MARCELO DE QUEIROZ LEITE

**ARQUITETURA DE SISTEMAS ORIENTADA A SERVIÇOS PARA
ANÁLISE DE EXPERIMENTOS AMBIENTAIS EM WEBLABS 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

MARCELO DE QUEIROZ LEITE

**ARQUITETURA DE SISTEMAS ORIENTADA A SERVIÇOS PARA
ANÁLISE DE EXPERIMENTOS AMBIENTAIS EM WEBLABS DE
ABELHAS**

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

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, de setembro de 2011.

Assinatura do autor _____

Assinatura do orientador _____

FICHA CATALOGRÁFICA

Leite, Marcelo Queiroz

Arquitetura de sistemas orientada a serviços para análise de experimentos ambientais em weblabs de abelhas / M.Q. Leite. – ed.rev. -- São Paulo, 2011.

104 p.

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

1. Arquitetura de software 2. Computação 3. Biodiversidade 4. Weblabs I. Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Computação e Sistemas Digitais II. t.

DEDICATÓRIA

Aos meus familiares, principalmente a minha avó e aos professores que ao longo desta jornada sempre incentivaram minha formação acadêmica, profissional e principalmente à de ser humano.

AGRADECIMENTOS

A Deus por estar presente em minha vida e me fortalecer a cada momento de fraqueza.

Ao meu orientador, Prof. Dr. Pedro Luiz Pizzigatti Corrêa, pela paciência, boa vontade e pela confiança em mim depositado, sem a qual, não haveria a possibilidade de execução deste trabalho.

Aos meus familiares, principalmente a minha avó que sempre esteve presente em grande parte dos momentos de minha vida, apoiando e buscando me proporcionar os recursos que me possibilitaram o crescimento pessoal e profissional.

Aos coordenadores do programa em especial ao Prof. Dr. Marco Túlio pelo entusiasmo e sempre disposição para ajudar nas dúvidas do ocorridas durante o programa.

Ao Prof. Dr. Carlos Eduardo Cugnasca e ao Prof. Dr. André Riyuiti Hirakawa, pelo apoio, críticas e sugestões no processo de qualificação e também pelo incentivo neste período de convivência no Laboratório de Automação Agrícola (LAA).

Ao amigo Leandro Halle companheiro de pesquisa e um dos responsáveis pela elaboração e funcionamento do laboratório virtual sem o qual, não seria possível a conclusão deste trabalho.

Ao amigo Edson de Souza responsável pela Infraestrutura do LAA, por sua boa vontade em sempre estar presente auxiliando nos problemas relacionados ao laboratório.

A Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – CAPES, pela criação e regulamentação do programa de Mestrado Interinstitucional, Minter.

A Fundação de Amparo e Pesquisas do Amazonas – FAPEAM, pela concessão da Bolsa de Mestrado ao autor.

A Superintendência da Zona Franca de Manaus – SUFRAMA, pelo apoio financeiro prestado ao programa Minter.

A Escola Politécnica da USP e a Universidade Estadual do Amazonas, que definitivamente, transformaram minha vida.

RESUMO

O processo de elaboração, execução, coleta e análise de dados de experimentos ambientais com abelhas utilizando meios computacionais como suporte a pesquisa, é uma atividade complexa por exigir do pesquisador conhecimentos específicos aos quais ele não foi preparado para atuar. Laboratórios virtuais conhecidos como Weblabs são ferramentas de apoio a pesquisa, pois permitem a realização de experimentos sem a necessidade da presença física da equipe de pesquisadores. O modelo matemático utilizado junto com a sua interpretação são partes do experimento, porém não são tratados dessa maneira atualmente nos Weblabs. Este trabalho propõe a especificação de uma arquitetura de referência a ser empregada como modelo para permitir o desenvolvimento de componentes que disponibilizem serviços com a capacidade de efetuar a integração de dados coletados por meio de experimentos científicos executados com o auxílio dos Weblabs com ferramentas especializadas voltadas a experimentos ambientais e de biodiversidade. A especificação é baseada no paradigma de arquitetura orientada a serviços, permitindo o gerenciamento de análise de experimentos utilizando Weblabs e disponibilizando os serviços especificados para funcionamento em ambientes distribuídos. Este modelo é a base para um estudo de caso com a elaboração de uma prova de conceito, por meio do desenvolvimento de um componente que publique, disponibilize e integre os dados coletados em um Weblab com uma ferramenta especialista em análises ambientais.

Palavras-chave: Computação e Biodiversidade, Weblabs, Computação Orientada a Serviços, Metadados de Experimentos.

ABSTRACT

The preparation, execution, collection and closing of an experiment using computational means to support the research, is a complex activity because it requires specific knowledge of the researcher to whom he was not prepared to act. WebLabs are known support tools to researchers, because they allow performing experiments without the physical presence of the research team. Among the activities undertaken by the researcher, after collecting data, performs the analysis of data, in which mathematical and statistical models are applied to interpret the data. In a Weblab experiments are performed through web in remote laboratories, and also needed to analyze the data obtained with the data collection. The mathematical model used along with their interpretation are events that are part of the experiment, but are not treated that way today in Weblabs. This paper proposes the specification of reference architecture to be used as a template to allow the development of components that provide services with the ability to make the integration of data collected through scientific experiments carried out with the help of weblabs with tools aimed at environmental and biodiversity experiments. The specification is based on the paradigm of service oriented architecture, allowing management analysis using Weblab experiments and providing the services specified for operation in distributed environments. This model is the basis for a case study with the development of a proof of concept, through development of a component that publishes and distributes and integrates the data collected in a Weblab with a special tool in environmental analysis.

Keywords: Computation and Biodiversity, Weblabs, Service Oriented Computing, Experiments Metadata

Às vezes se descobre coisas que são mais importantes para você do que as que você achava que eram.

Steve Wozniak, co-fundador da Apple

LISTA DE FIGURAS

Figura 1 - Ciclo de experimento do myExperiment (MYEXPERIMENT, 2007)	7
Figura 2 - Site do projeto GBIF (GBIF, 2002).....	9
Figura 3 - Site do IABIN (IABIN, 2005)	11
Figura 4 - Arquitetura em camadas do projeto SEEK (apud Ferreira, 2007)	13
Figura 5 - Arquitetura de Weblab proposta para o Projeto VINCES (Ferreira, 2007)	14
Figura 6 - Modelo Conceitual para Weblabs.....	17
Figura 7 - Exemplo de um arquivo baseado no EML (fonte: http://knb.ecoinformatics.org/)	20
Figura 8 - Exemplo de um arquivo baseado no Darwin Core (fonte: http://www.tdwg.org)	21
Figura 9 - Exemplo de um arquivo baseado no Plinian Core (CARTOLANO, 2009)	22
Figura 10 - Exemplo de uso da ferramenta Kepler.	25
Figura 11 - Ambiente da ferramenta Taverna.....	26
Figura 12 - Tela inicial do Weblab Bioabelha. (BioAbelha, 2010)	27
Figura 13 - Biodiversity Analysis Tool (BAT)	29
Figura 14 - Interface da ferramenta MARTT	30
Figura 15 - Colaboração de serviços WEB (ENDREI, 2004).....	33
Figura 16 - Caracterização dos domínios utilizando Casos de Uso.	38
Figura 17 - Fragmento do código WSDL do serviço de disponibilização.....	44
Figura 18 - Arquitetura de Integração Proposta.....	46
Figura 19 – Modelo lógico de dados do Weblab.....	47
Figura 20 - Esquema lógico do banco de dados do Weblab.	48
Figura 21 - Mapeamento da Entidade ResourceGroup	49
Figura 22 - Fragmento de dados da entidade eml-datatable.....	51
Figura 23 - Fragmento do Código EML gerado pelo Componente de Disponibilização	51
Figura 24 - Arquitetura de middleware empregada na proposta.....	53
Figura 25 – Infraestrutura para execução do experimento com abelhas.....	54
Figura 26 - Estrutura de monitoramento interno da colônia de abelhas	56
Figura 28 – Informações sobre o EML gerado no experimento	59
Figura 28 - Interface de configuração da ferramenta Metacat.....	60

Figura 29 - Configurando o Kepler para acesso ao serviço de Disponibilização.....	61
Figura 30 - Efetuando a conexão entre o componente de serviço web e o componente de dados EML	62
Figura 31 - Executando o comando Preview para verificar dados coletados	63
Figura 32 - Componente de configuração de gráfico XY Plotter.....	64
Figura 33 - Gráfico gerado com o uso da ferramenta Kepler.....	64
Figura 34 - Modelo Físico do Banco de dados do Weblab	89

LISTA DE TABELAS

Tabela 1 - Casos de Uso do Domínio de Aquisição	39
Tabela 2 - Casos de Uso do Domínio de Integração	40
Tabela 3 - Casos de Uso do Domínio de Análise.....	40
Tabela 4 - Mapeamento de dados da seção ResourceGroup.....	50
Tabela 5 – Amostra de dados coletados da sala interna	57
Tabela 6 - Fragmento de dados Importados do Weblab.....	59

LISTA DE ABREVIATURAS E SIGLAS

ABCD – *Access to Biological Collections Data*

ABIF – *Australian Biodiversity Information Facility*

BAT – *Biological Analysis Tool*

BIS – *Biodiversity Information Standards*

BSD – *Berkeley Software Distribution*

DCMI – *Dublin Core Metadata Initiative*

DGMS – *Data Grid Management System*

DTD – *Document Type Definition*

DW – *Data Warehouse*

EML – *Ecological Metadata Language*

ESB – *Enterprise Service Bus*

GBIF – *Global Biodiversity Information Facility*

IABIN – *Inter American Biodiversity Information Network*

IP – *Internet Protocol*

JEE – *Java Enterprise Edition*

KAR – *Kepler Archive Format*

KNB – *Knowledge Network for Biocomplexity*

LAA – Laboratório de Automação Agrícola da Escola Politécnica da
Universidade de São Paulo

LDAP – *Lightweight Directory Access Protocol*

NSF – *National Science Foundation*

PTN – *Pollinators Thematic Network*

SEEK – *Science Environmental for Ecological Knowledge*

SGBD – Sistema Gerenciador de Banco de Dados

SGML - *Standard Generalized Markup Language*

SOA – *Service Oriented Architecture*

SOAP – *Simple Object Access Protocol*

SOC – *Service Oriented Computing*

SPM – *Species Profiles Model*

SRB – *Storage Resource Broker*

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

TDWG – *Taxonomic Database Working Group*

TI – *Tecnologia da Informação*

UDDI – *Universal Description, Discovery and Integration*

UML – *Unified Modeling Language*

URI – *Uniform Resource Identifier*

VINCES – *Virtual Network Center of Ecosystem Services*

XML – *Extensible Markup Language*

WSDL – *Web Service Description Language*

WWW – *World Wide Web*

SUMÁRIO

1 INTRODUÇÃO	1
1.1 JUSTIFICATIVA.....	2
1.2 OBJETIVOS	3
1.3 MATERIAL E MÉTODOS	4
1.4 ORGANIZAÇÃO DO TRABALHO	5
2 FUNDAMENTAÇÃO TEÓRICA	6
2.1 ARQUITETURA DE UM EXPERIMENTO	6
2.2 INFORMÁTICA PARA A BIODIVERSIDADE	7
2.2.1 GBIF	8
2.2.2 BIS (TWDG).....	9
2.2.3 IABIN.....	10
2.3 PROJETOS PARA BIODIVERSIDADE	11
2.3.1 PROJETO SEEK.....	11
2.3.2 PROJETO VINCES.....	13
2.4 REDES DE SENSORES	15
2.5 WEBLAB	16
2.6 METADADOS PARA APLICAÇÕES DE BIODIVERSIDADE	18
2.6.1 EML	19
2.6.2 DARWIN CORE	20
2.6.3 ABCD	21
2.6.4 PLINIAN CORE.....	22
2.7 FERRAMENTAS DE SOFTWARE	22
2.7.1 METACAT	23
2.7.2 KEPLER.....	23
2.7.3 TAVERNA.....	25
2.7.4 BIOABELHA.....	26
2.7.5 BIODIVERSITY ANALISYS TOOL (BAT)	27
2.7.6 MARTT.....	29
2.8 ARQUITETURA ORIENTADA A SERVIÇOS	30
2.9 WEB SERVICES	31
2.10 CONSIDERAÇÕES FINAIS	33
3 ARQUITETURA DE REFERÊNCIA PARA ANÁLISE DE EXPERIMENTOS EM	

WEBLABS	35
3.1 VISÃO GERAL	35
3.1.1 REQUISITOS FUNCIONAIS E NÃO FUNCIONAIS.....	36
3.1.2 CARACTERIZAÇÃO DOS DOMÍNIOS	37
3.1.3 DESCRIÇÃO DOS CASOS DE USO.....	39
3.2 DESCRIÇÃO DOS SERVIÇOS.....	40
3.2.1 NOMENCLATURA DOS SERVIÇOS	41
3.2.2 DEFINIÇÃO DOS COMPONENTES.....	44
3.3 MODELO DE INFORMAÇÃO	46
3.3.1 MAPEAMENTO DO CONCEITO DE EXPERIMENTO PARA EML	47
3.3.2 EXEMPLO DE MAPEAMENTO	51
3.4 QUESTÕES DE INTEGRAÇÃO COM O MÓDULO DE ANÁLISE	52
4 ESTUDO DE CASO	54
4.1 IMPLEMENTAÇÃO DA ARQUITETURA	55
4.2 DESCRIÇÃO DO EXPERIMENTO.....	55
4.2.1 CONSIDERAÇÕES SOBRE O EXPERIMENTO	56
4.3 MAPEAMENTO DOS DADOS DO EXPERIMENTO PARA EML	56
4.4 AVALIAÇÃO DOS RESULTADOS COM A FERRAMENTA DE ANÁLISE.....	59
5 CONCLUSÕES	65
5.1 CONTRIBUIÇÕES	65
5.2 AVALIAÇÃO.....	66
5.3 TRABALHOS FUTUROS	66
6 REFERÊNCIAS.....	68
APÊNDICE A.....	75
A.1 CÓDIGO WSDL GERADO NA PROPOSTA	75
A.2 MENSAGENS SOAP (SIMPLE OBJECT ACCESS PROTOCOL) DOS SERVIÇOS DISPONIBILIZADOS COMPONENTE DE DISPONIBILIZAÇÃO	80
GETEXPERIMENTMETADATA.....	80
GETEXPERIMENTSLIST	83
APÊNDICE B.....	86
CÓDIGO EML OBTIDO NA EXECUÇÃO DA FERRAMENTA.....	86
APÊNDICE C	88
MODELO FÍSICO BANCO DE DADOS WEBLAB.....	89

1 INTRODUÇÃO

Segundo Canhos (2003), a situação da biodiversidade neste início do século XXI pode ser caracterizada pela combinação de processos acelerados de destruição de ecossistemas primários, associados a esforços mobilizadores para a conservação e uso sustentável da biodiversidade e de grandes avanços em tecnologia de informação e comunicação de dados. Esta combinação de fatores está conduzindo ao surgimento de uma nova área de desenvolvimento científico e tecnológico, denominada Computação e Biodiversidade (referenciado em Inglês como *Biodiversity Informatics*). Além dos aspectos relacionados ao desenvolvimento de software para a análise e síntese de dados, avanços significativos estão ocorrendo nesta área relacionados a: especificações de metadados utilizados em pesquisas para biodiversidade, cuja utilização facilita o mecanismo de troca de informações dentre variados sistemas, o desenvolvimento de padrões e protocolos para integração de dados distribuídos, fundamentais para a construção da infraestrutura global de informação sobre biodiversidade e o desenvolvimento de novas ferramentas de software específicas para pesquisas em biodiversidade.

