

**NEILOR FERMINO CAMARGO**

**VFT: UMA ESTRUTURA DE APLICAÇÃO PARA VISUALIZAÇÃO DE  
FENÔMENOS TERRESTRES EM 3D VIA WEB**

**Dissertação apresentada como requisito parcial  
à obtenção do grau de Mestre em Informática,  
Curso de Pós-Graduação em Informática, Setor  
de Ciências Exatas, Universidade Federal do  
Paraná.**

**Orientador: Prof. Dr. Carlos Alberto Picanço de  
Carvalho**

**CURITIBA  
2003**



Ministério da Educação  
Universidade Federal do Paraná  
Mestrado em Informática

## PARECER

Nós, abaixo assinados, membros da Banca Examinadora da defesa de Dissertação de Mestrado em Informática, do aluno Neilor Firmino Camargo, avaliamos o trabalho intitulado, "*VFT – Uma Estrutura de Aplicação para Visualização de Fenômenos Terrestres em 3D via Web*", cuja defesa foi realizada no dia 26 de setembro de 2003, às dez horas, no Auditório da Informática da Universidade Federal do Paraná. Após a avaliação, decidimos pela aprovação do candidato.

Curitiba, 26 de setembro de 2003.



Prof. Dr. Carlos Alberto Picânço de Carvalho  
DINF/UFPR - Orientador



Prof. Dr. Daltro José Nunes  
UFRGS – Membro Externo

*Hélio Pedrini*  
Prof. Dr. Hélio Pedrini  
DINF/UFPR – Membro Interno

## **DEDICATÓRIA**

Dedico este trabalho a perseverança e a todos aqueles que perseveraram.

## **AGRADECIMENTOS**

Ao professor Carlos Alberto Picanço de Carvalho, pela dedicação e confiança em mim depositada.

Ao professor Antônio José Berutti Vieira por partilhar seus conhecimentos e visão inovadora.

Ao professor Alexandre Ibrahim Direne pela dedicação aos alunos da pós-graduação.

A Prof<sup>a</sup>. Cláudia R. Sluter, ao Prof. Henrique Firkowski, a Prof<sup>a</sup>. Luciene S. Delazari e ao Departamento de Geomática pelo apoio dado.

A Natália T. Camargo, a Célia S. de Q. Camargo, a Henrique de Q. Camargo e a Carolina de Q. Camargo pela motivação.

## SUMÁRIO

<b>LISTA DE ILUSTRAÇÕES .....</b>	<b>V</b>
<b>LISTA DE ABREVIATURAS E LISTA DE SIGLAS.....</b>	<b>VII</b>
<b>RESUMO .....</b>	<b>VIII</b>
<b>ABSTRACT .....</b>	<b>IX</b>
<b>1 INTRODUÇÃO .....</b>	<b>10</b>
1.1 O PROBLEMA.....	10
1.2 JUSTIFICATIVA .....	10
1.3 OBJETIVOS .....	11
1.3.1 Objetivo Geral .....	11
1.3.2 Objetivos Específicos.....	11
<b>2 UMA VISÃO GERAL DE ESTRUTURAS DE APLICAÇÃO E TECNOLOGIAS PARA VISUALIZAÇÃO 3D VIA WEB .....</b>	<b>12</b>
2.1 ESTRUTURAS DE APLICAÇÃO ORIENTADAS A OBJETOS .....	12
2.1.1 CARACTERÍSTICAS .....	13
2.1.2 CLASSIFICAÇÃO .....	15
2.1.3 COMO DESENVOLVER.....	18
2.2 VISUALIZAÇÃO DE OBJETOS EM 3D VIA WEB (WEB 3D).....	21
2.2.1 VRML.....	21
2.2.2 Java3D.....	23
2.2.3 X3D.....	23
<b>3 ANÁLISE DE DOMÍNIO .....</b>	<b>25</b>
3.1 FATOS, FENÔMENOS, DADOS E MODELOS .....	25
3.2 FENÔMENOS TERRESTRES .....	26
3.3 O MODELO DE FENÔMENOS TERRESTRES .....	27
3.3.1 A Semântica.....	27
3.3.2 A Superfície .....	28
3.3.3 Área de Levantamento.....	29
3.3.4 Janela de Visualização .....	30
3.4 O MODELO DE FENÔMENOS TERRESTRES EM UML .....	31
3.4.1 A Classe Fenômeno Terrestre .....	31
3.4.2 A Classe Superfície .....	32
3.4.3 A Classe Área de Levantamento e a Classe Janela de Visualização .....	33
<b>4 PROJETO DA ARQUITETURA.....</b>	<b>35</b>
4.1 A ARQUITETURA CLIENTE/SERVIDOR .....	35
4.2 O PARADIGMA MODELO-VISÃO-CONTROLE (MVC).....	36
4.3 A ARQUITETURA DA ESTRUTURA DE APLICAÇÃO .....	37
<b>5 PROJETO DA ESTRUTURA DE APLICAÇÃO.....</b>	<b>39</b>
5.1 O PACOTE DO MODELO MODEL .....	40
5.1.1 O pacote fenômeno <b>fen</b> .....	41
5.1.2 O pacote superfície <b>sup</b> .....	41
5.1.3 O pacote coordenadas <b>coord</b> .....	43
5.1.4 O pacote área <b>area</b> .....	44
5.2 O PACOTE VISÃO <b>VISAO</b> .....	44
5.3 O PACOTE CONTROLE <b>CONTROL</b> .....	44
5.4 O PACOTE PERSISTÊNCIA <b>PERSIST</b> .....	45

<b>6 IMPLEMENTAÇÃO</b> .....	<b>48</b>
6.1 OS DADOS .....	48
6.2 O MODELO.....	51
6.3 O CONTROLE.....	51
6.4 A VISUALIZAÇÃO.....	53
<b>7 APLICAÇÃO TESTE</b> .....	<b>56</b>
7.1 PROPÓSITO.....	56
7.2 IDENTIFICAÇÃO DE CASOS DE USO .....	56
7.2.1 Caso de uso: Cadastrar Área de Levantamento .....	57
7.2.2 Caso de uso: Visualizar Área de Levantamento .....	57
7.2.3 Caso de uso: Selecionar Janela de Visualização .....	58
7.3 A INTERFACE COM O USÁRIO.....	59
7.4 INSTALAÇÃO E USO .....	62
<b>8 CONCLUSÃO</b> .....	<b>63</b>
8.1 RESULTADOS.....	63
8.2 TRABALHOS FUTUROS .....	65
<b>REFERÊNCIAS</b> .....	<b>66</b>
<b>APÊNDICES</b> .....	<b>68</b>

## LISTA DE ILUSTRAÇÕES

FIGURA 1 – ATIVIDADES DE DESENVOLVIMENTO .....	19
FIGURA 2 – RELAÇÃO ENTRE FATOS, FENÔMENOS, MODELOS E DADOS .....	26
FIGURA 3 – SUPERFÍCIE TERRESTRE .....	27
FIGURA 4 – ÁREA DE LEVANTAMENTO.....	29
FIGURA 5 – RETÂNGULO ENVOLVENTE .....	30
FIGURA 6 – JANELA DE VISUALIZAÇÃO .....	30
FIGURA 7 – A CLASSE <b>FENOMENOTERRESTRE</b> .....	32
FIGURA 8 – A CLASSE <b>SUPERFÍCIE</b> .....	33
FIGURA 9 – AS CLASSES AREALEVANTAMENTO E JANELAVISUALIZACAO.....	34
FIGURA 10 – PARADIGMA CLIENTE/SERVIDOR .....	35
FIGURA 11 – PARADIGMA MVC .....	36
FIGURA 12 – VISÃO GERAL DA ARQUITETURA.....	37
FIGURA 13 – A ESTRUTURA DE PACOTES .....	40
FIGURA 14 – O PACOTE MODELO <b>MODEL</b> .....	40
FIGURA 15 – O PACOTE FENÔMENO TERRESTRE <b>FEN</b> .....	41
FIGURA 16 – O PACOTE SUPERFÍCIE TERRESTRE <b>SUP</b> .....	42
FIGURA 17 – O PACOTE COORDENADAS <b>COORD</b> .....	43
FIGURA 18 – O PACOTE ÁREA <b>AREA</b> .....	44
FIGURA 19 – O PACOTE CONTROL.....	45
FIGURA 20 – O PACOTE <b>PERSIST</b> - CLASSE BDFTBEAN .....	45
FIGURA 21 – O PACOTE <b>PERSIST</b> –CLASSES DE ACESSO A ÁREAS DE LEVANTAMENTO.....	46
FIGURA 22 – O PACOTE <b>PERSIST</b> - CLASSES DE ACESSO A ÁREAS DE LEVANTAMENTOMENTO.....	47
FIGURA 23 – ESTRUTURA DA TABELA <b>AREASLEVANTAMENTO</b> .....	48
FIGURA 24 – ESTRUTURA DA TABELA <b>FENÔMENOS</b> .....	49

FIGURA 25 – ESTRUTURA DE DOCUMENTO XML PARA DESCREVER UMA ÁREA DE LEVANTAMENTO.....	50
FIGURA 26 – O CÓDIGO FONTE DA CLASSE CONTROLE.....	52
FIGURA 27 – EXEMPLO DE PÁGINA JSP COM VRML.....	54
FIGURA 28 – UM EXEMPLO DE IMPLEMENTAÇÃO DO MÈTODO <b>GETVRML()</b> .....	55
FIGURA 29 – CASOS DE USO DA APLICAÇÃO <b>VFTAPP</b> .....	56
FIGURA 30 – CADASTRAR ÁREA DE LEVANTAMENTO .....	57
FIGURA 31 – VISUALIZAR ÁREA DE LEVANTAMENTO .....	58
FIGURA 32 – SELECIONAR JANELA DE VISUALIZAÇÃO.....	59
FIGURA 33 - PÁGINA PRINCIPAL DA APLICAÇÃO .....	59
FIGURA 34 – PÁGINA CADASTRAR ÁREA DE LEVANTAMENTO.....	60
FIGURA 35 – PÁGINA SELECIONAR JANELA DE VISULIZAÇÃO .....	60
FIGURA 36 – PÁGINA ÁREAS DE LEVANTAMENTO.....	61
FIGURA 37 – PÁGINA DE VISUALIZAÇÃO DE ÁREA DE LEVANTAMENTO .....	61
FIGURA 38 – COMPARATIVODE LINHAS DE CÓDIGO JAVA, HTML, JSP .....	64
FIGURA 39 – COMPARATIVO NUMERO DE CLASSES JAVA E PÁGINAS JSP .....	65

## LISTA DE ABREVIATURAS E LISTA DE SIGLAS

AL	- Área de Levantamento
API	- Application Programming Interface
DTD	- Document Type Definition
EAI	- External Authoring Interface
FT	- Fenômeno Terrestre
GCC	- Geocentric Coordinate System
GDC	- Geodetic Coordinate System
GEOVRML	- Extend VRML to support geographic applications
GEOX3D	- X3D Geospatial Component
HP-UX	- Sistema operacional da Hewlett Packard semelhante ao UNIX
HTML	- Hypertext Markup Language
IBM-AIX	- Sistema operacional da IBM semelhante ao UNIX
J2EE	- Java 2 Platform Enterprise Edition
JAVA	- Linguagem de Programação Java
JSP	- Java Server Page
Linux	- Sistema Operacional semelhante ao UNIX
MPEG	- Moving Picture Experts Group
MVC	- Modelo Visualização Controle
Solaris	- Sistema operacional da SUN semelhante ao UNIX
UML	- Unified Modeling Language
UNIX	- Sistema operacional multitarefa e multiusuário
UTM	- Universal Transverse Mercator
VFT	- Visualização de Fenômenos Terrestres
VRML	- Virtual Reality Modeling Language
VRML97	- VRML97 Specification ISO/IEC 14772-1:1997
WEB	- World Wide Web
WGS84	- World Geodetic System 1984
Windows	- Família de sistemas operacionais da Microsoft
X3D	- Extended 3D
XML	- Extended Markup Language



## RESUMO

Esta dissertação apresenta uma Estrutura de Aplicação Orientada a Objetos para a construção de aplicações de visualização de fenômenos terrestres em 3D via Web. Faz uma investigação sobre os métodos, técnicas e processos de construção de Estruturas de Aplicação Orientadas a Objetos. Descreve um modelo conceitual para a representação de fenômenos terrestres. Determina mecanismos para organização, manipulação e visualização em 3D destes fenômenos utilizando a VRML. E, finalmente mostra uma aplicação de visualização em 3D de fenômenos terrestres via Web utilizando a estrutura apresentada.

Palavras-chave: Estruturas de Aplicação Orientadas a Objetos; Visualização de objetos 3D via Web; VRML; JAVA; JSP; Fenômenos Terrestres, Ciências Geodésicas.

## ABSTRACT

This work presents an object oriented application framework for terrestrial phenomena visualizations in 3D. Based in a formal model to express terrestrial phenomena as software objects and, in a model to persist collections of terrestrial phenomena in a computer is presented architecture to construct object-oriented applications. The architecture allows creating applications for 3D terrestrial phenomena visualizations using the Virtual Realty Modeling Language.

**Keywords:** Geodetic Science; Object Oriented Application Frameworks; 3D Computer Visualization, VRML, JAVA, JSP.

# 1 INTRODUÇÃO

## 1.1 O PROBLEMA

A criação de aplicações com interabilidade com visualização espacial de objetos via Web é uma tarefa complexa e demorada. Esta tarefa envolve o uso de diversas tecnologias, mecanismos de visualização, manipulação de objetos, e de interface com o usuário.

Como simplificar o desenvolvimento de aplicações que mostrem visualizações em 3D de superfícies geográficas e de objetos sobre elas em um computador conectado a Web?

Este trabalho é a proposta de uma Estrutura de Aplicação Orientada a Objetos contendo uma arquitetura de classes e objetos necessários para a criação de aplicações de visualização em 3D via Web de fenômenos terrestres.

## 1.2 JUSTIFICATIVA

Uma Estrutura de Aplicação Orientada a Objetos é uma aplicação reutilizável construída para um domínio específico de aplicações que deve ser estendida e especializada para produzir aplicações completas neste domínio (FAYAD; SCHMIDT, 1997, p. 32). As Estruturas de Aplicação Orientadas a Objeto estão assumindo um papel de importância na indústria e no meio acadêmico como uma forma de criar abstrações para facilitar o desenvolvimento de *software*. Este tipo de reutilização permite a redução de custos e a melhoria da qualidade no desenvolvimento de *software* (FAYAD; SCHMIDT 1997).

Uma Estrutura de Aplicação Orientada a Objetos com uma arquitetura de classes e objetos que cooperem entre si e que possa ser reutilizada para a criação de aplicações de exploração, manipulação e visualização de fenômenos terrestres em 3D, poderá resultar em uma melhoria na qualidade, produtividade e

confiabilidade.

Outros benefícios que este trabalho pode trazer:

- a) Um modelo de referência para o projeto e o desenvolvimento de Estruturas de Aplicação Orientadas a Objetos para construção de aplicações de visualização via Web;
- b) Uma maneira de aumentar a produtividade e reduzir custos na criação de aplicações de visualização na Web;
- c) Um modelo de estudo de técnicas de reuso na concepção, projeto e implementação de aplicações de visualização baseadas na Web.

### 1.3 OBJETIVOS

#### 1.3.1 Objetivo Geral

Desenvolver uma Estrutura de Aplicação Orientada a Objetos para a construção de aplicações de visualização de fenômenos terrestres em 3D via Web.

#### 1.3.2 Objetivos Específicos

- a) Investigar e identificar os métodos, técnicas e processos de construção de Estruturas de Aplicação Orientadas a Objetos.
- b) Compreender os mecanismos de estruturação e organização de dados para visualização em 3D da linguagem VRML.
- c) Compreender o modelo conceitual de representação de fenômenos terrestres e determinar mecanismos para organização, manipulação e visualização em 3D destes fenômenos utilizando a VRML.
- d) Aplicar a Estrutura de Aplicação Orientada a Objetos desenvolvida na construção de uma aplicação de visualização em 3D de fenômenos terrestres via Web.

## **2 UMA VISÃO GERAL DE ESTRUTURAS DE APLICAÇÃO E TECNOLOGIAS PARA VISUALIZAÇÃO 3D VIA WEB**

Neste capítulo são apresentadas as tecnologias e soluções encontradas na literatura para a construção de estruturas de aplicação orientadas a objetos e para visualização de objetos em 3D na Web e como estas têm sido utilizadas para visualização de fenômenos terrestres. O conteúdo está baseado em CAMARGO (2002) e CAMARGO e CARVALHO (2002).

### **2.1 ESTRUTURAS DE APLICAÇÃO ORIENTADAS A OBJETOS**

Estruturas de Aplicação e Padrões de Projeto vêm sendo utilizados como meios para o reuso de componentes e soluções já comprovadas (LARSEN, 1999, p. 39) e (FAYAD; SCHMIDT, 1997, p. 34). O projeto de sistemas usando componentes e soluções já comprovadas permite que os projetistas trabalhem em um nível de abstração mais alto e com maior confiabilidade evitando o contínuo redescobrimiento e reinvenção de conceitos e componentes. A redução de custos, aumento de produtividade e a melhoria na qualidade são as principais melhorias que este enfoque traz.

Uma Estrutura de Aplicação Orientada a Objetos é uma aplicação reutilizável que deve ser detalhada para produzir uma aplicação específica completa. Uma Estrutura de Aplicação Orientada a Objetos descreve a arquitetura de um sistema orientado a objetos e provê um contexto reutilizável de componentes e padrões de projeto para a construção de aplicações em um domínio específico (JOHNSON, 1997, p. 39). A Estrutura de Aplicação descreve como uma aplicação é composta de objetos e como estes objetos interagem entre si. A representação da arquitetura é feita através de um conjunto de classes, em geral abstratas, uma para cada tipo de objeto, e de um conjunto de padrões de interação entre os objetos (FAYAD;

SCHMIDT; JOHNSON, 1999, p. 5). As Estruturas de Aplicação Orientadas a Objetos substantificam as abstrações de projetos e implementações já comprovadas para reduzir custos e melhorar a qualidade do *software*. (FAYAD; SCHMIDT, 1997, p. 32).

Tradicionalmente, Estruturas de Aplicação reutilizáveis têm sido desenvolvidas pela generalização de sistemas e aplicações existentes. Este processo incremental de desenvolvimento é freqüentemente lento, já que, princípios e padrões de projeto fundamentais devem ser descobertos por síntese.

### 2.1.1 CARACTERÍSTICAS

As principais características das Estruturas de Aplicação orientadas a objetos residem na modularidade, reusabilidade, extensibilidade, e inversão de controle que proporcionam aos desenvolvedores de aplicações.

#### 2.1.1.1 Modularidade

Estruturas de Aplicação melhoram a modularidade pelo encapsulamento de detalhes mutáveis em interfaces estáveis. Esta modularidade ajuda a aprimorar a qualidade da aplicação desenvolvida restringindo a um só local o impacto de mudanças do projeto e da implementação. Esta localização reduz o esforço necessário para entender e manter a aplicação desenvolvida.

#### 2.1.1.2 Reusabilidade

As interfaces estáveis proporcionadas pelas Estruturas de Aplicação melhoram a reutilização através da definição de componentes genéricos que podem ser reaplicados na criação de novas aplicações. A reutilização da Estrutura de Aplicação sedimenta o conhecimento do domínio e justifica os esforços anteriores

feitos por aqueles que têm experiência em desenvolvimento de aplicações no domínio em questão. Esta reutilização evita a recriação e revalidação de soluções comuns para os requisitos de aplicação e para os desafios e problemas de projeto que reaparecem em aplicações diferentes no mesmo domínio. O reuso dos componentes da Estrutura de Aplicação pode levar a uma substancial melhoria na produtividade dos programadores, na melhoria da qualidade, no desempenho, na confiabilidade e na interoperabilidade da aplicação (FAYAD; SCHMIDT; JOHNSON, 1999, p. 8).

#### 2.1.1.3 Extensibilidade

Uma Estrutura de Aplicação melhora a extensibilidade disponibilizando métodos explícitos, chamados de métodos gancho (*hook methods*), que permitem que as aplicações estendam as interfaces estáveis da Estrutura de Aplicação. A função dos métodos gancho é de desacoplar das interfaces e comportamentos os aspectos variáveis específicos de cada aplicação dos aspectos comuns a todas as aplicações do domínio para o qual a Estrutura de Aplicação foi desenvolvida. Estes aspectos variáveis específicos da Estrutura de Aplicação são chamados de pontos de variabilidade (*hot spots*) (SCHMID, 1999 p. 356). O desacoplamento permite que as variações necessárias para uma aplicação particular do domínio possam ser implementadas quando da criação de aplicações específicas do domínio. A extensibilidade da Estrutura de Aplicação é essencial para assegurar a adequação de novos serviços e características necessárias a cada caso particular de uso da Estrutura de Aplicação.

#### 2.1.1.4 Inversão de Controle

A arquitetura em tempo de execução de uma Estrutura de Aplicação é

caracterizada por uma inversão de controle. Esta arquitetura permite que, os passos de processamento específicos da aplicação sejam associados a objetos de tratamento de eventos. Quando um evento ocorre, o mecanismo de despacho reativo da Estrutura de Aplicação reage chamando os métodos gancho (*hook methods*) pré-registrados nestes objetos de tratamento de eventos. Esta inversão de controle permite que a Estrutura de Aplicação determine e tenha controle do fluxo de ações a serem executadas pela aplicação específica. Esta característica é chamada de Princípio de Hollywood: “Não nos chame, nós chamaremos você” (MATTSSON e BOSCH, 1999 p. 471).

### 2.1.2 CLASSIFICAÇÃO

FAYAD, SCHMIDT e JOHNSON (1999, p. 9), fazem uma classificação das Estruturas de Aplicação de duas maneiras, uma levando em conta o espaço de abrangência para o qual são projetadas e outra pelas técnicas usadas para sua extensibilidade.

#### 2.1.2.1 Classificação Segundo a Abrangência

Esta classificação leva em conta o propósito para o qual a Estrutura de Aplicação é projetada.

##### 2.1.2.1.1 Estruturas de Aplicação de Infra-Estrutura de Sistemas

As Estruturas de Aplicação de Infra-Estrutura de Sistemas são destinadas a simplificar o desenvolvimento da infra-estrutura portátil e eficiente de sistemas tais como sistemas operacionais, sistemas de comunicação, interfaces com usuário. Estas Estruturas de Aplicação são utilizadas principalmente na organização interna de *softwares* e não são vendidas diretamente a usuários finais.



#### 2.1.2.1.2 Estruturas de Aplicação para *Middleware*

As Estruturas de Aplicação para *Middleware* são comumente projetadas para integrar aplicações com componentes distribuídos. Estas Estruturas de Aplicação são usadas por aqueles que desenvolvem *software* para modularizar, reusar e estender a infra-estrutura de *softwares* para funcionarem integrados em um ambiente distribuído.

#### 2.1.2.1.3 Estruturas de Aplicação Empresarial

As Estruturas de Aplicação Empresariais destinam-se a amplos domínios de aplicação tais como, telecomunicações, manufatura e finanças, e são peças fundamentais nas atividades de negócios. Estruturas de Aplicação empresariais são caras para desenvolver e comprar, porém podem determinar substanciais retornos de investimento já que suportam diretamente o desenvolvimento de aplicações para o usuário final.

#### 2.1.2.2 Classificação Segundo as Técnicas de Extensibilidade

Esta classificação leva em conta as técnicas usadas para realizar a extensão da Estruturas de Aplicação. Isto é, como adaptá-las para um caso específico do domínio do problema.

##### 2.1.2.2.1 Extensão por *Whitebox*

Estruturas de Aplicação *Whitebox* usam exclusivamente técnicas de orientação a objetos tais como herança e ligação dinâmica para prover mecanismos de extensibilidade (JOHNSON; FOOTE, 1997) e (ROBERTS; JOHNSON, 2000).

Uma funcionalidade é estendida por herança das classes abstratas básicas

definidas na Estrutura de Aplicação e sobrepondo métodos predefinidos, chamados métodos gancho, usando padrões de projeto semelhantes ao Método *Template* e ao Método *Factory* (GAMMA et al., 1995 p. 325). Estas Estruturas de Aplicação requerem dos que desenvolvem a aplicação um conhecimento detalhado da estrutura interna da Estrutura de Aplicação, o que tende que as aplicações sejam fortemente dependentes de detalhes específicos da hierarquia de herança da Estrutura de Aplicação.

#### 2.1.2.2.2 Extensão por *Blackbox*

Estruturas de Aplicação *Blackbox* realizam extensibilidade pela definição de interfaces para componentes cujos objetos podem ser conectados a Estrutura de Aplicação por composição (GAMMA, 1995, p. 19) ou integrando estes componentes usando padrões de projeto tais como *Strategy* (GAMMA et al., 1995 p.315) e *Command* (GAMMA et al., 1995, p. 233). As Estruturas de Aplicação *Blackbox* são estruturadas usando composição de objetos e delegação em vez de herança. Estas Estruturas de Aplicação são geralmente fáceis de usar e estender, no entanto são as mais difíceis de desenvolver porque exigem dos que as desenvolvem a definição de interfaces e métodos ganchos (*hook methods*) os quais devem antecipar uma grande variedade de casos de uso em potencial (SCHMID, 1999, p. 358).

#### 2.1.2.2.3 Extensão por *Graybox*

As Estruturas de Aplicação *Graybox* usam as duas técnicas, *Whitebox* e *Blackbox*, para implementar extensibilidade. As Estruturas de Aplicação *graybox* usam componentes *blackbox* conectados, para esconder detalhes julgados desnecessários de serem conhecidos pelos que desenvolvem as aplicações específicas. As técnicas *whitebox* são utilizadas para expor detalhes que os desenvolvedores devem conhecer para poder adaptar a Estrutura de Aplicação.

