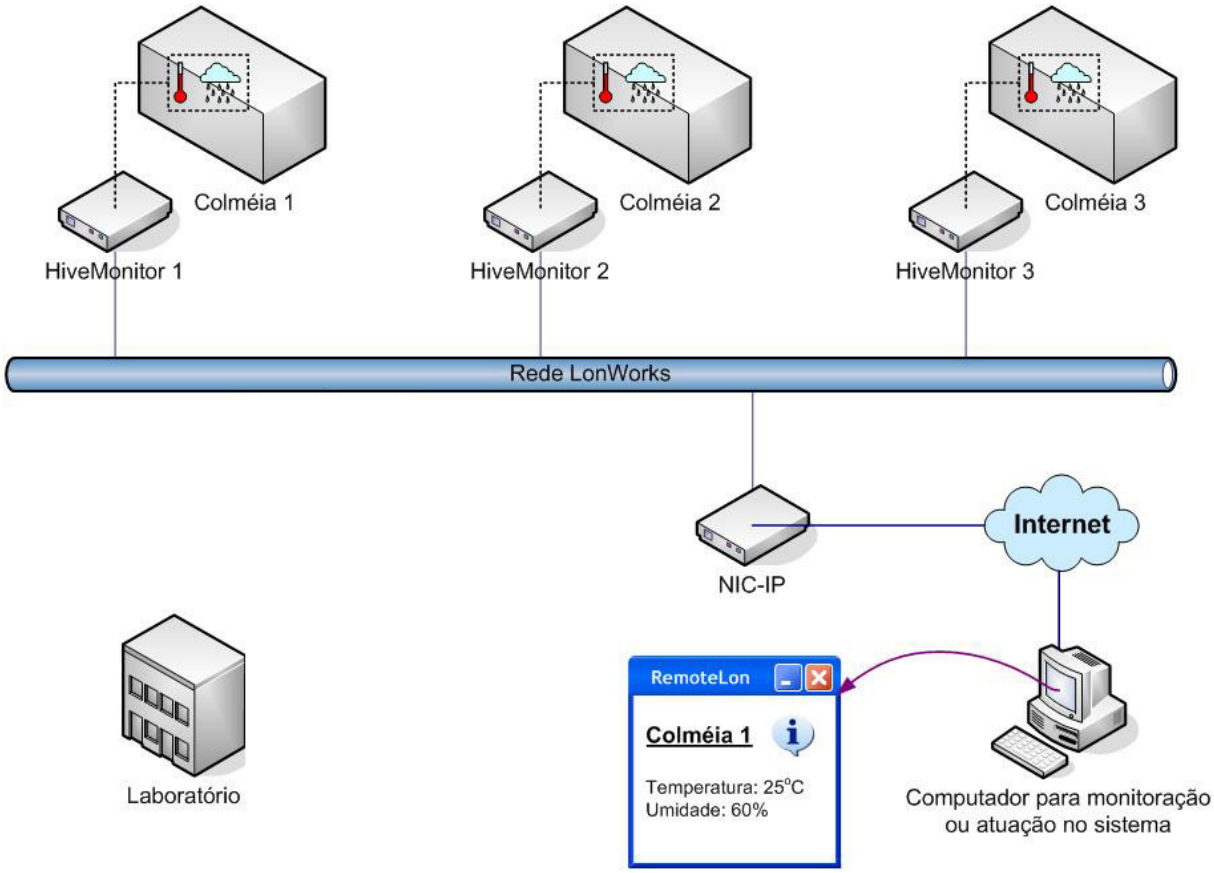

Figura 33: Estrutura dos sensores da rede LonWorks para o experimento

4.4.2.2 Redes de sensores sem fio

A rede de sensores sem fio montada no laboratório de manufatura da empresa *Crossbow* possui um software específico para configuração, monitoramento e

inserção dos dados coletados em uma base de dados própria, gerenciada pelo SGBD *PostgreSQL* e sua estrutura pode ser visualizada na Figura 34.

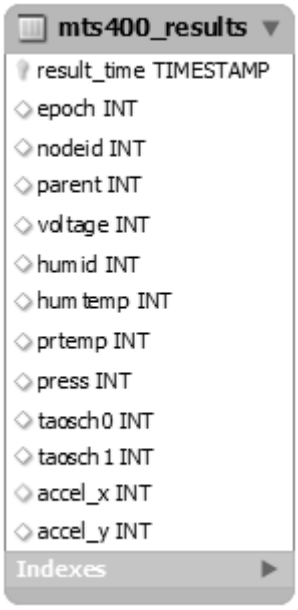

Figura 34: Estrutura da base de dados da RSSF

Como pode ser observado há somente uma relação com os atributos de armazenamento dos dados.

4.4.3 Análise de requisitos

Neste item são discutidos os requisitos funcionais e não funcionais para a especificação da arquitetura de aquisição dos dados de experimentos com abelhas.

Para a realização do experimento, todas as redes foram corretamente configuradas para coletar dados no mesmo intervalo de tempo.

As dificuldades encontradas foram justamente no trabalho posterior à coleta, que é o de tratamento desses dados, visto que cada base de dados, de diferentes redes e diferentes tecnologias armazena a informação em formatos diferentes.

Além disso, os pesquisadores para retirar essas informações das bases de dados precisavam fazer consultas (*queries*) utilizando linguagens de bancos de dados para obterem os resultados necessários e exportá-los para um formato mais habitual de uso, como planilhas.

No caso específico dos dados da base da rede LonWorks, esses dados ficam armazenados em formato Hexadecimal, tendo então que ser convertido para valores binários e então convertidos em base decimal, nos dados referentes a umidade, considerando que os dados ficam armazenados no mesmo atributo da relação, o atributo *RawValue*.

Um exemplo de visualização dos dados tratados pelos pesquisadores pode ser visto na Tabela 4.

Tabela 4: Tratamento dos dados coletados da base de dados da rede LonWorks

Id_NV	SampleTime	RawValue	Conversão Binária	Decimal
202	2009-11-17 18:16:19	426405CD	01000010011001000000010111001101	57
202	2009-11-17 18:21:19	42608C93	01000010011000001000110010010011	56
203	2009-11-17 18:01:24	424D217E	01000010010011010010000101111110	51
203	2009-11-17 18:06:24	42590BF6	01000010010110010000101111110110	54
203	2009-11-17 18:11:24	42693333	01000010011010010011001100110011	58
203	2009-11-17 18:16:24	42661F7E	01000010011001100001111101111110	58
203	2009-11-17 18:21:24	4262A8FF	01000010011000101010100011111111	57

Depois de realizado o experimento, foi constatada a necessidade de automatização de alguns processos, como a aquisição e visualização dos dados, onde surgiu a proposta de alteração na modelagem da base de dados central para armazenar os dados e a disponibilização de serviços *web* para criação e monitoramento de experimentos com abelhas.

Para evitar ou minimizar problemas em um experimento em andamento, é necessário disponibilizar serviços de alerta ao pesquisador responsável pelo experimento, como no caso de algum sensor para de funcionar ou caso deixem de coletar dados cadastrados no experimento.

É necessário destacar também que a utilização de uma arquitetura padronizada permite a interoperabilidade de sistemas, visto que novos sensores de diferentes arquiteturas aos existentes poderão ser incluídos ou substituídos no sistema.

A conversão automática dos dados torna-se necessária, considerando que independente do formato com que os dados são coletados pelos sensores, eles são disponibilizados em um mesmo formato no Weblab.

O sistema deve gerenciar os dados coletados em experimentos distintos e estes devem ser organizados adequadamente pelo sistema, uma vez que mais de um experimento pode estar sendo executado em um mesmo período.

5. ARQUITETURA PARA COLETA AUTOMÁTICA DE DADOS

5.1 APRESENTAÇÃO

Este capítulo apresenta a proposta de arquitetura orientada a serviços para aquisição automática de dados de experimentos em Weblab de abelhas.

Observando os sistemas individuais existentes para aquisição de dados no laboratório de pesquisa de abelhas, identificou-se a necessidade de integrar os diferentes dados dos sistemas e definir um esquema lógico de base de dados do Weblab para que pudesse integrar dados de diferentes bases, além das que já estavam em funcionamento, como LonWorks e RSSF.

5.2 ESQUEMA LÓGICO DO BANCO DE DADOS DO WEBLAB

O esquema lógico do banco de dados centralizado permitiu integrar dados de aquisição de variáveis ambientais e permitiu a criação de uma relação para receber dados geoespaciais, possibilitando futuras consultas a localização dos laboratórios ou dos sensores utilizados nos experimentos.

Na Figura 35 é mostrado o esquema lógico da base de dados central para a aquisição automática dos dados, utilizando com atributos específicos para mapeamento de padrões de metadados, para facilitar a interação com outras bases de experimentos.

As relações que compõe o esquema lógico da base de dados podem ser classificadas de acordo com o tipo de descrição que possuem. Os dados cadastrais do experimento (*Experiment)* estão relacionados aos dados do laboratório (*Laboratory*).

Os dados do pesquisador (*Researcher*) estão relacionados com o laboratório (*Laboratory*) e o experimento (*Experiment)*.

Os dados ambientais (*Data*) estão relacionados com o experimento (*Experiment)* e o sensor (*Sensor*), assegurando que dados coletados em um experimento não poderiam interferir em algum outro experimento em execução.

A relação de acesso (*AccessLevel*) ao sistema permite que o responsável pelo laboratório altere a permissão dos usuários, para que possam cadastrar experimentos, monitorar a coleta dos dados e visualizar os dados de experimentos anteriores. Pode-se também ocultar a disponibilização dos dados coletados de um experimento caso exista a necessidade.