As Tecnologias da Informação e Comunicação (TIC) são empregadas como um aliado aos pesquisadores, cientistas e biólogos em suas respectivas áreas de atuação, permitindo o planejamento e execução de experimentos com maiores níveis de detalhamento, e consequentemente obtenção de dados em grandes volumes, porém, ao mesmo tempo em que visa facilitar a elaboração de experimentos, a complexidade, e o nível de conhecimento em áreas referentes à computação, podem tornar-se mais um obstáculo do que um facilitador aos pesquisadores, devido ao fato de que esses assuntos não fazem parte das áreas fundamentais de formação dos pesquisadores que atuam com foco em biodiversidade. Os Weblabs, laboratórios que podem ser acessados remotamente por meio da Web (ZUBIA *et al.*, 2005) permitem que pesquisadores realizem seus experimentos mesmo em locais remotos por meio da Internet. Desta maneira os recursos materiais de um laboratório podem ser compartilhados com vários pesquisadores, mesmo que a distância, durante o período em que o experimento ocorrer, não havendo necessidade da presença “física” das pessoas direta ou indiretamente envolvidas no experimento. Seu uso permite o acesso aos dispositivos integrados ao experimento de uma forma balanceada entre os integrantes

responsáveis pela pesquisa, ou seja, cada um dos pesquisadores tem a capacidade de utilizar a infraestrutura do laboratório nas mesmas condições de qualquer outro pesquisador, desde que ele tenha as permissões necessárias para efetuar estas operações.

Weblabs definem um método padrão para realização de experimentos em locais distintos, permitindo por meio de esta forma reduzir a preocupação dos pesquisadores nos aspectos técnicos relacionados à infraestrutura, coleta e armazenamento dos dados obtidos na pesquisa. Todo este esforço é voltado para que o cientista possa concentrar-se na análise dos dados obtidos por meio do experimento.

Como base para o desenvolvimento deste trabalho, pode-se citar a utilização do Weblab de estudo de polinização por meio de abelhas “sem ferrão” como agentes polinizadores. Este Weblab teve como base o trabalho elaborado por Ferreira & Corrêa (2007), porém enfocando aspectos específicos relacionados à etapa de Análise de Experimentos realizados com abelhas.

1.1 JUSTIFICATIVA

A polinização representa um fator de produção na condução de culturas agrícolas ao redor do mundo e as abelhas como agentes polinizadores dos vegetais; em troca, os vegetais produzem substâncias que atraem as abelhas, as quais levam em seus pelos o pólen dessa planta. O pólen possui fundamental importância para o desenvolvimento da colméia, pois é uma das fontes de proteína das abelhas; sendo assim, ao garantir o alimento de sua espécie, as abelhas também perpetuam a espécie vegetal (SOUZA, EVANGELISTA-RODRIGUES e PINTO, 2007).

De acordo com o estudo de comportamento de abelhas, têm-se em destaque aqueles que utilizam tecnologias de monitoramento com o objetivo de identificar comportamentos em diferentes momentos do dia, considerando variações de temperatura, luminosidade, umidade e até quantificar o número de abelhas de uma colméia.

As abelhas sem ferrão possuem a capacidade de controlar a termorregulação de seus ninhos, determinando dentro de certos limites, condições de temperatura dentro do ninho (NOGUEIRA-NETO, 1997), porém este processo ainda é pouco estudado.

Dentro deste cenário, é efetuado o estudo do comportamento da temperatura nas abelhas sem ferrão e da reação destas à variação da temperatura e umidade do ambiente com maior precisão. Isto permite a construção de uma base de informações que possibilite elaborar previsões das reações destes insetos às mudanças climáticas.

Para este estudo ocorre a necessidade de aplicação de tecnologias de computação, com o intuito de evitar problemas de coleta, armazenamento e análise de dados coletados, podendo ainda dispor os resultados por meio da adoção de métodos padronizados de armazenamento, transferência, e recuperação dos dados acumulados pela base de dados do Weblab.

Apesar da utilização de laboratórios com controle por meio de acesso remoto (Weblabs) ser considerado uma ferramenta de suporte para pesquisadores como também para o ensino, sua utilização tem como principais elementos o controle a distância dos dispositivos integrados ao laboratório, e a coleta de grandes volumes de dados obtidos de sensores, interligados na infraestrutura do Weblab. A utilização das informações coletadas por meio destas ferramentas fica limitada às análises provenientes do próprio Weblab de abelhas, que neste caso, fornece pouco ou nenhum recurso de análise aos dados coletados. Quando disponíveis, estão disponíveis análises mais simples e com poucos níveis de complexidade, apresentando desta forma, resultados com baixa relevância científica.

O estudo do Weblab aplicado a abelhas tem como justificativa o fato do Brasil apresentar a maior diversidade de meliponíneos, conhecidos popularmente como abelhas sem ferrão, e que recebem esse nome pelo fato de possuírem um ferrão vestigial, que não pode ser usado na defesa (WILSON, 1971). São um grupo de abelhas eussociais, que apresentam distribuição pela região tropical e subtropical do planeta (NOGUEIRA-NETO, 1997). Isso facilita a obtenção de parcerias entre laboratórios e pesquisadores dessas áreas específicas já com necessidade de pesquisas e experimentos.

1.2 OBJETIVOS

O objetivo geral deste trabalho consiste na especificação de uma arquitetura baseada em orientação a serviços, para Weblabs, que permita o compartilhamento e análise de experimentos ambientais em larga escala, envolvendo polinizadores.

Os objetivos específicos do trabalho são:

- ✓ Definição de um padrão de metadados para experimentos com polinizadores: por meio desta definição têm-se como objetivo permitir a integração dos dados coletados utilizando padrões baseados em XML para integração com outros sistemas de informação de biodiversidade;
- ✓ Elaboração da arquitetura de referência para serviços de análise: Desenvolver uma arquitetura de referência orientada a serviços (SOA, *Service Oriented Architecture*) que permita a integração de dados obtidos com experimentos executados em ambientes remotos com ferramentas específicas de análise;
- ✓ Realizar um estudo de caso com experimentos em Weblabs de abelhas para validação da proposta: Desenvolver em parceria com pesquisadores do Instituto de Biologia e da USP de Ribeirão Preto a execução da coleta com o uso dos laboratórios virtuais e posteriormente as análises empregando as tecnologias desenvolvidas neste projeto para integração com os módulos de análise.

1.3 MATERIAL E MÉTODOS

Para desenvolvimento deste trabalho, foram feitas revisões de literatura, especificação e implementação de sistemas computacionais detalhados a seguir:

- Revisão de literatura sobre padrões de metadados e integração entre sistemas de informação de biodiversidade;
- Estudo do Weblab de abelhas do projeto Bioabelha (BIOABELHA, 2004), sua base de dados e quais dificuldades para integração de suas informações;
- Identificar os principais requisitos de análise para permitir a integração de experimentos obtidos por meio de Weblabs;
- Desenvolvimento de um protótipo para validação da arquitetura proposta por este trabalho.

Empregou-se o paradigma de arquitetura de sistemas baseado em serviços (*Service Oriented Architecture*) (OASIS, 2006)

Os serviços de análise e integração de experimentos foram projetados tendo como base o modelo definido pelo projeto SEEK (SEEK, 2004) em especial o

Metacat, para integração dos serviços, *Ecological Metadata Language* (EML) como padrão de metadados, e o Componente Kepler (ALTINTAS *et al.*, 2004) para realização das análises.

Para demonstrar a viabilidade desta proposta, foi realizado um estudo de caso iniciado com os experimentos efetuados pelo pesquisador e mestrando em Entomologia Ayrton Vollet da Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto da Universidade de São Paulo (FFCLRP), como também baseado em experimentos apresentados pela Dr. Favízia Freitas de Oliveira do Departamento de Ciências Biológicas, Laboratório de Sistemática de Insetos (LASIS) da Universidade Estadual de Feira de Santana (UEFS). As análises foram realizadas inicialmente sem o uso do sistema de análise, em seguida foram realizadas com o Componente de Análise (Kepler) para verificar a abrangência do sistema em relação aos requisitos de análise.

1.4 ORGANIZAÇÃO DO TRABALHO

A organização deste trabalho é descrita a seguir.

No Capítulo 2 são apresentados a fundamentação teórica de maior relevância ao trabalho, com ênfase nos projetos SEEK e VINCES (VINCES, 2007), apresentam-se conceitos sobre os principais metadados utilizados na área de Informática para a Biodiversidade, por fim é feita uma contextualização de componentes de software empregados em sistemas de biodiversidade com ênfase no projeto Webbee (Webbee, 2001) que é parte do estudo apresentado neste trabalho.

No Capítulo 3 é apresentada uma proposta de uma arquitetura de referência baseada em orientação a serviços, cujo propósito é o de permitir a integração entre os laboratórios virtuais com as ferramentas empregadas em análises voltadas a experimentos ambientais e biodiversidade.

No Capítulo 4 é apresentado um estudo de caso envolvendo um experimento ambiental envolvendo abelhas. Foi utilizado um protótipo desenvolvido a partir da arquitetura para a validação da proposta deste trabalho.

No Capítulo 5 são apresentadas as conclusões, ressaltando os principais resultados obtidos com o trabalho, possíveis melhorias e propostas para continuidade dessa pesquisa.

2 FUNDAMENTAÇÃO TEÓRICA

Este Capítulo apresenta uma revisão sobre os tópicos envolvidos no desenvolvimento deste trabalho, focando nos experimentos e projetos de biodiversidade de maior relevância a esta pesquisa, apresentando as ferramentas utilizadas para as análises de experimentos em biodiversidade e padrões de metadados voltados para pesquisas em biodiversidade.

Dentre as tecnologias computacionais utilizadas no desenvolvimento, são apresentados conceitos referentes à Arquitetura Orientada a Serviços (SOA, *Service Oriented Architecture*) (BUENO *et al.*, 2006) e de tecnologias de serviços Web, que contém os componentes que se adéquam aos requisitos necessários para elaboração, especificação da arquitetura de sistemas proposta.

2.1 ARQUITETURA DE UM EXPERIMENTO

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. A Figura 1 apresenta a arquitetura proposta pelo projeto *myExperiment* (GOBLE *et al.*, 2007). Este projeto apresenta um modelo de referência para definição das etapas a serem executadas em experimentos científicos. A seguir é apresentado um resumo sobre as atividades envolvidas em cada camada:

- Criação do experimento: identificam os objetivos da pesquisa, relação da equipe com os pesquisadores envolvidos com o experimento, definição das hipóteses a serem validadas pelo experimento;
- Personalização: definição da infraestrutura utilizada durante a execução do experimento, definição do período de execução, 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: esta etapa consiste na execução do experimento propriamente dito, operações de coleta e monitoramento dos dados são efetuadas nesta etapa;
- Análise e resultado do experimento: esta etapa tem como principal objetivo utilizar os dados coletados na etapa de execução do experimento e

disponibilizar tais dados com as ferramentas computacionais que possam efetuar cálculos a partir de modelos matemáticos e execução de outros tipos de análises, de acordo com as necessidades do experimento executado;

- Compartilhamento dos resultados obtidos: ao término da execução do experimento e após a execução das análises dos dados obtidos com a pesquisa, deve ser possível disponibilizar estas informações para avaliação, validação ou base de conhecimento a serem utilizadas por outros pesquisadores;
- Reutilização dos dados para novos experimentos: as informações do experimento realizado, agora podem ser novamente utilizadas como parâmetros para elaboração e execução de novos experimentos científicos.

Figura 1 - Ciclo de experimento do myExperiment (MYEXPERIMENT, 2007)

2.2 INFORMÁTICA PARA A BIODIVERSIDADE

A complexidade e a abrangência dos estudos sobre a Biodiversidade exigem

o suporte da Tecnologia da Informação e Comunicação (TIC). O surgimento da área de Computação e Biodiversidade é uma indicação do papel decisivo da tecnologia no apoio às decisões sobre as ações de conservação e uso sustentável da biodiversidade. (SARAIVA *et al.*, 2003) (CORREA, 2005).

Esta seção apresenta algumas iniciativas de projetos que possuem como objetivo facilitar o acesso e a utilização de informações, neste caso, pesquisas e trabalhos relacionados à biodiversidade.

2.2.1 GBIF

Global Biodiversity Information Facility (GBIF, 2002) é uma iniciativa internacional financiada por vários governos focada em produzir e disponibilizar dados sobre biodiversidade para todos que trabalhem com pesquisas científicas, conservação e desenvolvimento sustentável, de acesso livre e gratuito. Na Figura 2 visualiza-se o *site* do GBIF.

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

Figura 2 - Site do projeto GBIF (GBIF, 2002)

2.2.2 BIS (TWDG)

Inicialmente conhecido como *Taxonomic Database Working Group* (TDWG) o *Biodiversity Information Standards* (BIS, 2005) é uma organização sem fins lucrativos e também uma associação científica e educacional, afiliada a União Internacional de Ciências Biológicas (IUBS, 1997).

O TDWG foi criado para estabelecer uma colaboração internacional entre projetos do banco de dados biológicos. TDWG promoveu a mais ampla e eficiente divulgação de informações sobre o patrimônio mundial de organismos biológicos para o benefício do mundo em geral. *Biodiversity Information Standards* (TDWG) trabalha atualmente no desenvolvimento de especificações para a troca de informações sobre dados biológicos ou de biodiversidade.

2.2.3 IABIN

É um fórum especializado (*Inter American Biodiversity Information Network*) (IABIN, 2005) 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 para promover o desenvolvimento sustentável na região. Na Figura 3 pode-se visualizar o portal de acesso as informações da IABIN.

Seus principais objetivos são:

- Desenvolver uma infraestrutura para intercâmbio de informações de biodiversidade;
- Desenvolver a capacidade técnica para troca de informações sobre biodiversidade e o conhecimento sobrepondo as fronteiras políticas, lingüísticas e institucionais;
- Fornecer acesso a informações úteis aos tomadores de decisão (*decision makers*) para aperfeiçoar a conservação da biodiversidade;
- Apoiar na capacidade de armazenamento, utilização e distribuição de dados científicos, mantendo também constante a atualização destas informações;
- Produzir ou adaptar informações para 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 do IABIN (IABIN, 2005)

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 projeto *Science Environment for Ecological Knowledge* (SEEK, 2004), é um sistema projetado não apenas para permitir a aquisição e o armazenamento de dados, mas também para permitir a integração, transformação, análise e síntese de dados de biodiversidade.

A meta do projeto SEEK é a de construção de uma infraestrutura virtual para criar melhorias fundamentais na forma como os pesquisadores podem:

- Ter acesso global a dados e informações ecológicas;

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

A infraestrutura virtual SEEK abrange três sistemas; de acordo com a Figura 4:

- *Ecogrid* – considerada como a “próxima geração” (SEEK, 2004) 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. Esta camada é capaz de permitir a integração de dados provenientes de fontes heterogêneas de informações que serão utilizadas para análises nas camadas superiores.
- *Semantic Mediation System* – um sistema capaz de determinar se dados relevantes e componentes analíticos podem ser automaticamente transformados para uso com o *workflow* selecionado (SEEK, 2004). Esta camada tem a finalidade de permitir uma visão integrada sobre conjuntos de dados ecológicos e de biodiversidade a partir de fontes atualmente incompatíveis. Esta 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 nesta camada para permitir tal acesso uniforme (FERREIRA, 2007).
- *Analysis and Modeling System* – um ambiente visual automatizado (SEEK, 2004) onde ecologistas podem criar modificar e incorporar análises que compõem os fluxos de trabalho e novos modelos. Nesta camada o componente utilizado como apoio para o pesquisador é o Kepler (ALTINTAS, 2004), uma interface visual para desenvolvimento de análises baseados em

dados ecológicos, nas próximas seções serão comentados mais informações a respeito deste componente.

Figura 4 - Arquitetura em camadas do projeto SEEK (apud Ferreira, 2007)

2.3.2 PROJETO VINCES

O projeto *Virtual Network Center of Ecosystem Services* (VINCES, 2007) é um consórcio de laboratórios virtuais (Weblabs) voltados para 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. Este projeto é coordenado pelo Laboratório de Automação Agrícola da Universidade de São Paulo, e entre seus parceiros estão o Laboratório de Abelhas do Instituto de Biociências da USP, a seção de Fisiologia vegetal e Bioquímica do Instituto de Botânica da Secretaria de Estado do Meio Ambiente, grupo ORBIS, coordenado pelo Prof. Dr. James C. Nieh da Universidade da Califórnia, San Diego. Na Figura 5 pode-se visualizar a arquitetura de referência proposta para o projeto VINCES (FERREIRA, 2007).