### 2.1.3 COMO DESENVOLVER

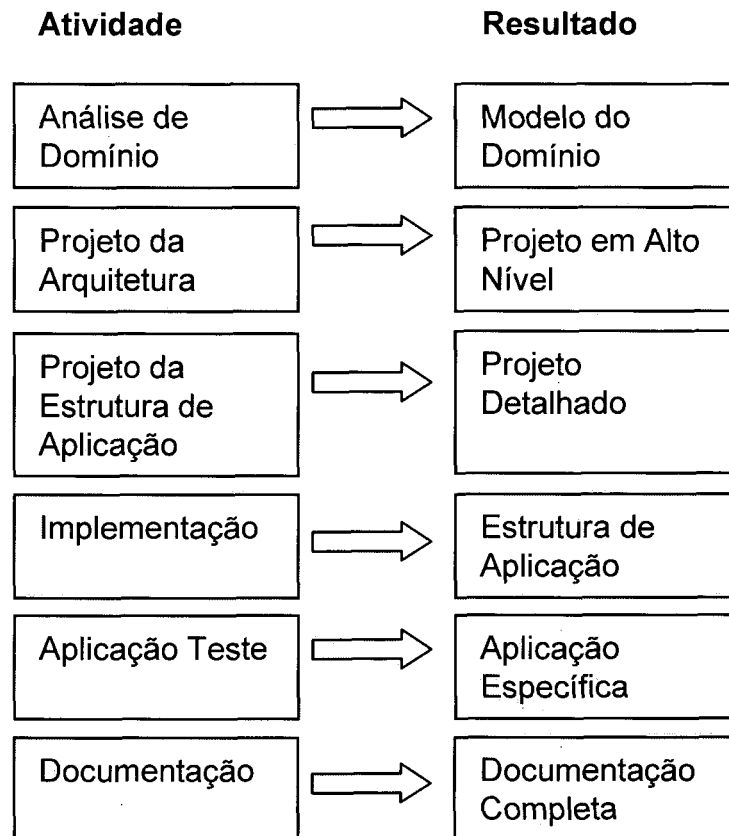
O desenvolvimento de Estruturas de Aplicação é um tanto diferente do desenvolvimento de outras aplicações. O desenvolvimento de Estruturas de Aplicação deve preocupar-se com todos os conceitos relevantes de um domínio, enquanto o desenvolvimento de uma aplicação deve somente se preocupar com os requisitos solicitados para aquela aplicação específica.

BOSCH et al. (1999, p. 62) propõem um modelo com as seguintes atividades para o desenvolvimento de Estruturas de Aplicação Orientada a Objetos: Análise de Domínio, Projeto da Arquitetura, Projeto da Estrutura de Aplicação, Implementação da Estrutura de Aplicação, Testes e Documentação (fig. 1).

#### 2.1.3.1 Análise de Domínio

Esta atividade concentra-se em descrever o domínio que é coberto pela Estrutura de Aplicação. Para capturar os requisitos e identificar os conceitos o analista pode se referir a aplicações previamente desenvolvidas, consultar especialistas e padrões existentes para o domínio. O resultado desta atividade é um modelo do domínio contendo os requisitos, os conceitos, e as relações entre estes conceitos no domínio coberto pela Estrutura de Aplicação.

FIGURA 1 – ATIVIDADES DE DESENVOLVIMENTO



### 2.1.3.2 Projeto da Arquitetura

A principal contribuição de uma Estrutura de Aplicação para uma aplicação é a arquitetura que ela define. A qualidade de uma Estrutura de Aplicação depende diretamente de seu projeto (GOEDICKE; MEYER, 1999, p. 345). Nesta atividade o projetista, usando o modelo da análise do domínio, deve decidir por uma arquitetura adequada para a Estrutura de Aplicação. O projetista, com base na arquitetura selecionada e em outras considerações sobre o modelo resultante da análise de domínio, desenvolve um plano em nível geral para a Estrutura de Aplicação. O resultado desta atividade é um projeto em alto-nível com as principais classes, relacionamentos, funcionalidades e pontos de variabilidade (*hot spots*) da Estrutura de Aplicação.

### 2.1.3.3 Projeto da Estrutura de Aplicação

No Projeto da Estrutura de Aplicação o projeto da arquitetura é refinado, e classes adicionais são incorporadas. O resultado desta atividade é uma descrição detalhada contendo o escopo das funcionalidades, as interfaces para reuso, regras que devem ser obedecidas e soluções adotadas.

### 2.1.3.4 Implementação

A atividade de implementação está relacionada com a codificação das classes abstratas e concretas da Estrutura de Aplicação. O resultado desta atividade é o conjunto de classes que implementam o comportamento básico da Estrutura de Aplicação e expõem os pontos de variabilidade (*hot spots*).

### 2.1.3.5 Aplicação Teste

Esta atividade visa determinar se a Estrutura de Aplicação provê as funcionalidades previstas, e serve também para avaliar a usabilidade da Estrutura de Aplicação. Neste estágio são desenvolvidas aplicações teste usando a Estrutura de Aplicação. O objetivo é decidir se a Estrutura de Aplicação deve ser re-projetada ou está suficientemente madura para ser liberada. Esta atividade requer o desenvolvimento de aplicações que usem a Estrutura de Aplicação. O resultado desta atividade é uma ou mais aplicações específicas construídas utilizando-se a Estrutura de Aplicação desenvolvida.

### 2.1.3.6 Documentação

A documentação da Estrutura de Aplicação é a atividade que produz o

resultado mais importante na construção de Estruturas de Aplicação. Nesta atividade é gerada uma documentação de suporte aos desenvolvedores de aplicações para que estes possam entender como funciona a Estrutura de Aplicação e de como usá-la no desenvolvimento de aplicações específicas. O resultado desta atividade é uma documentação clara e correta, composta de uma descrição do funcionamento da Estrutura de Aplicação e de um manual de usuário descrevendo como utilizá-la no desenvolvimento de aplicações específicas no domínio.

## 2.2 VISUALIZAÇÃO DE OBJETOS EM 3D VIA WEB (WEB 3D)

Segundo o *WEB3D Consortium* (2002), o termo Web3d descreve qualquer linguagem descritiva ou de programação que pode ser usada para troca interativa de modelos de objetos e mundos em 3D (três dimensões) através da Internet. A linguagem *Virtual Reality Modeling Language* (VRML), a extensão Java3D da linguagem Java e a linguagem *Extended 3D* (X3D) foram desenvolvidas com esta perspectiva.

### 2.2.1 VRML

Segundo o *WEB3D Consortium* (2002), VRML (*Virtual Reality Modeling Language*) é uma linguagem usada para descrever objetos e conjuntos de objetos de três dimensões (3D). Estas descrições, chamadas de “mundos”, armazenadas em arquivos texto.

A VRML foi projetada para que estas descrições possam ser utilizadas e visualizadas por meio da Internet, Intranets e sistemas locais. A VRML tem o propósito de ser uma forma padronizada de intercâmbio de dados em 3D e multimídia (WEB3D Consortium, 2002).

As cenas VRML podem ser visualizadas por um *software* visualizador VRML embutido no paginador Web padrão (Internet Explorer, Opera, Netscape, etc.). A

cena VRML e o visualizador VRML são modelados em classes, e são os componentes centrais na arquitetura da estrutura de visualização.

Em VRML, a representação de objetos em 3D utiliza um sistema de coordenadas cartesianas, onde o eixo Y é orientado para cima. O problema básico é o de transformar dados com referência geográfica que não usam um sistema de coordenadas cartesianas, para o sistema de coordenadas cartesianas utilizadas pela VRML. A biblioteca de métodos e classes GeoVRML provê facilidades e funcionalidades para a utilização coordenadas georeferenciadas diretamente dentro de um mundo VRML (GeoVRML.org, 2003). Esta capacidade estende a VRML para utilização em um contexto de visualização global georeferenciado. A GeoVRML suporta 3 sistemas de coordenadas georeferenciadas, *Geodetic Coordinate System* (GDC), *Geocentric Coordinate System* (GCC) e *Universal Transverse Mercator* (UTM), 21 elipsóides e 1 modelo de geóide *World Geodetic System 1984* (WGS84) (GeoVRML.org, 2003).

#### 2.2.1.1 EAI

A VRML não define qualquer interface de programação. A interface externa de Autoria (EAI – *External Authoring Interface*) foi projetada para permitir um ambiente externo para acessar e interagir com os nós em um mundo VRML. Usualmente, a linguagem de programação Java é utilizada para comunicação com mundo VRML por meio da EAI. A razão disto é porque Java é suportada pelos *browsers* Netscape e Microsoft, é orientada a objetos, é independente de plataforma e é largamente utilizada em aplicações para a Internet.

#### 2.2.1.2 VRML e Java

É possível criar aplicações heterogêneas de visualização 3D usando VRML e

Java. Para isto, as aplicações de visualização 3D devem ser divididas em múltiplas partes. Um *plug-in* de VRML é usado para a visualização e interação com a aplicação. Um *applet* Java rodando no mesmo *browser* que o *plug-in* pode acessar o mundo VRML por meio da interface EAI. E, o *applet* Java pode ser projetado para se comunicar com uma ferramenta externa de simulação (KYLÖNEN, 1999).

### 2.2.2 Java3D

Java3D é uma API para construção de aplicações *applets* gráficas 3D interativas em Java (SOWIZRAL; NADEAU, 2002). Java 3D é uma plataforma para a implementação de um ambiente para a criação de aplicações de realidade virtual (WOLTERS, 2002). A API Java3D não define nenhum formato para arquivos de descrição de mundos 3D. Porém, Java3D suporta a definição de componentes para a carga e intercâmbio de arquivos em diferentes formatos em tempo de execução, incluindo VRML.

A criação de aplicações de visualização usando somente a Java3D é possível. A aplicação fica mais leve pela não necessidade de utilização da camada EAI.

Aplicações em Java3D podem ser executadas de forma independente evitando as restrições de segurança imposta pelos *applets*. É possível usar estereoscopia e dispositivos de entrada de imagem de vídeo. Java3D oferece independência de plataforma e tem integração completa com uma grande variedade de outras API's Java.

### 2.2.3 X3D

*Extended 3D* (X3D) é uma linguagem definida nos padrões da *Extended Markup Language* (XML) que inclui todas as características da VRML. X3D é uma arquitetura aberta em desenvolvimento pelo consórcio Web3D para integrar as



tecnologias de criação e visualização gráfica em 3D. A arquitetura é composta de camadas em que no nível mais baixo estão implementados os recursos básicos da VRML para gráficos em 3D. O nível acima é composto das extensões da VRML97: GeoX3D, Java3D, MPEG-4, entre outras, que ampliam as funcionalidades do nível um. A principal motivação para a indústria é a de ter uma representação para visualização e interação com objetos 3D mais leve dentro dos padrões da Web, e assim eliminar a necessidade de sobrecarregar os navegadores com a instalação de diversos *plug-ins*.

### 3 ANÁLISE DE DOMÍNIO

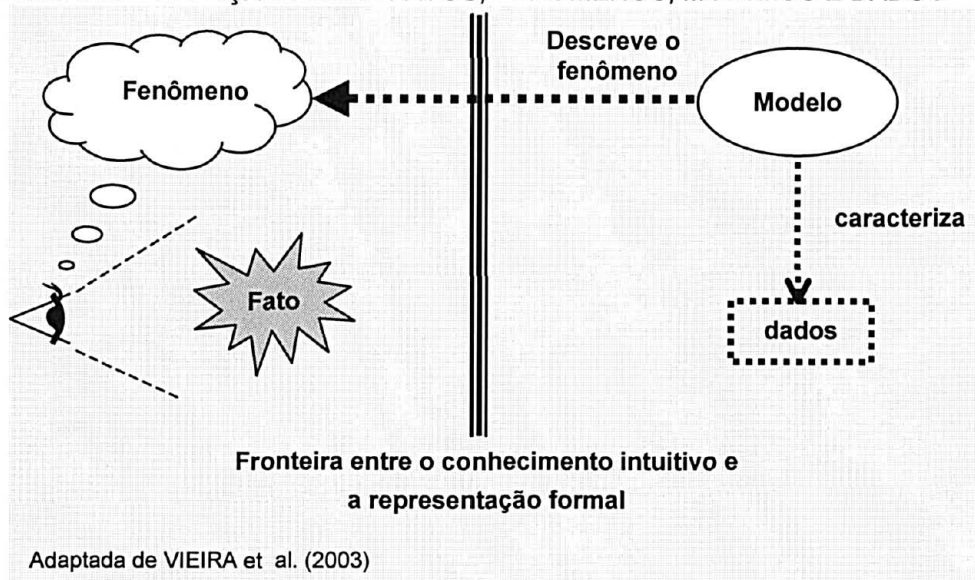
O conteúdo deste capítulo tem como fonte os estudos realizados sobre a representação de objetos que estão sobre a superfície terrestre. A principal fonte é o trabalho que vem sendo realizado sobre este tema por VIEIRA (VIEIRA et al., 2003); (CARMARGO et al., 2003); (LIMA et al., 2003).

#### 3.1 FATOS, FENÔMENOS, DADOS E MODELOS

Fatos são coisas que existem no mundo real e fenômenos são os entendimentos que se tem sobre estes fatos com base em uma certa idealização (VIEIRA, 2003). No contexto do planeta Terra, quando se reconhece que o planeta tem uma superfície tem-se um entendimento de um aspecto do planeta. O entendimento deste aspecto é o fenômeno chamado de superfície. Sobre esta superfície ocorrem fatos que, baseados em outros entendimentos, dão origem a outros fenômenos.

Para que um fenômeno represente um fato de forma precisa é necessário que se determine um conjunto de propriedades que o caracterize. A estas propriedades são atribuídos valores. O conjunto de propriedades com seus respectivos valores caracterizam os dados sobre o fenômeno. Um modelo é uma descrição das propriedades dos fenômenos tendo em vista um certo ponto de vista (fig. 2) (VIEIRA, 2003).

FIGURA 2 – RELAÇÃO ENTRE FATOS, FENÔMENOS, MODELOS E DADOS



### 3.2 FENÔMENOS TERRESTRES

Fenômenos Terrestres são entendimentos que se tem de fatos que ocorrem na superfície terrestre. A *superfície terrestre* representa a parte externa do planeta Terra. Sobre esta superfície existem alguns fenômenos (oceanos, rios, lagos, etc.) que são caracterizados por possuírem uma forma que se amolda à superfície terrestre criando regiões em que a superfície fica submersa, sendo denominados de *corpos hídricos* (fig. 3). As regiões em que a superfície não fica submersa é denominada de *superfície topográfica*. Os fatos que ocorrem sobre a *superfície topográfica* são chamados de *fenômenos topográficos*. Dada esta interpretação os fenômenos terrestres podem ser classificados em três categorias distintas: *superfície topográfica*, *fenômeno topográfico* e *corpo hídrico*.

FIGURA 3 – SUPERFÍCIE TERRESTRE



Adaptada de VIEIRA et al. (2003)

A organização dos fenômenos terrestres em uma estrutura conceitual que os classifica e os define como um conjunto de atributos e operações que podem ser aplicados sobre eles, define um modelo conceitual formal (VIEIRA et al., 2003).

### 3.3 O MODELO DE FENÔMENOS TERRESTRES

O modelo de fenômenos terrestres é um modelo conceitual formal para descrição fenômenos terrestres proposto por VIEIRA et al. (2003). A descrição do modelo é feita pelo uso do conceito de classes de objetos. Neste modelo um fenômeno terrestre é descrito por dois grupos de atributos. Um grupo diz respeito à semântica do fenômeno e o outro ao espaço ocupado pelo fenômeno e sua superfície.

#### 3.3.1 A Semântica

A semântica caracteriza o tipo de fenômeno terrestre dentro da categoria que ele pertence, por exemplo: lago, edificação ou outro tipo de fenômenos terrestres. A cada tipo está associado um conjunto de atributos que caracterizam os objetos daquele tipo. Por exemplo, para os objetos do tipo Imóvel pode se ter como atributos: registro, proprietário e o tipo do imóvel. Para objetos do tipo Solo:

composição, acidez, impermeabilidade e uso.

Os atributos de um fenômeno terrestre podem variar no tempo. Para descrever o aspecto temporal os atributos são representados por três elementos: nome, valor e o instante no tempo em que foi observado este valor.

### 3.3.2 A Superfície

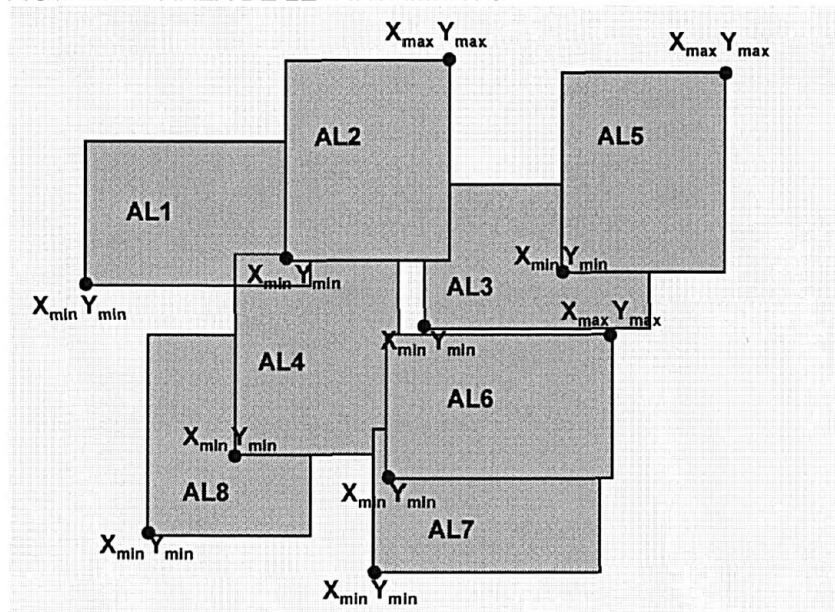
O espaço ocupado por um fenômeno terrestre é o espaço ocupado pela sua superfície e esta é composta por um conjunto de faces planas conexas. Isto se verifica quando se deseja descrever um fenômeno terrestre em sua forma tridimensional (3D). Entretanto, existem situações em que se pode admitir que um fenômeno terrestre tem uma dimensionalidade, 0D (ponto), 1D (linha) ou 2D (plano). Por exemplo, é possível se adotar uma dimensionalidade 1D para representar uma rede de drenagem numa certa interpretação, na qual são relevantes somente as conexões e os caminhos pela rede. Entretanto, em uma outra interpretação, pode ser necessário ter, além das conexões e caminhos, as margens dos rios representadas em verdadeira grandeza, na dimensionalidade 2D ou mesmo em 3D. Uma dimensionalidade 0D pode ser usada para representar a sede administrativa de cada cidade dentro de um mapa estadual. Neste caso, com apenas um ponto geográfico e os atributos semânticos, podem caracterizar a sede no espaço e no tempo. Portanto, a dimensionalidade está relacionada com a idealização do fenômeno terrestre e não com a posição geográfica deste fenômeno que é dada por uma lista de uma ou mais coordenadas geográficas.

A superfície de um fenômeno terrestre pode variar no tempo, assim, um atributo tempo é incorporada à descrição da superfície para registrar o instante no tempo em que foi observada.

### 3.3.3 Área de Levantamento

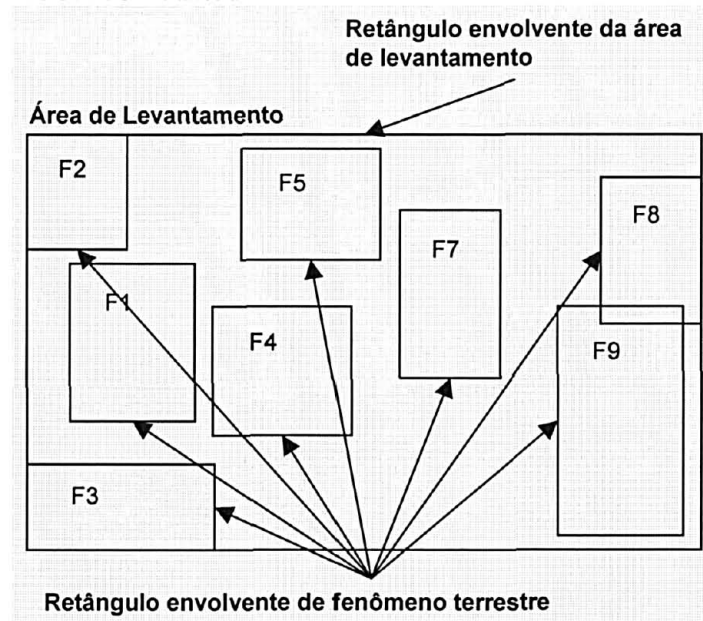
Área de Levantamento é um conjunto de dados obtidos sobre fenômenos terrestres. Uma Área de Levantamento é caracterizada por uma região retangular e data da obtenção dos dados. A região retangular é representada pela diagonal de pontos topográficos com coordenadas ( $X$  mínimo,  $Y$  mínimo) e ( $X$  máximo,  $Y$  máximo). Estas coordenadas são obtidas pela inspeção de todos os pontos topográficos de todos os fenômenos (fig. 4).

FIGURA 4 – ÁREA DE LEVANTAMENTO



A região retangular de uma Área de Levantamento define um Retângulo Envolvente que contém um ou mais fenômenos terrestres. Cada fenômeno terrestre tem o seu próprio Retângulo Envolvente que é obtido pela inspeção de todos os pontos topográficos do fenômeno (fig. 5).

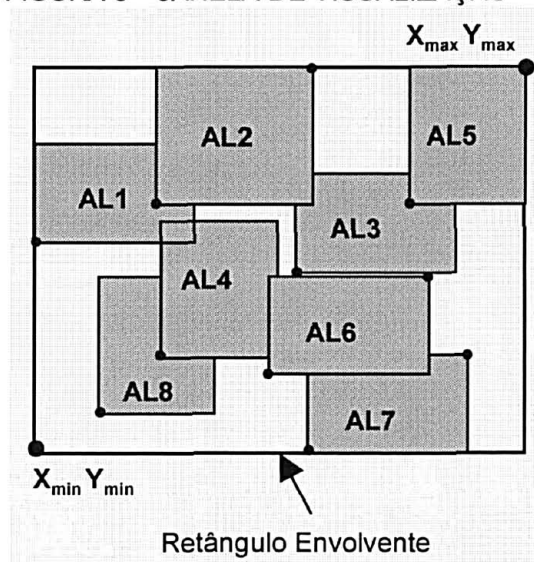
FIGURA 5 – RETÂNGULO ENVOLVENTE



### 3.3.4 Janela de Visualização

A Janela de Visualização é uma área definida pelo usuário para visualizar um fenômeno terrestre ou uma composição de fenômenos terrestres (fig. 6). A Janela de Visualização é dada pelo usuário por meio da especificação das coordenadas geográficas da diagonal do Retângulo Envolvente no qual deverão estar contidos os fenômenos terrestres que deseja visualizar.

FIGURA 6 – JANELA DE VISUALIZAÇÃO



### 3.4 O MODELO DE FENÔMENOS TERRESTRES EM UML

O modelo de fenômenos terrestres é descrito aqui pela utilização da *Unified Modeling Language* (UML). A UML é a linguagem gráfica padronizada internacionalmente para visualização, especificação, construção e documentação de sistemas baseados em *software* (BOOCH; RUMBAUGH; JACOBSON, 1999).

#### 3.4.1 A Classe Fenômeno Terrestre

A classe **FenomenoTerrestre** representa todos os fenômenos que ocorrem sobre a superfície terrestre. Um fenômeno terrestre é descrito pelas seguintes propriedades (fig. 7):

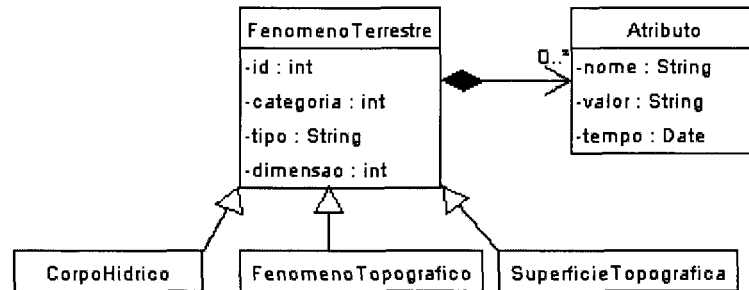
- a) **tipo**: define a semântica do fenômeno: imóvel, rua, construção, entre outros dentro da categoria do fenômeno;
- b) **dimensão**: indica a dimensionalidade da forma espacial do fenômeno, isto é, se a sua forma geométrica é composta de 0, 1, 2 ou 3 dimensões;
- c) **atributos**: conjunto de triplas com nome, valor e tempo para cada característica do fenômeno;
- e) **superfície**: descrição geométrica da forma espacial do fenômeno associada a um instante de observação que a define no tempo.
- f) **retEnvolvente**: representa as coordenadas geográficas do menor retângulo em que o fenômeno está contido.

As categorias de fenômenos: *corpo hídrico*, *fenômeno topográfico* e *superfície topográfica* são modeladas como sub-classes da classe **FenomenoTerrestre**. A sub-classe **CorpoHidrico** representa a categoria de fenômenos que estão sobre a superfície terrestre e são constituídos de massa líquida. Esta classe agrupa os tipos de fenômenos tais como: oceanos, lagos, rios entre outros. A sub-classe **SuperfícieTopografica** representa as áreas da superfície terrestre não cobertas por corpos hídricos em geral. A sub-classe **FenomenoTopografico** representa os



fenômenos terrestres que estão sobre a superfície topográfica tais como: árvores, edificações, vias dentre outros.

FIGURA 7 – A CLASSE *FenomenoTerrestre*



### 3.4.2 A Classe Superfície

A superfície de um fenômeno terrestre é um objeto complexo (fig. 8). A representação desta propriedade é dada por um conjunto de classes que através de seus relacionamentos e associações definem as superfícies possíveis para os fenômenos terrestres em geral.

A classe **Ponto** é a classe central nesta definição. Esta classe representa um ponto geográfico baseado no sistema de coordenadas geocêntricas. As suas propriedades são:

- a) **x, y, z**: valores da coordenada geocêntrica do ponto;
- b) **medido**: indica se o ponto foi observado em loco ou se teve origem a partir de uma simulação ou cálculo;
- c) **topográfico**: indica se o ponto está sobre a superfície topográfica;

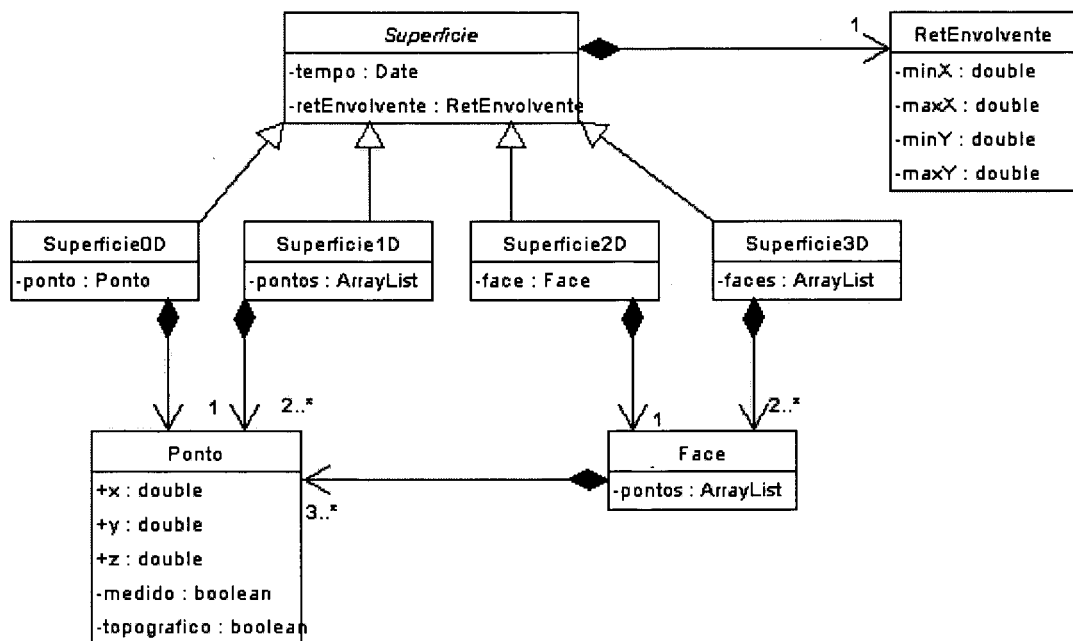
A classe **Superfície** representa as propriedades comuns das quatro dimensionalidades (0D, 1D, 2D, 3D) possíveis para um fenômeno terrestre. Estas propriedades são:

- a) **tempo**: instante no tempo em que a superfície foi observada e medida;

b) **retEnvolvente**: pontos topográficos que definem o menor retângulo em que a superfície está contida.