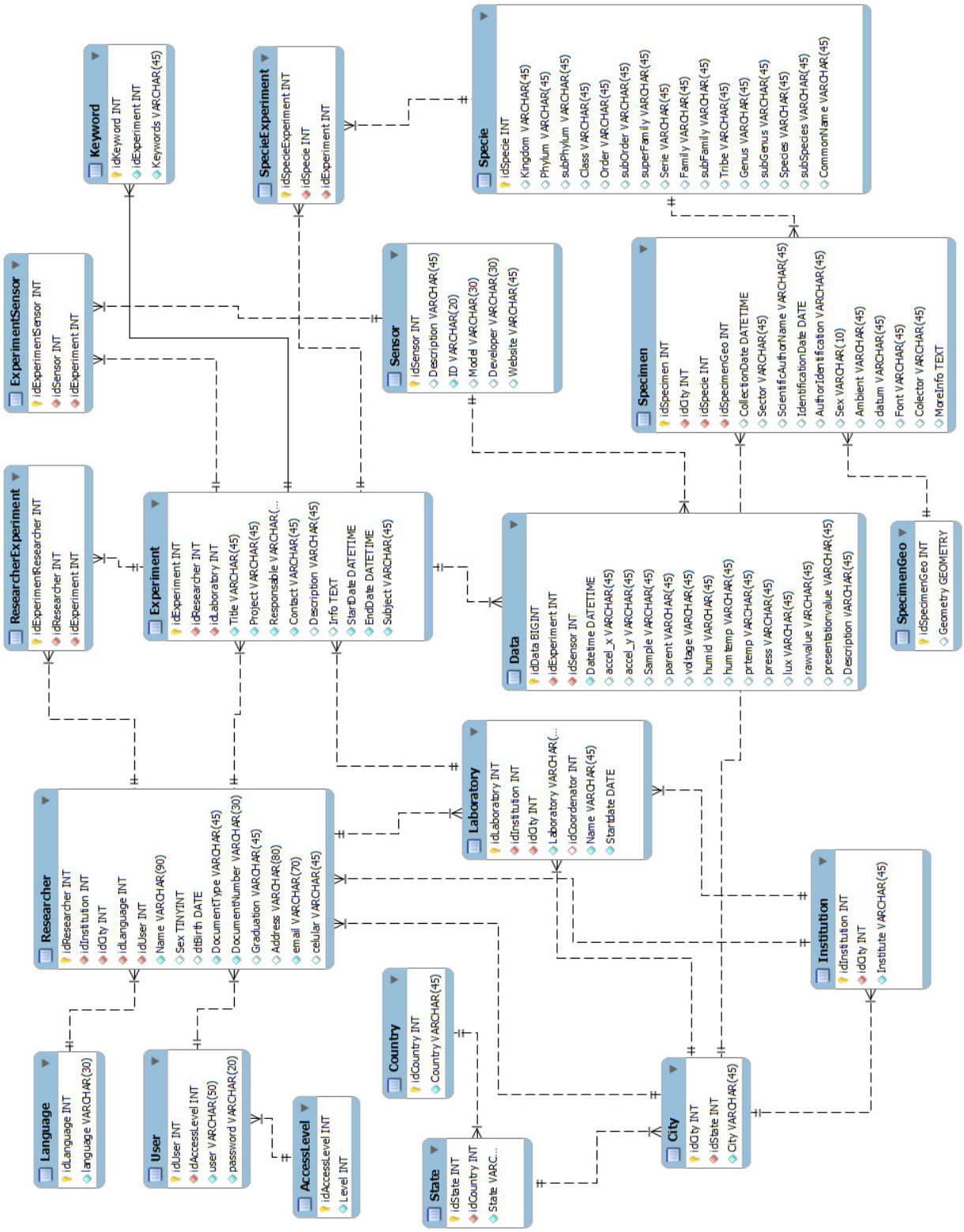

Figura 35: Esquema lógico da base de dados central do Weblab.

De acordo com o método definido para especificação de serviços (ENDREI et al., 2004) as seguintes etapas foram utilizadas para definição da arquitetura.

5.3 DEFINIÇÃO DE DOMÍNIO, COMPONENTES E SERVIÇOS

Adaptando o modelo de Endrei et al. (2004), discutido no Capítulo 2, é apresentado na Figura 36 uma série de etapas para a definição de componentes e serviços a serem utilizados no Weblab de abelhas.

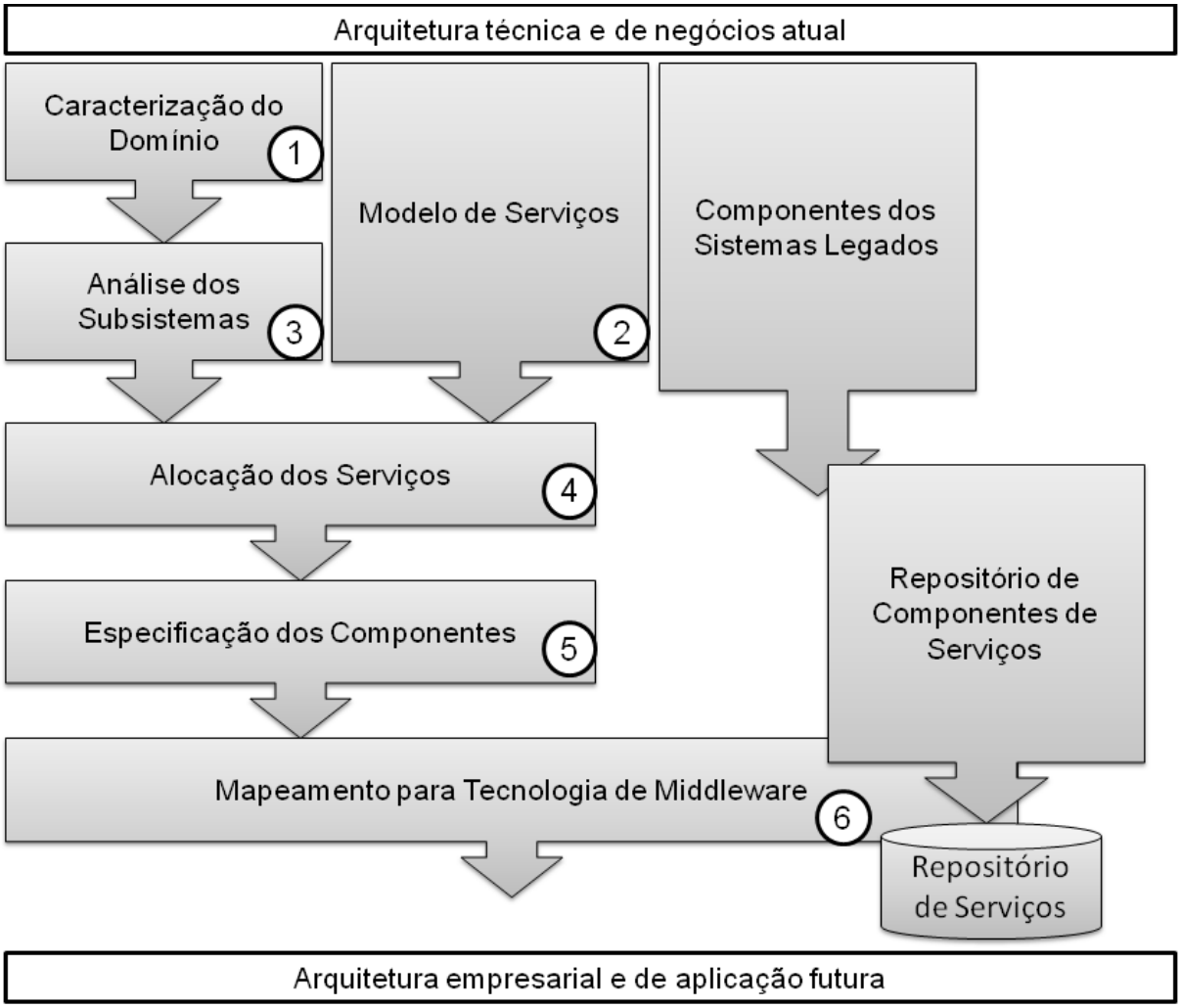

Figura 36: Principais passos da abordagem SOA (adaptado de ENDREI, 2004).

Nas etapas 1 e 2, de caracterização do domínio e modelo de serviços, baseado na perspectiva do negócio, são apresentadas na Figura 37 as áreas funcionais (domínio) encontradas.

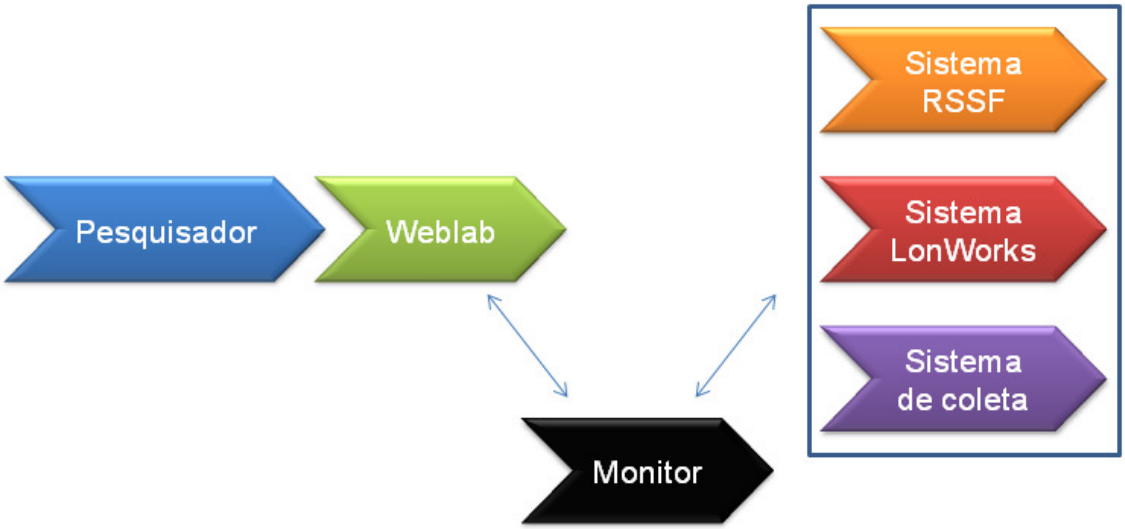

Figura 37: Caracterização do domínio do sistema.

Após a caracterização do domínio os seguintes casos de uso foram levantados:

Domínio Pesquisador:

- UC1 - Consulta disponibilidade do Weblab: este caso de uso verifica se o *website* do portal está acessível;
- UC2 - Realiza a autenticação: o pesquisador insere suas credenciais de usuário e senha para acessar o sistema;
- UC3 - Cadastra um experimento: o objetivo deste caso de uso é de cadastrar um experimento, informando um período de coleta de dados, assim como os sensores a serem utilizados.

Domínio Weblab:

- UC4 - Verifica autenticação do pesquisador: este caso de uso verifica as credenciais informadas pelo pesquisador, autorizando ou não seu acesso;
- UC5 - Aprova o experimento: alguns tipos de experimentos podem causar danos aos sensores ou as abelhas do laboratório, portanto devem ser autorizadas pelo responsável;
- UC6 - Reserva os sensores selecionados no experimento: este caso de uso

tem como objetivo não permitir que um mesmo sensor seja reservado para diferentes experimentos, em um mesmo período;

- UC7 - Disponibiliza os dados do experimento: este estudo de caso objetiva a disponibilização dos dados aos usuários do portal.