Figura 5 - Arquitetura de Weblab proposta para o Projeto VINCES (Ferreira, 2007)

O projeto VINCES é por si só um exemplo de trabalho cooperativo desenvolvido por equipes interdisciplinares, e é também um exemplo de pesquisa sobre a aplicação de sistemas e instrumentação baseados na Internet para o controle de experimentos científicos (SARAIVA *et al.*, 2003). O exemplo detalhado abaixo foca no serviço ambiental de polinização, no qual um dos Weblabs do VINCES está direcionado.

O projeto Webbee (WEBBEE, 2001), é uma rede de informação sobre biodiversidade de abelhas, integrado ao Projeto VINCES. Ele pode ser considerado um sistema de informação e um sistema 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 umidade. A atividade de vôo é monitorada com sensores nas entradas das colônias. As condições externas são monitoradas com uma estação meteorológica. Todos os dados obtidos estão

disponíveis em tempo real na Internet. Uma rede de instrumentação inteligente fora desenvolvida para substituição das placas de instrumentação virtuais (baseadas em PC), permitindo desta 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), onde a identificação de abelha pode ser realizada remotamente. Vídeos sobre o comportamento das abelhas são recursos de pesquisa relevantes que precisam se tornar disponíveis na Web para experimentos remotos.

2.4 REDES DE SENSORES

Sensores são dispositivos eletrônicos capazes de monitorar e coletar dados de um ambiente específico. Ao ser integrado a outros sensores forma-se então uma rede de sensores que tem neste momento a capacidade de unir as informações coletadas por cada um dos sensores para monitorar um determinado ambiente ou fenômeno (AQUINO, 2003).

Sensores podem ser vistos como componentes de dimensões reduzidas que combina energia computacional, capacidade de computação sem fio e sensores especializados, estes dispositivos podem ser utilizados para estudar ambientes de difícil acesso ou com algum perigo para os pesquisadores, tais como oceanos e áreas vulcânicas entre outros.

Dentre os equipamentos disponíveis para coleta e armazenamento temporário de dados os sensores são dispositivos que melhor se adequam as necessidades do experimento, por permitir sua utilização em diversos tipos de ambientes e também pelo fato de não causarem impactos visíveis ou funcionais no ambiente utilizado para a execução do experimento.

Atualmente existe uma variedade de tecnologias de sensores disponíveis para os mais diversos tipos de estruturas e ambientes, como sensores de posição e velocidade, sensores ópticos e sensores de deslocamento entre outros, porém para

a configuração do ambiente de experimentação adotado para coleta de dados do laboratório de abelhas foram empregadas tecnologias de sensores com as seguintes características: capacidade de comunicação em redes, reduzidas dimensões e poder de processamento local compatível com o fluxo de informações coletadas.

2.5 WEBLAB

Weblabs são sistemas tradicionalmente projetados por engenheiros para controle de dispositivos eletrônicos; naturalmente estes têm como tendência colocar uma maior atenção ao lado do hardware do sistema (ZUBIA *et al.*, 2005). Um Weblab é um sistema de hardware e software que permite o controle e o acompanhamento de um dispositivo eletrônico programável via Web. Weblabs são usualmente empregados nas universidades para que estudantes possam acessar os recursos dos laboratórios de qualquer lugar por meio de uma conexão com a Internet. (ZUBIA *et al.*, 2005)

Um modelo conceitual de domínio para Weblabs é apresentado na Figura 6. Os elementos que compõem um Weblab bem como as relações entre estes elementos são apresentados segundo a notação UML (*Unified Modeling Language*) (BOOCH *et al.*, 2005). Um participante pode constituir-se de um usuário individual ou também por um grupo de participantes ou usuários. Um Weblab oferece experimentos e a relação de um Weblab para com si próprio, indica que Weblabs podem ser federados para aumentar a gama de experimentos oferecidos e/ou para beneficiar um número maior de usuários.

Figura 6 - Modelo Conceitual para Weblabs

Experimentos com Weblabs são atividades executadas a distância pelos participantes. Experimentos utilizam recursos, tanto físicos (robôs, câmeras, etc.) quanto lógicos (software de simulação, de visualização, etc.), durante a sua realização. Cada participante possui um conjunto de credenciais para utilizar um Weblab.

Credenciais comumente estabelecem, além da identidade do usuário, um conjunto de papéis e permissões. São exemplos de papéis: aluno, instrutor e administrador. Permissões definem atribuições, usualmente associadas a papéis, tais como permissão para cadastrar usuários, para disponibilizar e executar experimentos. A interação do participante com o Weblab é regida por uma ou mais sessões. Uma sessão armazena o estado da interação do participante com o Weblab. Sistemas interativos como Weblabs necessitam de pelo menos três classes de sessões (ILAB, 2006):

- Sessão de Acesso: gerencia o acesso do participante ao Weblab de acordo com suas credenciais e, dependendo do experimento, pode ser efetuado

reserva de horário. A sessão de acesso encerra o experimento por determinação do usuário ou quando o tempo de execução expira;

- Sessão de Interação: gerencia o uso dos recursos oferecidos pelo Weblab para a execução do experimento, tais como robôs, telescópios e sensores;
- Sessão de comunicação: gerencia o uso de recursos que suportam comunicação na presença do participante do Weblab tais como câmeras, microfones e sistemas de comunicação interpessoal.

As três sessões anteriores são implementadas por um conjunto de objetos:

Objetos de acesso são responsáveis pelo gerenciamento de usuários, grupos, credenciais, recursos, experimentos e do próprio Weblab. O controle de acesso, a autenticação de usuários e a reserva de horários para acesso exclusivo também é de responsabilidade destes objetos. As concepções dos objetos de acesso os tornam gerais para serem empregados em qualquer aplicação de Weblab, independentemente de seu domínio.

Objetos de interação suportam a execução remota de experimentos, oferecendo métodos para manipulação, configuração e operação de recursos necessários aos experimentos; seleção de formas de interação, submissão remota de tarefas (scripts, código executável); aquisição remota de dados; e acompanhamento e registro da interação. Os objetos de interação são específicos para cada Weblab.

Objetos de comunicação suportam os diversos estilos de comunicação “um para muitos” tais como comunicação multimídia em tempo real, notificação assíncrona de eventos, comunicação em grupo e difusão de mensagens. De forma semelhante aos objetos de acesso, os objetos de comunicação são gerais e podem ser reusados para a construção de qualquer Weblab.

2.6 METADADOS PARA APLICAÇÕES DE BIODIVERSIDADE

Metadados são informações necessárias para compreensão e utilização eficiente dos dados, e inclui em sua documentação do conjunto, informações do tipo conteúdo, contexto, qualidade, estrutura e acessibilidade. Do ponto de vista da origem dos dados, metadados são necessários para descrever o processamento e análise destas informações. Quando um cientista tem como objetivo a reutilização

de dados coletados por outros (cientistas, pesquisadores), os metadados globais podem ser essenciais para apoiar a identificação e aquisição de dados apropriados, bem como para facilitar seu processamento e análises adicionais. (MICHENER, 1998).

A seguir é apresentada uma breve descrição e especificações de metadados empregados em experimentos voltados a biodiversidade e ecologia.

2.6.1 EML

Padrão de metadados baseado em *extended markup language* (MALER *et al.*, 1998) utilizado para catalogar informações de dados ambientais e ecológicos (*Ecological Metadata Language*) (EML, 1997). É baseado no trabalho realizado pela Sociedade Ecológica dos Estados Unidos e esforços associados (MICHENER *et al.*, 1998). 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 exemplo de documento EML contendo informações sobre o título do experimento e o usuário que o criou é apresentado na Figura 7. Cada módulo EML é projetado para descrever uma parte lógica do metadados completo que deve ser incluído no conjunto de dados ecológicos. O modelo de dados empregado em EML foi definido para atender as necessidades de pesquisadores e comunidades sobre ecologia, beneficiando-se de trabalhos anteriores sobre metadados. Algumas das características do EML são:

- Modularidade: EML foi projetado como um conjunto de módulos ao invés de um padrão extenso, com objetivo de facilitar o crescimento da linguagem no futuro. Ao implementar EML como um modelo extensível, é possível escolher quais módulos são pertinentes para descrever os dados de um domínio de aplicação específico e os recursos do software;
- Estrutura detalhada: EML considera troca de informações com detalhes suficientes para permitir serviços evoluções em termos de processamento de dados por meio da análise de metadados;
- Compatibilidade: EML adota grande parte de sua sintaxe de outras especificações de metadados, que evoluíram por meio da experiência de grupos em outros domínios de aplicação. Sempre que possível EML adiciona árvores inteiras de informação com objetivo de facilitar a conversão de um

documento EML em outras linguagens de metadados;

- Validação de tipos de dados: EML é implementado em linguagem XML, que define as regras de sintaxe do EML. O esquema XML (XML *schema*) é uma recomendação do W3C. Assim um documento de metadados que respeite a sintaxe EML, cumpre também os requisitos definidos no esquema XML para EML.

Atualmente sua especificação encontra-se na versão 2.1.0.

```
<?xml version="1.0"?>
<eml:eml
  packageId="eml.1.1" system="knb"
  xmlns:eml="eml://ecoinformatics.org/eml-2.1.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="eml://ecoinformatics.org/eml-2.1.0 eml.xsd">

  <dataset id="ds.1">
    <title>Sample Dataset Description</title>
    <creator id="23445" scope="document">
      <individualName>
        <surName>Smith</surName>
      </individualName>
    </creator>
    <creator id="23446" scope="document">
      <individualName>
        <surName>Smith</surName>
      </individualName>
    </creator>
    ...
    <contact>
      <references>23446</references>
    </contact>
    <contact>
      <references>23445</references>
    </contact>
  </dataset>
</eml:eml>
```

Figura 7 - Exemplo de um arquivo baseado no EML (fonte: <http://knb.ecoinformatics.org/>)

2.6.2 DARWIN CORE

O Darwin Core (BIS, 2010) foi projetado para facilitar a troca de informações sobre a ocorrência geográfica de organismos e da existência física de amostras em coleções bióticas. O núcleo do Darwin Core e suas extensões são minimamente restritivas, pelo fato de possuírem um numero resumido de itens de conteúdo da informação por projeto, pois isso tornaria inútil a norma para a implementação de ferramentas de qualidade dos dados. Um modelo baseado em Darwin Core pode ser visto na Figura 8.

Segundo Kakodkar *et al.* (2009) O padrão Darwin Core busca facilitar o

intercambio de informações entre robustos bancos de dados distribuídos. Ferramentas de recuperação de dados foram meticulosamente projetados 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 8 - Exemplo de um arquivo baseado no Darwin Core (fonte: <http://www.tdwg.org>)

2.6.3 ABCD

ABCD – *Access Biological Collection Data* (ABCD, 2006) é uma especificação de dados para unidades de coleção biológica, incluindo espécimes preservados e vivos, juntamente com observações de campo que não produziram documentos comprovativos (*voucher*) sobre outros espécimes. Ele destina-se a apoiar o intercâmbio e a integração de coleções primárias de dados detalhados e dados de observação. Todas as coleções biológicas do mundo contêm um número de itens de dados, incluindo dados específicos de espécime (ex. taxonomia, altura, sexo) e coleções específicas (por exemplo, em instituições) de elementos. O conjunto de elementos utilizados varia de coleção para coleção. ABCD fornece um conjunto de nomes de elementos reconciliados e sua definição para que cientistas e professores possam utilizar. Não se espera (mesmo que possível) para qualquer coleção de usar mais do que uma fração dos elementos definidos na especificação.

A meta inicial da especificação de dados do projeto deveria ser abrangente e geral, para incluir uma ampla gama de conceitos que pudessem estar disponíveis em uma coleção de banco de dados, mas para uso apenas o mínimo de elementos necessários para tornar a especificação funcional. ABCD deliberadamente não

abrange dados taxonômicos, tais como sinonímia, além do uso de nomes nas identificações. Da mesma forma, informações relacionadas a taxonomia, tais como áreas de distribuição, indicadores de valores, entre outros, também não estão incluídos. Estes elementos e conceitos são utilizados para fornecer uma maior compatibilidade com outras especificações no campo de coleções de dados biológicos tal como Darwin Core e EML entre outros.

2.6.4 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). Ele também 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 e referências entre outros. 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. O esquema Plinian Core v.2.0 oferece um suporte natural e estruturado ao relacionamento entre espécies. Na Figura 9 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="gisi: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 9 - Exemplo de um arquivo baseado no Plinian Core (CARTOLANO, 2009)

2.7 FERRAMENTAS DE SOFTWARE

Neste tópico, são apresentadas ferramentas de software empregadas em

projetos de biodiversidade, com ênfase nas ferramentas que fazem parte do projeto SEEK, utilizado como modelo de referência para o desenvolvimento desta pesquisa.

2.7.1 METACAT

Segundo Jones *et al.* (2006) o sistema *Metacat* fornece uma base de dados distribuída com facilidades de pesquisa que são configuradas para as necessidades específicas de uma organização. Este conjunto interligado de bases de dados fornece um mecanismo para que os pesquisadores possam compartilhar dados publicamente ou para compartilhar somente com um conjunto limitado de pesquisadores. Na comunidade do GBIF o portal centraliza coleções por meio de um subconjunto federado de dados que é realizado nas coleções. Ambos os recursos permitem novos tipos de estudos colaborativos, que antes eram considerados inacessíveis.

2.7.2 KEPLER

Segundo Ferreira (2007) apud (ALTINTAS, 2004) Kepler é uma aplicação baseada em Java que executa análises envolvendo coleções e dados ecológicos, permitindo a modelagem da sequência de processos utilizada em análises científicas.

Algumas características do Kepler são:

- E baseado no sistema Ptolemy (PTOLEMY, 1999), uma plataforma de apoio para modelos computacionais adequados para tipos distintos de análise (processamento de dados de sensor ou equações diferenciais integrais) (KEPLER, 2006);
- Baseado em Bibliotecas do Sistema R, uma linguagem e ambiente para computação estatística e gráfica. Fornece uma variedade de técnicas estatísticas e gráficas como modelagem linear e não linear, testes estatísticos, análise de séries temporais, classificação, *clustering*, entre outras. (PROGRAMMINGR, 2005).
- Está disponível para uso livre sobre a licença BSD¹;
- Fornece ao usuário uma interface gráfica e um ambiente de tempo de

¹ Licença que deve incluir no código fonte o seguinte reconhecimento: “Este produto inclui software

execução que pode executar *workflows* tanto de dentro da interface gráfica como a partir de linha de comando;

- Seus *workflows* podem ser aninhados, permitindo que tarefas complexas possam ser compostas a partir de componentes simples, permitindo ainda que os *workflows* projetados possam ser reutilizáveis por meio de *sub-workflows* modulares que podem ser guardados e utilizados por aplicações diferentes;
- Workflows desenvolvidos no Kepler podem ser integrados com tecnologias para processamento distribuído (ex. Globus (GLOBUS, 2003), SRB (STFC, 2008) e serviços Web), como aproveitar o suporte nativo do Kepler para processamento paralelo;
- Workflows projetados em Kepler e componentes específicos podem ser salvos, reutilizados e compartilhados com outros usuários usando o formato KAR (*Kepler Archive Format*);
- O Kepler fornece um repositório de componentes por meio de um servidor centralizado onde os componentes e *workflows* podem ser carregados, descarregados, pesquisados e compartilhados com a comunidade ou com usuários selecionados.

Graças a sua flexibilidade da linguagem EML, pode ser utilizado em outras áreas do conhecimento além da Ecoinformática. Em (KEPLER, 2006) são mostrados exemplos de outras áreas beneficiadas com o uso da ferramenta como, por exemplo, outras áreas da Biologia, Geologia, Oceanografia e Química. Na Figura 10 pode ser visualizada a interface da ferramenta com um *workflow* de exemplo carregado.

Figura 10 - Exemplo de uso da ferramenta Kepler.