As dimensionalidades estão representadas pelas classes **Superficie0D**, **Superficie1D**, **Superficie2D** e **Superficie3D**. A classe **Superficie0D** representa as superfícies pontuais definidas por uma instância da classe **Ponto**. A classe **Superficie1D** representa as superfícies lineares definidas por uma seqüência de duas ou mais instâncias da classe **Ponto**. A classe **Superficie2D** representa as superfícies planas definidas por uma instância da classe **Face**. A classe **Superficie3D** representa as superfícies multifacetadas definidas por um conjunto de duas ou mais instâncias da classe **Face**. A classe **Face** representa uma região plana definida por uma seqüência de instâncias da classe **Ponto** as quais determinam os vértices da face. Os pontos de uma face devem ser co-planares e não coincidentes.

FIGURA 8 – A CLASSE *Superficie*



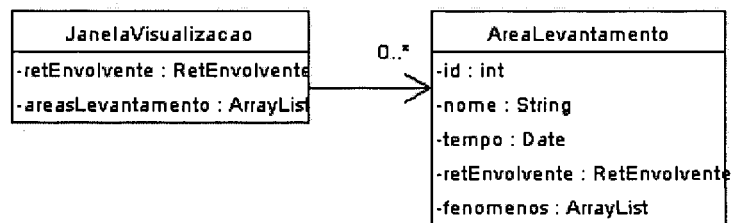
### 3.4.3 A Classe Área de Levantamento e a Classe Janela de Visualização

A classe **AreaLevantamento** representa uma Área de Levantamento com

seus fenômenos terrestres, retângulo envolvente e tempo (fig. 9).

A classe **JanelaVisualizacao** representa uma Janela de Visualização definida pelo usuário por meio das coordenadas de um retângulo envolvente (fig. 9). Esta classe agrupa todas as áreas de levantamento cujos retângulos envoltentes estejam dentro do retângulo envolvente definido pela Janela de Visualização.

FIGURA 9 – AS CLASSES AREALEVANTAMENTO E JANELAVISUALIZACAO



## 4 PROJETO DA ARQUITETURA

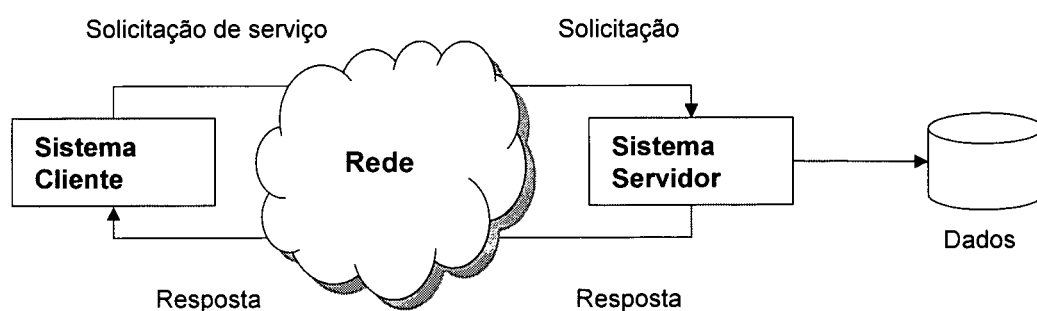
Uma arquitetura de *software* organiza grupos de classes em uma solução coerente, e mostra como as partes se relacionam entre si. A ênfase é na estrutura e nos relacionamentos entre as partes componentes, e não nos detalhes de cada parte (ARRINGTON, 2001).

A arquitetura apresentada nesta seção define um modelo de estrutura de *software* para visualização de fenômenos terrestres em 3D via Web. O modelo propõe uma estrutura de classes baseada no paradigma de projeto Modelo-Visualização-Control (MVC) e no modelo de arquitetura cliente/servidor para disponibilizar os serviços de acesso e visualização via Web.

### 4.1 A ARQUITETURA CLIENTE/SERVIDOR

Uma arquitetura cliente/servidor (fig. 10) é composta de um subsistema “servidor” que provê serviços para múltiplas instâncias de subsistemas “cliente”. O cliente se conecta ao servidor através de uma rede e solicita serviços ao servidor. O servidor provê os serviços de acesso aos dados e mantém a integridade dos dados. Para suportar o aumento da demanda por serviços podem existir diversos servidores (MURPHY, 2001).

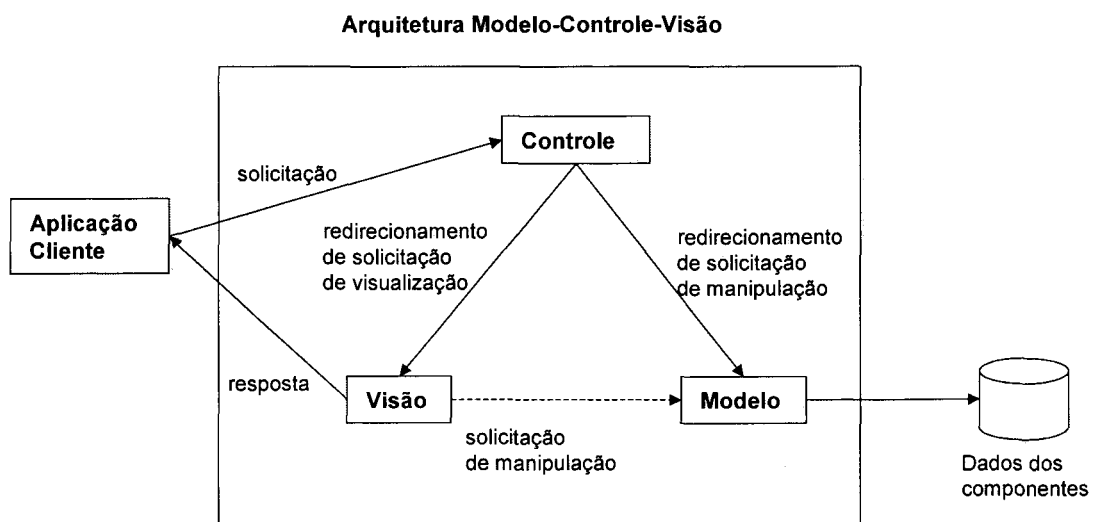
FIGURA 10 – PARADIGMA CLIENTE/SERVIDOR  
Arquitetura Cliente/Servidor



## 4.2 O PARADIGMA MODELO-VISÃO-CONTROLE (MVC)

O paradigma Modelo-Visão-Controlador (MVC) (fig. 11) é um modelo clássico de organizar aplicações que necessitam manter múltiplas visualizações dos mesmos objetos. O paradigma MVC faz uma separação clara das responsabilidades dos objetos de uma aplicação em três categorias: *modelo*, objetos com a responsabilidade de manterem os dados, as regras de acesso e a lógica de manipulação dos componentes de uma aplicação; *visão*, objetos que têm a responsabilidade de apresentar visualizações dos componentes da aplicação em diferentes maneiras; *controlador* objetos que têm a responsabilidade no tratamento de solicitações e eventos de acesso, manipulação e visualização dos componentes, encaminhando-as para os objetos do tipo *modelo* ou *visão* apropriados. Em razão desta separação, múltiplos objetos dos tipos *visão* e *controlador* podem acessar os objetos do tipo *modelo* sem que haja a necessidade de alteração do projeto destes objetos, os quais representam a lógica e o estado dos principais componentes da aplicação (GAMMA et al., 1995, p. 4); (ALLMARAJU et al., 2001, p. 525); (KURNIWAN, 2002, p. 397); (AHMED; UMRYSH, 2002, p. 139).

FIGURA 11 – PARADIGMA MVC

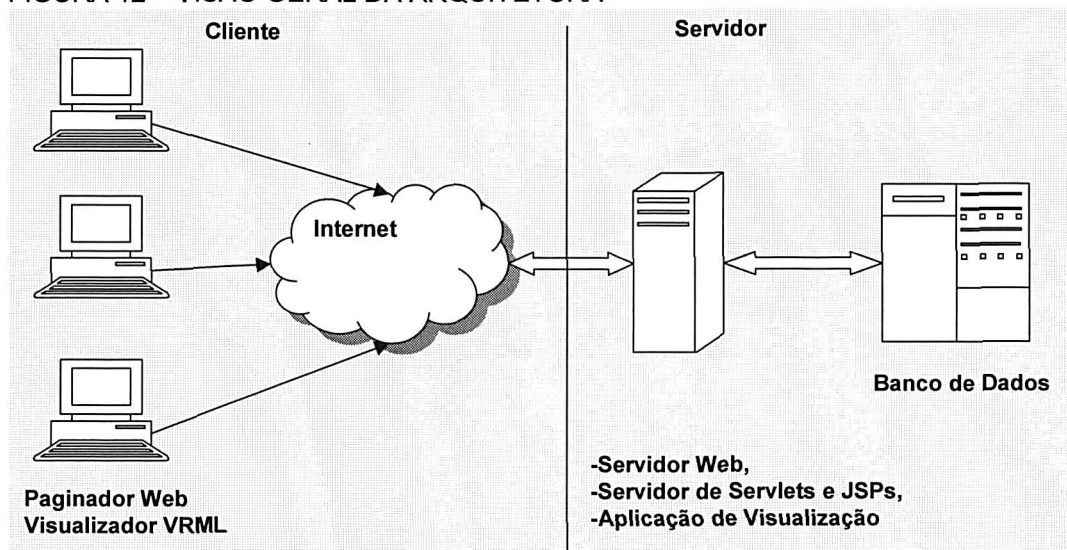


### 4.3 A ARQUITETURA DA ESTRUTURA DE APLICAÇÃO

A arquitetura da estrutura de aplicação de visualização é composta por paginador Web em um computador cliente que faz solicitações de serviços de visualização a uma aplicação de visualização instalada em um servidor Web (fig. 12). A aplicação para visualização de fenômenos terrestres é construída baseada no paradigma MVC e na utilização das tecnologias *Java 2 Platform, Enterprise Edition* (J2EE) para desenvolvimento de aplicações Web. A razão da escolha do J2EE é justificada nos seguintes aspectos:

- a) Existe uma boa quantidade de produtos e ferramentas no mercado, tanto para o lado do servidor como do cliente, para o desenvolvimento de aplicações e, grande parte delas são de acesso livre e com suporte fonte aberto (*open source*);
- b) A especificação J2EE cobre todos os aspectos que a plataforma Microsoft oferece;
- c) J2EE é independente de plataforma, pode-se desenvolver em um ambiente Windows ou Linux e distribuir a aplicação para ser executada em ambientes HP-UX, Solaris, IBM-AIX sem problemas.

FIGURA 12 – VISÃO GERAL DA ARQUITETURA



A aplicação de visualização é organizada segundo o paradigma MVC. Os componentes do tipo controle, em geral um único para a aplicação, são responsáveis pela execução da aplicação. Todas as solicitações de visualização passam pelo controle, e este tem a responsabilidade de fazer todo o tipo de validação de segurança e de conteúdo realizando algum tipo de pré-processamento antes de delegar o trabalho para componentes visão ou modelo. Em J2EE estes componentes são *servlets* ou JSP's. Os componentes do tipo visão mostram os resultados das solicitações de visualização. Estes componentes, em J2EE são páginas *Java Server Pages* (JSP) ou *servlets* que geram páginas HTML ou VRML como resposta às solicitações para enviar ao paginador Web. Os componentes do tipo modelo são os que realizam o trabalho real da aplicação. Estes componentes representam a lógica central da aplicação e mantêm os dados da aplicação e as regras de manipulação dos fenômenos terrestres armazenados em um banco de dados.

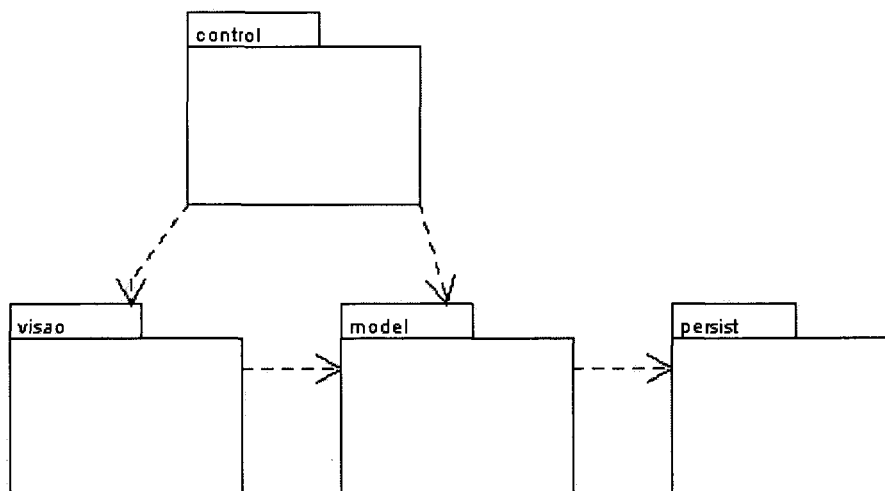
## 5 PROJETO DA ESTRUTURA DE APLICAÇÃO

O projeto é um detalhamento da arquitetura da estrutura de aplicação proposta, especificando em detalhe as classes e componentes necessários para a construção da estrutura de aplicação. O projeto é descrito por meio de um conjunto de pacotes e suas dependências. O conceito de pacote permite agrupar classes semelhantes ou classes relacionadas simplificando o entendimento e a implementação do sistema.

A estrutura de aplicação é uma composição de quatro pacotes de classes com os nomes **control**, **model**, **visao** e **persist**. A figura 13 mostra os pacotes e suas dependências. As classes do pacote **model** representam os componentes do tipo modelo que fazem o acesso e tratamento dos fenômenos terrestres segundo o modelo de fenômenos terrestre. O pacote **persist** é um pacote de apoio ao pacote **model** provendo classes para os serviços de persistência dos objetos criados pelos componentes do pacote **model**. As classes do pacote **visao** representam os componentes do tipo visão e têm a responsabilidade de gerar cenas VRML e páginas HTML para atender as solicitações de serviços. O grupo de classes do pacote **control** representa os componentes do tipo controle, os quais são responsáveis pelo controle da aplicação recebendo e encaminhando solicitações de serviços de visualização (CAMARGO et al., 2003).

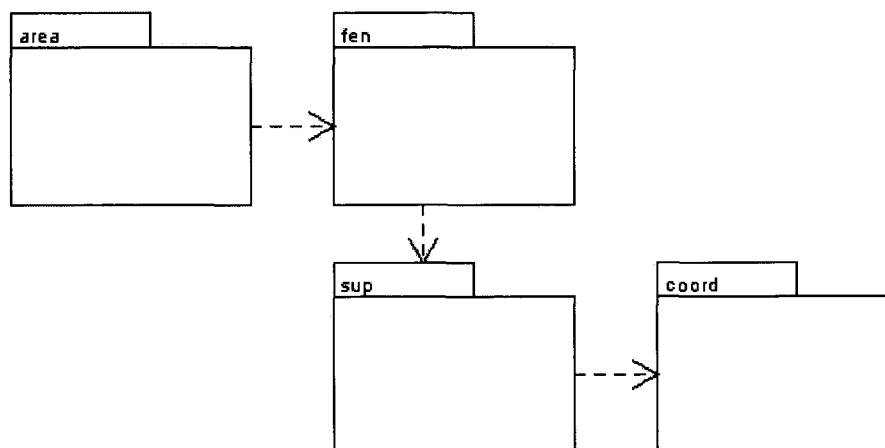


FIGURA 13 – A ESTRUTURA DE PACOTES



### 5.1 O PACOTE DO MODELO **model**

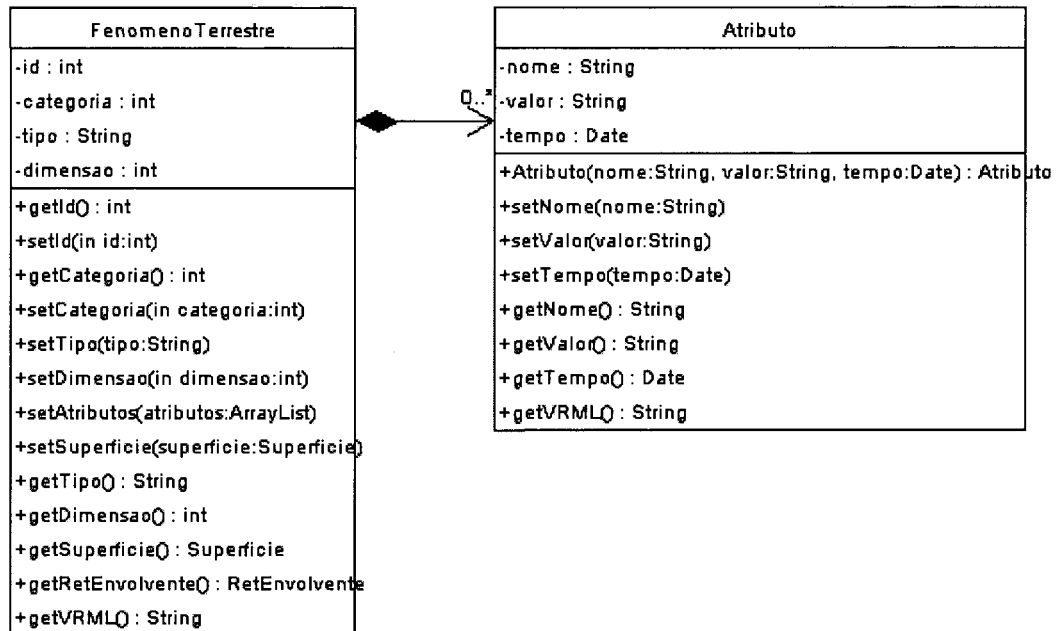
O pacote **model** (fig. 14) é composto de três pacotes de classes, os pacotes: **fen**, **sup**, **coord** e **area**. O pacote **fen** agrupa classes básicas da definição de fenômeno terrestre e é dependente do pacote **sup** que contém as classes necessárias para a representação da superfície do fenômeno terrestre. O pacote **coord** define um grupo de classes para representar os diferentes sistemas de coordenadas geográficas. O pacote **area** agrega classes que representam grupos de fenômenos terrestres.

FIGURA 14 – O PACOTE MODELO **model**

### 5.1.1 O pacote fenômeno **fen**

O pacote **fen** (fenômeno) (fig. 15) é constituído de classes que representam os fenômenos terrestres em geral.

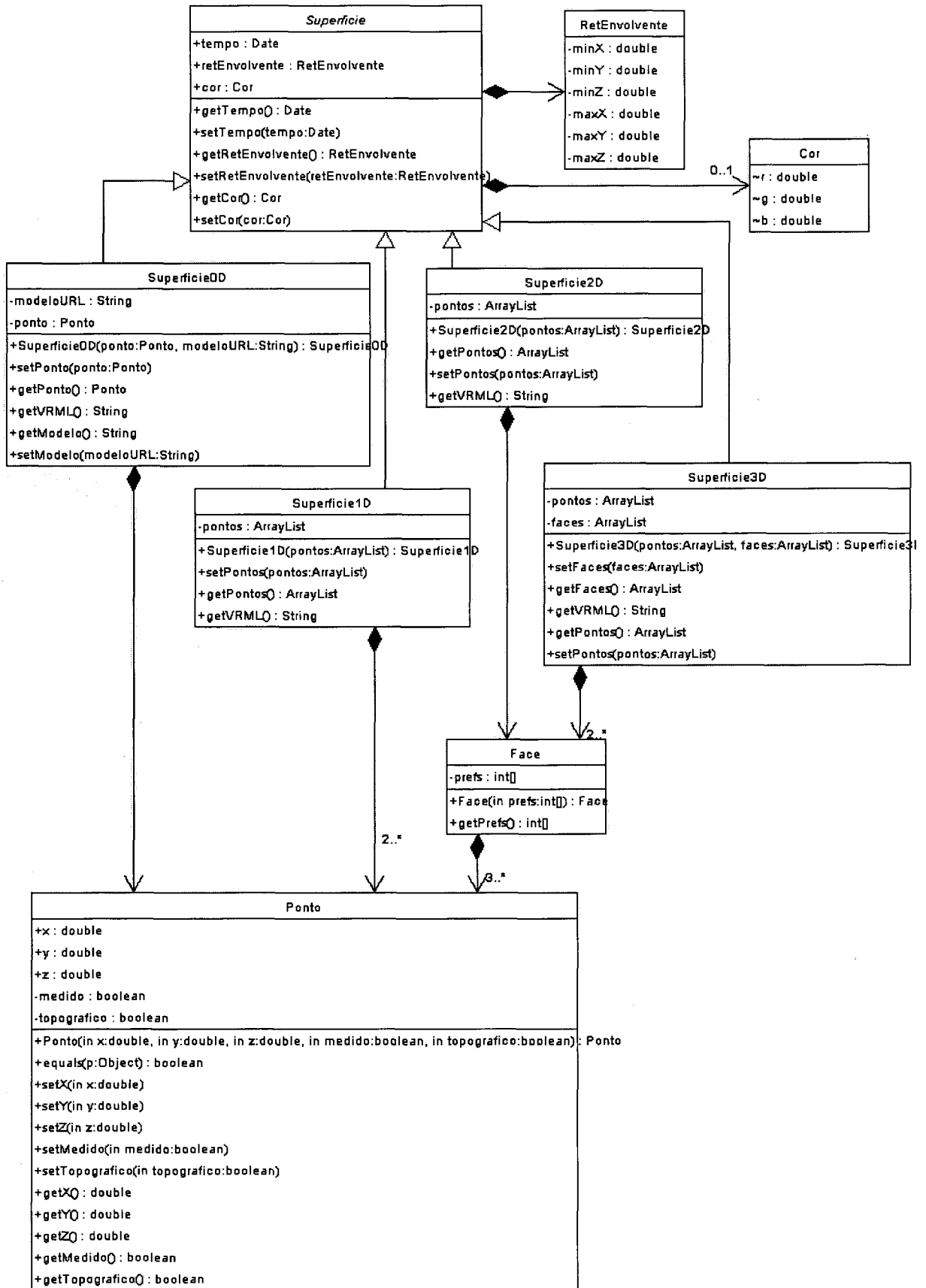
FIGURA 15 – O PACOTE FENÔMENO TERRESTRE **fen**



### 5.1.2 O pacote superfície **sup**

O pacote **sup** (superfície) (fig. 16) contém as classes que representam as diversas de formas de superfícies de um fenômeno terrestre.

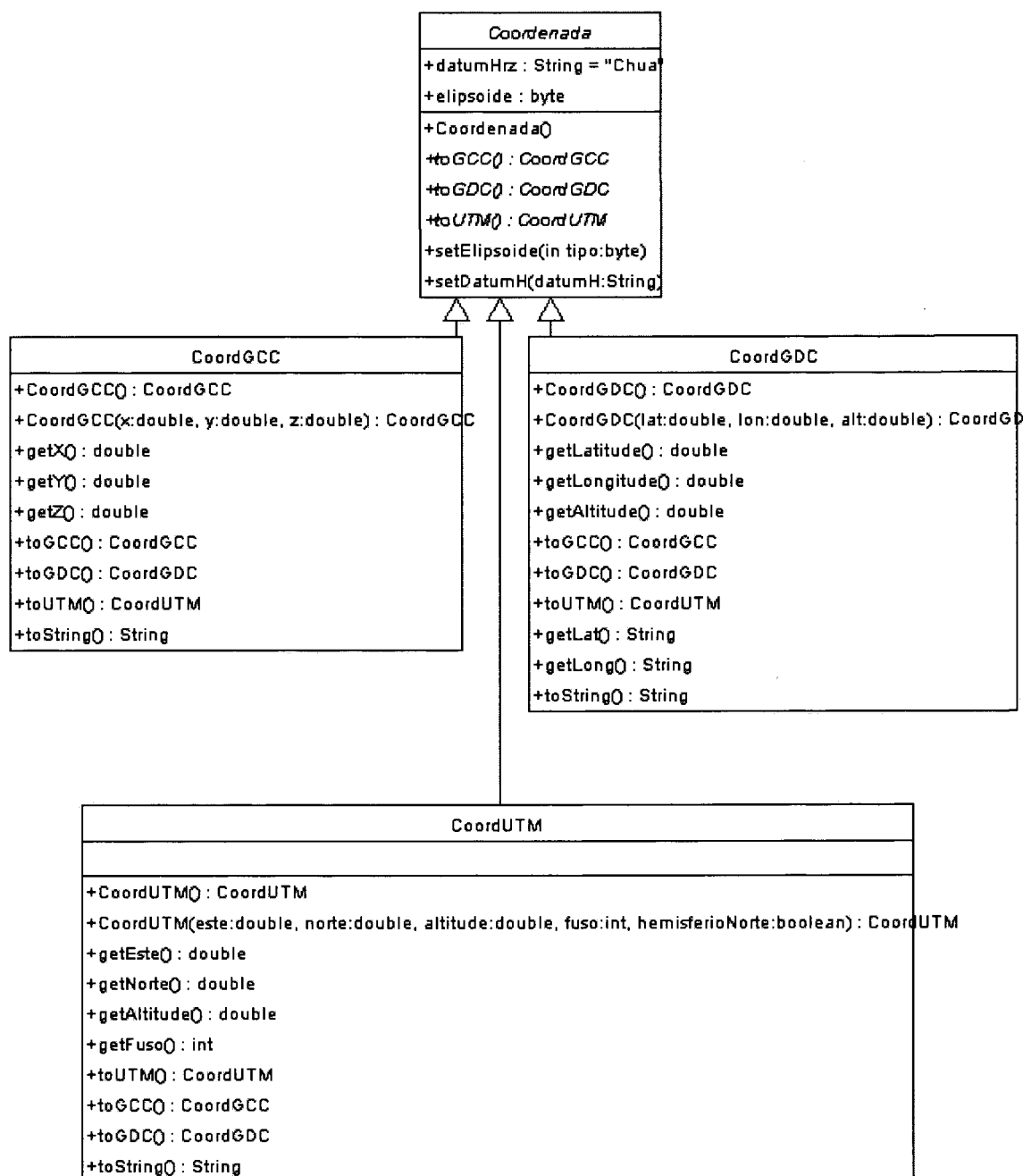
FIGURA 16 – O PACOTE SUPERFÍCIE TERRESTRE sup



### 5.1.3 O pacote coordenadas **coord**

As classes neste pacote representam os sistemas de coordenadas geográficas: *Geodetic Coordinate System* (GDC), *Geocentric Coordinate System* (GCC) e *Universal Transverse Mercator* (UTM) que dão suporte à representação de pontos e a conversão entre sistemas de coordenadas.