Domínio Monitor:

- UC8 - Consulta o cadastro de experimentos: este caso de uso tem como objetivo consultar a lista de experimentos do Weblab para realizar a aquisição automática dos dados;
- UC9 - Realiza a aquisição dos dados dos BD dos sensores: é a aquisição de dados automática que extrai os dados dos sistemas coletores;
- UC10 - Realiza a conversão dos dados: consiste em converter os dados dos sistemas coletores, caso haja necessidade, para o formato padrão do Weblab;
- UC11 - Realiza o mapeamento de metadados dos dados: representa a padronização dos dados na base de dados central, mapeando os dados no padrão Dublin Core;
- UC12 - Insere os dados no banco de dados do Weblab: este caso de uso abrange a inserção do dado padronizado na base de dados central;
- UC13 - Gerencia falhas na coleta dos sensores: engloba o gerenciamento de falhas na coleta dos dados, verificando se os sensores estão ativos e se estão coletando no tempo de amostragem correto.

Domínio dos sistemas de coleta:

- UC14 - Coleta os dados no ambiente: este caso de uso objetiva a coleta de dados dos sensores no ambiente;
- UC15 - Armazena os dados no BD: representa a inserção dos dados coletados pelos sensores em seus sistemas.

Na Figura 38 estão representados graficamente os casos de uso especificados, cujo domínio “Pesquisador” representa o pesquisador, que acessa a *URL* do Weblab e já identifica se está em funcionamento ou não. Ao constatar sua disponibilidade pode realizar sua autenticação e então o cadastro do experimento.

No domínio “Weblab” são os casos de uso do Weblab, autorizando o acesso do pesquisador, aguardando um aval do pesquisador responsável para aprovar o experimento cadastrado e no final do processo, disponibilizar os dados coletados.

Os casos de uso do domínio “Monitor” ele é responsável por consultar se há um novo experimento cadastrado, verifica se os sensores selecionados pelo pesquisador para o período estão disponíveis, os reserva para e no período do experimento, acessa os sistemas de coleta e extrai os dados, os converte para o padrão do Weblab, faz o mapeamento no banco de dados central utilizando o padrão de metadados Dublin Core e insere os dados no banco de dados central.

O domínio sistema de coleta representa os casos de uso das redes de sensores sem fio, que coletam e armazenam os dados ambientais.

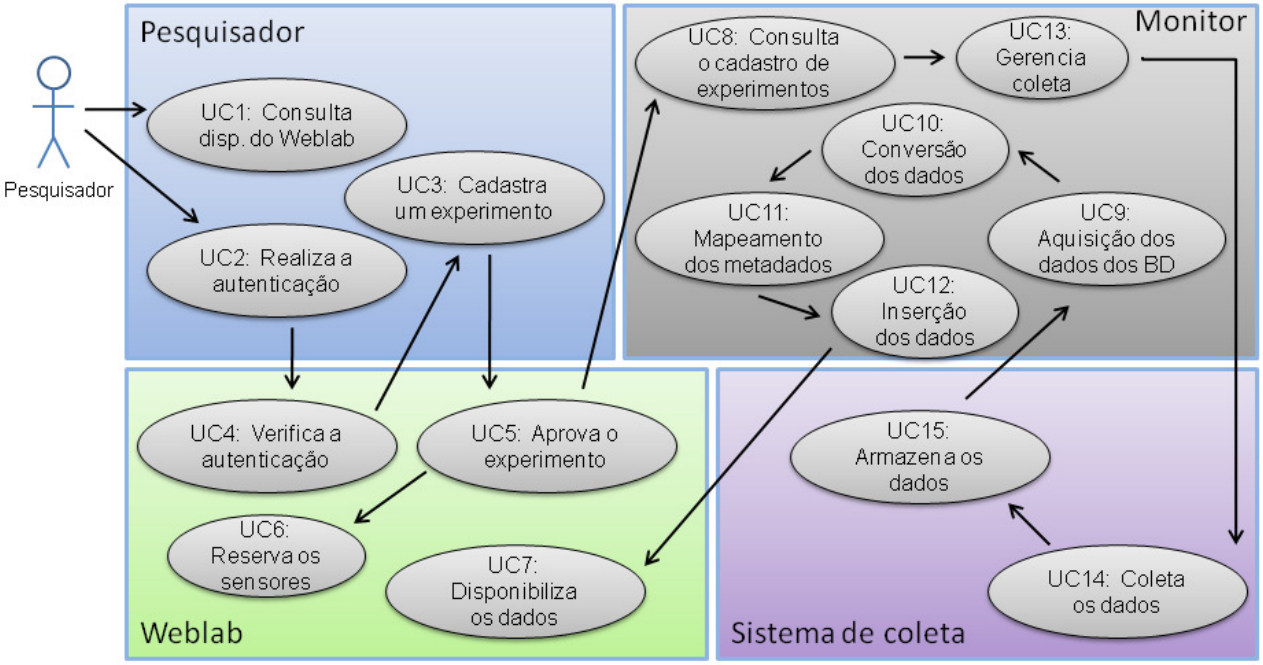

Figura 38: Casos de uso do sistema.

Aplicando o negócio e padrões de integração, obtêm-se a Tabela 5 com a descrição dos casos de uso dos domínios que serão detalhados neste trabalho.

Tabela 5: Casos de uso da caracterização do domínio

Nome do Caso de uso	Descrição	Invocador	Implementador
UC8: Consulta o cadastro de experimentos	Verifica se há novos experimentos cadastrados	Monitor	Weblab
UC9: Realiza a aquisição dos dados no sistema dos sensores	Verifica o período de duração do experimento e realiza a aquisição dos dados coletados pelos sensores	Monitor	Sistema coletor
UC10: Realiza a conversão dos dados	Verifica o formato do dado e se necessário o converte para base decimal	Monitor	Monitor
UC11: Realiza o mapeamento dos metadados	Realiza o mapeamento dos dados para sua inserção no BD do Weblab	Monitor	Monitor
UC12: Insere os dados	Insere os dados no banco de dados central do Weblab	Monitor	Monitor
UC13: Gerencia falhas na coleta	Verifica o status da coleta dos dados, e envia um alerta em caso de falha	Monitor	Monitor
UC14: Coleta os dados no ambiente	Coleta os dados ambientais	Monitor	Sistema coletor
UC15: Armazena os dados no BD	Armazena os dados coletados	Sistema coletor	Sistema coletor

Nas etapas 3 e 4 de análise de subsistemas e alocação dos serviços, com a estrutura de dados montada e preparada para receber os dados dos demais subsistemas, foi necessário a definição da arquitetura do sistema para o Weblab, e na Figura 39 é representada essa arquitetura.

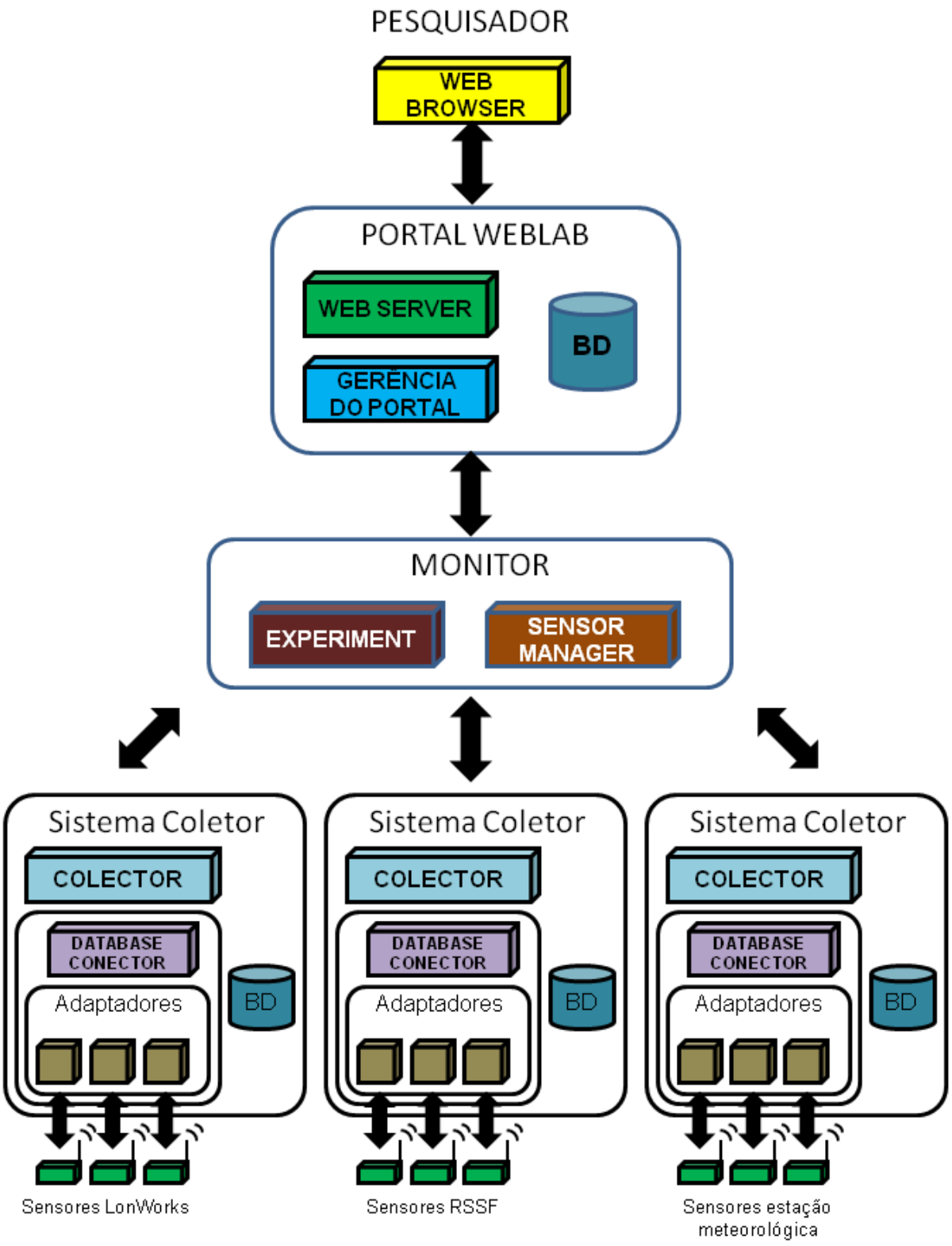

Figura 39: Arquitetura do Weblab de abelhas.