2.7.3 TAVERNA

O ambiente de gerenciamento de *workflow* Taverna é uma ferramenta baseada em software livre. Foi desenvolvida pelo projeto *myGrid* sendo a ferramenta responsável em realizar os projetos e a execução de workflows científicos. O ambiente foi direcionado principalmente para a área de ciências biológicas, na qual os cientistas não são especialistas em programação e serviços Web (OINN *et al.*, 2006).

A ferramenta permite a automatização dos métodos experimentais por meio da utilização de um número de diferentes serviços (tais como serviços Web) de um conjunto diversificado de domínios - a partir de biologia, química e medicina para a música, meteorologia e ciências sociais. Efetivamente permite que um cientista com pouco conhecimento em computação e poucos recursos técnicos, trabalhe para a produção de análises de alta complexidade por meio de dados de acesso público e privado, a partir de um computador pessoal. A Figura 11 apresenta o ambiente da ferramenta Taverna utilizada na elaboração de um fluxo de trabalho. Neste exemplo foi representando uma sequência de atividades para processamento de imagens que podem ser obtidas por meio de dispositivos eletrônicos como câmeras e

scanners entre outros.

Figura 11 - Ambiente da ferramenta Taverna

2.7.4 BIOABELHA

O projeto temático financiado pela FAPESP chamado BioAbelha desenvolveu um Weblab para estudo de polinizadores. Neste Weblab estão incluídos: uma base de dados de imagens de alta definição e vídeos para estudos taxonômicos e comportamentais em abelhas; um Weblab de monitoramento em tempo real de uma colônia de abelhas, e também material para trabalhos de educação à distância.

Um banco de dados com padrões definidos de vídeos e imagens em alta definição de abelhas foi desenvolvido para ampliar a base de dados disponível no site Webbee (WEBBEE, 2010). Este site contém informações sobre a biodiversidade de abelhas brasileiras em textos e imagens. Seu principal objetivo é gerar informações para pesquisas e estudos sobre a biologia das abelhas, a sistemática e o seu comportamento, podendo ainda ser utilizado para cursos de educação à distância. Seu banco de dados original é restrito a imagens de baixa resolução, uma vez que é suposto ser acessado por uma variedade de usuários com diversos tipos

de conexão à Internet. Na Figura 12 pode-se visualizar a tela inicial de acesso ao Weblab Bioabelha.

Figura 12 - Tela inicial do Weblab Bioabelha. (BioAbelha, 2010)

2.7.5 BIODIVERSITY ANALISYS TOOL (BAT)

É uma ferramenta disponibilizada por meio de serviços Web (*Biodiversity Analysis Tool*) (ABIF, 2009), permite ao usuário realizar pesquisas em base de dados sobre elementos de biodiversidade. Esta ferramenta utiliza grades (*grids*) computacionais pré-definidos para mostrar mapas de riquezas de espécies, endemismo – distribuição de espécies de animais ou vegetais existentes em uma área mais isolada - chamados de diversidade taxonômica. É uma ferramenta baseada na Web que têm como principal objetivo auxiliar os usuários (cientistas, biólogos entre outros) a entender os padrões da distribuição de biodiversidade. Esta ferramenta utiliza registros de observações de espécies publicado por meio da web pelos provedores de dados do projeto GBIF. Possui ainda a capacidade de analisar a distribuição de um determinado grupo de espécies em uma área particular, apresentando ao usuário um mapa contendo as áreas que podem ser significativos

para esse grupo de espécies. A Figura 13 apresenta a tela de seleção de dados da ferramenta. Com o uso desta ferramenta se torna possível:

- Ajudar a identificar áreas biologicamente exclusivas para determinados grupos de espécies;
- Fornecer uma avaliação objetiva da singularidade de uma área para um determinado grupo de espécies, pois o resultado é calculado diretamente a partir de registros de observação, sendo desta forma objetivo e independente de visões de especialistas ou de interesses particulares;
- Tornar seus resultados disponíveis publicamente a todos que possam se interessar (Governos, cientistas, ONGs entre outros), pois utiliza apenas informações disponíveis ao público;
- Permanecer sempre atualizado, como é uma ferramenta executada diretamente a partir de dados disponíveis na Web, com a correção ou adição de novos dados, estes dados serão refletidos nos resultados após a próxima atualização programada.

Australian Biodiversity Info... x

Google

http://www.environment.gov.au/cgi-bin/abif/bat.pl

Personalizar Links Ferramentas de idiomas Twitter / Home Universidade de São ...

Home | Contact us | What's new

ABIF

AUSTRALIAN BIODIVERSITY INFORMATION FACILITY

Biodiversity Analysis Tool

[Run a BAT query](#)

[BAT home](#)

[What is BAT](#)

[How to use BAT](#)

[BAT technical background](#)

[Contact BAT](#)

ABIF links

[ABIF home](#)

GBIF links

[GBIF home](#)

You are here: [ABIF home](#) > [BAT home](#) > BAT query

Biodiversity Analysis Tool - Define Analysis

Try an [example analysis](#) or define your own analysis below.

Data Source: Freshwater Fish - Sao Paulo, Brazil - CRIA

Grid Size: 1 Degree

Class: Osteichthyes

Order: Perciformes

Family: Cichlidae

Analysis Type: ☐ Richness ☐ Endemism ☒ Taxonomic Diversity

Top

Accessibility Disclaimer Privacy © Commonwealth of Australia

Department of the Environment, Water, Heritage and the Arts
GPO Box 787 Canberra ACT 2601 Australia
Telephone: +61 (0)2 6274 1111

Last updated: Friday, 02-Feb-2007 16:38:42 EST

Figura 13 - Biodiversity Analysis Tool (BAT)

2.7.6 MARTT

O Sistema Martt (MARTT, 2009) é uma máquina de aprendizagem baseado em um sistema de marcação semântica que pode ser usado para reformatar documentos de texto de biodiversidade em formato XML para bibliotecas digitais. O sistema permite dois diferentes motores (*engines*) de marcação automática; o primeiro necessita de treinamento e o outro não. Este sistema permite ainda, funções tais como dados de treinamento de preparação e correção de marcação. O MARTTII pode marcar uma descrição taxonômica ao nível de cláusulas, sem qualquer intervenção humana. Espera-se que em futuras versões o software permita marcar no nível de caracteres. Na Figura 14 apresenta a interface de uso da ferramenta empregada para a elaboração de uma seleção de itens dentro de arquivos XML que estejam de acordo com as regras especificadas, neste caso a seleção refere-se a buscas por sementes que se encontram cadastradas no sistema.

Figura 14 - Interface da ferramenta MARTT

2.8 ARQUITETURA ORIENTADA A SERVIÇOS

A Arquitetura Orientada a Serviços (SOA) (OASIS, 2006) é um paradigma para organização e utilização de competências distribuídas que estão sob controle de diferentes domínios proprietários.

Profissionais de Tecnologia da Informação e Comunicação (TIC) têm sido confrontados com o desafio de reduzir os custos maximizando a utilização da tecnologia existente, ao mesmo tempo em que eles têm que se esforçar continuamente para atender melhor aos clientes, serem mais competitivos, e serem mais sensíveis às prioridades estratégicas do negócio em que atuam. Há dois temas subjacentes a estas pressões: Heterogeneidade e mudança. A maioria das empresas hoje contém uma variedade de diferentes sistemas, aplicações e arquiteturas de diferentes idades e tecnologias. Integração de produtos de vários fornecedores e em diferentes plataformas era quase sempre um pesadelo. Mas também não podemos dar ao luxo de ter uma abordagem um único fornecedor de TIC, porque suítes de aplicativos e infraestruturas de apoio são tão inflexíveis (MIN *et al.*, 2004). Soluções baseadas em SOA não requerem uma implementação para a

Web, normalmente uma abordagem melhor para sistemas que têm grandes necessidades de distribuição e tratamento de dados importados de provedores de serviço que usualmente aplicam serviços Web como seu principal padrão de comunicação. 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 chaves para descrever o paradigma SOA. A visibilidade refere-se à capacidade para aqueles com necessidades e aqueles com competências estarem aptos a se verem mutuamente. Isto é tipicamente feito pelo oferecimento de descrições acerca destes 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 (OASIS, 2006).

2.9 WEB SERVICES

Serviços Web é 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 (MALER *et al.*, 2004) determinou a seguinte definição operacional de um serviço da Web:

"Um serviço Web é uma aplicação de software identificado por um URI, cujas interfaces e ligações são capazes de serem definidas, descritas e descobertas como artefatos XML. Um serviço Web permite interações diretas com outros agentes de software baseada em XML usando mensagens trocadas por meio de protocolos baseados na Internet."

Web services = XML + transporte protocolo (como HTTP)

Algumas das principais características dos serviços Web são as seguintes:

- Auto-suficientes: na camada cliente, nenhum software adicional é necessário. Uma linguagem de programação com suporte a XML e um cliente HTTP, já são os requisitos mínimos para começar;
- Auto-descritivos: 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 com a própria mensagem;
- Modularidade: os serviços Web podem ser implementados utilizando tecnologias de *middlewares* orientados a objetos como JEE (INDERJEET *et al.*, 2007), CORBA (KRAFZIG *et al.*, 2004) entre outros são padrões para a execução destes serviços na Web;
- Interoperabilidade e independência de linguagem: a interação entre um fornecedor de serviço e um cliente é projetada para ser completamente independente de plataforma e de linguagem. Esta interação requer um documento WSDL para definir a interface e descrever o serviço, juntamente com um protocolo de rede (geralmente HTTP). A interoperabilidade ocorre pelo fato de que o prestador do serviço e do solicitante do serviço não terem idéia 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. Uma grande parte da tecnologia de serviços Web fora construído por meio de iniciativas de projetos *open source*, portanto a independência de fornecedor e a interoperabilidade são metas realistas alcançadas com esta tecnologia;
- Dinâmicos: por meio das tecnologias UDDI e WSDL o processo de descoberta de serviços pode se tornar um recurso automatizado;
- 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 15 mostra um típico serviço de colaboração Web baseado no modelo SOA

Figura 15 - Colaboração de serviços WEB (ENDREI, 2004)

2.10 CONSIDERAÇÕES FINAIS

Este capítulo apresentou uma variedade de tecnologias empregadas no âmbito da informática para a biodiversidade, sendo que algumas delas estão envolvidas diretamente na elaboração deste projeto. Um dos aspectos de maior relevância das tecnologias descritas consiste no uso de plataformas independentes que se beneficiam de tecnologias de interoperabilidade como é o caso do XML, permitindo assim o desenvolvimento de projetos que tenham como características a

integração destes elementos. No próximo capítulo é apresentado o modelo arquitetural Componente de Integração, descrevendo seus casos de uso, requisitos e apresentando um modelo de referência.

3 ARQUITETURA DE REFERÊNCIA PARA ANÁLISE DE EXPERIMENTOS EM WEBLABS

Neste capítulo é apresentada a arquitetura de sistemas proposta para análise de experimentos ambientais com abelhas, os serviços definidos e o padrão de metadados empregado na proposta.

3.1 VISÃO GERAL

Um modelo de referência não está diretamente relacionado a um padrão, tecnologia ou outro detalhe de implementação concreta (OASIS, 2006). A arquitetura proposta neste trabalho descreve um modelo de referência para análise de experimentos com Weblabs utilizando o paradigma de orientação a serviços.

Por meio do uso da arquitetura orientada a serviços permite-se o reuso das funcionalidades das aplicações envolvidas (serviços). No contexto de uma arquitetura orientada a serviços, estas funcionalidades que antes estariam contidas em programas e sistemas de maneira isolada, agora estão expostas na forma de serviços por meio de interfaces, utilizando tecnologias interoperáveis e padrões abertos. Para definição dos serviços foram seguidas as seguintes etapas (CORREA, 2006):

- a) Levantamento de requisitos: definiu-se uma relação dos requisitos funcionais e não funcionais para análise de experimentos ambientais em Weblabs;
- b) Caracterização dos domínios: definiu-se um modelo baseado em Diagrama UML (*Unified Modeling Language*) de Caso de Uso (BOOCH *et al.*, 2005) para identificar as interações entre as camadas e uma visualização lógica da comunicação entre seus módulos;
- c) Descrição dos serviços: definiram-se as funcionalidades presentes em cada um dos serviços presentes na proposta, e apresentar a descrição empregada na camada juntamente com seus parâmetros exigidos para execução dos serviços disponibilizados;
- d) Definição da arquitetura de referência: definiu-se proposta de arquitetura que atenda os requisitos elaborados no estudo.
- e) Estudo sobre o uso dos metadados: adequaram-se os dados obtidos do Weblab de acordo com as especificações exigidas no metadados empregado

para integração.

3.1.1 REQUISITOS FUNCIONAIS E NÃO FUNCIONAIS

A seguir são relacionados os requisitos funcionais e não funcionais para o domínio de integração:

- Permitir uma análise prévia dos dados originalmente coletados: Efetuar uma validação das informações coletadas antes de disponibilizá-los as ferramentas de análise. Este requisito tem como foco, verificar se o conteúdo das informações disponibilizadas pelo Weblab está dentro dos intervalos válidos para análise antes da geração do EML a ser disponibilizado para a etapa de análise propriamente dita;
- Integração de análise de dados obtidos por diferentes Weblabs: capacidade de permitir que dados destes diferentes Weblabs sejam analisados em conjunto. O domínio de integração tem como recurso a geração de arquivos de metadados contendo as informações coletadas dos Weblabs envolvidos nos experimentos. Na arquitetura proposta pode-se definir que estes dados coletados sejam distribuídos em arquivos XML independentes ou que estes dados sejam consolidados em um único arquivo XML, e em sequência, disponibilizando este conteúdo ao domínio de análise;
- Publicação dos experimentos: permitir que os dados coletados possam ser compartilhados por pesquisadores que se encontrem em outros Weblabs, desde que tenham autorização para executar esta solicitação;
- Permitir que os dados coletados sejam acessados por diferentes ferramentas de análise: dispor os dados do experimento realizado por meio do uso de metadados, para que estes possam ser processados com as ferramentas específicas de análise;
- Disponibilizar serviços de autenticação e autorização: no qual os pesquisadores do Weblab possam ter perfis de acesso bem definidos, dessa forma, permitindo o acesso aos experimentos em que estes pesquisadores estejam autorizados, definindo também qual o perfil de acesso que este pesquisador possui aos dados que serão disponibilizados para análise;

Os requisitos citados foram considerados para a especificação da arquitetura do WebLab apresentados por Ferreira (2007) que enfoca a arquitetura para elaboração de experimentos remotos, cuja ênfase encontra-se nas funcionalidades de gerenciamento do Weblab e também no controle dos dispositivos integrados a este Weblab. Esta pesquisa está direcionada a complementar as funcionalidades propostas, focando em serviços específicos de análise de experimentos.

3.1.2 CARACTERIZAÇÃO DOS DOMÍNIOS

Baseado nos estudos de arquiteturas de Weblabs; definidas por Ferreira (2007) e Leite (2010), foram identificados 3 domínios distintos em Weblabs, de acordo com a Figura 16.

- a) Domínio de aquisição: consiste no domínio que abrange os itens diretamente gerenciados pelo Weblab, neste caso efetuam-se o controle dos dispositivos empregados para coleta de dados (redes de sensores), das atividades relacionadas ao monitoramento das informações obtidas, do controle de permissão de acesso dos pesquisadores autorizados a efetuar mudanças de configurações nos dispositivos de coleta ou outros parâmetros pertinentes ao experimento;
- b) Domínio de integração: finalizado as atividades executadas pelo Domínio de Aquisição, observa-se a necessidade de interoperabilidade entre Weblabs para permitir que os experimentos possam ser acessados por ferramentas em análises de dados. Neste domínio estas informações obtidas de um ou mais Weblabs podem ser disponibilizados por meio de metadados ao Domínio de Análise, permitindo assim o uso destes dados por ferramentas de análises específicas de experimentos ambientais e de biodiversidade. Este domínio possui também como funcionalidade efetuar o controle de acesso aos dados dos experimentos aos quais os pesquisadores estejam previamente autorizados;
- c) Domínio de análise: ao receber os dados do experimento gerados no Domínio de Integração utilizando um padrão de metadados, pode-se agora disponibilizar estas informações para serem processadas por ferramentas de análise em Biodiversidade ou ambientais, ou seja, efetuar o trabalho de análise desses experimentos empregando modelos matemáticos definidos

pelos pesquisadores. Estas análises podem ser feita de forma individual utilizando os dados obtidos de apenas um Weblab, ou em conjunto efetuando análises consolidadas dos dados coletados dos Weblabs associados ao experimento executado.