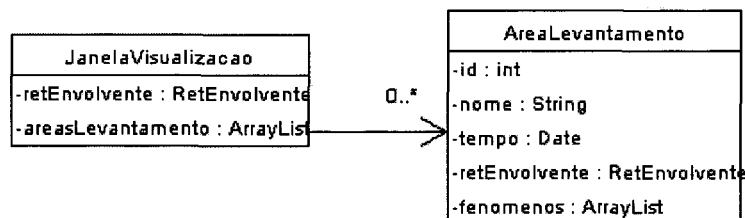
FIGURA 17 – O PACOTE COORDENADAS **coord**



#### 5.1.4 O pacote área **area**

Este pacote é composto das classes que representam grupos de fenômenos terrestres (fig. 18).

FIGURA 18 – O PACOTE ÁREA **area**



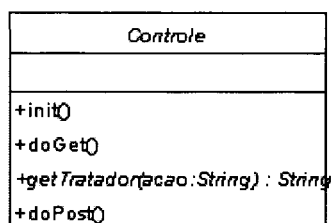
#### 5.2 O PACOTE VISÃO **visao**

Este pacote é destinado a manter as classes e componentes que geram visualizações do modelo. O desenvolvedor da aplicação final deve definir dentro deste pacote classes e componentes de visualização específicos da aplicação que está desenvolvendo.

#### 5.3 O PACOTE CONTROLE **control**

O pacote controle é destinado a conter as classes que representam o componente de controle da aplicação sendo desenvolvida (fig. 19). A classe abstrata **Controle** é responsável pelo controle da aplicação para receber e encaminhar solicitações de serviços.

FIGURA 19 – O PACOTE CONTROL



#### 5.4 O PACOTE PERSISTÊNCIA **persist**

Este pacote contém as classes e interfaces que fazem acesso ao banco de dados contendo os objetos do modelo. A classe BdFTBean (fig. 20) é a interface para persistência dos dados sobre áreas de levantamento e fenômenos terrestres. As classes AreaLevantamentoBD e FenomenoTerrestreBD (fig. 21) representam os dados sobre áreas de levantamento e fenômenos terrestres. As classes AreaLevantamentoParser, AreaLevantamentoHandler, FenomenoTerrestreParser, FenomenoTerrestreHandler (fig. 22) implementam os métodos necessários para leitura e armazenamento de áreas de levantamento informadas em documentos em formato XML.

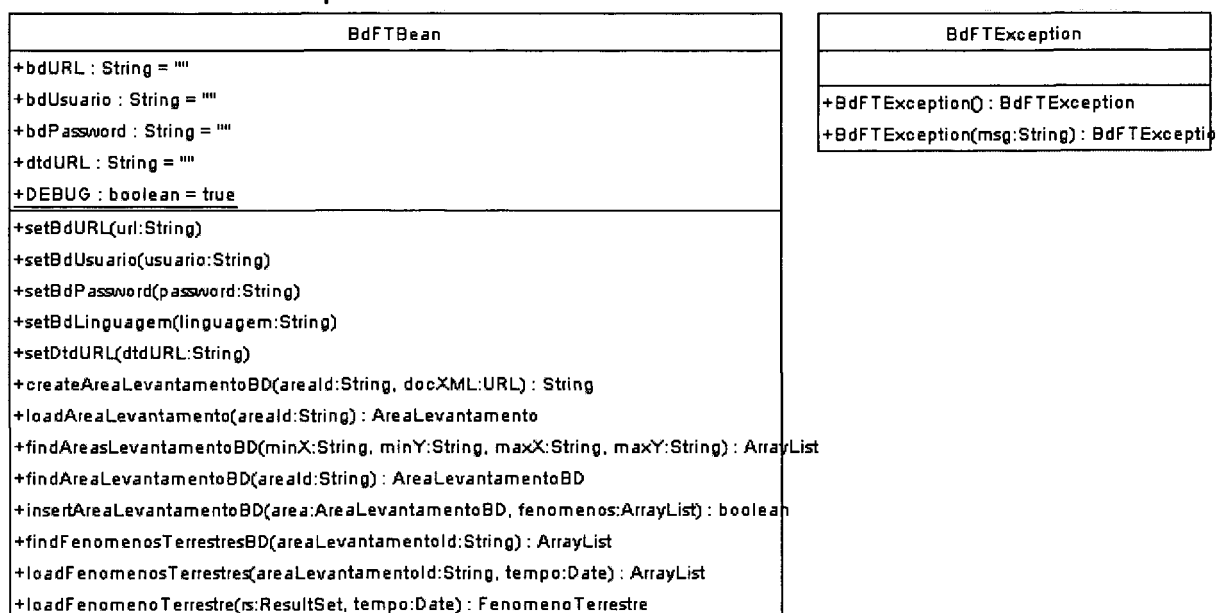
FIGURA 20 – O PACOTE **persist** - CLASSE BdFTBean

FIGURA 21 – O PACOTE persist –CLASSES DE ACESSO A ÁREAS DE LEVANTAMENTO

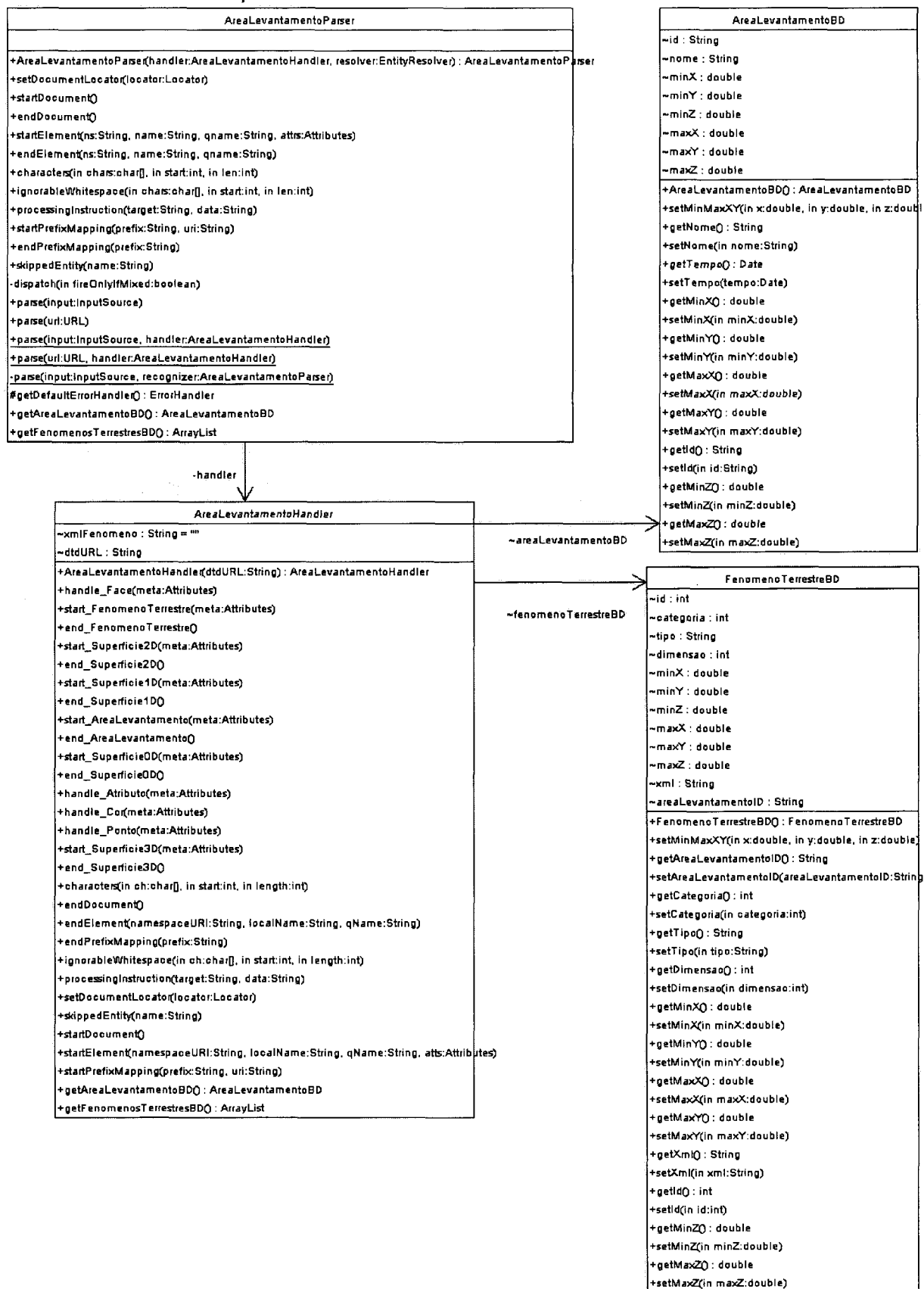
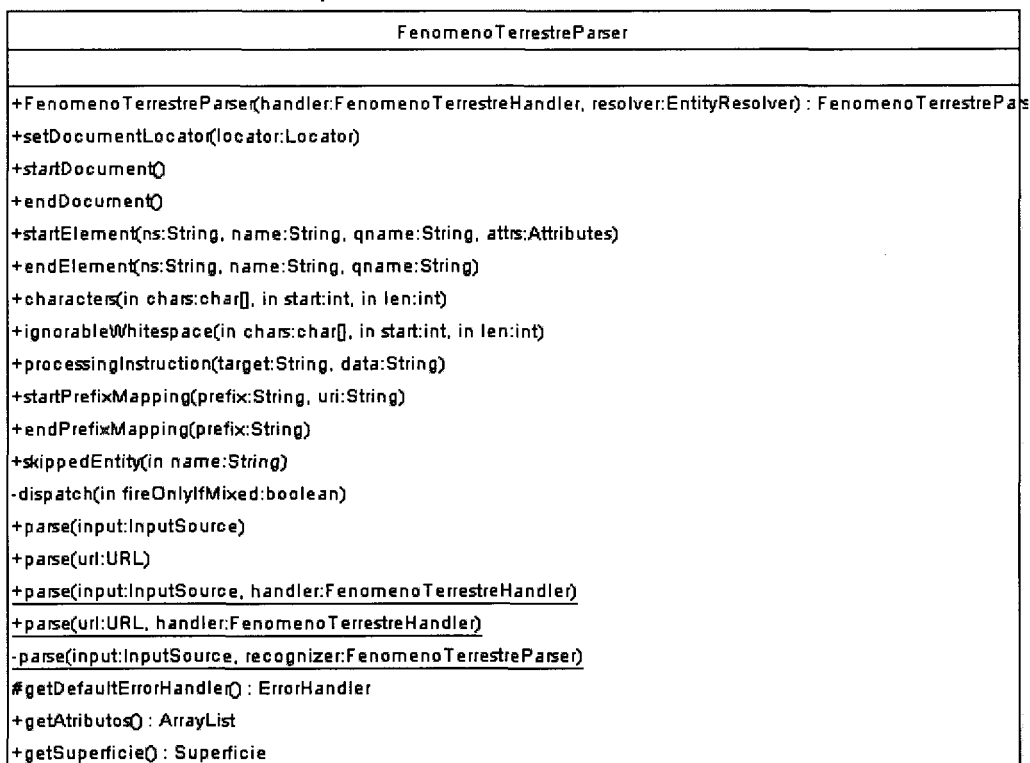


FIGURA 22 – O PACOTE persist - CLASSES DE ACESSO A ÁREAS DE LEVANTAMENTO



-handler





## 6 IMPLEMENTAÇÃO

Este capítulo mostra a implementação da estrutura de aplicação projetada no capítulo anterior. É mostrada a implementação das classes para obtenção dos dados, do modelo de fenômenos terrestres, do controle da aplicação e de visualização.

### 6.1 OS DADOS

Os dados sobre os fenômenos terrestres são capturados por meio de arquivos texto contendo a descrição de áreas de levantamento. A descrição é feita por meio da linguagem XML (*Extend Markup Language*). As descrições são então armazenadas em um banco de dados relacional.

A persistência dos dados das áreas de levantamento é feita em duas tabelas, **AreasLevantamento** (fig. 23) e **Fenomenos** (fig. 24) de um banco de dados relacional Microsoft Access.

FIGURA 23 – ESTRUTURA DA TABELA **AreasLevantamento**

Coluna	Tipo de dados
id	Auto Numeração
nome	Texto
tempo	data
minX	Número de precisão dupla
minY	Número de precisão dupla
maxX	Número de precisão dupla
maxY	Número de precisão dupla

A tabela **AreasLevantamento** consiste das colunas:

- a) **id**: um número identificador único para cada área de levantamento;

- b) **nome**: descritivo da área;
- c) **tempo** momento do registro da observação;
- d) **minX, minY, maxX e maxY**: coordenadas do retângulo envolvente da área de levantamento;

FIGURA 24 – ESTRUTURA DA TABELA **Fenômenos**

Coluna	Tipo de dados
id	Auto Numeração
categoria	Número inteiro
tipo	Texto
dimensao	Número inteiro
minX	Número de precisão dupla
minY	Número de precisão dupla
maxX	Número de precisão dupla
maxY	Número de precisão dupla
xml	Memorando
areaLevantamentold	Número inteiro

A tabela **Fenomenos** consiste das colunas:

- a) **id**: um número identificador único para cada fenômeno terrestre dentro do sistema;
- b) **categoria**: indicador da categoria do fenômeno 0-Fenômeno Topográfico, 1- Corpo Hídrico, 2 – Superfície Topográfica;
- c) **tipo**: o nome do tipo de fenômeno terrestre: casa, rua, lago ou outro;
- d) **dimensão**: dimensão da forma do fenômeno terrestre 0, 1, 2 ou 3 dimensões;
- e) **minX, minY, maxX e maxY**: coordenadas do retângulo envolvente do fenômeno terrestre;
- f) **xml**: texto em XML com a descrição do fenômeno terrestre;

- g) **areaLevantamentoid**: referência para a linha da tabela **AreasLevantamento** na qual está descrita área de levantamento que contem o fenômeno.

O banco de dados para áreas de levantamento está representado pela classe **BdFTBean** do pacote **persist**. Esta classe é responsável por qualquer acesso ao banco de dados.

Uma área de levantamento com seus fenômenos, é uma estrutura de dados complexa. A *Extended Markup Language* (XML) é uma meta-linguagem utilizada para definir outras linguagens e foi escolhida para definir e descrever áreas de levantamento.

Um documento escrito em XML deve respeitar certas regras de formato e gramática, estas regras são expressas em um texto chamado Definição de Tipo de Documento (DTD) ou esquema. O DTD para área de levantamento é dado na figura 25.

A operação **createAreaLevantamento(String docXML)** da classe **BdFTBean** tem a responsabilidade de verificar se o documento que descreve uma área de levantamento contido no arquivo, cujo nome é dado em **docXML**, é válido. Se o conteúdo for válido, cria e armazena a área de levantamento nas tabelas **AreasLevantamento** e **Fenomenos** do banco de dados.

FIGURA 25 – ESTRUTURA DE DOCUMENTO XML PARA DESCREVER UMA ÁREA DE LEVANTAMENTO

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT AreaLevantamento (FenomenoTerrestre+)>
<!ATTLIST AreaLevantamento
    id NMTOKEN #REQUIRED
    nome CDATA #IMPLIED
    tempo CDATA #REQUIRED
>
<!ELEMENT FenomenoTerrestre (Cor?, Atributo*, (Superficie3D | Superficie0D
| Superficie1D | Superficie2D))>
<!ATTLIST FenomenoTerrestre
    categoria (SuperficieTopografica | FenomenoTopografico |
CorpoHidrico) #REQUIRED
    tipo NMTOKEN #IMPLIED
    dimensao (0 | 1 | 2 | 3) #REQUIRED
>
<!ELEMENT Atributo EMPTY>
```

```

<!ATTLIST Atributo
  nome NMTOKEN #REQUIRED
  valor CDATA #REQUIRED
  tempo CDATA #IMPLIED
>
<!ELEMENT Superficie3D (Ponto+, Face+)>
<!ELEMENT Superficie2D (Ponto+)>
<!ELEMENT Superficie1D (Ponto+)>
<!ELEMENT Superficie0D (Ponto)>
<!ATTLIST Superficie0D
  modeloVRML NMTOKEN #IMPLIED
>
<!ELEMENT Ponto EMPTY>
<!ATTLIST Ponto
  x NMTOKEN #REQUIRED
  y NMTOKEN #REQUIRED
  z NMTOKEN #REQUIRED
  medido (S | N) #FIXED "S"
  topografico (S | N) #FIXED "S"
>
<!ELEMENT Face EMPTY>
<!ATTLIST Face
  prefs NMTOKENS #REQUIRED
>
<!ELEMENT Cor EMPTY>
<!ATTLIST Cor
  r NMTOKEN #REQUIRED
  g NMTOKEN #REQUIRED
  b NMTOKEN #REQUIRED

```

## 6.2 O MODELO

O modelo de fenômenos terrestres é implementado de acordo com as definições feitas no projeto. Em todas as classes são implementados as operações **get** e **set** para cada atributo. Este critério permite que as classes possam ser tratadas como *JavaBeans*.

## 6.3 O CONTROLE

A classe **Controle** (fig. 26) é uma classe abstrata e é a peça central da estrutura de aplicação. Esta classe é um *servlet* que implementa o método **doPost()** como um método *template* (GAMMA et al., 1995 p. 325). O método *template* usa o método gancho **getTratador(String ação)** para determinar a página HTML, JSP ou *servlet* apropriado que trata a ação solicitada ao controle. O método gancho é um

método abstrato, e deve ser implementado em uma subclasse da classe **Controle** na aplicação final.

FIGURA 26 – O CÓDIGO FONTE DA CLASSE CONTROLE

```

package com.vft3d.vftserv.control;

import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;
import java.util.*;
import com.vft3d.vftserv.persist.BdFTBean;

public abstract class Controle extends HttpServlet {

    /** Inicializa as variáveis globais */
    public void init(ServletConfig config) throws ServletException {
        System.out.println("inicializando o controlador.");
        ServletContext context = config.getServletContext();
        context.setAttribute("baseURL", config.getInitParameter("baseURL"));
        context.setAttribute("imageURL", config.getInitParameter("imageURL"));

        // cria uma instância da classe BdFTBean para acesso ao banco de dados
        BdFTBean bdBean = new BdFTBean();
        // inicializa os campos do BdFTBean
        bdBean.setBdURL(config.getInitParameter("bdURL"));
        bdBean.setBdUsuario(config.getInitParameter("bdUsuario"));
        bdBean.setBdPassword(config.getInitParameter("bdPassword"));

        // Coloca o BdFTBean no contexto do servlet
        // Para que o BdFTBean possa ser acessado a partir das paginas JSP
        context.setAttribute("bdBean", bdBean);

        // Carrega o driver jdbc do banco de dados
        try {
            Class.forName(config.getInitParameter("jdbcDriver"));
        }
        catch (ClassNotFoundException e) {
            System.out.println(e.toString());
        }
        super.init(config);
    }

    /**Processa as solicitações request Get do HTTP */
    public void doGet(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException
    {
        doPost(request, response);
    }

    /** Operacao primitiva chamada pelo metodo template doPost.
    * Esta operacao deve ser implementada pela aplicação.
    * Recebe uma string com um nome de acao e retorna o nome
    * da pagina JSP ou HTML que responde a acao solicitada.
    */
    public abstract String getTratador(String acao);

```

```

/**Procesa a solicitação request Post do HTTP
 * este metodo é um metodo template que usa a operação
 * getTratador() para executar a acao solicitada como parametro
 * no Post request enviado pela Web.
 */
public void doPost(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException
{
    String base = "/jsp/"; // local default para as paginas JSP
    String url = base + "Default.jsp"; // pagina jsp default

    String acao = request.getParameter("acao");

    if (acao!=null) {
        url = base + getTratador(acao);
    }

    RequestDispatcher requestDispatcher;
    requestDispatcher = getServletContext().getRequestDispatcher(url);
    requestDispatcher.forward(request, response);
}
}

```

## 6.4 A VISUALIZAÇÃO

A classe **FenomenoTerrestre** e suas subclasses implementam a interface VRML. Esta interface determina uma operação chamada **getVRML()** a qual retorna uma *string* contendo a representação na linguagem VRML do objeto que a classe representa. Desta forma, a representação visual de qualquer fenômeno terrestre fica encapsulada na classe que o representa e é implementada na estrutura de aplicação.

A criação de visualizações pelas aplicações é feita por meio de páginas JSP. Estas páginas usam a instância da classe **BDFTBean** criada pelo controle para fazer acesso aos dados sobre o modelo de fenômenos terrestres e as representações em VRML dos fenômenos terrestres. As páginas JSP devem ser disponibilizadas dentro do diretório **jsp** de distribuição da aplicação. Páginas JSP que geram uma cena VRML devem conter obrigatoriamente na primeira linha de texto a seguinte declaração:

```
<%@ page contentType="model/vrml"%>#VRML V2.0 utf8
```

Este procedimento é necessário para que o visualizador VRML entenda que o texto é uma cena VRML. A figura 27 mostra o exemplo do código fonte de uma página que gera uma cena VRML para visualização de áreas de levantamento pertencentes a uma janela de visualização. Uma versão simplificada do método **getVRML()** implementado na classe **AreaLevantamento** é dado na figura 28.

FIGURA 27 – EXEMPLO DE PÁGINA JSP COM VRML

```
<%@ page contentType="model/vrml"%>#VRML V2.0 utf8
<jsp:useBean id="bdBean" scope="application"
class="com.vft3d.vftserv.persist.BdFTBean"/>
<%@ page import="com.vft3d.vftserv.model.fen.AreaLevantamento" %>
<%@ page import="java.sql.*" %>
<%@ page import="java.util.*" %>
<%
String baseURL = (String) application.getAttribute("baseURL");
%>
<%
String minX = request.getParameter("minX");
String minY = request.getParameter("minY");
String maxX = request.getParameter("maxX");
String maxY = request.getParameter("maxY");
if (
    (minX!=null && !minX.trim().equals(""))&&
    (minY!=null && !minY.trim().equals(""))&&
    (maxX!=null && !maxX.trim().equals(""))&&
    (maxY!=null && !maxY.trim().equals(""))
)
{
%>
<%
    ArrayList areasLevantamento =
bdBean.getJanelaVisualizacao(minX,minY,maxX,maxY);
    Iterator iterator = areasLevantamento.iterator();
    while (iterator.hasNext()) {
        AreaLevantamento areaLevantamento = (AreaLevantamento)
iterator.next();
%>
        <%=areaLevantamento.getVRML()%>
<%
    }
} else
    out.println("Entre com coordenadas válidas.");
%>
```

FIGURA 28 – UM EXEMPLO DE IMPLEMENTAÇÃO DO MÉTODO `getVRML()`