Nesta etapa ocorre a alocação dos serviços, onde os objetivos mapeados em cada componente são identificados, abaixo serão descritos os principais serviços de interesse desse trabalho.

O componente “Web Server” disponibiliza o serviço de experimentos disponíveis que fornece as informações cadastradas de experimentos, os sensores a serem utilizados, o período e o responsável; disponibiliza também o serviço de status do experimento, em que emite um alerta automático, por e-mail ou SMS (*short message service*), para o pesquisador responsável pelo experimento, comunicando que algum sensor deixou de enviar os dados coletados; outro serviço disponibilizado é o de inserção de dados em sua base, para que o Monitor insira os dados adquiridos dos sistemas coletores.

O componente “Gerência do Portal” disponibiliza serviços de cadastro de laboratórios, pesquisadores, experimentos e autenticação de usuários.

O domínio “Monitor” possui dois componentes distintos, o componente “Experiment” fornece serviços para obter informações sobre os sistemas coletores disponíveis; realiza a conexão e aquisição dos dados dos sistemas; converte os dados em formato estabelecido pelo Weblab; realiza o mapeamento dos metadados dos atributos; insere os dados na base central; o componente “Sensor Manager” verifica o status da coleta dos sensores; valida se o dado coletado está dentro de uma faixa aceitável de valor.

O componente “Colector” fornece serviços de consulta aos dados do sistema coletor.

Os componentes “Database Conector” e “Adaptadores” são de especificação e domínio dos fabricantes dos sistemas dos sensores utilizados.

5.4 DEFINIÇÃO DOS METADADOS

O padrão de metadados utilizado foi o Dublin Core, por possuir elementos que condizem com as informações disponíveis no laboratório, sem a necessidade obrigatória de seu preenchimento, visto que os sistemas utilizados nos experimentos não fornecem informações de cadastro de pesquisadores, formas de contato ou descrição do experimento.

O padrão de metadados EML não foi utilizado como padrão no banco de dados, pois é muito detalhado e possui uma complexa estrutura de atributos, sendo muitos deles obrigatórios de preenchimento, o que tornariam a base de dados

central inflexível a experimentos básicos de coleta, como por exemplo, uma coleta de dados de temperatura de uma colmeia.

Levando em consideração os atributos do padrão Dublin Core, o mapeamento dos atributos da base de dados central do Weblab pode ser observado na Figura 40.

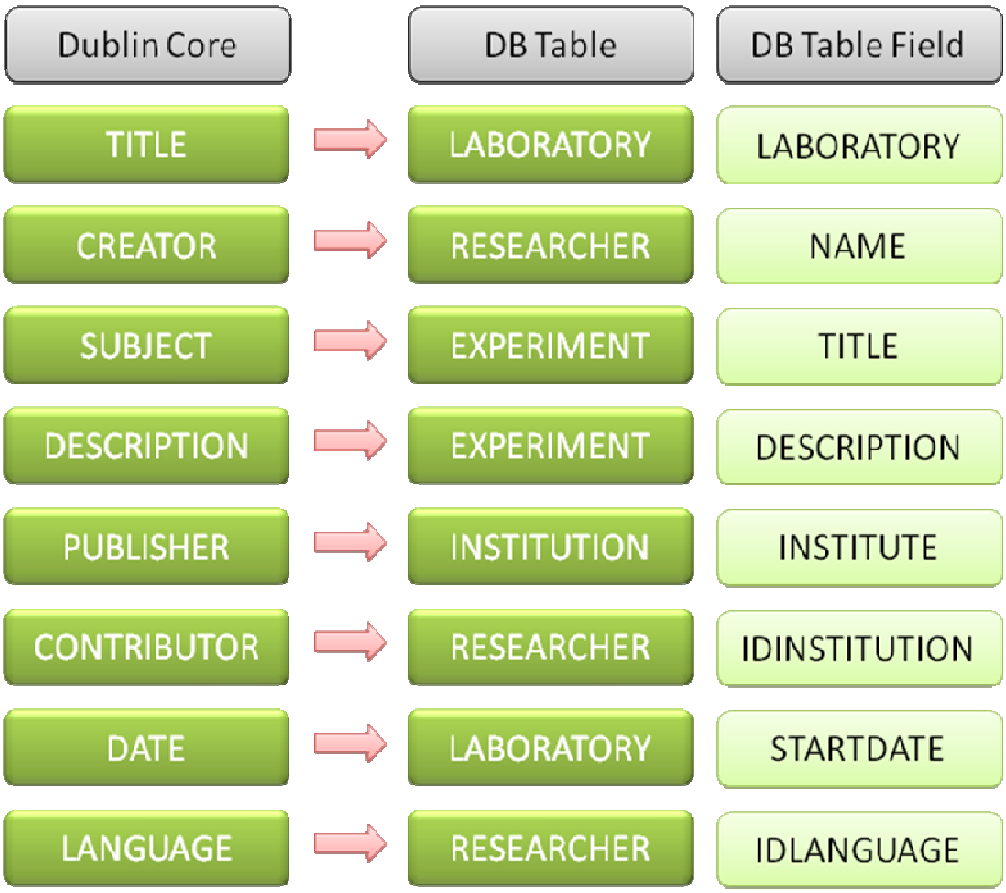

Figura 40: Mapeamento do padrão Dublin Core e BD Weblab.

Os sensores coletam os dados no ambiente onde estão localizados, armazenando em suas bases de dados locais, proprietárias do sistema acoplado no formato que foram projetados. O monitor conecta-se ao serviço disponibilizado no Weblab para inserir os dados no banco de dados central, padronizando os dados com o padrão de metadados definido para então ser disponibilizado no Portal do Weblab de abelhas.

5.5 DESCRIÇÃO DOS SERVIÇOS

Como visto anteriormente, a utilização de arquitetura orientada a serviços facilita a implementação de novas funcionalidades que se tornem necessárias.

A seguir são apresentadas as funcionalidades presentes nos serviços do Monitor *Experiment*:

a) *GetExperimentsList*

O serviço *GetExperimentList* é responsável em disponibilizar ao Monitor a lista de experimentos disponíveis no Weblab.

Este serviço exige um parâmetro informando a situação em que se encontram os experimentos que serão consultados pelo Monitor, este parâmetro pode receber os valores:

- Ativos – Solicita uma lista de todos os experimentos em execução no Weblab;
- Finalizados – Solicita a lista de todos os experimentos que se encontram no estado “finalizado”, e que podem disponibilizar seus dados.

O resultado obtido com este serviço é uma lista de todos os experimentos cadastrados no Weblab, de acordo com o parâmetro que foi passado.

b) *GetColeta*

O serviço *GetColeta* efetua uma requisição das informações provenientes da coleta de dados de um experimento.

Este serviço recebe como parâmetro de entrada o último dado coletado no experimento e retorna como resultado os dados que ainda não foram adquiridos do sistema de coleta.

c) *InsertExperimentData*

O serviço *InsertExperimentData* efetua a inserção dos dados convertidos nos padrões de formato e metadados na base de dados central do Weblab.

Na Figura 41 é apresentado o diagrama de sequência dos serviços do Monitor *Experiment* em um exemplo de coleta de dados.

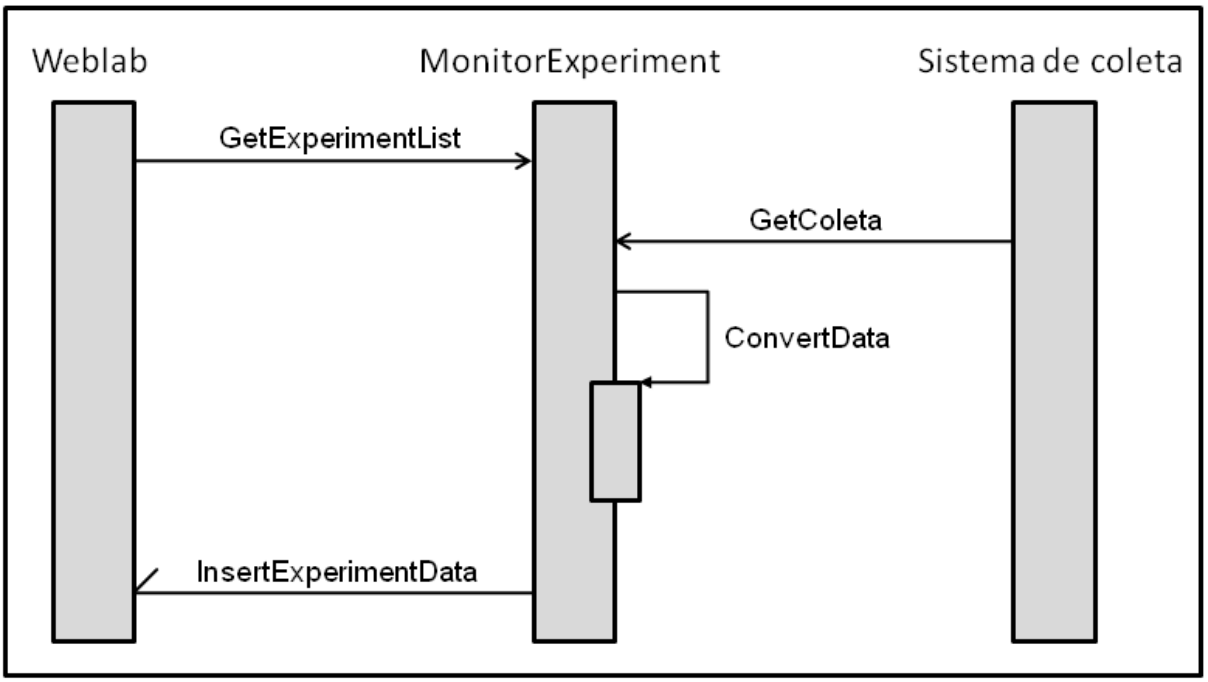

Figura 41: Diagrama de sequência do Monitor *Experiment*.

Descritos os serviços do Monitor *Experiment,* a seguir são apresentadas as funcionalidades presentes nos serviços do Monitor *Sensor Manager*:

a) *GetValidationData*

O serviço *GetValidationData* efetua uma validação dos dados coletados de um experimento para posterior disponibilidade destes dados.