A execução da análise destes dados é efetuada pelo Componente Kepler (seção 2.7.2). O uso do Kepler foi definido pelo fato deste componente possuir módulos que permitem acesso direto aos dados disponibilizados pela Componente de Integração – desde que o pesquisador possua autorização - com os modelos matemáticos elaborados pelos pesquisadores, como também permite seu uso por meio de acesso aos dados disponibilizados nos bancos de dados do *Metacat*. O modelo arquitetural proposto neste trabalho não se limita ao uso de uma ferramenta específica, para permitir a análise com outras ferramentas, devem-se definir as especificações de como os dados devem ser transferidos e a partir deste ponto, adicionar um novo componente para prover este serviço dentro do Domínio de Integração.

Figura 16 - Caracterização dos domínios utilizando Casos de Uso.

3.1.3 DESCRIÇÃO DOS CASOS DE USO

Definidos os domínios que compõem a arquitetura proposta neste projeto, são descritos a seguir, os casos de uso que definem as funcionalidades existentes em cada um destes domínios, apresentando em seguida, um breve resumo de suas características.

Domínio de Aquisição: Responsável pelo gerenciamento, configuração e a execução da coleta dos dados referentes a um experimento elaborado pela equipe de pesquisadores. Na Tabela 1 são apresentados os Casos de Uso definidos para este domínio:

Tabela 1 - Casos de Uso do Domínio de Aquisição

Domínio	Caso de Uso	Descrição
Aquisição	Controle de sensores	Englobam todos os elementos referentes à camada de acesso aos dispositivos físicos, redes de sensores, tendo como tarefas essenciais o manuseio e o controle de cada um destes dispositivos integrados ao laboratório.
	Gerência de sensores	O objetivo deste caso de uso consiste no gerenciamento e controle das informações coletadas por meio da rede de sensores em execução, dentre estas atividades tem se o armazenamento em banco de dados destas informações.
	Gerenciar experimento	Permitir o controle e gerenciamento da aquisição de dados, período de coleta, pesquisadores envolvidos com a pesquisa.

Domínio de Integração: representam os serviços responsáveis por disponibilizar os dados para as ferramentas específicas de análise, seus principais objetivos consistem na validação dos dados obtidos e na geração de metadados de acordo com a ferramenta de análise a ser utilizada. Na Tabela 2 são definidos os Casos de Uso deste domínio:

Tabela 2 - Casos de Uso do Domínio de Integração

Domínio	Caso de Uso	Descrição
Integração	Validação dos dados	Possui o objetivo de efetuar uma análise prévia dos dados fornecidos pelo experimento a serem disponibilizados ao serviço de integração de EML, sua interface de uso publicada neste serviço possibilita que outras fontes de dados possam se beneficiar da tecnologia, permitindo assim a sua integração com as ferramentas de análise em uso.
	Geração EML	Efetua o mapeamento e a criação do EML que permite a integração destes dados com o <i>Metacat</i> da arquitetura SEEK (Seção 2.3.1) ou diretamente com a ferramenta Kepler por meio dos componentes de conexão disponíveis nesta ferramenta.

Domínio de Análise: responsável pelos serviços de análise dos dados coletados no experimento. Este domínio acessa um serviço disponível pelo Domínio de Integração para acesso aos dados do experimento. Na Tabela 3 são apresentados os Casos de Uso definidos para este Domínio:

Tabela 3 - Casos de Uso do Domínio de Análise

Domínio	Caso de Uso	Descrição
Análise	Conexão Metacat	Componente responsável em efetuar a integração com a camada de acesso aos dados da arquitetura SEEK, permitindo a integração de diversos tipos de fontes de dados para execução de suas respectivas análises
	Análise Matemática	Efetuar a execução da análise definida na configuração do experimento pela ferramenta especialista.

3.2 DESCRIÇÃO DOS SERVIÇOS

Neste item, são apresentados os serviços definidos para o domínio de integração. Os serviços foram definidos de acordo com as seguintes características:

- Validação de acesso: permite o acesso do pesquisador aos dados coletados

pelo experimento encerrado no Weblab, em que ele se encontra fazendo parte da equipe de pesquisa;

- Relação de experimentos: disponibilizam a aplicação que interage com o pesquisador os experimentos em execução no Weblab, permitindo assim sua análise com as ferramentas apropriadas;
- Validação dos dados do experimento: o conjunto de dados obtidos pela coleta deve ser validado antes de serem disponibilizados para análise;
- Verificação de experimentos finalizados: análises dos experimentos cuja aquisição de dados foi encerrada;
- Disponibilização dos dados coletados: permite publicar os dados coletados pelo Weblab utilizando padrões de metadados baseados em XML como o EML (*Ecological Metadata Language*), compartilhando os dados coletados por meio do *Metacat* ou realizar análises com o componente Kepler.

3.2.1 NOMENCLATURA DOS SERVIÇOS

O modelo proposto define um grupo de funcionalidades que permite ao pesquisador desde a verificação do estado do experimento, até a disponibilidade destes dados em formato EML.

A seguir um breve resumo das funcionalidades presentes no Componente de Disponibilização dos dados coletados:

a) *GetExperimentsList*

O serviço *GetExperimentsList* é responsável por disponibilizar ao consumidor do serviço – neste caso a aplicação de interface com o pesquisador - a lista de experimentos disponíveis no Weblab.

Este serviço recebe como parâmetro de entrada um valor que define a situação em que se encontram os experimentos que serão consultados pelo pesquisador, este parâmetro pode possuir os seguintes 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 para serem integrados com o Componente de Análise.

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) *GetExperimentStatus*

O serviço *GetExperimentStatus* lista o estado em que se encontra o experimento solicitado pelo pesquisador. Diferente do serviço anterior, necessita do nome do experimento para consulta de seu estado.

Este serviço recebe como parâmetro de entrada um experimento em execução no Weblab, e retorna como resultado o estado em que ele se encontra podendo ser:

- Ativo – O experimento se encontra em execução no Weblab;
- Finalizado – O experimento finalizou o processo de coleta de dados.

c) *GetAuthorizationProfile*

O serviço *GetAuthorizationProfile* retorna o nível de autorização que o pesquisador possui nos experimentos cadastrados no Weblab.

Este serviço recebe como parâmetro de entrada um código do pesquisador e o do experimento a ser consultado, retornando como resultado o nível de autorização que o pesquisador possui neste experimento.

d) *GetValidationData*

O serviço *GetValidationData* efetua a validação dos dados coletados de um experimento que se encontra no estado “finalizado”.

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

e) *GetExperimentMetadata*

O serviço *GetExperimentMetadata* usando um padrão de metadados efetua uma requisição do experimento solicitado e em estado “finalizado”. Com estes dados disponíveis em EML, podem ser acessadas pelo Componente Kepler diretamente ou utilizando o Metacat para acesso a experimentos remotos.

Este serviço recebe como parâmetro o nome de um experimento que já foi

validado pelo serviço *GetValidationData* e disponibiliza um arquivo de dados no formato EML.

f) *GetSelectedExperiments*

O serviço *GetSelectedExperiments* possui as mesmas características do serviço *GetExperimentMetadata*, porém, com a capacidade de seleção de dados provenientes de mais de um Weblab, permitindo ao pesquisador a execução de análises envolvendo vários experimentos. Esse serviço viabiliza a realização de experimentos distribuídos geograficamente em vários Weblabs.

Este serviço recebe como parâmetros de entrada uma lista contendo os Weblabs e os seus respectivos experimentos encerrados, retornando como resultado desta operação os metadados EML, de cada experimento.

Na Figura 17 pode-se verificar um fragmento do código WSDL gerado, contendo as interfaces que permitem a integração com os serviços disponibilizados. Nesta figura são descritos os serviços publicados pelo Componente de Disponibilização.

```

<?xml version="1.0" encoding="utf-8" ?>

- <wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
  xmlns:tns="http://weblab.com.br/" xmlns:s="http://www.w3.org/2001/XMLSchema"
  xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
  xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
  targetNamespace="http://weblab.com.br/"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">

- <wsdl:types>
- <s:schema elementFormDefault="qualified" targetNamespace="http://weblab.com.br/">
- <s:element name="GetExperimentsList">
  <s:complexType />
</s:element>
- <s:element name="GetExperimentsListResponse">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="GetExperimentsListResult" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>
- <s:element name="GetValidationData">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="experimentname" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>
- <s:element name="GetValidationDataResponse">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="GetValidationDataResult" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>
- <s:element name="GetExperimentEMLFormat">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="experimentname" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>
- <s:element name="GetExperimentEMLFormatResponse">
- <s:complexType>
- <s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="GetExperimentEMLFormatResult"
    type="s:string" />
</s:sequence>
</s:complexType>

```

Figura 17 - Fragmento do código WSDL do serviço de disponibilização.

3.2.2 DEFINIÇÃO DOS COMPONENTES

No modelo apresentado na Figura 18, a utilização do paradigma de serviços considera a autonomia dos domínios, integrados por meio de serviços. Isso torna facilitada a integração de novos domínios, desde que considerem os serviços especificados. A seguir são apresentados os componentes que realizam os serviços da arquitetura de integração proposta:

- Componentes de Aquisição – disponibilizam serviços responsáveis pela aquisição dos dados de arquiteturas de redes heterogêneas de sensores, na Figura 18 representada pelos componentes “Redes de Sensores” (os serviços desse componente não são abordados nesse trabalho);
- Componente de Gerenciamento do experimento (Weblab) – responsável pelo gerenciamento dos dados obtidos pelos componentes de Aquisição, controle de acesso aos dados, monitoramento de falhas durante a aquisição, alocação dos sensores envolvidos no experimento e período de aquisição dos dados do experimento;
- Componente de Disponibilização – após o termino das atividades de coleta de dados do experimento, estes dados estão disponíveis para análise. Esta atividade é executada por meio deste componente, cuja responsabilidade neste momento é de integrar as informações obtidas com as ferramentas de análise de dados;
- Componente de Disponibilização de Metadados – para integração dos dados coletados e validados com a ferramenta de análise deve ser solicitado o uso do serviço de geração do arquivo EML (Seção 2.6.1). Este é o formato empregado pelo componente de acesso aos dados pelo Componente *Metacat*;
- Componente *Metacat* – Serviço com a funcionalidade de permitir o compartilhamento de dados distribuídos para serem utilizados por outros pesquisadores e ferramentas de análise.

Figura 18 - Arquitetura de Integração Proposta

3.3 MODELO DE INFORMAÇÃO

Como parte da arquitetura proposta, torna-se necessário especificar o modelo de informação de um experimento. O modelo de informação tem como o objetivo viabilizar a interoperabilidade de dados entre os diferentes Weblabs e assim definir um padrão de metadados para o compartilhamento de dados. Para isto elaborou-se o mapeamento dos dados gerados no banco de dados do Weblab para um padrão de metadados. A Figura 19 apresenta os principais conceitos de um experimento, relacionando os dispositivos para aquisição de dados como: vídeo, áudio, sensores e informações fornecidas manualmente pelos pesquisadores, além da infraestrutura disponível (laboratório) e responsáveis pelo experimento (pesquisadores).

Figura 19 – Modelo lógico de dados do Weblab

3.3.1 MAPEAMENTO DO CONCEITO DE EXPERIMENTO PARA EML

O mapeamento de dados consiste em permitir mapear o esquema do banco de dados coletados com o esquema de metadados EML. Isto é necessário para que os dados possam ser compartilhados pelo Componente de Disponibilização ao *Metacat* e também ser manipulado pelo Componente de Análise *Kepler*. A elaboração deste mapeamento foi feito por meio de uma análise do esquema lógico do banco de dados do Weblab com o metadados adotado, o EML. A Figura 20 apresenta o esquema lógico relacional empregado para o mapeamento.

Figura 20 - Esquema lógico do banco de dados do Weblab.

Para mapeamento dos dados foram utilizados os seguintes módulos EML:

a. EML-Dataset:

O módulo EML-Dataset contém informações que descrevem cada uma das características dos recursos de um conjunto de dados (*dataset*), permitindo ainda que possam ser importadas informações de outros módulos, que podem ser utilizados para descrever estes conjuntos de dados com um maior nível de detalhamento das informações (EML, 1997). Nas especificações do EML estas informações encontram-se na seção *ResourceGroup*, que permite descrever informações sobre a equipe de pesquisadores e sobre o experimento realizado. Dados como título, criador, contatos, resumo sobre o experimento são adicionados nesta seção *ResourceGroup*. Outra característica é a de possuir também informações sobre as entidades de dados, neste caso o nome do

arquivo que contem as informações que são disponibilizadas em conjunto com o EML, na Figura 21 é apresentado os atributos desta entidade e na Tabela 4 são apresentados o mapeamento dos atributos que foram efetuados para elaboração desta entidade:

Figura 21 - Mapeamento da Entidade ResourceGroup

Tabela 4 - Mapeamento de dados da seção *ResourceGroup*

Seção EML	Tabela no Banco de Dados	Campo
<title>	Experimentos	Nome
<creator>	Pesquisadores	Nome
<abstract>	Experimentos	Objetivos
<keyword>	Experimentos	Palavras_chave
<additional Info>	Experimentos	Observações
<intellectualRights>	Pesquisadores	Nome
<distribution>	Experimentos	Projeto
<pubDate>	Experimentos	Data_inicio

b. EML-Datatable

No módulo EML-DataTable são descritas as características lógicas de cada conjunto de informações (*dataset*). Um conjunto de dados em arquivos texto separados por delimitadores pode ser considerado como um *dataset*, e cada um destes arquivos podem posteriormente ser considerados uma entidade *datatable* de um *dataset*. Desde que o módulo EML-Datatable estenda o módulo EML-Entity, ele utiliza todos os elementos comuns a entidade para descrever os dados, juntamente com alguns elementos específicos para as entidades *datatable*. O módulo EML-Datatable contém a descrição dos atributos (coluna / campo / variável) da tabela de dados por meio da utilização do módulo EML-Attribute. Da mesma forma, existem campos utilizados para descrever a distribuição física da tabela de dados, a sua cobertura global, o método utilizado na criação dos dados e informações de outras estruturas lógicas, como a sua orientação, maiúsculas e minúsculas, entre outros. Na Figura 22 pode se visualizar um fragmento da entidade EML-Datatable.