```
public String getVRML() {
    String larg, alt;
    larg=Double.toString(maxX-minX);
    alt =Double.toString(maxY-minY);
    String vrml = "\nDEF AL" + Integer.toString(id) + " Transform {\n"
        +"\t children [\n"
        +"\t\t Shape {\n"
        +"\t\t\t appearance Appearance {\n"
        +"\t\t\t\t material Material { diffuseColor 0.1 0.4 0.4 }\n"
        +"\t\t\t\t }\n"
        +"\t\t\t geometry Box { size " + larg + " "+alt + " 0.002 }\n"
        +"\t\t }\n"
        +"\t ]\n"
        +"}\n";
    return vrml;
}
```



## 7 APLICAÇÃO TESTE

Neste capítulo é desenvolvida uma aplicação exemplo utilizando a estrutura de aplicação apresentada nos capítulos anteriores.

### 7.1 PROPÓSITO

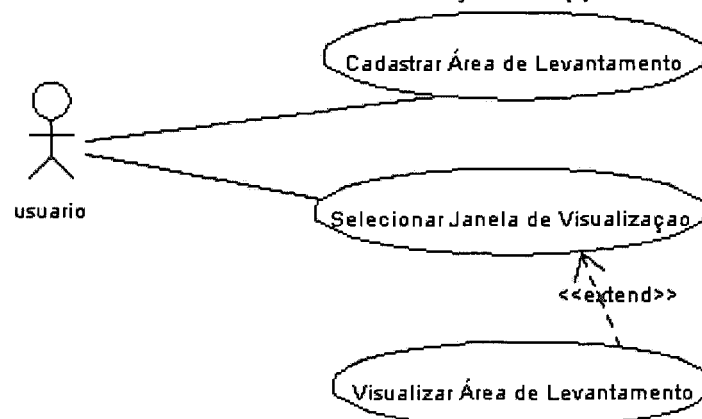
O modelo para representação de fenômenos terrestres é um modelo que está sob experimentação. Para atender às necessidades iniciais de estudos sobre o modelo surgiu a necessidade que uma aplicação para cadastramento e visualização de áreas de levantamento fosse desenvolvida. A aplicação chamada de **vftapp**, descrita a seguir, foi então desenvolvida para suprir esta necessidade e testar a arquitetura proposta.

### 7.2 IDENTIFICAÇÃO DE CASOS DE USO

Os seguintes casos de uso foram identificados como requisitos básicos para a aplicação (fig. 29):

- a) Cadastrar área de levantamento;
- b) Selecionar área de visualização;
- c) Visualizar área de levantamento.

FIGURA 29 – CASOS DE USO DA APLICAÇÃO **vftapp**

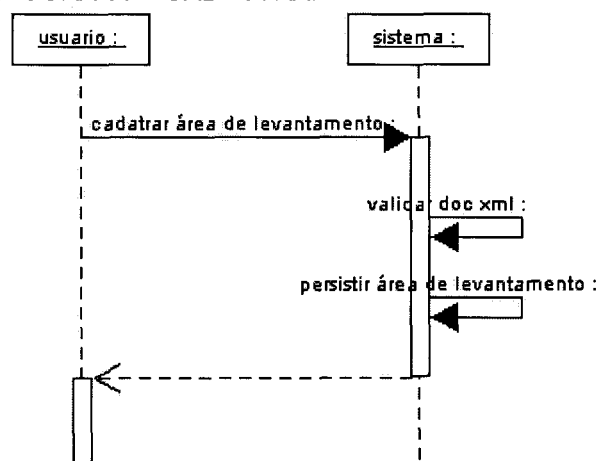


### 7.2.1 Caso de uso: Cadastrar Área de Levantamento

Fluxo principal de eventos: O caso de uso inicia quando o usuário entra na página principal da aplicação e seleciona a opção: Cadastrar Área de Levantamento. O sistema então apresenta ao usuário uma tela solicitando dados sobre a área de levantamento que deseja cadastrar e o nome do arquivo que contém a descrição em XML desta área. O usuário submete os dados e o nome do arquivo solicitado e o sistema responde com uma tela a confirmação que a área de levantamento foi cadastrada (fig. 30).

Fluxo alternativo de eventos: O sistema não é capaz de cadastrar a área de levantamento. Uma descrição do problema encontrado é então enviada ao usuário.

FIGURA 30 – CADASTRAR ÁREA DE LEVANTAMENTO



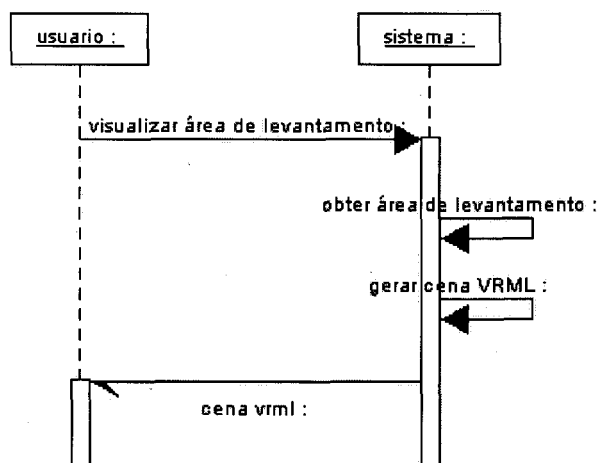
### 7.2.2 Caso de uso: Visualizar Área de Levantamento

Fluxo principal de eventos: O caso de uso se inicia quando o usuário seleciona com sucesso uma janela de visualização, ver caso de uso Selecionar Janela de Visualização, e seleciona uma área de levantamento para visualizar. O sistema obtém a área de levantamento solicitada e gera uma página com a cena VRML contendo todos os fenômenos da área. Uma página HTML com dados necessários para que o visualizador VRML do usuário mostre a cena VRML é então

gerada e enviada (fig. 31).

Fluxo alternativo de eventos: O sistema não é capaz de obter a área de levantamento solicitada. Uma descrição do problema encontrado é enviada ao usuário.

FIGURA 31 – VISUALIZAR ÁREA DE LEVANTAMENTO

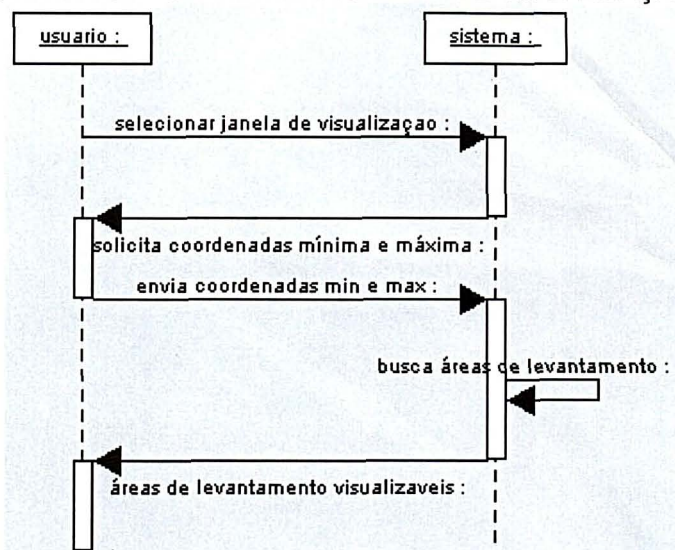


### 7.2.3 Caso de uso: Selecionar Janela de Visualização

Fluxo principal de eventos: O caso de uso inicia quando o usuário entra na página principal da aplicação e seleciona a opção: *Selecionar Janela de Visualização*. O sistema apresenta então ao usuário uma tela solicitando as coordenadas do retângulo no qual as áreas de levantamento que deseja visualizar devem estar contidas. O usuário envia as coordenadas solicitadas e o sistema apresenta uma lista com dados das áreas de visualização que estão dentro da janela solicitada (fig. 32).

Fluxo alternativo de eventos: O sistema não é capaz de obter as áreas de levantamento solicitadas. Uma descrição do problema encontrado é enviada ao usuário.

FIGURA 32 – SELECIONAR JANELA DE VISUALIZAÇÃO



### 7.3 A INTERFACE COM O USÁRIO

A interface com o usuário é implementada por meio de páginas HTML. Para consistência no aspecto das páginas, cada página usa o mesmo design. Há um cabeçalho no alto da página, um menu à esquerda e uma área para o conteúdo específico.

O modelo das páginas é dado a seguir.

FIGURA 33 - PÁGINA PRINCIPAL DA APLICAÇÃO

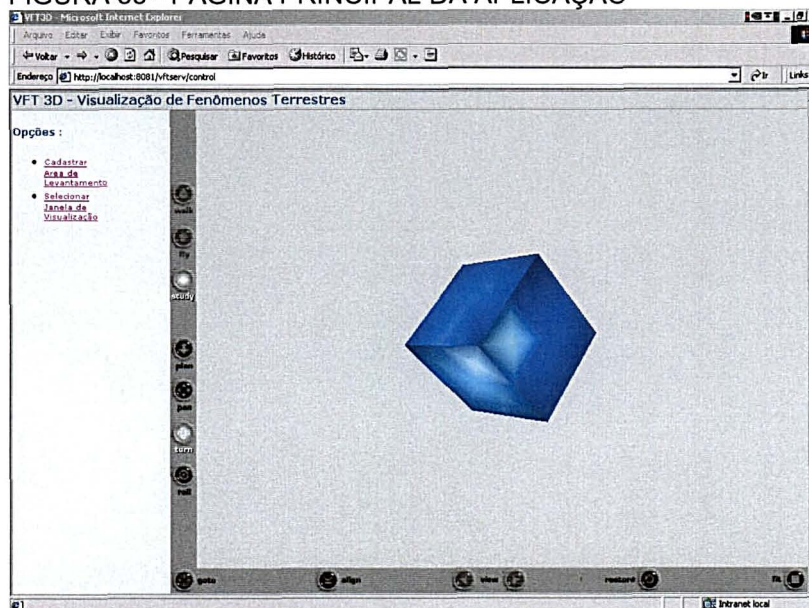


FIGURA 34 – PÁGINA CADASTRAR ÁREA DE LEVANTAMENTO

VFT 3D - Visualização de Fenômenos Terrestres

Opções:

- Cadastro Área de Levantamento
- Selecionar Janela de Visualização

**Cadastro de Área de Levantamento**

Nome do Área: AL01

Descrição em XML: C:\vft3d\ymldoc\AL01.xml

Procurar...

Enviar

Procurar...

FIGURA 35 – PÁGINA SELECIONAR JANELA DE VISUALIZAÇÃO

VFT 3D - Visualização de Fenômenos Terrestres

Opções:

- Cadastro Área de Levantamento
- Selecionar Janela de Visualização

**Selecionar Janela de Visualização**

	Latitude	N/S	Longitude	E/O
Coordenada mínima	24	Sul	49	Oeste
Coordenada máxima	26	Sul	48	Oeste

Enviar

FIGURA 36 – PÁGINA ÁREAS DE LEVANTAMENTO



FIGURA 37 – PÁGINA DE VISUALIZAÇÃO DE ÁREA DE LEVANTAMENTO

## 7.4 INSTALAÇÃO E USO

A instalação da Estrutura de Aplicação e da aplicação exemplo está descrita nos apêndices 1-Instalação da Estrutura de Aplicação e 2-Instalação da Aplicação Teste VFTapp1 respectivamente. Nestes apêndices estão contidos todos os arquivos fontes de classe, dados, demais recursos e instruções necessárias para o funcionamento da aplicação exemplo.

## 8 CONCLUSÃO

Este trabalho mostra que é possível disponibilizar visualizações de fenômenos terrestres utilizando a Web. A descrição de fenômenos terrestres em um documento XML é um aspecto promissor que este trabalho desperta para a troca de dados sobre fenômenos terrestres. A utilização da tecnologia JSP na criação e composição de cenas VRML no desenvolvimento de aplicações em VRML é uma contribuição importante deste trabalho. O modelo para a representação de fenômenos terrestres é a principal contribuição dada.

### 8.1 RESULTADOS

A tarefa de determinar o modelo conceitual para fenômenos terrestres não foi uma tarefa trivial e teve diversas iterações para se chegar a um modelo simples, consistente e significativo. Esta tarefa levou cerca de 60% do tempo do desenvolvimento da estrutura de aplicação. É importante dizer que uma alteração no modelo conceitual pode levar a alterações significativas na Estrutura de Aplicação, e por conseqüência nas aplicações desenvolvidas que a utilizam.

A diversidade de tecnologias envolvidas na construção da Estrutura de Aplicação exigiu um grande esforço de aprendizagem e experimentação para compreender, instalar, integrar e utilizar cada tecnologia. No entanto a utilização de tecnologias centradas em JAVA permitiu que a aprendizagem adquirida em uma tecnologia fosse aproveitada no entendimento das outras tecnologias.

Os dados sobre áreas de levantamento usados na aplicação **vftapp** foram obtidos de arquivos de levantamentos já existentes no Departamento de Geomática da UFPR. Os dados sobre fenômenos topográficos são coletados e registrados por meio de pontos georeferenciados. Cada ponto tem sua localização determinada por três componentes altitude, longitude e elevação. Estes dados são geralmente



visualizados em um ambiente 2D. Porém, quando visualizados em um ambiente 3D aparecem as inconsistências existentes na determinação da componente elevação de cada ponto. Esta inconsistência faz que os fenômenos topográficos apareçam, em alguns casos, acima da superfície topográfica e em outros abaixo desta. A aplicação desenvolvida se mostrou útil para revelar estas inconsistências nos dados

O esforço para desenvolvimento da aplicação teste **vftapp** ficou principalmente na especificação das páginas HTML com código JSP embutido para gerar as visualizações. A figura 38 mostra um comparativo entre o número de linhas de código Java, HTML e JSP escritos na Estrutura de Aplicação e na aplicação **vftapp**, e a figura 39 mostra uma comparação entre o número de classes e páginas JSP desenvolvidas.

FIGURA 38 – COMPARATIVO DE LINHAS DE CÓDIGO JAVA, HTML, JSP

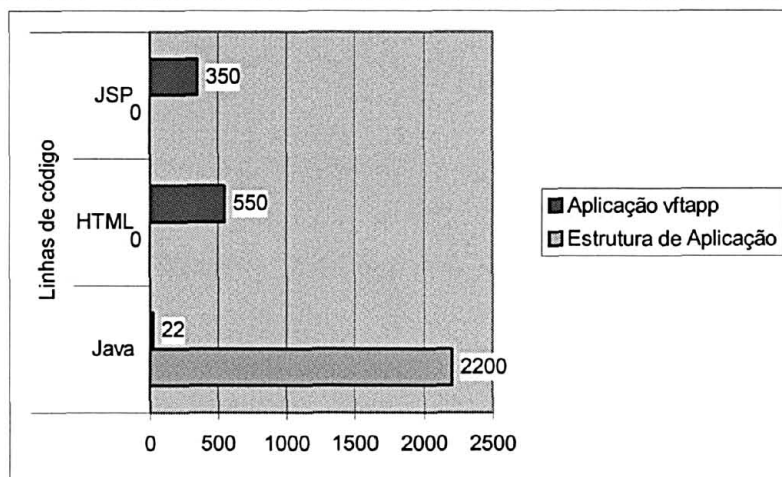
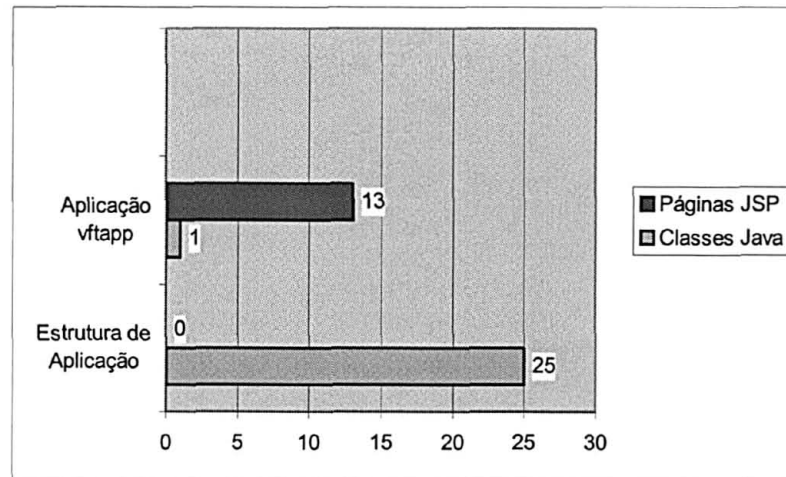


FIGURA 39 – COMPARATIVO NUMERO DE CLASSES JAVA E PÁGINAS JSP



## 8.2 TRABALHOS FUTUROS

Alguns novos estudos e desenvolvimentos podem ser iniciados a partir das idéias, conceitos e mecanismos apresentados neste trabalho. Entre eles:

- a) Construção de biblioteca de *tags* JSP específica para geração de páginas com cenas em VRML a partir do modelo de fenômenos terrestres;
- b) Inclusão de mecanismos mais sofisticados de interação do usuário com a cena VRML além dos permitidos pelos visualizadores VRML. A utilização da interface EAI para manipular a cena no computador cliente poderá permitir melhorias na interação e performance na utilização das visualizações geradas;
- c) O desenvolvimento de um sistema distribuído para o registro, edição, visualização e troca de levantamentos de fenômenos terrestres via Web.
- d) A persistência dos fenômenos terrestre em formato de texto XML consome recursos de processamento significativos a cada visualização gerada. Uma melhoria na forma de fazer o armazenamento da descrição dos fenômenos terrestres é uma melhoria desejável.

## REFERÊNCIAS

- AHMED, K. Z.; UMRYSH, C. E. **Desenvolvendo Aplicações Comerciais em Java com J2EE e UML**. Rio de Janeiro: Ciência Moderna, 2002.
- ALLAMARAJU, S.; ASHRI, R.; DARBY, C.; FLENNER, R.; KARSJENS, T.; KERZNER, M.; KROTOV, A.; LINDE, A.; MacINTOSH, J.; McGOVERN, J.; MIRCHANDANI, T.; PLASTER, B.; REAMY, D. SARANG, P. G.; WRITZ, D. **Professional Java E-Commerce**. Birmingham, UK: Wrox, 2001.
- ARRINGTON, CT. **Enterprise Java with UML**. New York: John Wiley & Sons, Inc. 2001.
- BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **The Unified Modeling Language User Guide**. Reading, MA: Addison-Wesley-Longman, Inc., 1999.
- BOSCH, J.; MOLIN, P.; MATTSSON, M.; BENGTSSON, P.; FAYAD, M. Frameworks Problems and Experiences. In: FAYAD, M. E.; SCHMIDT, D. C.; JOHNSON, R. E. (Ed.). **Building Application Frameworks: Object-Oriented Foundations of Framework Design**. New York: J. Wiley, 1999. p. 55-82.
- CAMARGO, N. F.; CARVALHO, C. A. P.; VIEIRA, A. J. B.; LIMA, J. D. Modelo de Visualização de Fenômenos Terrestres em 3D. In: III Colóquio Brasileiro de Ciências Geodésicas, 3., 2003, Curitiba. **Anais Novos Desenvolvimentos em Ciências Geodésicas**. Curitiba: UFPR – Curso de Pós Graduação em Ciências Geodésicas, 2003. Disponível em: CDROM.
- CAMARGO, N. F.; CARVALHO, C. A. P. Estruturas de Aplicação Orientadas a Objetos: Uma Tecnologia para Reutilização de Projetos de Software. In: 1º. **Simpósio Sul-Brasileiro de Matemática e Informática**, 2002, Curitiba. Anais disponível em: <http://www.uniandrade.br/simposio/anais.htm>. Acesso em: 20 nov.. 2002.
- FAYAD, E. M.; SCHMIDT, D. C.; JOHNSON, R. E. Application Frameworks. In: FAYAD, M. E.; SCHMIDT, D. C.; JOHNSON, R. E. (Ed.). **Building Application Frameworks: Object-Oriented Foundations of Framework Design**. New York: J. Wiley, 1999. p. 3-28.
- FAYAD, M. E.; SCHMIDT, D. C. **Object-Oriented Application Frameworks**. Communications of the ACM, New York, v. 40, n. 10, p. 32-38, oct. 1997.
- GAMMA, E.; HELM, R.; JOHNSON, R.; VLISSIDES, J. **Design Patterns: Elements of Reusable Object-Oriented Software**. Reading, Massachusetts: Addison-Wesley, 1995.
- GeoVRML.org. **GeoVRML 1.1 Specification**. Disponível em: <http://www.geovrml.org/1.1/doc/> Acesso em 10 jun. 2003.
- GOEDICKE, M.; MEYER, T. Formal Design and Performance Evaluation. In: FAYAD, M. E.; SCHMIDT, D. C.; JOHNSON, R. E. (Ed.). **Implementing Application Frameworks: Object-Oriented Foundations of Framework Design**. New York: J. Wiley, 1999. p. 345.

- JOHNSON, R. E. **Frameworks = (Componentes+Patterns)**. Communications of the ACM, New York, v. 40, n.10, p. 39-42, oct. 1997.
- JOHNSON, R. E.; FOOTE, B. **Designing Reusable Classes**. Journal of Object-Oriented Programming, v. 1, n. 2, p. 22-35, 1997.
- KURNIAWAN, B. **Java para Web com Servlets, JSP e EJB**. Rio de Janeiro: Ciência Moderna, 2002.
- KYLLÖNEN, H. **Framework for Developing VRML-Based Virtual Prototypes**. Master's Thesis. University of Oulu, Department of Electrical Engineering. p. 54, 1999.
- LARSEN, G. **Designing Component-Based Frameworks Using Patterns in the UML**. Communications of the ACM, New York, v. 42, n.10, p. 38-45, oct./1999.
- LIMA, J. D.; CARVALHO, C. A. P.; VIEIRA, A. J. B. Persistência de Fenômenos Terrestres Representados como Objetos. In: III Colóquio Brasileiro de Ciências Geodésicas, 3., 2003, Curitiba. **Anais Novos Desenvolvimentos em Ciências Geodésicas**. Curitiba: UFPR – Curso de Pós Graduação em Ciências Geodésicas, 2003. Disponível em: CDROM.
- MATTSON, M; BOSCH, J. Composition Problems, Causes, and Solutions. In: FAYAD, M. E.; SCHMIDT, D. C.; JOHNSON R. E. **Building Application Frameworks: Object-Oriented Foundations of Framework Design**. New York: J. Wiley, 1999. p. 467-488.
- MURPHY, G. **Software Architecture: Client/Server Architectural Style**. Disponível em: <<http://www.cs.ubc.ca/spider/murphy/cpsc310-2001/client-server-2up.pdf>> Acesso em: 09 nov. 2002.
- ROBERTS, D.; JOHNSON, R. **Evolving Frameworks: A Pattern Language for Developing Object-Oriented Frameworks**. Disponível em: <<http://st-www.cs.uiuc.edu/users/droberts/evolve.html>> Acesso em: 23 ago. 2000.
- SCHMID, H. A. Framework Design by Systematic Generalization. In: FAYAD, M. E.; SCHMIDT, D. C.; JOHNSON, R. E. **Building Application Frameworks: Object-Oriented Foundations of Framework Design**. New York: J. Wiley, 1999. p. 353-378.
- SOWIZRAL, H. A.; NADEAU, D. R. **Introduction to Programming with Java 3D**. Disponível em: <<http://www.web3d.org/xd3>>. Acesso em: 10 maio 2002.
- VIEIRA, A. J. B.; CARVALHO, C. A. P.; SLUTER, C. R. CAMARGO, N. F.; LIMA, J. D. Modelo Espaço-Temporal de Fenômenos Terrestres. In: III Colóquio Brasileiro de Ciências Geodésicas, 3., 2003, Curitiba. **Anais Novos Desenvolvimentos em Ciências Geodésicas**. Curitiba: UFPR – Curso de Pós Graduação em Ciências Geodésicas, 2003. Disponível em: CDROM.
- WEB3D Consortium. **VRML Specification**. Disponível em: <<http://www.web3D.org>> Acesso em: 10 jun. 2002.
- WOLTERS, H.J. **3D Graphics on the Internet**. Disponível em: <[www.hpl.hp.com/research/cp/cmsl/presentations/3d/3DWeb.pdf](http://www.hpl.hp.com/research/cp/cmsl/presentations/3d/3DWeb.pdf)>. Acesso em: 10 jun 2002.

## APÊNDICES

APÊNDICE 1 – INSTALAÇÃO DA ESTRUTUA DE APLICAÇÃO.....	69
APÊNDICE 2 – INSTALAÇÃO DA APLICAÇÃO TESTE vftAPP1.....	125

## **APÊNDICE 1 –INSTALAÇÃO DA ESTRUTUA DE APLICAÇÃO**

## SUMÁRIO

<b>1</b>	<b>INSTALAÇÃO DA ESTRUTURA DE APLICAÇÃO.....</b>	<b>72</b>
1.1.1	Criar a estrutura de diretórios.....	72
1.1.2	Instalar os arquivos fonte das classes da Estrutura de Aplicação.....	73
1.1.3	Instalar a biblioteca <b>geovrml</b> .....	73
1.1.4	Compilar as classes da Estrutura de Aplicação .....	73
1.1.5	Instalar o arquivo de protótipos VRML .....	74
1.1.6	Instalar o banco de dados.....	74
1.1.7	Instalar os arquivos DTD (Document Type Definition) .....	74
<b>2</b>	<b>ARQUIVOS FONTES DAS CLASSES DA ESTRUTURA DE APLICAÇÃO .....</b>	<b>75</b>
2.1	PACOTE CONTROL .....	75
2.1.1	A CLASSE Controle.....	75
2.2	PACOTE MODEL.....	77
2.2.1	Pacote area.....	77
2.3	PACOTE COORD .....	79
2.3.1	A classe Coordenada.....	79
2.3.2	A classe CoordGCC.....	80
2.3.3	A classe CoordGDC.....	80
2.3.4	A classe CoordUTM.....	82
2.4	PACOTE FEN .....	83
2.4.1	A classe Atributo .....	83
2.4.2	A classe FenomenoTerrestre.....	84
2.4.3	A interface VRML .....	86
2.5	PACOTE SUP .....	86
2.5.1	A classe Face.....	86
2.5.2	A classe Ponto .....	86
2.5.3	A classe RetEnvolvente .....	87
2.5.4	A classe Superficie.....	89
2.5.5	A classe Superficie0D .....	90
2.5.6	A classe Superficie1D .....	91
2.5.7	A classe Superficie2D .....	91
2.5.8	A classe Superficie3D .....	93
2.5.9	A classe Cor.....	94
2.6	PACOTE PERSIST .....	95
2.6.1	A classe BDFTBEAN .....	95
2.6.2	A classe BdFTEException .....	101
2.6.3	A classe AreaLevantamentoBD .....	101
2.6.4	A classe AreaLevantamentoParser.....	103
2.6.5	A classe AreaLevantamentoHandler.....	106
2.6.6	A classe FenomenoTerrestreBD.....	110
2.6.7	A classe FenomenoTerrestreParser .....	112
2.6.8	A classe FenomenoTerrestreHandler .....	115
<b>3</b>	<b>ARQUIVO DE PROTOTIPOS VRML .....</b>	<b>119</b>
3.1	ARQUIVO PROTOS.WRL.....	119
<b>4</b>	<b>DEFINIÇÃO DO BANCO DE DADOS .....</b>	<b>122</b>
4.1	AREASLEVANTAMENTO .....	122

4.2 FENOMENOS .....	122
<b>5 ARQUIVOS DTD (DOCUMENT TYPE DEFINITION) .....</b>	<b>123</b>
5.1 AREALEVANTAMENTO.DTD .....	123
5.2 FENOMENOTERRESTRE.DTD.....	123



## 1 INSTALAÇÃO DA ESTRUTURA DE APLICAÇÃO

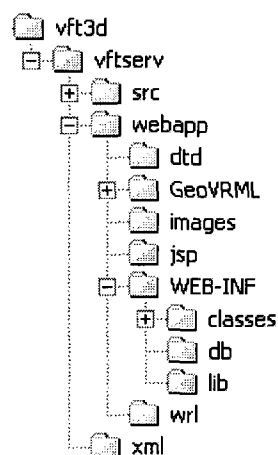
Para a instalação da Estrutura de Aplicação os seguintes preparativos são necessários:

- Criar a estrutura de diretórios.
- Instalar os arquivos fonte das classes da Estrutura de Aplicação
- Instalar a biblioteca **geovrml**
- Compilar as classes
- Instalar o arquivo de protótipos VRML
- Instalar o banco de dados
- Instalar os arquivos DTD (Document Type Definition)

### 1.1.1 Criar a estrutura de diretórios

Criar uma estrutura de diretórios em um disco para os arquivos da aplicação conforme a figura 1.

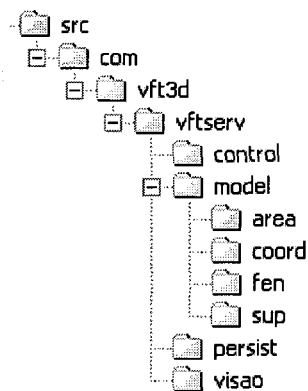
FIGURA 1- A ESTRUTURA DE DIRETÓRIOS DA APLICAÇÃO



### 1.1.2 Instalar os arquivos fonte das classes da Estrutura de Aplicação

Instalar os arquivos fonte das classes da Estrutura de Aplicação, descritas na seção 2, em uma estrutura de subdiretórios do diretório **vtf3d\vftserv\src** conforme a figura 2. Os arquivos (.java) devem ser colocados nos subdiretórios do pacote a que pertencem.

FIGURA 2- A ESTRUTURA DO DIRETORIO **src** -CÓDIGO FONTE DAS CLASSES



### 1.1.3 Instalar a biblioteca **geovrml**

Fazer download do *run-Time* da biblioteca GeoVRML 1.1 do site <http://www.geovrml.org/1.1/> . As instruções de instalação estão contidas no próprio site. Durante a instalação selecione o diretório **vtf3d\vftserv\webapp\GeoVRML** como destino para os arquivos da GeoVRML.

### 1.1.4 Compilar as classes da Estrutura de Aplicação

Compilar os fontes (.java) do diretório **vtf3d\vftserv\src** direcionando os arquivos de saída bytecode (.class) gerados para o diretório **vtf3d\vftserv\webapp\WEB-INF\classes**. A variável de ambiente CLASSPATH deve estar configurada apropriadamente para que o arquivo

**vtf3d\lvftserv\webapp\GeoVRML\1.1\classes\geovrml.jar** possa ser acessado pelo compilador Java durante a compilação

#### 1.1.5 Instalar o arquivo de protótipos VRML

O arquivo **protos.wrl**, descrito na seção 3, deve ser inserido no diretório **vtf3d\lvftserv\webapp\wrl**.

#### 1.1.6 Instalar o banco de dados

Criar um banco de dados usando o Microsoft Access segundo os comandos SQL descritos na seção 4. O banco de dados deve ficar disponível no diretório **vtf3d\lvftserv\webapp\WEB-INF\db** com o nome **bdvft.mdb**. Criar uma fonte de dados para o banco de dados com o nome **BDFVT**.

#### 1.1.7 Instalar os arquivos DTD (Document Type Definition)

Instalar os arquivos Document Type Definition descritos na seção 5 no diretório **vtf3d\lvftserv\webapp\dtd**.

## 2 ARQUIVOS FONTES DAS CLASSES DA ESTRUTURA DE APLICAÇÃO

### 2.1 PACOTE CONTROL

#### 2.1.1 A CLASSE Controle