Este serviço recebe como parâmetro o nome do experimento que possui os dados que serão validados antes de serem disponibilizados para análise.

b) *VerifyStatus*

O serviço *VerifyStatus* está em constante consulta no domínio sistema de coleta, verificando o status dos sensores, ou seja, se estão coletando os dados corretamente.

c)*NotifySensorStatus*

O serviço *NotifySensorStatus* notifica ao domínio Weblab o *status* dos sensores, se estão em funcionamento ou se algum sensor não está coletando corretamente, verificando no sistema de coleta os dados coletados pelos

sensores.

d) *NotifySensorFail*

O serviço *NotifySensorFail* notifica ao domínio Weblab uma falha na rede de sensores, informando que sensor está com falhas na coleta.

Na Figura 42 é apresentado o diagrama de sequência dos serviços do Monitor *Experiment e* Monitor *Sensor Manager* em um exemplo de coleta de dados.

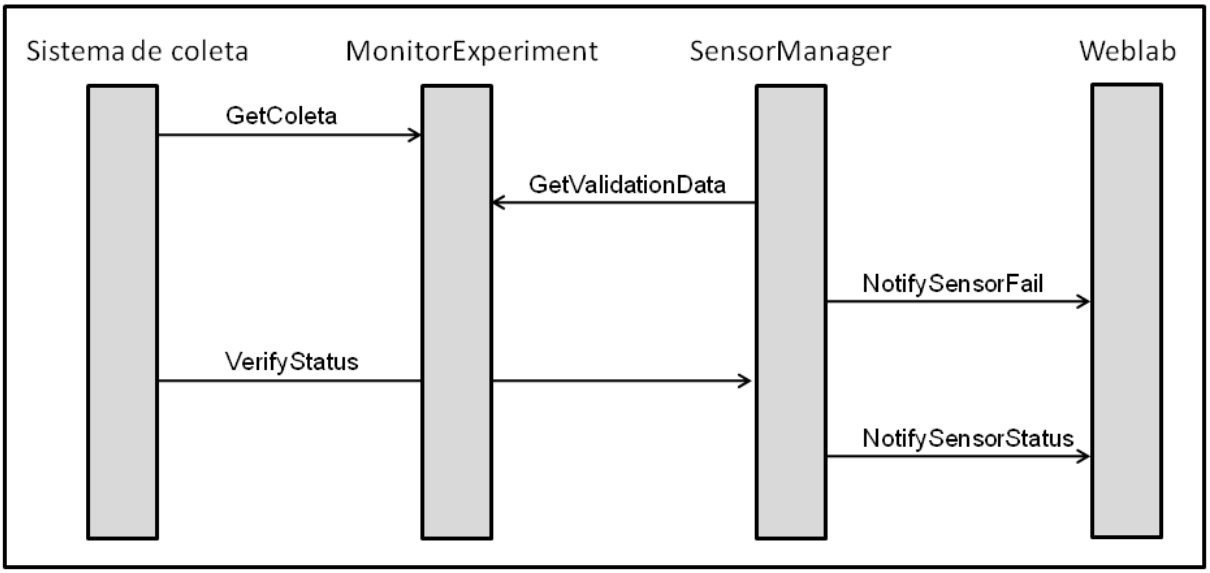

Figura 42: Diagrama de sequência dos serviços do Monitor.

6. CONCLUSÃO

Neste capitulo são apresentados as considerações finais sobre este trabalho com uma retrospectiva discutida nos capítulos anteriores e na sequência opções de trabalhos futuros.

6.1 CONSIDERAÇÕES FINAIS

Com os experimentos realizados em parceria com o Laboratório de Abelhas da Faculdade de Ciências e Letras da USP em Ribeirão Preto, foi possível identificar as principais dificuldades encontradas pelos pesquisadores de abelhas na área de informática, principalmente no tratamento dos dados coletados pelos sensores.

Foram apresentados nos Capítulo 2 e 3 que diferentes tecnologias estão envolvidas ao se realizar experimentos ambientais com abelhas, e com a constante evolução da computação, com o surgimento de novas tecnologias tornou-se necessário definir uma arquitetura mais preparada para incorporar essas inovações.

Além disso, foi apresentado no Capítulo 4 o experimento realizado desde a elaboração da infraestrutura computacional, a disposição dos sensores no ambiente, a coleta dos dados até seu armazenamento e análise, e que foi realizado para entender os processos e identificar como poderiam ser melhorados.

A decisão de propor uma arquitetura de sistemas orientada a serviços foi definida pela necessidade de integrar diferentes bases de dados de um mesmo experimento, a ser consultada e monitorada em um mesmo portal Weblab. Para isso foi necessário definir um esquema lógico de banco de dados mais amplo para suportar futuras alterações, como foi visto no Capítulo 5.

Fez-se necessário também identificar os domínios, componentes e os serviços que seriam disponibilizados pelo servidor, para que uma estação cliente pudesse ter um retorno favorável de sua requisição e que o pesquisador responsável pelo experimento não perdesse seus dados por mau funcionamento de algum sensor ou sistema.

Diferentes experimentos tornam a base de dados do portal mais atrativa para diversos pesquisadores que tenham necessidade de comparar os próprios dados e até mesmo inseri-los dentro da base central, para que outros pesquisadores possam consultar, dentro de um padrão de metadados facilitando sua busca e entendimento.

Apesar de o trabalho ter-se desenvolvido para utilização em experimentos de abelhas, pode-se considerar que com pequenas modificações, é possível sua utilização para outros tipos de experimentos ambientais.

6.2 TRABALHOS FUTUROS

A continuidade do trabalho deve ser realizada por meio da implementação da arquitetura proposta, pela disponibilização de um servidor com os serviços apresentados e a sua utilização por uma estação cliente, para inserir os dados armazenados em diferentes bases de dados em um Weblab.

A realização de um experimento utilizando a arquitetura proposta e nas mesmas características do executado neste trabalho, comparando os valores obtidos e a automatização de todo o processo, é uma sugestão para validar a proposta.

Ainda como trabalho de expansão, uma sugestão seria a busca por novos serviços a serem disponibilizados que auxiliem os pesquisadores a entenderem os dados coletados pelos sensores.

A análise automática dos dados, com geração de gráficos comparativos entre diferentes períodos poderia ser uma implementação a ser realizada.

Outra sugestão seria a de integração com *Data Warehouse*, permitindo assim a aplicação de *Data Mining* que facilitem os pesquisadores na procura de padrões de comportamento dos dados.

A análise dos requisitos não funcionais, como segurança dos dados coletados e a escalabilidade dos sensores na rede devem realizadas a fim de se garantir a constante atualização e funcionamento do Weblab,

6. REFERÊNCIAS BIBLIOGRÁFICAS

AKYILDIZ, I.F.; et al. A Survey on sensor Networks. In: **Communications Magazine, IEEE, USA, august 2002.** (DOI: 10.1109/MCOM.2002.1024422)

ALESHEIKH, A.A.; GHORBANI, M.; MOHAMMADI, H. Design and implementation of sensor metadata on internet. In: **XXth ISPRS Congress, Commission 4, p.1148-1152, 2004.**

ALTINTAS, I. et al., Kepler: an extensible system for design and execution of scientific workflows In: **Proceedings of 16th International Conference on Scientific and Statistical Database Management, June 2004.**

AMÂNCIO, S.M., Redução da taxa de transmissão de vídeo digital por meio da aplicação de segmentação de objetos aplicado ao estudo de abelhas. 82 p. Dissertação de Mestrado. Escola Politécnica da Universidade de São Paulo, 2008.

BENTHEM, V.F.D.J.; IMPERATRIZ-FONSECA, V.L., Biology of the stingless bee p*lebeia remota* (Holmberg): observations and evolutionary implications. *Insectes Sociaux*. 42(1): 71-87. 1995.

BIOABELHA Biodiversidade e uso sustentável dos polinizadores com ênfase em abelhas. Disponível em <http://pitanga.pcs.usp.br/bioabelha> Acessado em 17 de janeiro de 2010.

BIOLCHINI, F. et al. Systematic review in software engineering. Technical Report ES 679/05. In: System **Engineering and Computer Science Departament**. **COPPE, UFRJ, Rio de Janeiro, 2005**.

BIS Biodiversity Information Standards TDWG. Disponível em <http://www.tdwg.org/activities/darwincore> Acessado em 06 de abril de 2010.

BORGES, F.D.B.; BLOCHTEIN, B. Variação sazonal das condições internas de colônias de *Melipona marginata obscurior* (Moure) no Rio Grande do Sul. *Revista Brasileira de Zoologia*, Curitiba, v.23, n. 3, p. 711-714. 2006.

BUENO, J.F. et al. OASIS, Modelo de Referência para Arquitetura Orientada a Serviço 1.0. Escola Politécnica da Universidade de São Paulo, 2006.

CARTOLANO Jr., E.A., et al. Uma proposta de esquema de dados de relacionamento entre espécimes. Pp. 1-8. In: **Anais da XXXIII Conferência Latinoamericana de Informática e Taller Latinoamericano de Informática para la Biodiversidad, San Jose, SJ, Costa Rica, 2007.**

CERPA, A. et al.; Habitat monitoring: Application driver for wireless communications technology. **In: ACM SIGCOMM Workshop on Data Communications in Latin America and the Caribbean, Costa Rica, 2001**.

CERPA, A.; ESTRIN, D.; “ASCENT: Adaptive self-configuring sensor networks topologies”. In: **Proceedings of IEEE INFOCOM, New York, NY, june 2002**.

CHONG, C.; KUMAR, S.P. Sensor networks: evolution, opportunities and challenges. In: **Proceedings of the IEEE, v.91, n.8, august 2003**. (DOI: 10.1109/JPROC.2003.814918)

CORTOPASSI-LAURINO, M. et al. Global meliponiculture: challenges and opportunities**.** *Apidologie* v.37, p. 275–292. 2006.

COSTA, L., VENTURIERI, G.C., Caixas incubadoras para formação de colônias de abelhas sem ferrão (Apidae: Meliponina). *Bioscience Journal*. Uberlândia. V.23 (1): 141-146. 2007.

DAWES, K. N. et al. Sensor management and its application in collaborative environmental research. In: **Fourth IEEE International Conference on eScience. p.143-150. 2008**. (DOI: 10.1109/eScience.2008.27)

DCMI - Dublin Core Metadata Initiative. Disponível em http://dublincore.org/. Acessado em 21 de outubro de 2010.