Data Table:						
Name:		Datos Meteorologicos				
Description:		Dtos Estacion meteorologica La Hechicera para e? 2001				
Physical Structure Description:						
Object Name:		sample.dat				
Size:		188860 bytes				
Character Encoding:		ASCII				
Text Format:		Number of Header Lines:		1		
		Record Delimiter:		\n		
		Maximum Record Length:		column		
		Simple Delimited:		Field Delimeter: ,		
Case Sensitive?		no				
Number Of Records:		100				
Attribute(s) Info:						
Attribute Name	Column Label	Definition	Type of Value	Measurement Type	Measurement Domain	Missing Value
DATE	DATE	Date of collection	string	datetime	Format MM/DD/YY Precision 1	
TIME	TIME	Time of collection	string	datetime	Format HH:MM Precision 1	
T_AIR	T_AIR	Air temperature	float	interval	Unit celsius Precision 1 Type real	
RH	RH	Relative Humidity	int	ratio	Unit dimensionless Precision 1 Type integer	
DEW	DEW	Dew point	float	interval	Unit celsius Precision 1 Type real	
BARO	BARO	Barometric preasure	float	interval	Unit millibar Precision 1 Type real	
					Unit degree	

Figura 22 - Fragmento de dados da entidade eml-datable

3.3.2 EXEMPLO DE MAPEAMENTO

Na Figura 23 é apresentado um fragmento do código EML obtido com o uso do Componente de Integração.

```
<?xml version="1.0" encoding="utf-8" ?>
- <eml:eml packageId="eml.2.1" system="knb" xmlns:eml="eml://ecoinformatics.org/eml-2.1.0"
  xmlns:xsl="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ds="eml://ecoinformatics.org/dataset-2.1.0"
  xsl:schemaLocation="eml://ecoinformatics.org/eml-2.1.0 eml.xsd">
- <access authSystem="ldap://ldap.ecoinformatics.org:389/dc=ecoinformatics,dc=org"
  order="AllowFirst">
- <allow>
  <principal>uid=mleite,o=unaffiliated,dc=ecoinformatics,dc=org</principal>
  <permission>read</permission>
  <permission>write</permission>
- <dataset>
  <title>Estudo de Polinizadores</title>
  - <creator>
    - <individualName>
      <surname>Vollet</surname>
    </individualName>
  </creator>
  - <contact>
    - <individualName>
      <surname>Ayrton</surname>
    </individualName>
  </contact>
</dataset>
- <dataTable>
  <entityName>dados_lonworks</entityName>
```

Figura 23 - Fragmento do Código EML gerado pelo Componente de Disponibilização

3.4 QUESTÕES DE INTEGRAÇÃO COM O MÓDULO DE ANÁLISE

O Componente de Integração responsável pela conexão entre o Weblab e o Componente de Análise tem como características os seguintes requisitos:

- ✓ Acesso ao banco de dados contendo as informações obtidas por meio da rede de sensores integradas ao Weblab;
- ✓ Acesso ao banco de dados do Weblab, responsável pelo controle de execução do experimento, período de coleta de dados, equipe responsável, entre outros;
- ✓ Regras de avaliação de dados, a serem executadas antes da elaboração dos dados em formato de metadados;
- ✓ Integração com o módulo de geração da linguagem ecológica de metadados (EML).

Esta integração ocorre por meio do Componente *Metacat*, sendo este o elemento de integração que possibilita o compartilhamento dos dados para as componentes de análise. Os dados são representados no padrão de metadados EML gerados pelo Componente de Disponibilização.

O mapeamento de tecnologias de *middleware* que efetuam a comunicação com os serviços presentes na proposta baseiam-se em protocolos amplamente empregados na Internet, este acesso deve ser efetuado pelo pesquisador por meio de uma ferramenta de acesso Web. Toda a operação da ferramenta Weblab utiliza-se do protocolo HTTP para permitir a configuração e controle de seus componentes, o uso deste protocolo também é responsável pelo acesso do Componente de Disponibilização aos dados coletados e disponíveis no banco de dados gerenciado pela ferramenta de Weblab, a comunicação do Componente Kepler com o Componente de Disponibilização baseia-se na utilização de serviços Web que provê as funcionalidades necessárias para que o Domínio de Análise possa solicitar as pesquisas a serem analisadas dentro da ferramenta, estes dados são transferidos em formato EML (Seção 2.6.1). Com o modelo proposto nesta arquitetura os acessos aos dados do Weblab podem ser feitos de forma direta pela ferramenta de análise como também por meio do *Metacat* que provê integração com bases de dados contendo as informações coletadas pelo Weblab. A Figura 24 apresenta este mapeamento de tecnologias que efetuam a comunicação entre os serviços.

Figura 24 - Arquitetura de *middleware* empregada na proposta

4 ESTUDO DE CASO

Neste Capítulo é apresentado um estudo de caso com objetivo de validar a arquitetura e as funcionalidades propostas. Para o estudo de caso foi considerado o Weblab Bioabelha (Seção 2.7.4). Este estudo foi realizado em cooperação com a Profa. Dra. Vera Lucia Imperatriz Fonseca e seu orientado de mestrado Airtton Vollet do Departamento de Biologia da Universidade de São Paulo, Campus de Ribeirão Preto. Foram coletados dados de monitoramento de variáveis ambientais em experimentos com abelhas e utilizados para análise.

A execução da análise foi realizada empregando o Componente Kepler (Seção 2.5.2), no qual foi elaborado um modelo matemático envolvendo os dados provenientes do Weblab para processamento e análise. O uso da arquitetura de referência proposta por este trabalho é demonstrado pela elaboração de um serviço Web que fornece os serviços propostos e é utilizado para integração do Weblab com a ferramenta de análise, por meio do uso do padrão de metadados EML. A Figura 25 representa a infraestrutura de sensores e de rede empregada em sua execução.

Figura 25 – Infraestrutura para execução do experimento com abelhas.

4.1 IMPLEMENTAÇÃO DA ARQUITETURA

O modelo proposto na Seção 3.2.2 (Definição dos Componentes) apresenta os módulos divididos de acordo com a caracterização dos domínios definidos na Seção 3.1.2, na qual se observa que a coleta de dados é uma atividade diretamente ligada ao gerenciamento do experimento, tarefa efetuada pelo Weblab, ambos utilizando sistemas de gerenciamento de banco de dados (SGBD) para armazenamento das informações. Com o encerramento da etapa de aquisição de dados, os dados coletados estão disponíveis ao Componente de Disponibilização serviço Web que permite a disponibilidade destes dados ao Componente Kepler. Ao ser requisitado os dados coletados de um experimento, o Componente de Disponibilização efetua uma validação dos dados coletados verificando-se, por exemplo, se os dados coletados pelo Weblab encontram-se dentro de intervalos estabelecidos na definição do experimento. Ao término desta validação o Componente de Disponibilização efetua a geração de um arquivo em formato EML de acordo com as especificações definidas na Seção 3.3.1 (Mapeamento do conceito de experimento para EML), este conteúdo pode ser disponibilizado a ferramenta Kepler por meio de seu componente de acesso a serviços Web. Na etapa de análise estas informações estão disponíveis para pesquisadores envolvidos com o experimento.

4.2 DESCRIÇÃO DO EXPERIMENTO

Um dos estudos efetuados pelo laboratório de abelhas na Universidade de São Paulo – campus Ribeirão Preto – utilizou o ambiente de coleta de dados por meio de uma infraestrutura de redes de sensores, baseadas em tecnologia *lonworks*, responsável pela coleta de informações na execução do experimento. A Figura 26 mostra a infraestrutura de monitoramento nas colméias; Nos círculos em destaque pode se observar os sensores sem fio (Lonworks) modelo MPR2400-MICAZ que foram adicionados para monitoramento de temperatura e de umidade no experimento.

Figura 26 - Estrutura de monitoramento interno da colônia de abelhas

4.2.1 CONSIDERAÇÕES SOBRE O EXPERIMENTO

A termorregulação consiste em um mecanismo presente nas abelhas sociais que permite sua adaptação e sobrevivência em diferentes ambientes climáticos. A distribuição geográfica das abelhas sem ferrão apresenta uma diversidade de questões para a sobrevivência sobre diferentes regiões climáticas. O experimento efetuado pelos pesquisadores do Instituto de Biologia em conjunto com a equipe de Ribeirão Preto, tem como propósito estudar o mecanismo de termorregulação das abelhas *Melípona quadrifasciata*, o foco é avaliar esta espécie quanto aos mecanismos de sobrevivência em relação a altas temperaturas e também diferentes umidades relativas de ar. Para a execução do experimento foram empregados sensores eletrônicos para monitoramento sobre como se comporta a temperatura e a umidade interna das regiões analisadas, região do favo, potes de alimentos e na entrada e saída da colméia.

O objetivo do experimento foi avaliar as hipóteses relacionadas com a termorregulação desta espécie, evidenciando-se questões comportamentais sob condições de temperaturas elevadas.

4.3 MAPEAMENTO DOS DADOS DO EXPERIMENTO PARA EML

Para validação da proposta, foi elaborado um roteiro para o experimento no qual é demonstrado o funcionamento de um dos serviços propostos na arquitetura. Para execução da análise foi efetuado uma seleção de dados obtidos no período de

10 dias de coleta de informações do experimento gerando uma base de dados de aproximadamente quinze mil registros. Estes dados foram disponibilizados pela Componente de Disponibilização ao Componente de Análises Kepler, responsável pela execução de análises matemáticas e demonstração por meio de gráficos do resultado obtido.

A Tabela 5 apresenta parte dos dados coletados pela rede *Lonworks* (Seção 2.5) como data, horário, temperatura ambientes e dados de umidade de cada uma das colméias monitoradas no experimento.

Tabela 5 – Amostra de dados coletados da sala interna

Período		Estação Meteorológica		RSSF – Nó 102		RSSF – Nó 202		Rede LON – Nose	
Data	Hora	Temp.	Umid.	Temp.	Umid.	Temp.	Umid.	Temp.	Umid.
05/05/2009	1:10 PM	24,2	56	27,3	49	27,4	49	27,1	44
05/05/2009	1:11 PM	24,2	54	27,4	48	27,4	48	27,1	43
05/05/2009	1:12 PM	24,2	52	27,4	47	27,3	48	27,1	41
05/05/2009	1:13 PM	24,3	50	27,4	47	27,3	47	27,0	40
05/05/2009	1:14 PM	24,3	49	27,4	47	27,3	47	27,0	41
05/05/2009	1:15 PM	24,3	49	27,5	48	27,3	48	27,1	42
05/05/2009	1:16 PM	24,3	49	27,5	48	27,3	48	27,1	40
05/05/2009	1:17 PM	24,3	48	27,5	48	27,2	48	27,1	42
05/05/2009	1:18 PM	24,3	48	27,5	48	27,2	48	27,1	42

Os dados de temperatura e umidade podem ser mapeados em EML utilizando as regras de mapeamento definidas no Capítulo 3, seção 3.3.1. A Figura 27 apresenta o arquivo EML gerado pelo Componente de Disponibilização:

Data Set Citation		
Vollet .Estudo sobre termorregulação de abelhas.		
mleite.5.3		
Data Tables, Images, and Other Entities:		
Metadata download:	Ecological Metadata Language (EML) File	
Data Table:		
Name:	dados_lonworks	
Description:	Dados obtidos da rede lonwork	
Online Distribution Info:		
Download File:	ecogrid://knb/mleite.3.3	
Physical Structure Description:		
Object Name:	Unnamed-table-6592582516504223340.tmp	
Text Format:	Maximum	column
	Record Length:	
	Simple	Field
	Delimited:	Delimiter: ,
Attribute(s) Info:		
	Attribute Description (label and definition)	Type of Value Measurement Type and Domain

data_time data_time - Data e horário de coleta de dados nos sensores		integer	Format YYYY-MM-DD hh:mm:ss Precision 1 minute	dateTime
Umidade Umidade - Dados de umidade		integer	Unit humidity Precision 1 Type natural Min 0 Max 100	ratio
Temperatura Temperatura - Temperatura coletada nas colmeias		float	Unit celsius Precision 0.1 Type natural	ratio
Data Set Owner(s):				
Individual:	Vollet			
Position:	Ayrton			
Abstract:				
Estudo elaborado para monitoramento da termorregulação das abelhas <i>Melípona quadrifasciata</i>				
Keywords:				
	<ul style="list-style-type: none">• weblab• metadados• informatica para biodiversidade			
Temporal Coverage:				
Date:	2009-05-09			
Contact:				
Individual:	Vollet			
Position:	Avrton			

additionalMetadata

```

|__element 'metadata'
|  |__element 'unitList'
|  |  |__element 'unitType'
|  |  |  |__attribute 'id' = 'humidity'
|  |  |  |__attribute 'name' = 'humidity'
|  |  |  |__element 'dimension'
|  |  |  |  |__attribute 'name' = 'Length'
|  |  |  |__text '\n '
|  |  |__text '\n '
|  |  |__element 'unit'
|  |  |  |__attribute 'id' = 'humidity'
|  |  |  |__attribute 'multiplierToSI' = ''
|  |  |  |__attribute 'name' = 'humidity'
|  |  |  |__attribute 'parentSI' = ''
|  |  |  |__attribute 'unitType' = 'humidity'
|  |  |  |__element 'description'
|  |  |  |  |__text 'air humidity'
|  |  |  |__text '\n '
|  |  |__text '\n '
|  |__text '\n '
|__text '\n '

```

Access Control:	
Auth System:	knb
Order:	allowFirst
Access Rules:	
ALLOW:	[read] public

Figura 27 – Informações sobre o EML gerado no experimento

A Tabela 6 apresenta um fragmento dos dados obtidos com a execução do serviço:

Tabela 6 - Fragmento de dados Importados do Weblab

Data	Hora	Umidade	Temperatura
2009-05-05	13:10:00	83	27.4
2009-05-05	13:11:00	83	27.5
2009-05-05	13:12:00	83	27.5
2009-05-05	13:13:00	83	27.5
2009-05-05	13:14:00	83	27.5
2009-05-05	14:15:00	83	27.5
2009-05-05	14:16:00	83	27.5
2009-05-05	14:17:00	83	27.6
2009-05-05	14:18:00	83	27.6
2009-05-05	14:19:00	83	27.6
2009-05-05	14:20:00	83	27.6
2009-05-05	15:21:00	83	27.7
2009-05-05	15:22:00	83	27.7
2009-05-05	15:23:00	83	27.6
2009-05-05	15:24:00	83	27.6

4.4 AVALIAÇÃO DOS RESULTADOS COM A FERRAMENTA DE ANÁLISE

Para análise do experimento foi definido um fluxo de tarefas (*workflow*), definindo quais dados serão utilizados, as operações matemáticas e os mecanismos de visualização dos resultados

A seguir são descritas as ferramentas de software que fazem parte da arquitetura empregada:

a) *Metacat*

O *Metacat* gerencia os arquivos XML em uma hierarquia de nós, sendo

que cada nó é armazenado como um registro em tabelas de banco de dados. Como os dados de muitos esquemas XML estão divididos em vários DTDs (*Document Type Definition*) são exigidos vários arquivos XML relacionados, porém, armazenados separadamente no sistema. Este sistema utilizado pacotes para interligar estes documentos relacionados. A Figura 28 apresenta a tela inicial de configuração da ferramenta *Metacat*.

Metacat Configuration

All of the following sections must be in a configured state for Metacat to run properly:
 -- User logged in as: uid=mleite,o=unaffiliated,dc=ecoinformatics,dc=org

[configured]	Metacat Global Properties	Reconfigure Now
[configured]	Authentication Configuration	Reconfigure Now
[configured]	Skins Specific Properties	Reconfigure Now
[configured]	Database Installation/Upgrade	Version: 1.9.1
[bypassed]	Geoserver Configuration	Reconfigure Now

Restarting Metacat

Since this is a reconfiguration, you will need to restart Metacat after any changes.

The simplest way to restart metacat is to restart the entire servlet engine. For Tomcat, this would mean calling "sudo /etc/init.d/tomcat5.5 restart" or an equivalent command appropriate to your operating system. After restarting, you can access your new Metacat server at the URL: <http://localhost:8080/knb>

You are logged in as: uid=mleite,o=unaffiliated,dc=ecoinformatics,dc=org

Figura 28 - Interface de configuração da ferramenta *Metacat*

b) Kepler

Para manipular os dados coletados do experimento em EML, é adicionado um módulo de acesso a serviços Web chamado de "*Web Service Actor*" do *framework* disponibilizado pelo Kepler. Este módulo permite o acesso aos serviços, podendo ser configurado de acordo com as especificações de cada fornecedor de serviço. O Kepler deve ter acesso ao endereço do servidor dos serviços Web, utilizando o WSDL do serviço que disponibiliza as funcionalidades existentes no Componente de Disponibilização permitindo ao Kepler, na execução da análise, efetuar a

solicitação dos dados coletados pelo Weblab e em sequência processar e efetuar o trabalho de análise destas informações, esta configuração pode ser visualizada na Figura 29:

Figura 29 - Configurando o Kepler para acesso ao serviço de Disponibilização.