```

package com.vft3d.vftserv.control;

import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;
import java.util.*;
import com.vft3d.vftserv.persist.BdFTBean;

/** Esta classe representa o controle da aplicação Web.
 * A aplicação final deve estender esta classe e implementar
 * o método getTratador(). A função do método getTratador é de determinar
 * qual pagina HTML ou JSP ou Servlet deve ser usada para tratar a
 * solicitação recebida.
 */
public abstract class Controle extends HttpServlet {

    /** Inicializa as variáveis globais
     * @param config
     * @throws ServletException
     */
    String jspDir="/"; // diretorio default para paginas jsp

    public void init(ServletConfig config) throws ServletException {
        System.out.println("inicializando o controlador.");
        jspDir=config.getInitParameter("jspDir")+"/";
        ServletContext context = config.getServletContext();
        context.setAttribute("baseUrl", config.getInitParameter("baseUrl"));
        context.setAttribute("imageURL", config.getInitParameter("imageURL"));
        context.setAttribute("xmlPath", config.getInitParameter("xmlPath"));
        context.setAttribute("wrlURL", config.getInitParameter("wrlURL"));
        context.setAttribute("wrlPath", config.getInitParameter("wrlPath"));

        // cria uma instância da classe BdFTBean para acesso ao banco de dados
        BdFTBean bdBean = new BdFTBean();
        // inicializa os campos do BdFTBean
        bdBean.setBdURL(config.getInitParameter("bdURL"));
        bdBean.setBdUsuario(config.getInitParameter("bdUsuario"));
        bdBean.setBdPassword(config.getInitParameter("bdPassword"));
        bdBean.setDtdURL(config.getInitParameter("dtdURL"));
        bdBean.setBdLinguagem(config.getInitParameter("bdLinguagem"));

        // Coloca o BdFTBean no contexto do servlet
        // Para que o BdFTBean possa ser acessado a partir das paginas JSP
        context.setAttribute("bdBean", bdBean);

        // Carrega o driver jdbc do banco de dados
        try {
            Class.forName(config.getInitParameter("jdbcDriver"));
        }
        catch (ClassNotFoundException e) {
            System.out.println(e.toString());
        }
    }
}

```

```

    }
    super.init(config);
}

/** Processa as solicitações request Get do HTTP
 * @param request
 * @param response
 * @throws ServletException
 * @throws IOException
 */
public void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException
{
    doPost(request, response);
}

/** Operacao primitiva chamada pelo metodo template doPost.
 * Esta operacao deve ser implementada pela aplicacao.
 * Recebe uma string com um nome de acao e retorna o nome
 * da pagina JSP, HTML ou Servlet que trata a acao solicitada.
 */
public abstract String getTratador(String acao);

/** Processa a solicitação request Post do HTTP.
 * Este metodo é um metodo template que usa a operacao
 * getTratador() para executar a acao solicitada como parametro
 * no Post request enviado pela Web.
 */
public void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException
{
    String url = jspDir + "Default.jsp"; // pagina jsp default

    String acao = request.getParameter("acao");

    if (acao!=null) {
        String nextPage=getTratador(acao);
        if (nextPage!=null) {
            url = jspDir + nextPage;
        } else {
            url = jspDir + "Erro.jsp?mens=solicitacao de acao invalida.";
        }
    }

    System.out.println(url);
    RequestDispatcher requestDispatcher;
    requestDispatcher = getServletContext().getRequestDispatcher(url);
    requestDispatcher.forward(request, response);
}
}

```

## 2.2 PACOTE model

### 2.2.1 Pacote area

#### 2.2.1.1 A classe AreaLevantamento

```
package com.vft3d.vftserv.model.area;

import java.util.Date;
import com.vft3d.vftserv.model.sup.RetEnvolvente;
import java.util.ArrayList;
import com.vft3d.vftserv.model.fen.FenomenoTerrestre;
import com.vft3d.vftserv.model.fen.VRML;

public class AreaLevantamento implements VRML {
    private String id;
    private String nome;
    private Date tempo;
    private RetEnvolvente retEnvolvente;
    private ArrayList fenomenos;

    public String getId(){
        return id;
    }

    public void setId(String id){
        this.id = id;
    }

    public String getNome() {
        return (this.nome);
    }

    public void setNome(String nome) {
        this.nome = nome;
    }

    public void setTempo(Date tempo) {
        this.tempo = tempo;
    }

    public String getTempo() {
        return (this.tempo.toGMTString());
    }

    public void setFenomenos(ArrayList fenomenos) {
        this.fenomenos = fenomenos;
    }

    public ArrayList getFenomenos() {
        return (this.fenomenos);
    }

    public void setRetEnvolvente(RetEnvolvente retEnvolvente) {
        this.retEnvolvente=retEnvolvente;
    }

    /**
     * Retorna o retângulo no qual a Area de Levantamento está contida.
     */
}
```

```

    * O retângulo envolvente é constituído pelos retangulos
    * envolventes de todos os fenomenos terrestre.
    */
    public RetEnvolvente getRetEnvolvente() {
        return this.retEnvolvente;
    }

    public String getVRML() {
        String vrml="";
        vrml+="#Inicio da area de levantamento: "+ id + "\r\n";
        vrml+="DEF " + id +" Transform { children [\r\n";

        for(int i=0; (!fenomenos.isEmpty() && i<fenomenos.size()) ; i++) {
            vrml+=((FenomenoTerrestre)fenomenos.get(i)).getVRML();
        }

        vrml+="] }\r\n";
        vrml+="#Fim da area de levantamento: "+ id + "\r\n";
        return vrml;
    }
}

```

### 2.2.1.2 A classe JanelaVisualizacao

```

package com.vft3d.vftserv.model.area;

import com.vft3d.vftserv.model.sup.RetEnvolvente;
import com.vft3d.vftserv.model.fen.VRML;
import com.vft3d.vftserv.model.fen.FenomenoTerrestre;
import java.util.ArrayList;

public class JanelaVisualizacao implements VRML {
    private RetEnvolvente retEnvolvente;
    private ArrayList fenomenos;

    public RetEnvolvente getRetEnvolvente(){
        return retEnvolvente;
    }

    public void setRetEnvolvente(RetEnvolvente retEnvolvente){
        this.retEnvolvente = retEnvolvente;
    }

    public ArrayList getFenomenos(){
        return fenomenos;
    }

    public void setFenomenos(ArrayList fenomenos){
        this.fenomenos = fenomenos;
    }

    public String getVRML() {
        String vrml="";
        for(int i=0; (!fenomenos.isEmpty() && i<fenomenos.size()) ; i++) {
            vrml+=((FenomenoTerrestre)fenomenos.get(i)).getVRML();
        }
        return vrml;
    }
}

```

## 2.3 PACOTE COORD

### 2.3.1 A classe Coordenada

```

package com.vft3d.vftserv.model.coord;

import geotransform.coords.*;
import geotransform.ellipsoids.*;

import java.util.*;

public abstract class Coordenada {
    public Ellipsoid e = new SA_Ellipsoid(); //inicializando (a,f) com elipsoide
    SAD-69
    public String datumHrz="Chua";

    public void Coordenada() {
    }

    public abstract CoordGCC toGCC();
    public abstract CoordGDC toGDC();
    public abstract CoordUTM toUTM();
    public void setElipsoide(byte tipo) {
        // Estamos seguindo o padrão da tabela da GeoVRML para Elipsoides
        //tipo = 15 ==> International 1924 || Hayford || CorregoAlegre
        //tipo = 18 ==> SAD-69
        //tipo = 20 ==> WGS-84
        if(tipo == 0) e =new AA_Ellipsoid();
        else if(tipo == 1) e =new AM_Ellipsoid();
        else if(tipo == 2) e =new AN_Ellipsoid();
        else if(tipo == 3) e =new BN_Ellipsoid();
        else if(tipo == 4) e =new BR_Ellipsoid();
        else if(tipo == 5) e =new CC_Ellipsoid();
        else if(tipo == 6) e =new CD_Ellipsoid();
        else if(tipo == 7) e =new EA_Ellipsoid();
        else if(tipo == 8) e =new EB_Ellipsoid();
        else if(tipo == 9) e =new EC_Ellipsoid();
        else if(tipo == 10) e =new ED_Ellipsoid();
        else if(tipo == 11) e =new EE_Ellipsoid();
        else if(tipo == 12) e =new FA_Ellipsoid();
        else if(tipo == 13) e =new HE_Ellipsoid();
        else if(tipo == 14) e =new HO_Ellipsoid();
        else if(tipo == 15) e =new IN_Ellipsoid();
        else if(tipo == 16) e =new KA_Ellipsoid();
        else if(tipo == 17) e =new RF_Ellipsoid();
        else if(tipo == 18) e =new SA_Ellipsoid();
        else if(tipo == 19) e =new WD_Ellipsoid();
        if(tipo == 20) e =new WE_Ellipsoid();
        else {
            System.out.print("\n Tipo de elipsoide não previsto :[" +tipo +"]");
            System.exit(0);
        }
    }

    public void setDatumH(String datumH) {
        this.datumHrz=datumH;
    }
}

```



### 2.3.2 A classe CoordGCC

```

package com.vft3d.vftserv.model.coord;

import java.util.*;
import geotransform.coords.*;
import geotransform.ellipsoids.*;
import geotransform.transforms.*;

public class CoordGCC extends Coordenada {

    Gcc_Coord_3d gcc;

    public CoordGCC() {
        gcc = new Gcc_Coord_3d();
    }
    public CoordGCC(double x, double y, double z) {
        gcc = new Gcc_Coord_3d(x,y,z);
    }

    public double getX() {
        return gcc.x;
    }

    public double getY() {
        return gcc.y;
    }

    public double getZ() {
        return gcc.z;
    }

    public CoordGCC toGCC() {
        return this;
    }
    public CoordGDC toGDC() {
        CoordGDC cGDC = new CoordGDC();
        Gcc_To_Gdc_Converter.Init(e);
        Gcc_To_Gdc_Converter.Convert(this.gcc,cGDC.gdc);
        return cGDC;
    }
    public CoordUTM toUTM() {
        CoordUTM cUTM = new CoordUTM();
        Gcc_To_Utm_Converter.Init(e);
        Gcc_To_Utm_Converter.Convert(this.gcc, cUTM.utm);
        return cUTM;
    }
    public String toString() {
        return "GCC(X:"+gcc.x+",Y:"+gcc.y+",Z:"+gcc.z+)";
    }
}

```

### 2.3.3 A classe CoordGDC

```

package com.vft3d.vftserv.model.coord;

import java.util.*;
import geotransform.coords.*;
import geotransform.ellipsoids.*;
import geotransform.transforms.*;
import java.text.DecimalFormat;

```

```

public class CoordGDC extends Coordenada
{
    public Gdc_Coord_3d gdc;

    public CoordGDC () {
        gdc=new Gdc_Coord_3d();
    }
    public CoordGDC (double lat, double lon, double alt) {
        gdc =new Gdc_Coord_3d(lat,lon,alt);
    }

    public double getLatitude() {
        return gdc.latitude;
    }

    public double getLongitude() {
        return gdc.longitude;
    }

    public double getAltitude() {
        return gdc.elevation;
    }

    public CoordGCC toGCC() {
        CoordGCC cGCC = new CoordGCC();
        Gdc_To_Gcc_Converter.Init(e);
        Gdc_To_Gcc_Converter.Convert(this.gdc, cGCC.gcc);
        return cGCC;
    }

    public CoordGDC toGDC() {
        return this;
    }

    public CoordUTM toUTM() {
        CoordUTM cUTM = new CoordUTM();
        Gdc_To_Utm_Converter.Init(e);
        Gdc_To_Utm_Converter.Convert(this.gdc, cUTM.utm);
        return cUTM;
    }

    public String getLat() {
        char lat=(gdc.latitude <0) ? 'S':'N';
        DecimalFormat duasDecimais =new DecimalFormat("000.000");
        return duasDecimais.format(Math.abs(gdc.latitude))+"-"+lat;
    }

    public String getLong() {
        char lon=(gdc.longitude <0) ? 'O':'L';
        DecimalFormat duasDecimais =new DecimalFormat("000.000");
        return duasDecimais.format(Math.abs(gdc.longitude))+"-"+lon;
    }

    public String toString() {
        char lon=(gdc.longitude <0) ? 'O':'E';
        char lat=(gdc.latitude <0) ? 'S':'N';
        return " "+Math.abs(gdc.longitude)+"-" + lon
            +", " +Math.abs(gdc.latitude)+"-"+lat
            +", "+gdc.elevation + " ";
    }
}

```

### 2.3.4 A classe CoordUTM

```
package com.vft3d.vftserv.model.coord;

import java.util.*;
import geotransform.coords.*;
import geotransform.ellipsoids.*;
import geotransform.transforms.*;

public class CoordUTM extends Coordenada {

    Utm_Coord_3d utm;

    public CoordUTM() {
        utm = new Utm_Coord_3d();
    }

    public CoordUTM(double este,
                    double norte,
                    double altitude,
                    int fuso,
                    boolean hemisferioNorte)
    {
        utm = new Utm_Coord_3d(este,norte,altitude, (byte) fuso, hemisferioNorte);
    }

    public double getEste() {
        return utm.x;
    }

    public double getNorte() {
        return utm.y;
    }

    public double getAltitude() {
        return utm.z;
    }

    public double getX() {
        return utm.x;
    }

    public double getY() {
        return utm.y;
    }

    public double getZ() {
        return utm.z;
    }

    public int getFuso() {
        return (int)utm.zone;
    }

    public CoordUTM(double este,
                    double norte,
                    double altitude,
                    byte fuso,
                    boolean hemisferioNorte,
                    byte tipo,
                    String datumH) {
        setElipsoide(tipo);
        setDatumH(datumH);
        utm = new Utm_Coord_3d(este,norte,altitude, fuso, hemisferioNorte);
    }
}
```

```

}

public CoordUTM toUTM() {
    return this;
}

public CoordGCC toGCC(){
    CoordGCC cGCC = new CoordGCC();
    Utm_To_Gcc_Converter.Init(e);
    Utm_To_Gcc_Converter.Convert(utm, cGCC.gcc);
    return cGCC;
}

public CoordGDC toGDC() {
    CoordGDC cGDC = new CoordGDC();
    Utm_To_Gdc_Converter.Init(e);
    Utm_To_Gdc_Converter.Convert(this.utm, cGDC.gdc);
    return cGDC;
}

public String toString() {
    return "UTM(Este:"+utm.x
        +", Norte:"+utm.y
        +", Altitude:"+utm.z
        +", Fuso:"+utm.zone
        +", "+utm.hemisphere_north
        +")";
}
}
}

```

## 2.4 PACOTE FEN

### 2.4.1 A classe Atributo

```

package com.vft3d.vftserv.model.fen;

import java.util.Date;

/**
 * Um atributo que caracteriza um fenômeno terrestre.
 */
public class Atributo {
    private String nome;
    private String valor;
    private Date tempo;

    public Atributo(String nome, String valor, Date tempo) {
        this.nome = nome;
        this.valor = valor;
        this.tempo = tempo;
    }

    public void setNome(String nome) {
        this.nome = nome;
    }

    public void setValor(String valor) {
        this.valor = valor;
    }
}

```

```

public void setTempo(Date tempo) {
    this.tempo = tempo;
}

public String getNome() {
    return (this.nome);
}

public String getValor() {
    return (this.valor);
}

public Date getTempo() {
    return (this.tempo);
}

public String toString() {
    return getNome() + "=" + getValor() + "(" + getTempo().toString() + ")";
}
}

```

## 2.4.2 A classe FenomenoTerrestre

```

package com.vft3d.vftserv.model.fen;

import java.util.ArrayList;
import java.util.ListIterator;

import com.vft3d.vftserv.model.sup.RetEnvolvente;
import com.vft3d.vftserv.model.sup.Superficie;
import com.vft3d.vftserv.model.fen.VRML;
import java.util.Date;

/** <P>
 * Esta classe representa um fenômeno terrestre
 * como um entendimento de um fato que ocorre
 * na superfície terrestre.
 * </P>
 * <P>
 * Superfície terrestre é o nome dado para a parte externa do planeta terra.
 * Sobre esta superfície encontram-se fenômenos em que esta superfície
 * fica submersa. Estes fenômenos, tais como, oceanos, lagos, rios são
 * denominados de corpos hídricos.
 * As regiões em que a superfície aflora são chamadas de superfícies
 * topográficas (VIEIRA et al., 2002).
 * </P>
 */

public class FenomenoTerrestre implements VRML {
    private int id;
    private int categoria;
    private String tipo;
    private int dimensao;
    private ArrayList atributos;
    private Superficie superficie;

    public int getId(){ return id; }

    public void setId(int id){ this.id = id; }

    public int getCategoria(){ return categoria; }
}

```

```

public void setCategoria(int categoria){ this.categoria = categoria; }

public void setTipo(String tipo) {
    this.tipo = tipo;
}

/**
 * Atribui uma dimensao da forma do fenômeno terrestre 0, 1, 2 ou 3
 * dimensões. O valor deve ser consistente com a forma da superficie
 * atribuida ao fenomeno.
 */
public void setDimensao(int dimensao) {
    this.dimensao = dimensao;
}

public void setAtributos(java.util.ArrayList atributos) {
    this.atributos = atributos;
}

/**
 * Atribui um objeto Superficie que define a forma do fenômeno terrestre.
 */
public void setSuperficie(Superficie superficie) {
    this.superficie = superficie;
}

public String getTipo() {
    return (this.tipo);
}

public int getDimensao() {
    return (this.dimensao);
}

/** <P>
 * Retorna um objeto Superficie que representa a forma ao fenômeno terrestre.
 * </P>
 * @return Superficie.
 */
public Superficie getSuperficie() {
    return (this.superficie);
}

/** <P>
 * Obtem o menor retângulo no qual o
 * fenomeno terrestre está contido.
 * </P>
 * <P>
 * O retângulo envolvente é obtido com base nos pontos topograficos
 * que definem a superficie do fenomeno terrestre.
 * </P>
 * @return RetEnvolvente.
 */
public RetEnvolvente getRetEnvolvente() {
    return superficie.getRetEnvolvente();
}

public String getVRML(){
    String vrml="#Inicio do fenomeno terrestre: "+ id + "\r\n";
    vrml+="DEF FT" + id + " Transform { children [\r\n";
    vrml+="\tDEF SupFT" + id+" "+superficie.getVRML();
    vrml+="] } # fim fenomeno terrestre\r\n";
    vrml+="#Fim do fenomeno terrestre: "+ id + "\r\n";
    return vrml;
}

```

```
}

```

### 2.4.3 A interface VRML

```
package com.vft3d.vftserv.model.fen;
import java.net.URL;

public interface VRML {
    String getVRML();
}

```

## 2.5 PACOTE SUP

### 2.5.1 A classe Face

```
package com.vft3d.vftserv.model.sup;

public class Face {
    private int[] prefs;

    public Face(int[] prefs) {
        this.prefs=prefs;
    }

    public int[] getPrefs() {
        return(this.prefs);
    }
}

```

### 2.5.2 A classe Ponto

```
package com.vft3d.vftserv.model.sup;

/**
 * Esta classe define a representação um ponto em um sistema de coordenadas
 geográficas.*/
public final class Ponto {
    public double x;
    public double y;
    public double z;
    private boolean medido;
    private boolean topografico;

    public Ponto(double x, double y, double z, boolean medido,
        boolean topografico) {
        this.x = x;
        this.y = y;
        this.z = z;
        this.medido = medido;
        this.topografico = topografico;
    }
}

```

```

    }

    public boolean equals(Object p) {

        return (p instanceof Ponto &&
            this.x == ((Ponto)p).x &&
            this.y == ((Ponto)p).y &&
            this.z == ((Ponto)p).z &&
            this.medido == ((Ponto)p).medido &&
            this.topografico == ((Ponto)p).topografico
        );

    }

    public void setX(double x) {
        this.x = x;
    }

    public void setY(double y) {
        this.y = y;
    }

    public void setZ(double z) {
        this.z = z;
    }

    public void setMedido(boolean medido) {
        this.medido = medido;
    }

    public void setTopografico(boolean topografico) {
        this.topografico = topografico;
    }

    public double getX() {
        return (this.x);
    }

    public double getY() {
        return (this.y);
    }

    public double getZ() {
        return (this.z);
    }

    public boolean getMedido() {
        return (this.medido);
    }

    public boolean getTopografico() {
        return (this.topografico);
    }

}

```

### 2.5.3 A classe RetEnvolvente

```

package com.vft3d.vftserv.model.sup;

import java.lang.Math;

/**

```



```
* Representa o menor retângulo no qual o
* fenomeno terrestre está contido.
* O retângulo envolvente é constituído pelos pontos topograficos
* que definem a superficie do fenomeno terrestre.*/

public class RetEnvolvente {
    private double minX;
    private double minY;
    private double minZ;
    private double maxX;
    private double maxY;
    private double maxZ;

    public RetEnvolvente() {
    }

    public RetEnvolvente(double minX, double minY, double minZ,
        double maxX, double maxY, double maxZ) {
        this.minX = minX;
        this.minY = minY;
        this.minZ = minZ;
        this.maxX = maxX;
        this.maxY = maxY;
        this.maxZ = maxZ;
    }

    public void setMinX(double minX) {
        this.minX = minX;
    }

    public double getMaxX() {
        return (this.maxX);
    }

    public void setMaxX(double maxX) {
        this.maxX = maxX;
    }

    public void setMinY(double minY) {
        this.minY = minY;
    }

    public void setMaxY(double maxY) {
        this.maxY = maxY;
    }

    public double getMinX() {
        return (this.minX);
    }

    public double getMinY() {
        return (this.minY);
    }

    public double getMaxY() {
        return (this.maxY);
    }

    public double getLargura(){
        return(maxX-minX);
    }

    public double getAltura() {
        return (maxY-minY);
    }
}
```

```

public double getDiagonal() {
    return
        (Math.sqrt(
            Math.pow(getLargura(),2)+
            Math.pow(getAltura(),2)
        )
    );
}

public double getMinZ() {
    return minZ;
}

public void setMinZ(double minZ) {
    this.minZ = minZ;
}

public double getMaxZ() {
    return maxZ;
}

public void setMaxZ(double maxZ) {
    this.maxZ = maxZ;
}
}

```

#### 2.5.4 A classe Superficie

```

package com.vft3d.vftserv.model.sup;

import java.util.Date;
import com.vft3d.vftserv.model.fen.VRML;

public abstract class Superficie implements VRML {
    public Date tempo;
    public RetEnvolvente retEnvolvente;
    public Cor cor;

    public Date getTempo() {
        return (this.tempo);
    }

    public void setTempo(Date tempo) {
        this.tempo = tempo;
    }

    public RetEnvolvente getRetEnvolvente() {
        return (this.retEnvolvente);
    }

    public void setRetEnvolvente(RetEnvolvente retEnvolvente) {
        this.retEnvolvente = retEnvolvente;
    }

    public Cor getCor() {
        return cor;
    }

    public void setCor(Cor cor) {
        this.cor = cor;
    }
}

```

### 2.5.5 A classe Superficie0D

```

package com.vft3d.vftserv.model.sup;

import java.util.*;
import com.vft3d.vftserv.model.sup.RetEnvolvente;
import com.vft3d.vftserv.model.fen.FenomenoTerrestre;

/**
 * Esta classe representa a superficie
 * de um fenomeno terrestre com dimensao 0 (ponto).
 */

public class Superficie0D extends Superficie {
    private String modeloURL; // url para a forma da superficie em vml
    private Ponto ponto;

    public Superficie0D(Ponto ponto, String modeloURL) {
        this.retEnvolvente=new RetEnvolvente(ponto.x, ponto.y, ponto.z
        ,ponto.x, ponto.y, ponto.z);

        this.ponto = ponto;
        this.modeloURL=modeloURL;
    }

    public void setPonto(Ponto ponto) {
        this.ponto = ponto;
    }

    public Ponto getPonto() {
        return (this.ponto);
    }

    public String getVRML() {
        String vrml=" Superficie0D {"
            +" ponto \""
            +Double.toString(ponto.getX())
            +" "
            +Double.toString(ponto.getY())
            +" "
            +Double.toString(ponto.getZ())
            +"\"";
        if (modeloURL!=null) {
            vrml+=" modelo "+ "\""+modeloURL+"\"";
        }
        if (getCor()!=null) {
            vrml+=" cor"
                +" "+getCor().getR()
                +" "+getCor().getG()
                +" "+getCor().getB()
                +"\r\n" ;
        }
        vrml+="}\r\n";
        return vrml;
    }

    public String getModelo() {
        return modeloURL;
    }

    public void setModelo(String modeloURL) {
        this.modeloURL = modeloURL;
    }
}

```

## 2.5.6 A classe Superficie1D