DEUTSCH C. A. et al. Impacts of climate warming on terrestrial ectotherms across latitude. In: **National Academy of Sciences, v.105, n.18. 2008**.

ECHELON CORP. **Introduction to the LonWorks System**, Echelon Corporation, Palo Alto, USA, 1999.

EML Ecological Metadata Language. Disponível em http://knb.ecoinformatics.org/software/eml Acessado em 06 de abril de 2010.

ENDREI, M. et al. **Patterns: service-oriented architecture and web services**. IBM Red Books, 2004

ENGELS W.; ROSENKRANZ P.; ENGELS E. Thermoregulation in the nest of the neotropical stingless bee Scaptotrigona postica and a hypothesis on the evolution of temperature homeostasis in highly eusocial bees. **Studies on Neotropical Fauna Environment**, v.30, n.4, p. 193–205. 1995.

FERREIRA, M.S.J. Uma arquitetura de sistemas distribuídos para weblabs de serviços ambientais. Dissertação de Mestrado. Escola Politécnica da Universidade de São Paulo, 2007.

FGDC – Federal Geographic Data Committee. Disponível em http://www.fgdc.gov/. Acessado em 21 de outubro de 2010.

GARCIA-ZUBIA, J.; DEL MORAL, A. Suitability and implementation of a weblab in engineering. In: **IEEE Conference on Emerging Technologies and Factory Automation, 2005**. **EFTA , v.2, p.49-56, 2005**. (DOI: 10.1109/ETFA.2005.1612662)

GBIF Global Biodiversity Information Facility. Disponível em <http://www.gbif.org>. Acessado em 06 de dezembro de 2009.

GOBLE, C. et al. myExperiment: social networking for workflow-using e-scientists. In: **WORKS, p. 1-2, Monterey, California, USA, june 2007.**

GUNZER, H. 2002. **Introduction to Web Services**, Março de 2002. Disponível em: <http://bdn.borland.com/java/webtech/0,1418,10018,00.html>. Acessado em 17 de agosto de 2010.

HARDISON, J. et al.; The microelectronics Weblab 6.0 – an implementation using web services and the iLab shared architecture. In: **Proceedings of the International Conference on Engineering Education and Research, março de 2005.**

HEARD, T.A. The role of stingless bees in crop pollination. In: **Annual Review of Entomology, v.44, p.183-206, Australia, january 1999.** (DOI: 10.1146/annurev.ento.44.1.183)

HEIDEMANN, J., GOVINDAN, R. Embedded sensor networks. In: *Handbook of Networked and Embedded Control Systems*, D. Hristu-Varsakelis and W.S. Levine, Editors, Springer Verlag, p.18, 2004. (DOI: 10.1007/0-8176-4404-0_31)

HUNHS M.N., SINGH M.P. Service-oriented computing: key concepts and principles. In: **IEEE Internet Computing, February 2005**. (DOI: 10.1109/MIC.2005.21)

HUNT, C. **TCP/IP network administration**. O'Reilly & Associates, 1998.

HUSTON, M.A., **Biological diversity: the coexistence of species on changing landscapes**. Cambridge University Press, 1994.

IABIN Inter-American Biobiversity Network. Disponível em <http://www.iabin.net>. Acessado em 09 de dezembro 2009.

JANSSEN, J.A.A.J.; MARIS, M.G. Self-configurable distributed control networks on naval ships. In: **Thirteenth International Ship Control Systems Symposium (SCSS). Orlando, Florida, USA, 2003.**

JONES J.C.; OLDROYD, B.P. Nest thermoregulation in social insects. In: **Advances in Insect Physiology, v.33, p.153-191, Australia, 2007.** (DOI: 10.1016/S0065-2806(06)33003-2)

KAKODKAR, A.P. et al., Darwin Core Based Data Streamlining with Digimus 2.0. In: **Council of Scientific and Industrial research (CSIR), Dona Paula, Goa 403 004, India, 2009.**

KIM, T.J. Metadata for geo-spatial data sharing: a comparative analysis. In: **37th Annual Meeting of the Werstern Regional Science Association, p.171-181, Monterey, USA, february, 1998.**

KISS, J. Casa Aquecida: Um sistema de colméia com calefação, mantém a temperatura em 28 graus e permite a criação de abelhas nordestinas no Sul do país. *Globo Rural, v.*245, n.32-37, 2006.

KLEIN, A.M. et al. Importance of pollinators in changing landscapes for world crops. In: **Proceedings of the Royal Society, v.274, n.1608, p.303-313, february, 2007.** (DOI: 10.1098/rspb.2006.3721)

KLEINHENZ M., BUJOK B., FUCHS S., TAUTZ J. Hot bees in empty broodnest cells: heating from within. In: **Journal of Experimental Biology, v.206, p.4217-4231, august, 2003**. (DOI: 10.1242/jeb.00680)

KLUMPP, J. **Australian stingless bees: a guide to sugarbag beekeeping**. Earthiling Enterprises, 2007.

LANDI, C.; LICCARDO, A.; POLESE, N.; Remote laboratory activities to support experimental session for undergraduate measurements courses. In: **Proceedings of the 23rd IEEE Instrumentation and Measurement Technology Conference, IMTC, p.851-856. April, 2006.** (DOI: 10.1109/IMTC.2006.328232)

MAINWARING, A. et al., Wireless sensor networks for habitat monitoring. In: **Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications, pages 88–97. ACM Press, september, 2002.**

MALER E. et al. Extensible markup language (xml) 1.1. Technical report, W3C. Disponível em <http://www.w3.org/TR/2004/REC-xml11-20040204/>. Acessado em 29 de março de 2010.

MARDAN, M., KEVAN, P.G., Critical temperatures for survival of brood and adult workers of the giant honeybee, *Apis dorsata* (Hymenoptera: Apidae). In: **Apidologie v.33, p.295-301. June, 2002.** (DOI: 10.1051/apido:2002017)

MCMULLAN, J.B.; BROWN, M.J.F. Brood pupation temperature affects the susceptibility of honeybees (*Apis mellifera*) to infestation by tracheal mites (*Acarapis woodi*). In: **Apidologie v.36, p.97-105. March, 2005.** (DOI: 10.1051/apido:2004073)

MEESE, R.J.; NERONOV, V.M.; ALESTCHENKO, G.M.; RUGGIERO, M. Rapid acquisition and dissemination of standardize biological inventories from Russian Biosphere Reserves. In: **Biodiversity and Conservation v.12, n.7, p.1421-1429, 2003.** (DOI: 10.1023/A:1023685928666)

MICHENER W.K. et al., Data and Information Management in the Ecological Sciences: A Resource Guide. Disponível em <http://www.ecoinformatics.org/pubs/guide/michener.fv2.htm> Acessado em 03 de outubro de 2010.

NEDIC, Z.; MACHOTKA, J.; NAFALSKI, A. Remote laboratories versus virtual and real laboratories. In: **Proceedings of 33rd Frontiers in Education v.1, p.T3E-1-T3E-6, Australia, november, 2003.** (DOI: 10.1109/FIE.2003.1263343)

NETO, A. V. A influência do aquecimento das colméias sobre o desenvolvimento de colônias de espécies de abelhas sem ferrão (Apidae: Meliponini) durante o período de inverno. Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto – USP, 2008.

NICOLSON, S.W. Water homeostasis in bees, with the emphasis on sociality. In: **The Journal of Experimental Biology n.212, p.429-434. January, 2009.** (DOI: 10.1242/jeb.022343)

NOGUEIRA-NETO, P. **Vida e criação das abelhas indígenas sem ferrão**. Editora Nogueirapis. 1997.

OASIS, Reference Model for Service Oriented Architecture. Disponível em : <http://www.oasisopen.org/committees/tc_home.php?wg_abbrev=soa-rm>. Acessado em 17 de novembro de 2009.

PAPAZOGLOU M.P., SOC, Service-oriented Computing: Concepts, Characteristics and Directions. In: **Proceedings of the 4th International Conference on Web Information System Engineering, 2003**. (DOI: 10.1109/WISE.2003.1254461)

PAPAZOGLOU M.P. et al., Service-oriented Computing: State of the Art and Research Challenges. In: **IEEE Computer Society, 2007**. (DOI: 10.1109/MC.2007.400)

PECKARSKY, B.L.; PENTON, M.A. Effects of enclosures on stream microhabitat and invertebrate community structure. In: **Journal of the North American Benthological Society 9(3), p.249-261. 1990.**

PLINIANCORE Project Plinian Core. Disponível em http://www.pliniancore.org/en/inicio.htm. Acessado em 06 de abril de 2010.

PULIER E., TAYLOR H. **Understanding Enterprise SOA**. Manning Publications Co. 2006.

QUEZADA-EUAN, J.J.G., Biologia y uso de las abejas sin aguijón de la Península de Yucatan, México: Hymenoptera, Meliponini. *Libros científicos. Tratados*; v.16. 2005.

RIBEIRO, M.F.; IMPERATRIZ-FONSECA, V.L.; FILHO, P.S.S., A interrupção da construção de células de cria e postura em *Plebeia remota* (Holmberg) (Hymenoptera, Apidae, Meliponini). In: **G.A.R. MELO & I. ALVES-DOS-SANTOS (Eds), p.177-188, 2003. *Apoidea Neotropica: homenagem aos 90 anos de Jesus Santiago Moure*. Editora UNESC, p.320, Criciúma, 2003.**

SANTANA, F.S. Uma Infraestrutura orientada a serviços para a modelagem de nicho ecológico. Tese de Doutorado, Escola Politécnica da Universidade de São Paulo, 2009.

SARAIVA et al. WebBee – a Web-based information network on bees. *Revista de Engenharia de Computação e Sistemas Digitais*, n.1, v.1, p.77-86, november, 2003.

SALVANHA, P. Modelo para análise de desempenho do processo de replicação de dados em portais de biodiversidade. Dissertação de Mestrado, USP. São Paulo, 2009.

SAVVIDES, A. et al.; Localization in sensor networks source. In: **RAGHAVENDRA, C.S. (Ed.), SIVALINGAM, K. M. (Ed.), ZNATI, T. (Ed.) Wireless Sensor Networks; Springer Science, NY, ISBN:1-4020-7883-8, cap. 15, pp. 327-349, 2005.**

SEEK Science Environment for Ecological Knowledge. Disponível em <http://seek.ecoinformatics.org>. Acessado em 30 de novembro de 2009.