O serviço utilizado neste exemplo é o *GetExperimentMetadata* do Componente de Disponibilização descrito na Seção 3.2.1, sendo consumido pelo Componente de Análise (Kepler). Os dados obtidos por este serviço tornam-se agora disponíveis a um módulo chamado “EML 2 Dataset” do Kepler, responsável em receber os dados em EML para realizar a análise do experimento. Para permitir o acesso a esse serviço, é definida uma porta (*Port*) de entrada de dados no módulo “EML 2 Dataset”, permitindo desta forma a conexão com o Componente de Disponibilização, fornecedor do serviço Web. A Figura 30 mostra o Componente de Análise e a interação com os componentes descritos:

Figura 30 - Efetuando a conexão entre o componente de serviço web e o componente de dados EML

Efetuada a conexão entre os componentes de extração e armazenamento de dados no Kepler, pode-se executar uma operação disponível no módulo “EML 2 Dataset”, chamada *Preview* que executa uma consulta que pode ser utilizada para verificar se a conexão entre os componentes envolvidos e o serviço web estão operando normalmente. A Figura 31 mostra o resultado da execução de *Preview*, apresentando os dados obtidos com o experimento.

Figura 31 - Executando o comando *Preview* para verificar dados coletados

Após validar a conexão com o serviço web e para a conclusão da validação da análise proposta, foi adicionado um módulo de saída de dados em modo gráfico chamado “XY Plotter”, este módulo tem como funcionalidade a geração de gráficos utilizando como parâmetros os dados disponibilizados pelo módulo de entrada, neste caso o módulo “EML 2 Dataset” que por sua vez, recebe as informações por meio da execução do serviço Web. Neste exemplo foi definido um gráfico, para demonstração da variação de temperatura dentro do período estabelecido de coleta de dados obtidos no interior das colméias, os parâmetros empregados para execução foram os dados de temperatura e de informações de tempo como data e hora da informação. A Figura 32 mostra este módulo gráfico conectado ao “EML 2 Dataset”, e na Figura 33 visualiza-se o gráfico gerado após a execução da análise. .

Figura 32 - Componente de configuração de gráfico XY Plotter

Figura 33 - Gráfico gerado com o uso da ferramenta Kepler

5 CONCLUSÕES

O caso de uso apresentado mostrou como ocorre a interação entre os componentes da arquitetura, disponibilizando as informações obtidas pela realização do experimento efetuado pela equipe da Profa. Dra. Vera Lucia Imperatriz Fonseca do Departamento de Biologia da Universidade de São Paulo, Campus de Ribeirão Preto, com o Componente de Análise Kepler mostrando a viabilidade do modelo proposto, pois foram realizadas as análises necessárias para realização do experimento para avaliação de termoregulação de abelhas.

5.1 CONTRIBUIÇÕES

A arquitetura proposta não se destina apenas a análises de experimentos ambientais com abelhas. Sua especificação pode ser adaptada a outros domínios de experimentos, sendo, portanto um modelo de referência, utilizado como base para os componentes que possuam a capacidade de permitir a integração dos dados coletados com os softwares empregados para a execução das análises e obtenção dos resultados, como por exemplo, Matlab, Sistema R e Excel.

A elaboração de uma arquitetura orientada a serviços para uso na integração de Weblabs com módulos de análise contribui, de forma geral, para diminuir a complexidade do trabalho dos pesquisadores. Esses usuários não possuem necessariamente conhecimentos específicos em Tecnologia da Informação e Comunicação, permitindo assim que seus esforços sejam direcionados a complexidade biológica e análise do experimento. Outras contribuições deste trabalho podem ser destacadas:

1. Desenvolvimento de um modelo para integração de Weblabs, com objetivo de disponibilizar as informações coletadas para uso em módulos específicos para análises em outros experimentos na área de biodiversidade;
2. A especificação da arquitetura é baseada em padrões abertos de metadados (EML) e de protocolos web para disponibilização e publicação de experimentos na WEB;
3. Permitir o uso da arquitetura SEEK por meio de suas ferramentas para um novo domínio de aplicação, voltado para experimentos ambientais com abelhas.

5.2 AVALIAÇÃO

A arquitetura proposta neste trabalho tornou possível integrar ao Weblab a etapa de Análise dos Experimentos.

A implementação da arquitetura de referência permitiu sua validação, demonstrando sua viabilidade para análises de experimentos.

O Componente de Disponibilização propiciou além da experiência adquirida em seu desenvolvimento, mas também o conhecimento necessário para efetuar a validação e a possibilidade de tornar este componente parte da arquitetura de pesquisas com Weblab.

O estudo da arquitetura SEEK, mais especificamente do *Metacat*, permitiu que as informações obtidas pelo experimento executado com o Weblab fossem disponibilizados e publicados no Ecogrid (seção 2.3.1), permitindo desta forma, que outros pesquisadores possam efetuar suas análises por meio dos dados obtidos com os experimentos realizados neste Weblab.

5.3 TRABALHOS FUTUROS

A arquitetura elaborada é uma primeira proposta para permitir o reuso das informações coletadas em ambientes experimentais gerenciados por Weblabs para integração com ferramentas de análise, porém, seu uso pode ser adequado sem muitos esforços para permitir a integração com outros módulos, como exemplo integração com Data Warehouse (DW), bastando para isto elaborar um componente nos mesmos moldes para esta finalidade.

Algumas possibilidades de extensão da arquitetura proposta podem ser estudadas para permitir melhoramentos que tornem o seu uso aceito em outros projetos gerenciados com o uso de Weblabs.

Uma sugestão de trabalho relacionado a está arquitetura é o desenvolvimento de uma arquitetura baseada em barramento, dessa forma, permitindo que novos serviços sejam acoplados de uma maneira mais simplificada bastando para isso, efetuar novas configurações no controle da orquestração efetuada pela ferramenta que controla o barramento.

Outra sugestão refere-se à elaboração de um estudo sobre os meios de execução de experimentos mais empregados pelos cientistas e o desenvolvimento de novos componentes que possam ser acoplados a arquitetura proposta neste trabalho, aumentando assim, as possibilidades de uso do modelo proposto.

6 REFERÊNCIAS

ABCD **ABCD Schema 2.06 – ratified TDWG Standard**. Disponível em <<http://bgbm3.bgbm.fu-berlin.de/TDWG/CODATA/Schema/default.htm>>. Acessado em 12 de Set. de 2009.

ABIF **Australian Biodiversity Information Facility**. Disponível em <<http://www.environment.gov.au/biodiversity/abif/bat>>. Acesso em 15 Set. 2009.

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, Jun 2004.

AQUINO, A. L. L. **Uma ferramenta para monitorar aplicações com redes de sensores sem fio**. Dissertação de Mestrado, UFMG, 2003.

BARSEGUIAN, D. *et al.* **Workflows and extensions to the Kepler scientific workflow system to support environmental sensor data access and analysis**. Ecological Informatics, 2009.

BIOABELHA **Biodiversidade E Uso Sustentável Dos Polinizadores Com Ênfase Em Abelhas**. Disponível em <<http://pitanga.pcs.usp.br/bioabelha>> Acesso em 10 de Nov. 2009.

BIS **Biodiversity Information Standards TDWG**. Disponível em <<http://www.tdwg.org/activities/darwincore>> Acesso em 25 Set. 2009.

BOOCH, G.; JACOBSON, I; RUMBAUGH, J. **UML: Guia do Usuário. 2 edição**. São Paulo: Editora Campus. 2005.

BUENO, J.F.; CORRÊA, P.L.P.; ONOE, A.Y.; BORSOI, B.T. KIRAMOTO A.T. **OASIS, Modelo de Referência para Arquitetura Orientada a Serviço 1.0**. Escola Politécnica da Universidade de São Paulo 2006.

CANHOS, V. P. **Informática para biodiversidade: padrões, protocolos e ferramentas.** *Cienc. Cult.*, São Paulo, v. 55, n. 2, Apr. 2003 . Disponível em: <http://cienciaecultura.bvs.br/scielo.php?script=sci_arttext&pid=S0009-67252003000200025&lng=en&nrm=iso>. Acesso em: 14 Set. 2009.

CARDOZO E. **Weblabs sobre redes de Alto Desempenho.** Technical Report, 2007.

CARTOLANO A.E.Jr.; Saraiva A.M.; Corrêa P.L.P.; Giannini T.C.; Giovanni R. **Uma Proposta de Esquema de Dados de Relacionamento Entre Espécimes.** INBI 2007.

CORRÊA, P.L.P. **Adaptação do método proposto pela IBM (Redbook).** PCS 5002 Notas de aula, Abr 2006.

CUI, H. **An Application for Semantic Markup of Biodiversity Documents.** JCDL'08, 2008.

DUBLINCORE **Dublin Core Metadata Initiative.** Disponível em <<http://dublincore.org/documents/dces/>>. Acesso em 10 Nov. 2009.

EML **Ecological Metadata Language.** Disponível em <<http://knb.ecoinformatics.org/software/eml>>. Acesso em 10 Nov. 2009.

ENDREI, M.; ANG, J.; ARSANJANI, A.; CHUA, S.; COMTE, P.; KROGDAHL, P.; LUO, M.; NEWLING, T. **Patterns: Service-Oriented Architecture and Web Services.** IBM Redbook, 2004.

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.

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

GBIF **Global Biodiversity Information Facility.** Disponível em <<http://www.gbif.org>>. Acesso em 4 Nov. 2009.

GLOBUS **A open source Grid software to distributed resource sharing.** Disponível em <www.globus.org>. Acesso em 10 Jan. 2010.

GURALNICK, R. P. *et al.* **A. Automated approaches for documenting global biodiversity patterns and processes.** Biodiversity informatics, 2009.

HUNHS M. N., SINGH M. P. **Service-oriented computing: key concepts and principles.** In: IEEE Internet Computing, February 2005.

ILAB **iLABS: Internet access to real labs – anywhere, anytime.** Disponível em <<http://lcampus.mit.edu/iLabs>>. Acesso em 10 Jan. 2010.

IABIN **Inter-American Biobiversity Network.** Disponível em <<http://www.iabin.net>>. Acesso em: 4 Nov. 2009.

INDERJEET, S.; STEARNS, B.; JOHNSON, M. **Design Enterprise Applications with the JEE.** 4.ed. Pearson 2007.

IUBS **International Union of Biological Sciences.** Disponível em <<http://iubs.org>>. Acesso em 6 Nov. 2009.

JONES, M.B.; SCHILDHAUER, M.P.; REICHMAN O.J.; BOWERS S. **The New Bioinformatics: Integrating Ecological Data from the Gene to the Biosphere.** The Annual Review of Ecology, Evolution, and Systematics, August 2006.

KAKODKAR, A.P.; KERKAR, S.S.; VARGHESE N.S., KAVLEKAR D.P. & C.T. ACHUTHANKUTTY **Darwin Core Based Data Streamlining with Digimus 2.0.** Bioinformatics Centre, National Institute of Oceanography, 2009.

KRAFZIG, D.; BANKE, K.; SLAMA, D. **Enterprise SOA: Service Oriented Architecture Best Practices.** Prentice Hall. 2004.

LEITE, M.Q.; NAJM, L.H.; CORRÊA, P.L.P. **Model to integration and analysis of remote experiments executed in bee weblabs**. 7 Contecsi, 2010.

MACKENZIE, C.M.; LASKEY, K.; MCCABE, F.; BROWN, P. F.; HAMILTON; METZ, R. **Oasis Reference Model for Service Oriented Architecture 1.0**. 2006.

MALER E. *et al.* **Extensible markup language (xml) 1.1. Technical report, W3C**. Disponível em <<http://www.w3.org/TR/2004/REC-xml11-20040204/>>. Acesso em 25 Set. 2009.

MARTT **An Application for Semantic Markup of Biodiversity Documents** Disponível em <<http://www.ieee-tcdl.org/Bulletin/v5n1/Cui/cui.html>> Acesso em 5 de Out. 2009.

MICHENER W.K.; PORTER J.H.; STAFFORD S.G. **Data and Information Management in the Ecological Sciences: A Resource Guide**. Disponível em <<http://www.ecoinformatics.org/pubs/guide/michener.fv2.htm>> Acesso em: 25 Set. 2009.

MYEXPERIMENT **The myExperiment Virtual Research Environment**. Disponível em <<http://www.myexperiment.org>>. Acesso em 18 dez. 2009.

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

OINN, T. *et al.*, **Taverna: Lessons in creating a workflow environment for the Experience life sciences**. In: Concurrency Computat. Pract. Exper., vol. 18. 2006 p. 1067-1100.

OKAJIMA H.S.S.; LEDEL L.C. **Weblab Development Using a LabView and Java Integrated Solution for Kyatera Network**. Tidia III Workshop, 2006.

OSI **Open Source Initiative OSI – The BSD License: Licensing**. Disponível em <<http://www.opensource.org/licenses/bsd-license.php>> Acesso em 10 Nov. 2009.

PAGE R.D.M. **Taxonomic Names, Metadata, and the Semantic Web.** Biodiversity Informatics, 2006.

PAPAZOGLU M. P., **SOC, Service-oriented Computing: Concepts, Characteristics and Directions.** In: Proceedings of the 4th International Conference on Web Information System Engineering, 2003.

PAPAZOGLU M. P.; HEUVEL W.J. **Web Services Management: A Survey** IEEE, 2005.

PAPAZOGLU M. P.; P. TRAVERSO; D. SCHABRAM; F LEYMAN **Service-oriented Computing: State of the Art and Research Challenges.** In IEEE Computer Society, 2007.

PLINIANCORE **Plinian Core Project.** Disponível em <<http://www.pliniancore.org/en/inicio.htm>>. Acesso em 30 Set. 2009.

PROGRAMMING R **Beginner to advanced resources for the R programming language.** Disponível em <www.programmingr.com>. Acesso em 5 Nov. 2009.

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

PTOLEMY **Ptolemy Project.** Disponível em <<http://ptolemy.eecs.berkeley.edu/ptolemyII>> Acesso em 5 Nov. 2009.

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, A. M. **Tecnologia da Informação na Agricultura de Precisão e Biodiversidade: Estudos e proposta de Utilização de Web Services para Desenvolvimento e Integração de Sistemas.** 2003. Tese (Livre Docência) -

Departamento de Engenharia de Computação e Sistemas Digitais, Escola Politécnica, Universidade de São Paulo, São Paulo, 2003.

SARAIVA, A. M. **Project Weblabs on Ecosystem Services** FAPESP Progress Report, 2006.

SEEK **Science Environment for Ecological Knowledge**. Disponível em <<http://seek.ecoinformatics.org>>. Acesso em 17 Set. 2009.

SOBERON J., PETERSON T. **Managing and applying primary biodiversity data**. Biodiversity Informatics, 2004.

SOUZA, D. L.; EVANGELISTA-RODRIGUES, A.; PINTO, M. P. As Abelhas Como Agentes Polinizadores (The Bees Agents Pollinizer's). **Revista electrónica de Veterinaria**, v. 8, n. 3, mar. 2007. ISSN 1695-7504.

SOUZA V.P.C.; GIOVANNI S.S.R.; CANHOS D.A.L. **Global Biodiversity Informatics: Setting the scene for a “New World” of Ecological Modeling**. Biodiversity Informatics, 2004.

STFC **Science and Technology Facilities Council**. Disponível em <<http://www.stfc.ac.uk/e-Science/projects/medium-term/storage-resource-broker/22430.aspx>> Acesso em 10 Jan. 2010.

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

UDDI **Online community for the Universal Description, Discovery and Integration OASIS Standard**. Disponível em <<http://uddi.xml.org/uddi-101>> Acesso em 7 Out. 2009.

VINCES **Virtual Networking Center of Ecosystem Services**. Disponível em <<http://www.ib.usp.br/vinces>>. Acesso em 17 Set. 2009.

WEBBEE Uma rede de Informações sobre diversidade brasileira em abelhas.
Disponível em <<http://www.webbee.org.br>>. Acesso em 5 Nov. 2009.

WILSON, E. O., Insect Societies. Harvard University Press. 1971. ISBN 9780674454958.

WSDL Web Services Description Language (WSDL) 1.1 Disponível em <<http://www.w3.org/TR/wsdl>>. Acesso em 11 Nov. 2009.

ZUBIA, J.G.; LÓPEZ-DE-IPÍÑA, D.; ORDUÑA, P. TOWARDS A Canonical Software Architecture for Multi-Devices Weblabs IEEE 2005.

APÊNDICE A

Este apêndice tem como objetivo relacionar os itens dos serviços Web elaborados no desenvolvimento desta arquitetura, apresentando uma lista dos códigos gerados para seu funcionamento.