```

package com.vft3d.vftserv.model.sup;

import java.util.ArrayList;
import com.vft3d.vftserv.model.fen.FenomenoTerrestre;

/**
 * Esta classe representa a superficie
 * de um fenomeno terrestre com dimensão 1 (polilinha).
 */

public class Superficie1D extends Superficie {

    private ArrayList pontos;

    public Superficie1D(ArrayList pontos) {
        this.pontos=pontos;
    }

    public void setPontos(ArrayList pontos) {
        this.pontos = pontos;
    }

    public ArrayList getPontos() {
        return (this.pontos);
    }

    public String getVRML() {
        String vrml=" Superficie1D {\r\n"
            +"pontos [\" \r\n"
            ;
        // declara os pontos
        for(int i=0; (!pontos.isEmpty() && i<pontos.size()) ; i++) {
            Ponto ponto = (Ponto)pontos.get(i);
            vrml+="\t"+ponto.x
                +" "+ponto.y
                +" "+ponto.z
                +"\r\n"
                ;
        }
        vrml+="\t"}\r\n";
        if (getCor()!=null) {
            vrml+=" cor"
                +" "+getCor().getR()
                +" "+getCor().getG()
                +" "+getCor().getB()
                +"\r\n" ;
        }
        vrml+=" } # fim superficie\r\n";
        return vrml;
    }
}

```

## 2.5.7 A classe Superficie2D

```

package com.vft3d.vftserv.model.sup;

import java.util.ArrayList;
import com.vft3d.vftserv.model.fen.FenomenoTerrestre;

```

```

/**
 * <p>
 * Esta classe representa a superficie
 * de um fenomeno terrestre com dimensao 2 (superficie plana).
 * </p>
 */

public class Superficie2D extends Superficie {

    private ArrayList pontos;

    public Superficie2D(ArrayList pontos) {
        this.pontos=pontos;
    }

    public ArrayList getPontos() {
        return pontos;
    }

    public void setPontos(ArrayList pontos) {
        this.pontos = pontos;
    }

    /** <P>
     * Gera descricao em VRML da superficie 2D.
     * </P>
     * @return String com a descricao em VRML.
     */
    public String getVRML() {
        String vrml=" Superficie2D {\r\n"
            +"pontos [\" \r\n"
            ;
        // declara os pontos
        for(int i=0; (!pontos.isEmpty() && i<pontos.size()) ; i++) {
            Ponto ponto = (Ponto)pontos.get(i);
            vrml+="\t"+ponto.x
                +" "+ponto.y
                +" "+ponto.z
                +"\r\n"
                ;
        }
        vrml+="\t\"]\r\n";
        // declara a face
        vrml+="face [ \r\n";
        vrml+="\t";
        for (int j=0; j<pontos.size();j++ ) {
            vrml+=" "+j;
        }
        vrml+=" -1\r\n";
        vrml+="]\r\n";

        if (getCor()!=null) {
            vrml+=" cor"
                +" "+getCor().getR()
                +" "+getCor().getG()
                +" "+getCor().getB()
                +"\r\n" ;
        }

        vrml+="} # fim superficie\r\n";
        return vrml;
    }
}

```

## 2.5.8 A classe Superficie3D

```

package com.vft3d.vftserv.model.sup;

import java.util.ArrayList;
import com.vft3d.vftserv.model.fen.FenomenoTerrestre;

/**
 * Esta classe representa a superficie
 * de um fenomeno terrestre com dimensão 3 (muti-faces).
 */

public class Superficie3D extends Superficie {

    private ArrayList pontos;
    private ArrayList faces;

    public Superficie3D(ArrayList pontos, ArrayList faces) {
        this.pontos=pontos;
        this.faces = faces;
    }

    public void setFaces(ArrayList faces) {
        this.faces = faces;
    }

    public ArrayList getFaces() {
        return (this.faces);
    }

    public String getVRML() {
        String vrml=" Superficie3D {\r\n"
            +"pontos [\r\n"
            ;
        // declara os pontos

        for(int i=0; (!pontos.isEmpty() && i<pontos.size()) ; i++) {
            Ponto ponto = (Ponto)pontos.get(i);
            vrml+="\t"+ponto.x
                +" "+ponto.y
                +" "+ponto.z
                +"\r\n"
                ;
        }
        vrml+="\t"}\r\n";
        vrml+="faces [ \r\n";
        // declara as faces
        for(int i=0; (!faces.isEmpty() && i<faces.size()); i++) {
            Face face = (Face)faces.get(i);
            int[] prefs=face.getPrefs();
            vrml+="\t";
            for (int j=0; j<prefs.length;j++ ) {
                vrml+=" "+prefs[j];
            }
            vrml+=" -1\r\n";
        }
        vrml+="]\n";

        if (getCor()!=null) {
            vrml+=" cor"
                +" "+getCor().getR()
                +" "+getCor().getG()
                +" "+getCor().getB()
                +"\n" ;
        }
    }
}

```

```

    }
    vrml+=") # fim superficie\r\n";
    return vrml;
}

public ArrayList getPontos() {
    return pontos;
}

public void setPontos(ArrayList pontos) {
    this.pontos = pontos;
}
}

```

### 2.5.9 A classe Cor

```

package com.vft3d.vftserv.model.sup;

public class Cor {
    double r;
    double g;
    double b;

    public Cor() {
    }
    public Cor(double r, double g, double b) {
        this.r=r;
        this.g=g;
        this.b=b;
    }

    public double getR() {
        return r;
    }

    public void setR(double r) {
        this.r = r;
    }

    public double getG() {
        return g;
    }

    public void setG(double g) {
        this.g = g;
    }

    public double getB() {
        return b;
    }

    public void setB(double b) {
        this.b = b;
    }
}

```

## 2.6 PACOTE PERSIST

### 2.6.1 A classe BDFTBEAN

```

package com.vft3d.vftserv.persist;

import java.util.Hashtable;
import java.util.ArrayList;
import java.util.Iterator;
import java.util.Date;
import java.io.StringReader;
import java.sql.*;
import com.vft3d.vftserv.model.area.AreaLevantamento;
import com.vft3d.vftserv.model.fen.*;
import com.vft3d.vftserv.model.sup.*;
import org.xml.sax.SAXParseException;
import org.xml.sax.SAXException;
import org.xml.sax.InputSource;
import com.vft3d.vftserv.persist.AreaLevantamentoBD;
import com.vft3d.vftserv.persist.FenomenoTerrestreBD;
import java.util.Locale;
import java.text.*;

public class BdFTBean {
    public String bdURL = "";
    public String bdUsuario = "";
    public String bdPassword = "";
    public String dtdURL="";
    public NumberFormat numFormat;

    public static final boolean DEBUG = true;

    public void setBdURL(String url) {
        bdURL = url;
    }
    public void setBdUsuario(String usuario) {
        bdUsuario = usuario;
    }
    public void setBdPassword(String password) {
        bdPassword = password;
    }

    public void setBdLinguagem(String linguagem) {
        Locale.setDefault(new Locale(linguagem));
        numFormat=DecimalFormat.getInstance();
    }

    public void setDtdURL(String dtdURL) {
        this.dtdURL=dtdURL;
    }

    public String createAreaLevantamentoBD(String areaId, java.net.URL docXML) {
        AreaLevantamentoParser parser;
        areaId=areaId.trim();
        if (this.findAreaLevantamentoBD(areaId)!=null)
            return "Área de levantamento já existente";
        try {
            parser =
                new AreaLevantamentoParser(new AreaLevantamentoHandler(dtdURL),null);
            parser.parse(docXML);
        }
    }
}

```



```

catch (SAXParseException e) {
    return e.getLineNumber() + ": " + e.getMessage();
}
catch (Exception e) {
    return e.toString();
}
}
AreaLevantamentoBD areaBD = parser.getAreaLevantamentoBD();
if (!(areaBD.getId().equals(areaId)))
    return
        "Nome da área ("
        + areaBD.getId()
        + ") diferente do atributo id ("
        + areaId
        + ")";
ArrayList fenomenosBD = parser.getFenomenosTerrestresBD();
try {
    insertAreaLevantamentoBD(areaBD, fenomenosBD);
}
catch (SQLException se){
    return se.toString();
}

return null;
}

public AreaLevantamento loadAreaLevantamento(String areaId)
throws BdFTEException {
    AreaLevantamento areaLevantamento;
    AreaLevantamentoParser parser;
    areaId=areaId.trim();

    AreaLevantamentoBD areaBD=findAreaLevantamentoBD(areaId);
    if (areaBD==null)
        throw new BdFTEException("Área de levantamento não existente");

    areaLevantamento=new AreaLevantamento();
    areaLevantamento.setId(areaBD.getId());
    areaLevantamento.setNome(areaBD.getNome());
    Date tempo =areaBD.getTempo();
    areaLevantamento.setTempo(tempo);
    // cria retangulo envolvente
    RetEnvolvente retEnv = new RetEnvolvente();
    retEnv.setMinX(areaBD.getMinX());
    retEnv.setMinY(areaBD.getMinY());
    retEnv.setMinZ(areaBD.getMinZ());
    retEnv.setMaxX(areaBD.getMaxX());
    retEnv.setMaxY(areaBD.getMaxY());
    retEnv.setMaxZ(areaBD.getMaxZ());
    areaLevantamento.setRetEnvolvente(retEnv);
    areaLevantamento.setFenomenos(loadFenomenosTerrestres(areaId, tempo));
    return areaLevantamento;
}

public ArrayList findAreasLevantamentoBD(
    String minX,
    String minY,
    String maxX,
    String maxY
)
{
    ArrayList areasLevantamento = new ArrayList();
    try {
        Connection connection =
            DriverManager.getConnection(bdURL, bdUsuario, bdPassword);
        Statement s = connection.createStatement();
        String sql =

```

```

        "SELECT id, nome, tempo, minX, minY, minZ, maxX, maxY, maxZ "+
        " FROM AreasLevantamento " +
        " WHERE minX >= " + minX.trim() +
        " AND   minY >= " + minY.trim() +
        " AND   maxX <= " + maxX.trim() +
        " AND   maxY <= " + maxY.trim() +
        " ORDER BY tempo " +
        " ";
    ResultSet rs = s.executeQuery(sql);
    while (rs.next()) {
        AreaLevantamentoBD areaBD = new AreaLevantamentoBD();
        areaBD.setId(rs.getString(1));
        areaBD.setNome(rs.getString(2));
        areaBD.setTempo(rs.getDate(3));
        areaBD.setMinX(rs.getDouble(4));
        areaBD.setMinY(rs.getDouble(5));
        areaBD.setMinZ(rs.getDouble(6));
        areaBD.setMaxX(rs.getDouble(7));
        areaBD.setMaxY(rs.getDouble(8));
        areaBD.setMaxZ(rs.getDouble(9));
        areasLevantamento.add(areaBD);
    }
    rs.close();
    s.close();
    connection.close();
}
catch (SQLException e) {}
return areasLevantamento;
}

public AreaLevantamentoBD findAreaLevantamentoBD(String areaId) {
    AreaLevantamentoBD areaBD=null;
    try {
        Connection connection =
            DriverManager.getConnection(bdURL, bdUsuario, bdPassword);
        Statement s = connection.createStatement();
        String sql = "SELECT id, nome, tempo, minX, minY, minZ, maxX, maxY, maxZ"+
            " FROM AreasLevantamento " +
            " WHERE id = '\" + areaId.trim() + "\""
            + " ";
        ResultSet rs = s.executeQuery(sql);
        if (rs.next()) {
            areaBD=new AreaLevantamentoBD();
            areaBD.setId(rs.getString(1));
            areaBD.setNome(rs.getString(2));
            areaBD.setTempo(rs.getDate(3));
            areaBD.setMinX(rs.getDouble(4));
            areaBD.setMinY(rs.getDouble(5));
            areaBD.setMinZ(rs.getDouble(6));
            areaBD.setMaxX(rs.getDouble(7));
            areaBD.setMaxY(rs.getDouble(8));
            areaBD.setMaxZ(rs.getDouble(9));
        }
        rs.close();
        s.close();
        connection.close();
    }
    catch (SQLException e) {}
    return areaBD;
}

public boolean insertAreaLevantamentoBD(
    AreaLevantamentoBD area,
    ArrayList fenomenos
)
throws SQLException

```

```

{
    boolean returnValue = false;
    Connection connection = null;
    try {

        connection =
            DriverManager.getConnection(bdURL, bdUsuario, bdPassword);
        connection.setAutoCommit(false);
        Statement s = connection.createStatement();

        String sql = "INSERT INTO AreasLevantamento"
            + " (id, nome, tempo, minX, minY, minZ, maxX, maxY, maxZ)"
            + " VALUES"
            + " ('" + area.getId()
            + "', '" + area.getNome()
            + "', '" + (area.getTempo()).toLocaleString()
            + "', '" + numFormat.format(area.getMinX())
            + "', '" + numFormat.format(area.getMinY())
            + "', '" + numFormat.format(area.getMinZ())
            + "', '" + numFormat.format(area.getMaxX())
            + "', '" + numFormat.format(area.getMaxY())
            + "', '" + numFormat.format(area.getMaxZ())
            + "')";

        if (DEBUG) System.err.println(sql);

        s.executeUpdate(sql);

        // insere os fenomenos na tabela FenomenosTerrestreBD
        Iterator iterator = fenomenos.iterator();
        while(iterator.hasNext()) {
            FenomenoTerrestreBD fenomeno =
                (FenomenoTerrestreBD)iterator.next();
            sql = "INSERT INTO Fenomenos ("
                + " Categoria"
                + ", Tipo"
                + ", Dimensao"
                + ", minX"
                + ", minY"
                + ", minZ"
                + ", maxX"
                + ", maxY"
                + ", maxZ"
                + ", xml"
                + ", AreaLevantamentoId"
                + ") VALUES "
                + " ('" + fenomeno.getCategoria()
                + "', '" + fenomeno.getTipo()
                + "', '" + fenomeno.getDimensao()
                + "', '" + numFormat.format(fenomeno.getMinX())
                + "', '" + numFormat.format(fenomeno.getMinY())
                + "', '" + numFormat.format(fenomeno.getMinZ())
                + "', '" + numFormat.format(fenomeno.getMaxX())
                + "', '" + numFormat.format(fenomeno.getMaxY())
                + "', '" + numFormat.format(fenomeno.getMaxZ())
                + "', '" + fenomeno.getXml()
                + "', '" + fenomeno.getAreaLevantamentoID()
                + "') ";

            if (DEBUG) System.err.println(sql);
            s.executeUpdate(sql);
        }
        s.close();
        connection.commit();
        connection.close();
        returnValue = true;
    }
}

```

```

    }
    catch (SQLException e) {
        if (DEBUG) System.err.println(e.toString());
        try {
            connection.rollback();
            connection.close();
            throw e;
        }
        catch (SQLException se) {throw se;}
    }
    return returnValue;
}

public ArrayList findFenomenosTerrestresBD(String areaLevantamentoId)
throws BdfTException
{
    ArrayList fenomenosBD = new ArrayList();
    try {
        Connection connection =
            DriverManager.getConnection.bdURL, bdUsuario, bdPassword);
        Statement s = connection.createStatement();
        String sql =
            "SELECT "+
            " id, categoria, tipo, dimensao, minX, minY, minZ, maxX, maxY, maxZ, xml"+
            " FROM Fenomenos " +
            " WHERE areaLevantamentoId = '" + areaLevantamentoId.trim()+"' ";

        ResultSet rs = s.executeQuery(sql);
        while (rs.next()) {
            FenomenoTerrestreBD fenomenoBD = new FenomenoTerrestreBD();
            fenomenoBD.setId(Integer.parseInt(rs.getString(1)));
            fenomenoBD.setCategoria(Integer.parseInt(rs.getString(2)));
            fenomenoBD.setTipo(rs.getString(3));
            fenomenoBD.setDimensao(Integer.parseInt(rs.getString(4)));
            fenomenoBD.setMinX(rs.getDouble(5));
            fenomenoBD.setMinY(rs.getDouble(6));
            fenomenoBD.setMinZ(rs.getDouble(7));
            fenomenoBD.setMaxX(rs.getDouble(8));
            fenomenoBD.setMaxY(rs.getDouble(9));
            fenomenoBD.setMaxZ(rs.getDouble(10));
            fenomenoBD.setXml(rs.getString(11));

            fenomenosBD.add(fenomenoBD);
        }
        rs.close();
        s.close();
        connection.close();
    }
    catch (SQLException e) {throw new BdfTException(e.getMessage());}
    return fenomenosBD;
}

public ArrayList loadFenomenosTerrestres(
    String areaLevantamentoId,
    Date tempo
)
throws BdfTException
{
    ArrayList fenomenos = new ArrayList();
    try {
        Connection connection =
            DriverManager.getConnection.bdURL, bdUsuario, bdPassword);
        Statement s = connection.createStatement();
        String sql =
            "SELECT "+

```

```

" id, categoria, tipo, dimensao, minX, minY, minZ, maxX, maxY, maxZ, xml"+
" FROM Fenomenos " +
" WHERE areaLevantamentoId = '" + areaLevantamentoId.trim()+"' ";

ResultSet rs = s.executeQuery(sql);
while (rs.next()) {
    fenomenos.add(loadFenomenoTerrestre(rs,tempo));
}
rs.close();
s.close();
connection.close();
}
catch (SQLException e) {throw new BdFTException(e.getMessage());}
return fenomenos;
}

public FenomenoTerrestre loadFenomenoTerrestre(ResultSet rs, Date tempo)
throws BdFTException {
    FenomenoTerrestre fenomeno = new FenomenoTerrestre();
    Superficie superficie;
    int dimensao;
    RetEnvolvente retEnvolvente;

    try {
        fenomeno.setId(Integer.parseInt(rs.getString(1)));
        fenomeno.setCategoria(Integer.parseInt(rs.getString(2)));
        fenomeno.setTipo(rs.getString(3));
        dimensao=Integer.parseInt(rs.getString(4));
        fenomeno.setDimensao(dimensao);

        // cria retangulo envolvente
        retEnvolvente=new RetEnvolvente(
            rs.getDouble(5),
            rs.getDouble(6),
            rs.getDouble(7),
            rs.getDouble(8),
            rs.getDouble(9),
            rs.getDouble(10)
        );

        // carrega atributos e superficie a partir da descricao em xml
        FenomenoTerrestreParser parser;

        InputSource xmlInput=
            new InputSource(new StringReader(rs.getString("xml")));
        parser =
            new FenomenoTerrestreParser(new FenomenoTerrestreHandler(),null);
        parser.parse(xmlInput);

        fenomeno.setAtributos(parser.getAtributos());
        superficie=parser.getSuperficie();
        superficie.setRetEnvolvente(retEnvolvente);
        superficie.setTempo(tempo);
        fenomeno.setSuperficie(superficie);
    }
    catch (SAXParseException e) {
        throw new BdFTException(
            "Erro em xml de FenomenoTerrestre: "
            +e.getLineNumber() + ": "
            +e.getMessage()
        );
    }
    catch (Exception e) {
        throw new BdFTException(
            "Erro em loadFenomenoTerrestre: "

```

```

        +e.toString()
    );
}

return fenomeno;
}
}

```

## 2.6.2 A classe BdFTEException

```

package com.vft3d.vftserv.persist;

public class BdFTEException extends java.lang.Exception {

    public BdFTEException() {
    }

    public BdFTEException(String msg) {
        super(msg);
    }
}

```

## 2.6.3 A classe AreaLevantamentoBD

```

package com.vft3d.vftserv.persist;

import java.util.*;

public class AreaLevantamentoBD {
    String id;
    String nome;
    Date tempo;
    double minX;
    double minY;
    double minZ;
    double maxX;
    double maxY;
    double maxZ;

    public AreaLevantamentoBD() {
        minX=Double.MAX_VALUE;
        minY=Double.MAX_VALUE;
        minZ=Double.MAX_VALUE;
        maxX=Double.MIN_VALUE;
        maxY=Double.MIN_VALUE;
        maxZ=Double.MIN_VALUE;
    }

    public void setMinMaxXY(double x, double y, double z) {
        if (x<minX) { minX=x; }
        if (y<minY) { minY=y; }
        if (z<minZ) { minZ=z; }
        if (x>maxX) { maxX=x; }
        if (y>maxY) { maxY=y; }
        if (z>maxZ) { maxZ=z; }
    }
}

```

```
public java.lang.String getNome() {
    return nome;
}

public void setNome(java.lang.String nome) {
    this.nome = nome;
}

public Date getTempo() {
    return tempo;
}

public void setTempo(Date tempo) {
    this.tempo = tempo;
}

public double getMinX() {
    return minX;
}

public void setMinX(double minX) {
    this.minX = minX;
}

public double getMinY() {
    return minY;
}

public void setMinY(double minY) {
    this.minY = minY;
}

public double getMaxX() {
    return maxX;
}

public void setMaxX(double maxX) {
    this.maxX = maxX;
}

public double getMaxY() {
    return maxY;
}

public void setMaxY(double maxY) {
    this.maxY = maxY;
}

public java.lang.String getId() {
    return id;
}

public void setId(java.lang.String id) {
    this.id = id;
}

public double getMinZ() {
    return minZ;
}

public void setMinZ(double minZ) {
    this.minZ = minZ;
}

public double getMaxZ() {
```

```

        return maxZ;
    }

    public void setMaxZ(double maxZ) {
        this.maxZ = maxZ;
    }
}

```

## 2.6.4 A classe AreaLevantamentoParser

```

package com.vft3d.vftserv.persist;

import org.xml.sax.*;
import java.util.ArrayList;

public class AreaLevantamentoParser implements ContentHandler {

    private java.lang.StringBuffer buffer;
    private AreaLevantamentoHandler handler;
    private java.util.Stack context;
    private EntityResolver resolver;

    public AreaLevantamentoParser( final AreaLevantamentoHandler handler
                                   ,final EntityResolver resolver) {
        this.handler = handler;
        this.resolver = resolver;
        buffer = new StringBuffer(111);
        context = new java.util.Stack();
    }

    public final void setDocumentLocator(Locator locator) {
        handler.setDocumentLocator(locator);
    }

    public final void startDocument() throws SAXException {
        handler.startDocument();
    }

    public final void endDocument() throws SAXException {
        handler.endDocument();
    }

    public final void startElement(
        String ns,
        String name,
        String qname,
        Attributes attrs
    ) throws SAXException
    {
        dispatch(true);
        context.push(
            new Object[] {qname, new org.xml.sax.helpers.AttributesImpl(attrs)}
        );
        if ("Face".equals(qname)) {
            handler.handle_Face(attrs);
        } else if ("FenomenoTerrestre".equals(qname)) {
            handler.start_FenomenoTerrestre(attrs);
        } else if ("Superficie2D".equals(qname)) {
            handler.start_Superficie2D(attrs);
        } else if ("Superficie1D".equals(qname)) {
            handler.start_Superficie1D(attrs);
        } else if ("AreaLevantamento".equals(qname)) {

```



```

        handler.start_AreaLevantamento(attrs);
    } else if ("Superficie0D".equals(qname)) {
        handler.start_Superficie0D(attrs);
    } else if ("Atributo".equals(qname)) {
        handler.handle_Atributo(attrs);
    } else if ("Cor".equals(qname)) {
        handler.handle_Cor(attrs);
    } else if ("Ponto".equals(qname)) {
        handler.handle_Ponto(attrs);
    } else if ("Superficie3D".equals(qname)) {
        handler.start_Superficie3D(attrs);
    }
    handler.startElement(ns, name, qname, attrs);
}

public final void endElement(String ns, String name, String qname)
throws SAXException {
    dispatch(false);
    context.pop();
    if ("FenomenoTerrestre".equals(qname)) {
        handler.end_FenomenoTerrestre();
    } else if ("Superficie2D".equals(qname)) {
        handler.end_Superficie2D();
    } else if ("Superficie1D".equals(qname)) {
        handler.end_Superficie1D();
    } else if ("AreaLevantamento".equals(qname)) {
        handler.end_AreaLevantamento();
    } else if ("Superficie0D".equals(qname)) {
        handler.end_Superficie0D();
    } else if ("Superficie3D".equals(qname)) {
        handler.end_Superficie3D();
    }
    handler.endElement(ns, name, qname);
}

public final void characters(char[] chars, int start, int len)
throws SAXException {
    buffer.append(chars, start, len); handler.characters(chars, start, len);
}

public final void ignorableWhitespace(char[] chars, int start, int len)
throws SAXException {
    handler.ignorableWhitespace(chars, start, len);
}

public final void processingInstruction(String target, String data)
throws SAXException {
    handler.processingInstruction(target, data);
}

public final void startPrefixMapping(final String prefix, final String uri)
throws SAXException {
    handler.startPrefixMapping(prefix, uri);
}

public final void endPrefixMapping(final String prefix)
throws SAXException {
    handler.endPrefixMapping(prefix);
}

public final void skippedEntity(String name) throws SAXException {
    handler.skippedEntity(name);
}

private void dispatch(final boolean fireOnlyIfMixed) throws SAXException {
    if (fireOnlyIfMixed && buffer.length() == 0) return; //skip it
}

```

```

        Object[] ctx = (Object[]) context.peek();
        String here = (String) ctx[0];
        Attributes attrs = (Attributes) ctx[1];
        buffer.delete(0, buffer.length());
    }

    public void parse(final InputSource input)
    throws
        SAXException,
        javax.xml.parsers.ParserConfigurationException,
        java.io.IOException
    {
        parse(input, this);
    }

    public void parse(final java.net.URL url)
    throws
        SAXException,
        javax.xml.parsers.ParserConfigurationException,
        java.io.IOException
    {
        parse(new InputSource(url.toExternalForm()), this);
    }

    public static void parse(
        final InputSource input,
        final AreaLevantamentoHandler handler
    )
    throws
        SAXException,
        javax.xml.parsers.ParserConfigurationException,
        java.io.IOException
    {
        parse(input, new AreaLevantamentoParser(handler, null));
    }

    public static void parse(
        final java.net.URL url,
        final AreaLevantamentoHandler handler)
    throws
        SAXException,
        javax.xml.parsers.ParserConfigurationException,
        java.io.IOException
    {
        parse(new InputSource(url.toExternalForm()), handler);
    }

    private static void parse(
        final InputSource input,
        final AreaLevantamentoParser recognizer
    )
    throws
        SAXException,
        javax.xml.parsers.ParserConfigurationException,
        java.io.IOException
    {
        javax.xml.parsers.SAXParserFactory
        factory = javax.xml.parsers.SAXParserFactory.newInstance();
        factory.setValidating(true); //the code was generated according DTD
        factory.setNamespaceAware(false); //the code was generated according DTD
        XMLReader parser = factory.newSAXParser().getXMLReader();
        parser.setContentHandler(recognizer);
        parser.setErrorHandler(recognizer.getDefaultErrorHandler());
        if (recognizer.resolver != null)
            parser.setEntityResolver(recognizer.resolver);
    }

```