SHAH, R.C.; ROY, S.; JAIN, S.; BRUNETTE, W.; Data MULEs: modeling a three-tier architecture for sparse sensor networks. In: Ad Hoc Networks v.1, p.215-233. August 2003. (DOI: 10.1016/S1570-8705(03)00003-9)

SLAA, E.J.; SANCHEZ, L.A.; SANDI, M.; SALAZAR, W. A scientific note on the use of stingless bees for commercial pollination in enclosures. In: **Apidologie v.31, n.1, p.141-142. January, 2000.** (DOI: 10.1051/apido:2000112)

SOYSAL, O. Computer integrated experimentation in electrical engineering education over distance. In: **Proceedings of ASEE 2000 Annual Conference, Saint Lois, MO, USA, june 2000.**

TAUTZ J. et al. Behavioral performance in adult honey bees is influenced by the temperature experienced during their pupal development. In: **Proceedings of National Academy of Sciences, v.100, n.12, p.7343-7347. USA, june 2003.** (DOI: 10.1073/pnas.1232346100)

THOMAS E. **Service-Oriented Architecture – Concepts, Technology and Design**. Prentice Hall, 1 edition, 2005.

TIDIA – Tecnologia da Informação no Desenvolvimento da Tecnologia Avançada. Apresentação do projeto TIDIA, coordenado pela Fapesp. Disponível em http://www.tidia.fapesp.br/. Acesso em 12 de maio de 2010.

VELTHUIS, H.H.W., KOEDAM, D., IMPERATRIZ-FONSECA, V.L., The rate of brood cell production in the Stingless bee *Melípona bicolor* fluctuates with nest box temperature. *Revista de Etologia* v.2, n.1, p.141-145. 1999.

VENTURIERI, G.C.; RAIOL, V.F.O.; PEREIRA, C.A.B. Avaliação da criação racional de *Melípona fasciculata* (Apidae: Meliponina), entre os agricultores familiares de Bragança, PA, Brasil. *Biota Neotropica*. 2003.

VINCES Virtual Network Center of Ecosystem Services. Disponível em <http://www.ib.usp.br/vinces>. Acessado em 12 de dezembro 2009.

W3C, Working Group Note. Web Services Architecture. Disponível em: <http://www.w3.org/TR/ws-arch>. Acessado em 28 de dezembro de 2009.

WEBBEE Uma rede de Informações sobre diversidade brasileira em abelhas. Disponível em <http://www.webbee.org.br>. Acesso em 13 de janeiro de 2010.

WERNER-ALLEN, G. et al. Firefly – inspired sensor network synchronicity with realistic radio effects. In: **Proceedings of the 3rd international conference on Embedded networked sensor systems, SenSys. New York, USA 2005.** (DOI: 10.1145/1098918.1098934)

WILSON, E.O.; PETER, F.M., **Biodiversity**. National Academies 1988. ISBN/ASIN: 0309037395

WSDL Web Services Description Language (WSDL) 1.1. Disponível em <http://www.w3.org/TR/wsdl>. Acessado em 23 de outubro de 2009.

YAN, Y. et al.; Putting labs online with Web services In: **IT Professional v.8, n.2, p.27-34, março de 2006.** (DOI: 10.1109/MITP.2006.45)

YE, F. et al. PEAS: a robust energy conserving protocol for long-lived sensor networks. In: **International Conference on Distributed Computing Systems (ICDCS), maio 2003.** (DOI: 10.1109/ICDCS.2003.1203449)

ZÜGE, P.V.; AIDAR, D.S. Colméias térmicas Züge-Aidar para meliponíneos (Hymenoptara, Apidae, Meliponinae). *Revista Mensagem Doce*, n.57. 2000.

APÊNDICE A - MODELAGEM DA BASE DE DADOS CENTRAL

O esquema lógico do banco de dados central está descrito conforme pode ser observado abaixo, com as relações e atributos necessários para o funcionamento do Weblab, em que os atributos em negrito e sublinhados são as chaves primárias e os itens sublinhados sem negrito são as chaves estrangeiras.

A relação AccessLevel (**idaccesslevel**, level) armazena os níveis de acesso dos usuários no sistema, como pode ser visto na Figura 43.

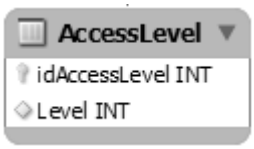

Figura 43: Relação AccessLevel.

Seu único atributo, *level*, trata do nível de acesso de cada usuário, podendo ser “admin” do laboratório, que cria ou remove laboratórios virtuais, adiciona ou remove sensores utilizados, aprova a criação de experimentos. Outra opção de nível de acesso é de “pesquisador” que pode criar experimentos (mediante autorização do administrador) monitorar os seus experimentos e visualizar outros experimentos. O outro nível de acesso é de usuário, que pode visualizar experimentos (mediante autorização do administrador).

A relação City (**idcity**, idstate, city) armazena a relação de cidades cadastradas, na Figura 44.

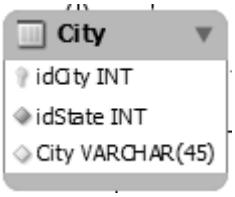

Figura 44: Relação City.

O atributo *city* trata uma lista completa das cidades contidas no estado correspondente.

A relação Country (**idcountry**, country) armazena uma relação de países cadastrados, apresentado na Figura 45.

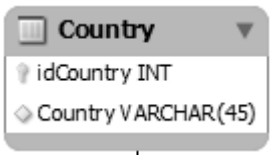

Figura 45: Relação Country.

O atributo *country* trata uma lista completa de países.

A relação Data, apresentada na Figura 46 (**iddata**, idexperiment, idsensor, datetime, accel_x, accel_y, sample, parent, voltage, humid, humtemp, prtemp, press, lux, rawvalue, presentationvalue, description) armazena os dados coletados pelos sensores.

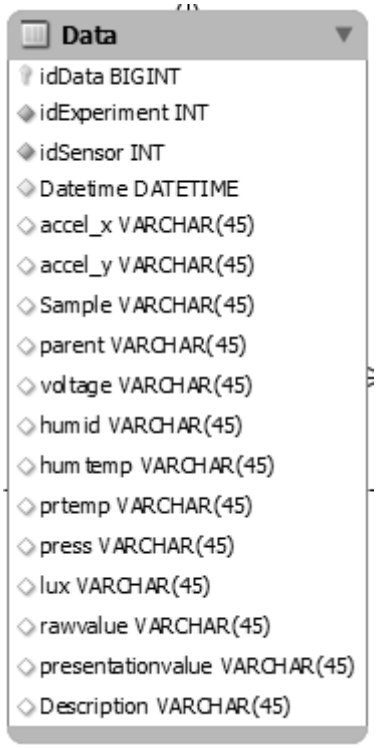

Figura 46: Relação Data.

O atributo *datetime* trata o dia e hora em que o dado foi coletado pelo sensor; os atributos *accel_x* e *accel_y* tratam de dados de aceleração do sensor; o atributo *sample* trata a taxa de amostragem que o sensor foi configurado, ou seja, qual o período de intervalo entre a coleta de dados; o atributo *parent* é mais específico para sensores que enviam seus dados a outros sensores para enviar ao *gateway* (*multihop*), onde fica armazenado o sensor que coletou o dado; o atributo *voltage*

trata da voltagem da bateria dos sensores (os que precisam utilizar); o atributo *humid* armazena os dados coletados pelo sensor de umidade; *humtemp* trata dos dados de umidade do sensor de temperatura, funcionando como uma espécie de *backup*, caso o outro falhe; *prtemp* trata os dados coletados referentes a temperatura; o atributo *press* armazena os dados coletados de pressão atmosférica; *lux* é o atributo que armazena informações sobre luminosidade; *rawvalue* é um atributo que armazena os dados brutos, em estado natural; *presentationvalue* é o atributo que trata os dados de alguma forma já tratados ou convertidos, ocorre principalmente na rede *lonworks*; o atributo *description* é utilizado para alguma descrição opcional que se torne necessária.

A relação Experiment que pode ser visualizada na Figura 47 (**idexperiment**, idresearcher, idlaboratory, title, project, responsible, contact, description, info, startdate, enddate, subject) armazena os dados dos experimentos cadastrados.

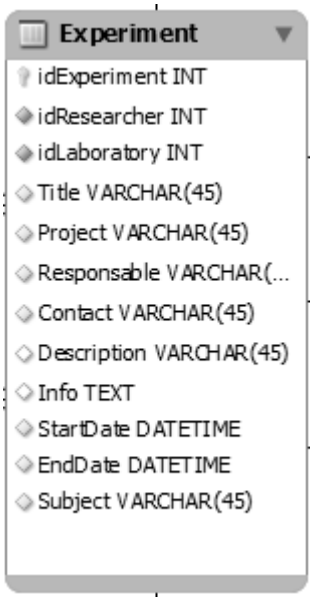

Figura 47**:** Relação Experiment.

O atributo *title* refere-se ao título do experimento; *project* é o atributo que descreve o projeto no qual o experimento está contido; *responsible* é o atributo que armazena o responsável; o atributo *contact* trata o dado referente ao contato do responsável; *description* é uma descrição do experimento; *info* é o atributo para armazenar mais informações sobre o experimento; *startdate* e *enddate* são atributos que armazenam as datas de início e fim do experimento; *subject* é o atributo que trata o assunto ou contexto em que está sendo feito o experimento.

A Figura 48 apresenta a relação ExperimentSensor (**idexperimentsensor**, idsensor, idexperiment) que armazena a relação de sensores utilizados no experimento.

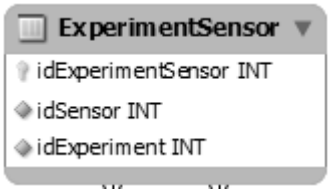

Figura 48: Relação ExperimentSensor.

A relação Institution, mostrada abaixo na Figura 49 (**idinstitution**, idcity, institute) armazena as instituições cadastradas.

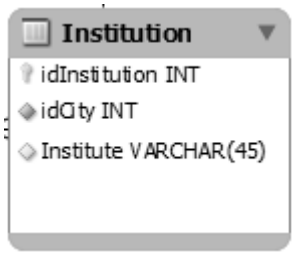

Figura 49: Relação Institution.

O atributo *institute* armazena os nomes das instituições cadastradas.

A relação Keyword (**idkeyword**, idexperiment, keywords) armazena as palavras-chave cadastradas nos experimentos. Ela pode ser vista na Figura 50.