A.1 CODIGO WSDL GERADO NA PROPOSTA

```
<wsdl:definitions targetNamespace="http://weblab.com.br/" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <s:schema elementFormDefault="qualified" targetNamespace="http://weblab.com.br/" xmlns:s="http://www.w3.org/2001/XMLSchema">
      <s:element name="GetExperimentsList">
        <s:complexType/>
      </s:element>
      <s:element name="GetExperimentsListResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetExperimentsListResult" type="s:string"/>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetValidationData">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="experimentname" type="s:string"/>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetValidationDataResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetValidationDataResult" type="s:string"/>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetExperimentStatus">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="experimentname" type="s:string"/>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetExperimentStatusResponse">
```

```

<s:complexType>
<s:sequence>
<s:element minOccurs="0" maxOccurs="1" name="GetExperimentStatusResult" type="s:string"/>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="GetExperimentMetadata">
<s:complexType>
<s:sequence>
<s:element minOccurs="0" maxOccurs="1" name="experimentname" type="s:string"/>
/>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="GetExperimentMetadataResponse">
<s:complexType>
<s:sequence>
<s:element minOccurs="0" maxOccurs="1" name="GetExperimentMetadataResult" type="s:string"/>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="GetExperimentXLSFormat">
<s:complexType>
<s:sequence>
<s:element minOccurs="0" maxOccurs="1" name="experimentname" type="s:string"/>
/>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="GetExperimentXLSFormatResponse">
<s:complexType/>
</s:element>
</s:schema>
</wsdl:types>
<wsdl:message name="GetExperimentsListSoapIn">
<wsdl:part name="parameters" element="tns:GetExperimentsList"/>
</wsdl:message>
<wsdl:message name="GetExperimentsListSoapOut">
<wsdl:part name="parameters" element="tns:GetExperimentsListResponse"/>
</wsdl:message>
<wsdl:message name="GetValidationDataSoapIn">
<wsdl:part name="parameters" element="tns:GetValidationData"/>
</wsdl:message>
<wsdl:message name="GetValidationDataSoapOut">
<wsdl:part name="parameters" element="tns:GetValidationDataResponse"/>
</wsdl:message>
<wsdl:message name="GetExperimentStatusSoapIn">
<wsdl:part name="parameters" element="tns:GetExperimentStatus"/>

```

```

</wsdl:message>
<wsdl:message name="GetExperimentStatusSoapOut">
<wsdl:part name="parameters" element="tns:GetExperimentStatusResponse"/>
</wsdl:message>
<wsdl:message name="GetExperimentMetadataSoapIn">
<wsdl:part name="parameters" element="tns:GetExperimentMetadata"/>
</wsdl:message>
<wsdl:message name="GetExperimentMetadataSoapOut">
<wsdl:part name="parameters" element="tns:GetExperimentMetadataResponse"/>
</wsdl:message>
<wsdl:message name="GetExperimentXLSFormatSoapIn">
<wsdl:part name="parameters" element="tns:GetExperimentXLSFormat"/>
</wsdl:message>
<wsdl:message name="GetExperimentXLSFormatSoapOut">
<wsdl:part name="parameters" element="tns:GetExperimentXLSFormatResponse"/>
>
</wsdl:message>
<wsdl:portType name="Service1Soap">
<wsdl:operation name="GetExperimentsList">
<wsdl:documentation>
Retorna a lista dos experimentos em execução no weblab
</wsdl:documentation>
<wsdl:input message="tns:GetExperimentsListSoapIn"/>
<wsdl:output message="tns:GetExperimentsListSoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetValidationData">
<wsdl:documentation>
Efetua uma validação dos dados coletados no weblab
</wsdl:documentation>
<wsdl:input message="tns:GetValidationDataSoapIn"/>
<wsdl:output message="tns:GetValidationDataSoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetExperimentStatus">
<wsdl:documentation>
Verifica o Status em que se encontra o experimento
</wsdl:documentation>
<wsdl:input message="tns:GetExperimentStatusSoapIn"/>
<wsdl:output message="tns:GetExperimentStatusSoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetExperimentMetadata">
<wsdl:documentation>
Retorna os dados do experimento em formato EML
</wsdl:documentation>
<wsdl:input message="tns:GetExperimentMetadataSoapIn"/>
<wsdl:output message="tns:GetExperimentMetadataSoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetExperimentXLSFormat">
<wsdl:documentation>
Retorna os dados do experimento em formato XLS
</wsdl:documentation>

```

```

<wsdl:input message="tns:GetExperimentXLSFormatSoapIn"/>
<wsdl:output message="tns:GetExperimentXLSFormatSoapOut"/>
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="Service1 Soap" type="tns:Service1 Soap">
<soap:binding transport="http://schemas.xmlsoap.org/soap/http" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
<wsdl:operation name="GetExperimentsList">
<soap:operation soapAction="http://weblab.com.br/GetExperimentsList" style="document" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
<wsdl:input>
<soap:body use="literal" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
</wsdl:input>
<wsdl:output>
<soap:body use="literal" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetValidationData">
<soap:operation soapAction="http://weblab.com.br/GetValidationData" style="document" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
<wsdl:input>
<soap:body use="literal" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
</wsdl:input>
<wsdl:output>
<soap:body use="literal" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetExperimentStatus">
<soap:operation soapAction="http://weblab.com.br/GetExperimentStatus" style="document" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
<wsdl:input>
<soap:body use="literal" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
</wsdl:input>
<wsdl:output>
<soap:body use="literal" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetExperimentMetadata">
<soap:operation soapAction="http://weblab.com.br/GetExperimentMetadata" style="document" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
<wsdl:input>
<soap:body use="literal" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
</wsdl:input>
<wsdl:output>
<soap:body use="literal" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetExperimentXLSFormat">
<soap:operation soapAction="http://weblab.com.br/GetExperimentXLSFormat" style="document" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />

```

```

<wsdl:input>
<soap:body use="literal" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
</wsdl:input>
<wsdl:output>
<soap:body use="literal" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" />
</wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:binding name="Service1Soap12" type="tns:Service1Soap">
<soap12:binding transport="http://schemas.xmlsoap.org/soap/http" xmlns:soap12="
http://schemas.xmlsoap.org/wsdl/soap12/" />
<wsdl:operation name="GetExperimentsList">
<soap12:operation soapAction="http://weblab.com.br/GetExperimentsList" style="d
ocument" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/" />
<wsdl:input>
<soap12:body use="literal" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12
/" />
</wsdl:input>
<wsdl:output>
<soap12:body use="literal" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12
/" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetValidationData">
<soap12:operation soapAction="http://weblab.com.br/GetValidationData" style="doc
ument" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/" />
<wsdl:input>
<soap12:body use="literal" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12
/" />
</wsdl:input>
<wsdl:output>
<soap12:body use="literal" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12
/" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetExperimentStatus">
<soap12:operation soapAction="http://weblab.com.br/GetExperimentStatus" style="
document" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/" />
<wsdl:input>
<soap12:body use="literal" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12
/" />
</wsdl:input>
<wsdl:output>
<soap12:body use="literal" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12
/" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetExperimentMetadata">
<soap12:operation soapAction="http://weblab.com.br/GetExperimentMetadata" styl
e="document" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/" />

```

```

<wsdl:input>
<soap12:body use="literal" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12
/">
</wsdl:input>
<wsdl:output>
<soap12:body use="literal" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12
/">
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetExperimentXLSFormat">
<soap12:operation soapAction="http://weblab.com.br/GetExperimentXLSFormat" st
yle="document" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/">
<wsdl:input>
<soap12:body use="literal" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12
/">
</wsdl:input>
<wsdl:output>
<soap12:body use="literal" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12
/">
</wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="Service1">
<wsdl:port name="Service1 Soap" binding="tns:Service1 Soap">
<soap:address location="http://localhost:49854/Service.asmx" xmlns:soap="http://sc
hemas.xmlsoap.org/wsdl/soap/">
</wsdl:port>
<wsdl:port name="Service1 Soap12" binding="tns:Service1 Soap12">
<soap12:address location="http://localhost:49854/Service.asmx" xmlns:soap12="htt
p://schemas.xmlsoap.org/wsdl/soap12/">
</wsdl:port>
</wsdl:service>
</wsdl:definitions>

```

A.2 MENSAGENS SOAP (*SIMPLE OBJECT ACCESS PROTOCOL*) DOS SERVIÇOS DISPONIBILIZADOS COMPONENTE DE DISPONIBILIZAÇÃO

GETEXPERIMENTMETADATA

Retorna os dados do experimento em formato EML

Parâmetro	Valor
experimentname:	<input type="text"/>

O exemplo a seguir mostra uma solicitação e uma resposta SOAP 1.1. Os espaços reservados mostrados devem ser substituídos por valores reais.

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

<?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>
    < GetExperimentMetadata xmlns="http://weblab.com.br/">
      <experimentname>string</experimentname>
    </ GetExperimentMetadata>
  </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>
    < GetExperimentMetadataResponse xmlns="http://weblab.com.br/">
      < GetExperimentMetadataResult>string</ GetExperimentMetadataResult>
    </ GetExperimentMetadataResponse>
  </soap:Body>
</soap:Envelope>
```

O exemplo a seguir mostra uma solicitação e uma resposta SOAP 1.2. Os espaços reservados mostrados devem ser substituídos por valores reais.

POST /Service.asmx HTTP/1.1

Host: localhost

Content-Type: application/soap+xml; charset=utf-8

Content-Length: [length](#)

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-
envelope">
```

```
<soap12:Body>
```

```
<GetExperimentMetadata xmlns="http://weblab.com.br/">
```

```
<experimentname>string</experimentname>
```

```
</GetExperimentMetadata>
```

```
</soap12:Body>
```

```
</soap12:Envelope>
```

HTTP/1.1 200 OK

Content-Type: application/soap+xml; charset=utf-8

Content-Length: [length](#)

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-
envelope">
```

```
<soap12:Body>
```

```
<GetExperimentMetadataResponse xmlns="http://weblab.com.br/">
```

```
<GetExperimentMetadataResult>string</GetExperimentMetadataResult>
```

```
</GetExperimentMetadataResponse>
```

```
</soap12:Body>
```

```
</soap12:Envelope>
```

A seguir é apresentado um exemplo de solicitação e resposta HTTP POST. Os espaços reservados mostrados devem ser substituídos por valores reais.

POST /Service.asmx/GetExperimentMetadata HTTP/1.1

Host: localhost

Content-Type: application/x-www-form-urlencoded

Content-Length: [length](#)

[experimentname](#)=[string](#)

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<string xmlns="http://weblab.com.br/">string</string>

```

GETEXPERIMENTSLIST

Retorna a lista dos experimentos em execução no weblab

O exemplo a seguir mostra uma solicitação e uma resposta SOAP 1.1. Os espaços reservados mostrados devem ser substituídos por valores reais.

```

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>

```

```

    </GetExperimentsListResponse>
  </soap:Body>
</soap:Envelope>

```

O exemplo a seguir mostra uma solicitação e uma resposta SOAP 1.2. Os espaços reservados mostrados devem ser substituídos por valores reais.

```

POST /Service.asmx HTTP/1.1
Host: localhost
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-
envelope">
  <soap12:Body>
    <GetExperimentsList xmlns="http://weblab.com.br/" />
  </soap12:Body>
</soap12:Envelope>
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-
envelope">
  <soap12:Body>
    <GetExperimentsListResponse xmlns="http://weblab.com.br/">
      <GetExperimentsListResult>string</GetExperimentsListResult>
    </GetExperimentsListResponse>
  </soap12:Body>
</soap12:Envelope>

```

A seguir é apresentado um exemplo de solicitação e resposta HTTP POST. Os espaços reservados mostrados devem ser substituídos por valores reais.

POST /Service.asmx/GetExperimentsList HTTP/1.1

Host: localhost

Content-Type: application/x-www-form-urlencoded

Content-Length: [length](#)

HTTP/1.1 200 OK

Content-Type: text/xml; charset=utf-8

Content-Length: [length](#)

<?xml version="1.0" encoding="utf-8"?>

<string xmlns="http://weblab.com.br/">[string](#)</string>

APÊNDICE B

CODIGO EML OBTIDO NA EXECUÇÃO DA FERRAMENTA

```
<?xml version="1.0"?>
<eml:eml packageId="mleite.5.3" system="knb"
xmlns:eml="eml://ecoinformatics.org/eml-2.1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="eml://ecoinformatics.org/eml-2.1.0 eml.xsd">
<access authSystem="knb" order="allowFirst">
<allow><principal>public</principal>
  <permission>read</permission>
</allow>
</access>
<dataset> <title>Estudo sobre termorregula;;o de
abelhas</title>
  <creator id="1277490991410">
<individualName><surName>Vollet</surName>
  </individualName>
  <positionName>Ayrton</positionName>
</creator>
  <abstract><para>Estudo elaborado para monitoramento da
termorregula;;o das abelhas Mel;pona
quadrifasciata</para>
  </abstract>
  <keywordSet><keyword>weblab</keyword>
  <keyword>metadados</keyword>
  <keyword>informatica para biodiversidade</keyword>
</keywordSet>
  <coverage><temporalCoverage><singleDateTime><calendarDate>2009-05-
09</calendarDate>
  </singleDateTime>
</temporalCoverage>
</coverage>
  <contact
id="1277491015334"><individualName><surName>Vollet</surName>
  </individualName>
  <positionName>Ayrton</positionName>
</contact>
  <dataTable
id="1277493711981"><entityName>dados_lonworks</entityName>
  <entityDescription>Dados obtidos da rede
lonwork</entityDescription>
  <physical><objectName>Unnamed-table-
6592582516504223340.tmp</objectName>
  <dataFormat>
<textFormat><attributeOrientation>column</attributeOrientation>
  <simpleDelimited><fieldDelimiter>,</fieldDelimiter>
</simpleDelimited>
</textFormat>
</dataFormat>
  <distribution><online><url>ecogrid://knb/mleite.3.3</url>
</online>
</distribution>
```

```

</physical>
<attributeList><attribute
id="1277493711982"><attributeName>data_time</attributeName>
<attributeLabel>data_time</attributeLabel>
<attributeDefinition>Data e horario de coleta de dados nos
sensores</attributeDefinition>
<storageType typeSystem="Postgresql">integer</storageType>
<measurementScale><dateTime><formatString>YYYY-MM-DD
hh:mm:ss</formatString>
<dateTimePrecision>1 minute</dateTimePrecision>
</dateTime>
</measurementScale>
</attribute>
<attribute
id="1277493711983"><attributeName>Umidade</attributeName>
<attributeLabel>Umidade</attributeLabel>
<attributeDefinition>Dados de umidade</attributeDefinition>
<storageType typeSystem="Postgresql">integer</storageType>
<measurementScale><ratio><unit><customUnit>humidity</customUnit>
</unit>
<numericDomain><numberType>natural</numberType>
<bounds><minimum exclusive="true">0</minimum>
<maximum exclusive="true">100</maximum>
</bounds>
</numericDomain>
</ratio>
</measurementScale>
</attribute>
<attribute
id="1277493711985"><attributeName>Temperatura</attributeName>
<attributeLabel>Temperatura</attributeLabel>
<attributeDefinition>Temperatura coletada nas
colmeias</attributeDefinition>
<storageType typeSystem="Postgresql">float</storageType>
<measurementScale><ratio><unit><standardUnit>celsius</standardUnit>
</unit>
<precision>0.1</precision>
<numericDomain><numberType>natural</numberType>
</numericDomain>
</ratio>
</measurementScale>
</attribute>
</attributeList>
</dataTable>
</dataset>
<additionalMetadata><metadata><unitList><unitType id="humidity"
name="humidity"><dimension name="Length"></dimension>
</unitType>
<unit id="humidity" multiplierToSI=" " name="humidity"
parentSI="ampere" unitType="humidity"><description>air
humidity</description>
</unit>
</unitList>
</metadata>
</additionalMetadata>
</eml:eml>

```

APÊNDICE C

MODELO FÍSICO BANCO DE DADOS WEBLAB

Figura 34 - Modelo Físico do Banco de dados do Weblab