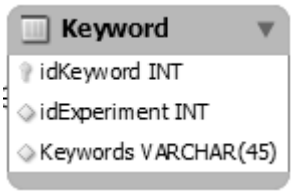

Figura 50: Relação Keyword.

O atributo *keywords* é utilizado para armazenar palavras-chave do experimento.

A Figura 51 apresenta a relação Laboratory (**idlaboratory**, idinstitution, idcity, laboratory, idcoordenator, name, startdate) que armazena as informações sobre os laboratórios.

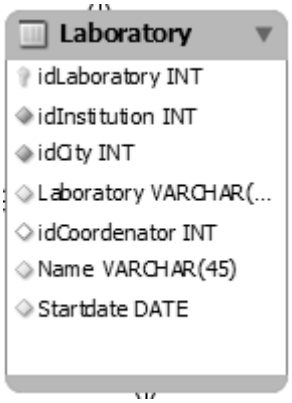

Figura 51**:** Relação Laboratory.

O atributo *laboratory* pode ser utilizado para armazenar informações específicas de cada instituição ao tratar seus laboratórios, através de abreviaturas ou referências; *idcoordenator* é o nome do coordenador do laboratório; o atributo *name* é o nome do laboratório; *startdate* é o atributo que armazena a data de criação do laboratório.

A relação Language (**idLanguage**, language) armazena os diferentes idiomas dos pesquisadores, ela pode ser visualizada na Figura 52.

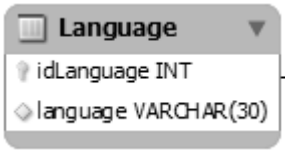

Figura 52**:** Relação Language.

O atributo *language* trata os dados da linguagem utilizada.

A Figura 53 apresenta a relação Researcher (**idresearcher**, idinstitution, idcity, idlanguage, iduser, name, sex, dtbirth, documenttype, documentnumber, graduation, address, email, celular) que armazena os dados cadastrais dos pesquisadores que utilização o Weblab.

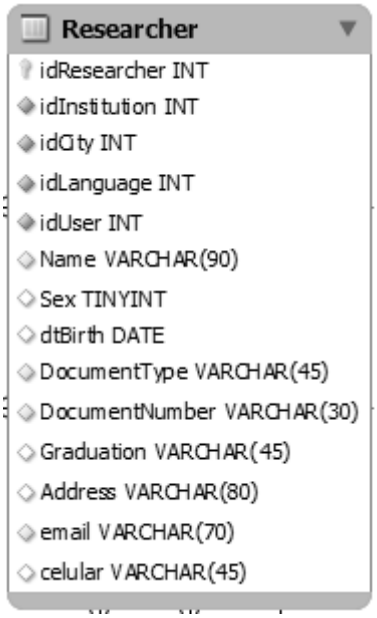

Figura 53**:** Relação Researcher.

O atributo *name* refere-se ao nome do pesquisador; o atributo *sex* trata do gênero, se masculino ou feminino; *dtbirth* armazena a data de nascimento do pesquisador; *documenttype* e *documentnumber* tratam os dados de tipo e número do documento; *graduation* é o atributo que trata da graduação do pesquisador; *address* é o atributo de endereço; o atributo email é para o cadastro do e-mail do pesquisador; celular é o atributo de cadastro do número do celular do pesquisador.

A relação ReseacherExperiment (**idexperimentresearcher**, idresearcher, idexperiment) armazena a relação de experimentos de cada pesquisador, pode ser visualizada na Figura 54.

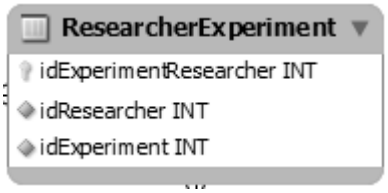

Figura 54**:** Relação ResearcherExperiment.

A Figura 55 apresenta a relação Sensor (**idsensor**, description, id, model, developer, website) que armazena as informações técnicas dos sensores.

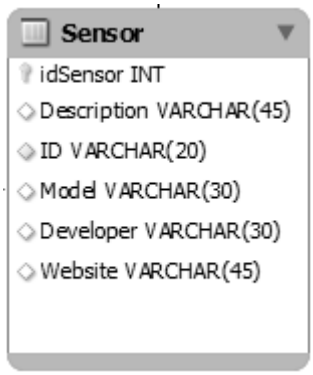

Figura 55**:** Relação Sensor.

O atributo *description* armazena dados sobre a descrição do sensor; *id* é o atributo que trata o *id* que o sensor obtém na rede; o atributo *model* armazena dados do modelo do sensor; *developer* é o atributo que trata o desenvolvedor do sensor; *website* é o atributo que armazena o site da empresa desenvolvedora do sensor.

A relação Specie (**idspecie**, kingdom, phylum, subphylum, class, order, suborder, superfamily, serie, family, subfamily, tribe, genus, subgenus, species, subspecies, commonname) armazena dados cadastrais da espécie a ser pesquisada em um experimento. Ela está representada na Figura 56.

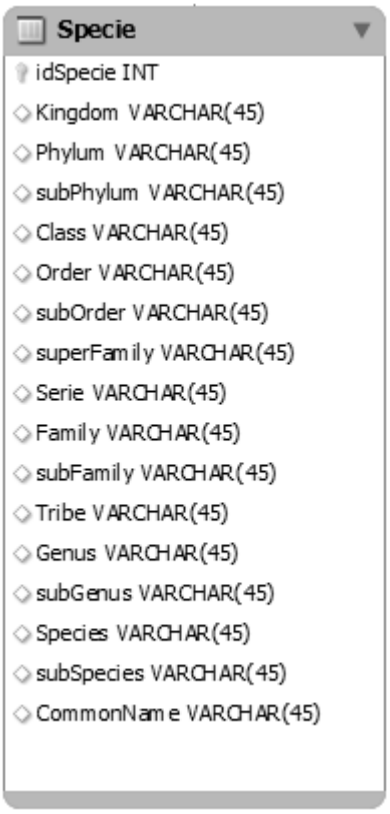

Figura 56**:** Relação Specie.

Os atributos dessa relação são referentes aos dados de taxonomia de uma espécie a ser estudada no laboratório, como reino, filo, subfilo, classe, ordem, subordem, superfamília, serie, família, subfamília, tribo, gênero, subgênero, espécie, subespécie e nome comum.

A Figura 57 apresenta a relação SpecieExperiment (**idspecieexperiment**, idspecie, idexperiment) que armazena a relação de espécies utilizadas em experimentos.

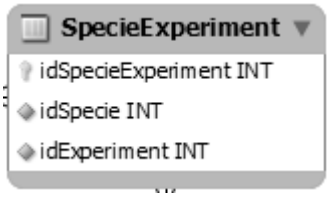

Figura 57**:** Relação SpecieExperiment.

A relação Specimen (**idspecimen,** idcity, idspecie, idspecimengeo, collectiondate, sector, scientificauthorname, identificationdate, authoridentification,

sex, ambient, datum, font, collector, moreinfo) armazena dados de espécimes em estudo dos experimentos no laboratório. Ela está representada na Figura 58.

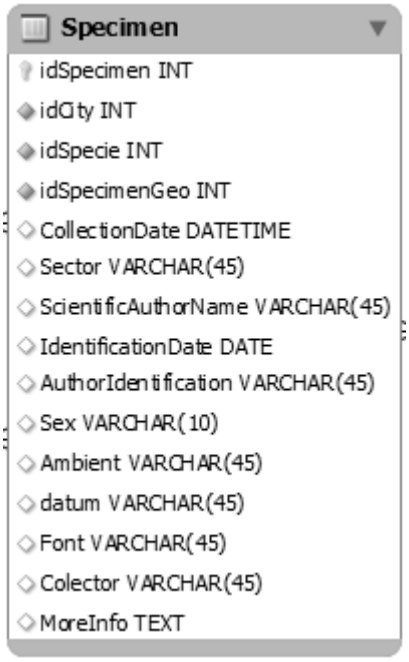

Figura 58**:** Relação Specimen.

O atributo *collectiondate* armazena os dados da data da coleção; *sector* é o setor onde o espécime foi encontrado; *scientificauthorname* trata o dado do autor do nome científico da espécie; o atributo *identificationdate* trata dados da data de identificação; *authoridentification* é o atributo que armazena os dados de autor da identificação; *sex* é o gênero do espécime; *ambient* é o ambiente em que foi encontrado; *datum* é a data da localização; *font* é o atributo que armazena dados de fonte de dados sobre o espécime; o atributo *collector* trata os dados do coletor; *moreinfo* é um atributo adicional para adicionar informações gerais sobre o espécime.

A Figura 59 apresenta a relação SpecimenGeo (**idspecimengeo**, geometry) que armazena dados geoespaciais para análises de localização dos experimentos.

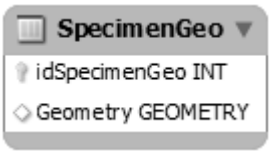

Figura 59**:** Relação SpecimenGeo.

O atributo *geometry* armazena dados de localização geoespacial, para tratar da localização.

A relação State (**idstate**, idcountry, state) armazena a relação de estados cadastrados. Ela está representada na Figura 60.

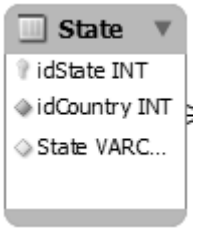

Figura 60: Relação State.

O atributo *state* trata uma lista completa dos estados contidos no país correspondente.

A Figura 61 apresenta a relação User (**iduser**, idaccesslevel, user, password) que armazena os dados de login e acesso dos usuários do sistema.

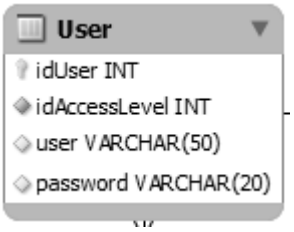

Figura 61: Relação User.

Os atributos *user* e *password* são para armazenar os dados de autenticação do usuário no Weblab.

APÊNDICE B – EXEMPLO DE MENSAGEM SOAP DE UM SERVIÇO

Um exemplo a seguir apresenta uma solicitação e resposta SOAP 1.2 do serviço *GetExperimentList*.

```
POST /Service.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://weblab.com.br/GetExperimentsList"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetExperimentsList xmlns="http://weblab.com.br/" />
  </soap:Body>
</soap:Envelope>
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetExperimentsListResponse xmlns="http://weblab.com.br/">
      <GetExperimentsListResult>string</GetExperimentsListResult>
    </GetExperimentsListResponse>
  </soap:Body>
</soap:Envelope>
```