```

        parser.parse(input);
    }

    protected ErrorHandler getDefaultErrorHandler() {
        return new ErrorHandler() {
            public void error(SAXParseException ex) throws SAXException {
                if (context.isEmpty()) System.err.println("Missing DOCTYPE.");
                throw ex;
            }

            public void fatalError(SAXParseException ex) throws SAXException {
                throw ex;
            }

            public void warning(SAXParseException ex) throws SAXException {
                // ignore
            }
        };
    }

    public AreaLevantamentoBD getAreaLevantamentoBD() {
        return handler.getAreaLevantamentoBD();
    }

    public ArrayList getFenomenosTerrestresBD() {
        return handler.getFenomenosTerrestresBD();
    }
}

```

### 2.6.5 A classe AreaLevantamentoHandler

```

package com.vft3d.vftserv.persist;

import org.xml.sax.*;
import java.util.Date;
import java.util.ArrayList;
import java.util.Enumeration;

public class AreaLevantamentoHandler implements ContentHandler {
    AreaLevantamentoBD areaLevantamentoBD;
    FenomenoTerrestreBD fenomenoTerrestreBD;
    ArrayList fenomenosBD;
    String xmlFenomeno="";
    String dtdURL;

    public static final boolean DEBUG = false;

    public AreaLevantamentoHandler(String dtdURL) {
        this.dtdURL=dtdURL;
    }

    public void handle_Face(final Attributes meta) throws SAXException {
        if (DEBUG) System.err.println("handle_Face: " + meta);
        xmlFenomeno+="

```

```

public void start_FenomenoTerrestre(final Attributes meta)
throws SAXException {
    if (DEBUG) System.err.println("start_FenomenoTerrestre: " + meta);
    // xml header
    xmlFenomeno="<?xml version=\"1.0\" encoding=\"UTF-8\"?>\r\n";
    xmlFenomeno+="<!DOCTYPE FenomenoTerrestre SYSTEM \""
        + dtdURL + "FenomenoTerrestre.dtd\">\r\n";

    xmlFenomeno+="<FenomenoTerrestre ";
    for (int i=0; i<meta.getLength(); i++) {
        xmlFenomeno+=" "+meta.getQName(i)+"=\""+meta.getValue(i)+"\" ";
    }
    xmlFenomeno+=">\r\n";

    fenomenoTerrestreBD= new FenomenoTerrestreBD();

    // categoria
    String categoria=meta.getValue("categoria");
    if (categoria.equals("FenomenoTopografico"))
        fenomenoTerrestreBD.setCategoria(0);
    else if (categoria.equals("CorpoHidrico"))
        fenomenoTerrestreBD.setCategoria(1);
    else if (categoria.equals("SuperficieTopografica"))
        fenomenoTerrestreBD.setCategoria(2);
    else throw new SAXException("Valor invalido para categoria");

    //tipo
    fenomenoTerrestreBD.tipo=meta.getValue("tipo");

    //dimensao
    try {
        int d = Integer.parseInt(meta.getValue("dimensao"));
        if (d<0 || d> 3)
            throw new SAXException("Valor inválido para dimensao");
        fenomenoTerrestreBD.setDimensao(d);
    } catch (Exception e) {
        throw new SAXException("Valor invalido para dimensao");
    }

    //areaLevantamentoId
    fenomenoTerrestreBD.areaLevantamentoID=areaLevantamentoBD.getId();
}

public void end_FenomenoTerrestre() throws SAXException {
    if (DEBUG) System.err.println("end_FenomenoTerrestre()");
    xmlFenomeno+="</FenomenoTerrestre>\r\n";

    fenomenoTerrestreBD.setXml(xmlFenomeno);
    fenomenosBD.add(fenomenoTerrestreBD);
}

public void start_Superficie2D(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("start_Superficie2D: " + meta);
    xmlFenomeno+="<Superficie2D";
    for (int i=0; i<meta.getLength(); i++) {
        xmlFenomeno+=" "+meta.getQName(i)+"=\""+meta.getValue(i)+"\" ";
    }
    xmlFenomeno+=">\r\n";
}

public void end_Superficie2D() throws SAXException {
    if (DEBUG) System.err.println("end_Superficie2D()");
    xmlFenomeno+="</Superficie2D>\r\n";
}

```

```

public void start_Superficie1D(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("start_Superficie1D: " + meta);
}

public void end_Superficie1D() throws SAXException {
    if (DEBUG) System.err.println("end_Superficie1D()");
    xmlFenomeno+="</Superficie1D>\r\n";
}

public void start_AreaLevantamento(final Attributes meta)
throws SAXException {
    if (DEBUG) System.err.println("start_AreaLevantamento: " + meta);
    areaLevantamentoBD=new AreaLevantamentoBD();
    fenomenosBD=new ArrayList();
    areaLevantamentoBD.setId(meta.getValue("id").trim());
    areaLevantamentoBD.setNome(meta.getValue("nome").trim());
    areaLevantamentoBD.setTempo(new Date(meta.getValue("tempo").trim()));
}

public void end_AreaLevantamento() throws SAXException {
    if (DEBUG) System.err.println("end_AreaLevantamento()");
}

public void start_Superficie0D(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("start_Superficie0D: " + meta);
    xmlFenomeno+="<Superficie0D";
    for (int i=0; i<meta.getLength(); i++) {
        xmlFenomeno+=" "+meta.getQName(i)+"=\""+meta.getValue(i)+"\" ";
    }
    xmlFenomeno+=">\r\n";
}

public void end_Superficie0D() throws SAXException {
    if (DEBUG) System.err.println("end_Superficie0D()");
    xmlFenomeno+="</Superficie0D>\r\n";
}

public void handle_Atributo(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("handle_Atributo: " + meta);

    xmlFenomeno+="<Atributo";
    for (int i=0; i<meta.getLength(); i++) {
        xmlFenomeno+=" "+meta.getQName(i)+"=\""+meta.getValue(i)+"\" ";
    }
    xmlFenomeno+=">\r\n";
}

public void handle_Cor(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("handle_Cor: " + meta);
    xmlFenomeno+="<Cor";
    for (int i=0; i<meta.getLength(); i++) {
        xmlFenomeno+=" "+meta.getQName(i)+"=\""+meta.getValue(i)+"\" ";
    }
    xmlFenomeno+=">\r\n";
}

public void handle_Ponto(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("handle_Ponto: " + meta);

    xmlFenomeno+="<Ponto";
    for (int i=0; i<meta.getLength(); i++) {
        xmlFenomeno+=" "+meta.getQName(i)+"=\""+meta.getValue(i)+"\" ";
    }
    xmlFenomeno+=">\r\n";
}

```

```

double x,y,z;
try {
    x= Double.parseDouble(meta.getValue("x"));
    y= Double.parseDouble(meta.getValue("y"));
    z= Double.parseDouble(meta.getValue("z"));
}
catch (Exception e) {
    throw new SAXException("Valor invalido para x, y ou z");
}

// determina os pontos min e max da area de levantamento
areaLevantamentoBD.setMinMaxXY(x,y,z);

// determina os pontos min e max do fenomeno terrestre
fenomenoTerrestreBD.setMinMaxXY(x,y,z);
}

public void start_Superficie3D(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("start_Superficie3D: " + meta);
    xmlFenomeno+="

```

```

        Attributes atts) throws SAXException {
    }

    public void startPrefixMapping(String prefix, String uri)
    throws SAXException {
    }

    public AreaLevantamentoBD getAreaLevantamentoBD() {
        return areaLevantamentoBD;
    }

    public ArrayList getFenomenosTerrestresBD() {
        return fenomenosBD;
    }
}

```

### 2.6.6 A classe FenomenoTerrestreBD

```

package com.vft3d.vftserv.persist;

public class FenomenoTerrestreBD {
    int id;
    int categoria;
    String tipo;
    int dimensao;
    double minX;
    double minY;
    double minZ;
    double maxX;
    double maxY;
    double maxZ;
    String xml;
    String areaLevantamentoID;

    public FenomenoTerrestreBD() {
        minX=Double.MAX_VALUE;
        minY=Double.MAX_VALUE;
        minZ=Double.MAX_VALUE;
        maxX=Double.MIN_VALUE;
        maxY=Double.MIN_VALUE;
        maxZ=Double.MIN_VALUE;
    }

    public void setMinMaxXY(double x, double y, double z) {
        if (x<minX) { minX=x; }
        if (y<minY) { minY=y; }
        if (z<minZ) { minZ=z; }
        if (x>maxX) { maxX=x; }
        if (y>maxY) { maxY=y; }
        if (z>maxZ) { maxZ=z; }
    }

    public String getAreaLevantamentoID() {
        return areaLevantamentoID;
    }

    public void setAreaLevantamentoID(String areaLevantamentoID) {
        this.areaLevantamentoID = areaLevantamentoID;
    }
}

```

```
public int getCategoria() {
    return categoria;
}

public void setCategoria(int categoria) {
    this.categoria = categoria;
}

public java.lang.String getTipo() {
    return tipo;
}

public void setTipo(java.lang.String tipo) {
    this.tipo = tipo;
}

public int getDimensao() {
    return dimensao;
}

public void setDimensao(int dimensao) {
    this.dimensao = dimensao;
}

public double getMinX() {
    return minX;
}

public void setMinX(double minX) {
    this.minX = minX;
}

public double getMinY() {
    return minY;
}

public void setMinY(double minY) {
    this.minY = minY;
}

public double getMaxX() {
    return maxX;
}

public void setMaxX(double maxX) {
    this.maxX = maxX;
}

public double getMaxY() {
    return maxY;
}

public void setMaxY(double maxY) {
    this.maxY = maxY;
}

public java.lang.String getXml() {
    return xml;
}

public void setXml(java.lang.String xml) {
    this.xml = xml;
}

public int getId() {
    return id;
}
```



```

    }

    public void setId(int id) {
        this.id = id;
    }

    public double getMinZ() {
        return minZ;
    }

    public void setMinZ(double minZ) {
        this.minZ = minZ;
    }

    public double getMaxZ() {
        return maxZ;
    }

    public void setMaxZ(double maxZ) {
        this.maxZ = maxZ;
    }
}

```

### 2.6.7 A classe FenomenoTerrestreParser

```

package com.vft3d.vftserv.persist;

import org.xml.sax.*;
import java.util.ArrayList;
import com.vft3d.vftserv.model.sup.Superficie;

public class FenomenoTerrestreParser implements ContentHandler {

    private java.lang.StringBuffer buffer;
    private FenomenoTerrestreHandler handler;
    private java.util.Stack context;
    private EntityResolver resolver;

    public FenomenoTerrestreParser( final FenomenoTerrestreHandler handler,
                                     final EntityResolver resolver) {

        this.handler = handler;
        this.resolver = resolver;
        buffer = new StringBuffer(111);
        context = new java.util.Stack();
    }

    public final void setDocumentLocator(Locator locator) {
        handler.setDocumentLocator(locator);
    }

    public final void startDocument() throws SAXException {
        handler.startDocument();
    }

    public final void endDocument() throws SAXException {
        handler.endDocument();
    }

    public final void startElement(
        String ns,

```

```

    String name,
    String qname,
    Attributes attrs)
throws SAXException {
    dispatch(true);
    context.push(
        new Object[] {qname, new org.xml.sax.helpers.AttributesImpl(attrs)}
    );
    if ("Face".equals(qname)) {
        handler.handle_Face(attrs);
    } else if ("FenomenoTerrestre".equals(qname)) {
        handler.start_FenomenoTerrestre(attrs);
    } else if ("Superficie2D".equals(qname)) {
        handler.start_Superficie2D(attrs);
    } else if ("Superficie1D".equals(qname)) {
        handler.start_Superficie1D(attrs);
    } else if ("Superficie0D".equals(qname)) {
        handler.start_Superficie0D(attrs);
    } else if ("Atributo".equals(qname)) {
        handler.handle_Atributo(attrs);
    } else if ("Cor".equals(qname)) {
        handler.handle_Cor(attrs);
    } else if ("Ponto".equals(qname)) {
        handler.handle_Ponto(attrs);
    } else if ("Superficie3D".equals(qname)) {
        handler.start_Superficie3D(attrs);
    }
    handler.startElement(ns, name, qname, attrs);
}

public final void endElement(
    String ns,
    String name,
    String qname
) throws SAXException {
    dispatch(false);
    context.pop();
    if ("FenomenoTerrestre".equals(qname)) {
        handler.end_FenomenoTerrestre();
    } else if ("Superficie2D".equals(qname)) {
        handler.end_Superficie2D();
    } else if ("Superficie1D".equals(qname)) {
        handler.end_Superficie1D();
    } else if ("Superficie0D".equals(qname)) {
        handler.end_Superficie0D();
    } else if ("Superficie3D".equals(qname)) {
        handler.end_Superficie3D();
    }
    handler.endElement(ns, name, qname);
}

public final void characters(char[] chars, int start, int len)
throws SAXException {
    buffer.append(chars, start, len); handler.characters(chars, start, len);
}

public final void ignorableWhitespace(char[] chars, int start, int len)
throws SAXException {
    handler.ignorableWhitespace(chars, start, len);
}

public final void processingInstruction(String target, String data)
throws SAXException {
    handler.processingInstruction(target, data);
}

```

```

public final void startPrefixMapping(final String prefix, final String uri)
throws SAXException {
    handler.startPrefixMapping(prefix, uri);
}

public final void endPrefixMapping(final String prefix)
throws SAXException {
    handler.endPrefixMapping(prefix);
}

public final void skippedEntity(java.lang.String name) throws SAXException {
    handler.skippedEntity(name);
}

private void dispatch(final boolean fireOnlyIfMixed) throws SAXException {
    if (fireOnlyIfMixed && buffer.length() == 0) return; //skip it

    Object[] ctx = (Object[]) context.peek();
    String here = (String) ctx[0];
    Attributes attrs = (Attributes) ctx[1];
    buffer.delete(0, buffer.length());
}

public void parse(final InputSource input)
throws
    SAXException,
    javax.xml.parsers.ParserConfigurationException,
    java.io.IOException
{
    parse(input, this);
}

public void parse(final java.net.URL url)
throws SAXException
    , javax.xml.parsers.ParserConfigurationException
    , java.io.IOException
{
    parse(new InputSource(url.toExternalForm()), this);
}

public static void parse(
    final InputSource input,
    final FenomenoTerrestreHandler handler
) throws
    SAXException,
    javax.xml.parsers.ParserConfigurationException,
    java.io.IOException
{
    parse(input, new FenomenoTerrestreParser(handler, null));
}

public static void parse(
    final java.net.URL url,
    final FenomenoTerrestreHandler handler
)
throws
    SAXException,
    javax.xml.parsers.ParserConfigurationException,
    java.io.IOException
{
    parse(new InputSource(url.toExternalForm()), handler);
}

private static void parse(
    final InputSource input,
    final FenomenoTerrestreParser recognizer

```

```

    )
    throws
        SAXException,
        javax.xml.parsers.ParserConfigurationException,
        java.io.IOException
    {
        javax.xml.parsers.SAXParserFactory
        factory = javax.xml.parsers.SAXParserFactory.newInstance();
        factory.setValidating(true); //the code was generated according DTD
        factory.setNamespaceAware(false); //the code was generated according DTD
        XMLReader parser = factory.newSAXParser().getXMLReader();
        parser.setContentHandler(recognizer);
        parser.setErrorHandler(recognizer.getDefaultErrorHandler());
        if (recognizer.resolver != null)
            parser.setEntityResolver(recognizer.resolver);
        parser.parse(input);
    }

    protected ErrorHandler getDefaultErrorHandler() {
        return new ErrorHandler() {
            public void error(SAXParseException ex) throws SAXException {
                if (context.isEmpty())
                    System.err.println("Missing DOCTYPE.");
                throw ex;
            }

            public void fatalError(SAXParseException ex) throws SAXException {
                throw ex;
            }

            public void warning(SAXParseException ex) throws SAXException {
                // ignore
            }
        };
    }

    public ArrayList getAtributos() {
        return handler.getAtributos();
    }

    public Superficie getSuperficie() {
        return handler.getSuperficie();
    }
}

```

### 2.6.8 A classe FenomenoTerrestreHandler

```

package com.vft3d.vftserv.persist;

import org.xml.sax.*;
import com.vft3d.vftserv.model.fen.Atributo;
import com.vft3d.vftserv.model.sup.*;
import java.util.Date;
import java.util.ArrayList;

public class FenomenoTerrestreHandler implements ContentHandler {

    public static final boolean DEBUG = false;

```

```

ArrayList atributos;
Superficie superficie;
ArrayList pontos;
ArrayList faces;
String modeloVRMLSuperficieOD=null;
Cor cor=null;

public void handle_Face(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("handle_Face: " + meta);

    String[] prefsSplit;
    String prefsStr = meta.getValue("prefs");
    prefsSplit=prefsStr.split(" ");
    int[] prefs= new int[prefsSplit.length];
    for (int i=0; i<prefsSplit.length; i++) {
        prefs[i]=Integer.parseInt(prefsSplit[i]);
    }
    faces.add(new Face(prefs));
}

public void start_FenomenoTerrestre(final Attributes meta)
throws SAXException {
    if (DEBUG) System.err.println("start_FenomenoTerrestre: " + meta);
    atributos=new ArrayList();
    superficie=null;
    pontos=new ArrayList();
    faces=new ArrayList();
}

public void end_FenomenoTerrestre() throws SAXException {
    if (DEBUG) System.err.println("end_FenomenoTerrestre()");
}

public void start_Superficie2D(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("start_Superficie2D: " + meta);
}

public void end_Superficie2D() throws SAXException {
    if (DEBUG) System.err.println("end_Superficie2D()");
    superficie=new Superficie2D(pontos);
    superficie.setCor(cor);
}

public void start_Superficie1D(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("start_Superficie1D: " + meta);
}

public void end_Superficie1D() throws SAXException {
    if (DEBUG) System.err.println("end_Superficie1D()");
    superficie=new Superficie1D(pontos);
    superficie.setCor(cor);
}

public void start_SuperficieOD(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("start_SuperficieOD: " + meta);
    modeloVRMLSuperficieOD=null;
    modeloVRMLSuperficieOD = meta.getValue("modeloVRML");
}

public void end_SuperficieOD() throws SAXException {
    if (DEBUG) System.err.println("end_SuperficieOD()");
    superficie=new SuperficieOD((Ponto)pontos.get(0),modeloVRMLSuperficieOD);
    superficie.setCor(cor);
    modeloVRMLSuperficieOD=null;
}

```

```

public void handle_Atributo(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("handle_Atributo: " + meta);

    Atributo atributo=new Atributo(
        meta.getValue("nome"),        //nome
        meta.getValue("valor"),       // valor
        meta.getValue("tempo").equals("")?null:new Date(meta.getValue("tempo"))
    );
    atributos.add(atributo);
}

public void handle_Cor(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("handle_Cor: " + meta);
    cor=new Cor(
        Double.parseDouble(meta.getValue(0)), //r
        Double.parseDouble(meta.getValue(1)), //g
        Double.parseDouble(meta.getValue(2))  //b
    );
}

public void handle_Ponto(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("handle_Ponto: " + meta);
    Ponto ponto= new Ponto(
        Double.parseDouble(meta.getValue(0)), //x
        Double.parseDouble(meta.getValue(1)), //y
        Double.parseDouble(meta.getValue(2)), //z
        (meta.getValue(3)=="S"|| meta.getValue(3)=="s"?true:false,
        (meta.getValue(4)=="S"|| meta.getValue(4)=="s"?true:false
    );
    pontos.add(ponto);
}

public void start_Superficie3D(final Attributes meta) throws SAXException {
    if (DEBUG) System.err.println("start_Superficie3D: " + meta);
}

public void end_Superficie3D() throws SAXException {
    if (DEBUG) System.err.println("end_Superficie3D()");
    superficie=new Superficie3D(pontos,faces);
    superficie.setCor(cor);
}

public void characters(char[] ch, int start, int length)
throws SAXException {
}

public void endDocument() throws SAXException {
}

public void endElement(String namespaceURI, String localName, String qName)
throws SAXException {
}

public void endPrefixMapping(String prefix) throws SAXException {
}

public void ignorableWhitespace(char[] ch, int start, int length)
throws SAXException {
}

public void processingInstruction(String target, String data)
throws SAXException {
}

public void setDocumentLocator(Locator locator) {
}

```

```
}

public void skippedEntity(String name) throws SAXException {
}

public void startDocument() throws SAXException {
}

public void startElement(String namespaceURI,
                        String localName,
                        String qName,
                        Attributes atts)
throws SAXException {
}

public void startPrefixMapping(String prefix, String uri)
throws SAXException {
}

public ArrayList getAtributos() {
    return atributos;
}

public Superficie getSuperficie() {
    return superficie;
}
}
```

### 3 ARQUIVO DE PROTOTIPOS VRML

#### 3.1 ARQUIVO protos.wrl

```

#VRML V2.0 utf8
EXTERNPROTO GeoViewpoint [
  eventIn      SFBool    set_bind
  eventIn      SFString  set_position
  eventIn      SFString  set_orientation
  exposedField SFFloat   fieldOfView
  exposedField SFBool    headlight
  exposedField SFBool    jump
  exposedField MFString  navType
  field        SFString  description
  field        SFNode    geoOrigin
  field        MFString  geoSystem
  field        SFString  position
  field        SFRotation orientation
  field        SFFloat   speedFactor
  eventOut     SFTIME    bindTime
  eventOut     SFBool    isBound
]
[
  "file:///C:/Arquivos%20de%20programas/GeoVRML/1.1/protos/GeoViewpoint.wrl"
  "file:///C:/Program%20Files/GeoVRML/1.1/protos/GeoViewpoint.wrl"
  "http://www.geovrml.org/1.1/protos/GeoViewpoint.wrl"
]

EXTERNPROTO GeoCoordinate [
  field SFNode    geoOrigin
  field MFString  geoSystem
  field MFString  point
]
[
  "file:///C:/Arquivos%20de%20programas/GeoVRML/1.1/protos/GeoCoordinate.wrl"
  "file:///C:/Program%20Files/GeoVRML/1.1/protos/GeoCoordinate.wrl"
  "http://www.geovrml.org/1.1/protos/GeoCoordinate.wrl"
]

EXTERNPROTO GeoLocation [
  exposedField SFString  geoCoords
  field        SFNode    geoOrigin
  field        MFString  geoSystem
  field        MFNode    children

  # most apps won't include this field in the EXTERNPROTO interface
  field        SFBool    debug
]
[
  "file:///C:/Arquivos%20de%20programas/GeoVRML/1.1/protos/GeoLocation.wrl"
  "file:///C:/Program%20Files/GeoVRML/1.1/protos/GeoLocation.wrl"
  "http://www.geovrml.org/1.1/protos/GeoLocation.wrl"
]

PROTO Superficie3D [
  field MFString  pontos []
  field MFInt32faces []
  field SFColorcor      0 0.8 0
]

```



```

{
  Collision {
    children [
      Shape {
        appearance Appearance {
          material Material {
            diffuseColor IS cor
          }
        }
        geometry IndexedFaceSet {
          coord GeoCoordinate {
            geoSystem [ "UTM", "Z22", "S", "WGS84"
"easting_first"]
            point IS pontos
          }
          coordIndex IS faces
          colorPerVertex TRUE
          solid FALSE
        }
      }
    ]
    collide TRUE
  }
}

PROTO Superficie2D [
  field MFString      pontos []
  field MFInt32face   []
  field SFColorcor    0 0.8 0
]
{
  Collision {
    children [
      Shape {
        appearance Appearance {
          material Material {
            diffuseColor IS cor
          }
        }
        geometry IndexedFaceSet {
          coord GeoCoordinate {
            geoSystem [ "UTM", "Z22", "S", "WGS84"
"easting_first"]
            point IS pontos
          }
          coordIndex IS face
          colorPerVertex TRUE
          solid FALSE
        }
      }
    ]
    collide TRUE
  }
}

PROTO Superficie1D [
  field MFString      pontos []
  field SFColorcor    0 0.8 0
]
{
  Collision {
    children [
      Shape {
        appearance Appearance {

```

```

        material Material {
            diffuseColor IS cor
        }
    }
    geometry IndexedLineSet {
        coord GeoCoordinate {
            geoSystem [ "UTM", "Z22", "S", "WGS84"
"eastings_first"]
            point IS pontos
        }
        colorPerVertex TRUE
    }
}
collide TRUE
}
}
PROTO Superficie0D [
    field SFString ponto ""
    field MFString modelo "ponto.wrl"
    field SFColorcor 0 0 0.8
]
{
    GeoLocation {
        geoSystem [ "UTM", "Z22", "S", "WGS84" "eastings_first"]
        geoCoords IS ponto
        children [ Inline { url IS modelo } ]
    }
}
}

```

## 4 DEFINIÇÃO DO BANCO DE DADOS

### 4.1 AreasLevantamento

```
create table `AreasLevantamento`  
(  
    `id` VARCHAR(10) unique,  
    `nome` VARCHAR(20),  
    `tempo` TIME STAMP,  
    `minX` DOUBLE,  
    `minY` DOUBLE,  
    `minZ` DOUBLE,  
    `maxX` DOUBLE,  
    `maxY` DOUBLE,  
    `maxZ` DOUBLE  
)
```

### 4.2 Fenomenos

```
create table `Fenomenos`  
(  
    `id` INTEGER not null unique,  
    `categoria` INTEGER,  
    `tipo` VARCHAR(30),  
    `dimensao` SMALLINT(5),  
    `minX` DOUBLE,  
    `minY` DOUBLE,  
    `minZ` DOUBLE,  
    `maxX` DOUBLE,  
    `maxY` DOUBLE,  
    `maxZ` DOUBLE,  
    `xml` LONGVARCHAR(1073741823),  
    `areaLevantamentoId` VARCHAR(10)  
)
```

## 5 ARQUIVOS DTD (DOCUMENT TYPE DEFINITION)

### 5.1 AreaLevantamento.dtd

```

<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT AreaLevantamento (FenomenoTerrestre+)>
<!ATTLIST AreaLevantamento
    id NMTOKEN #REQUIRED
    nome CDATA #IMPLIED
    tempo CDATA #REQUIRED
>
<!ELEMENT FenomenoTerrestre (Cor?, Atributo*, (Superficie3D | Superficie0D |
Superficie1D | Superficie2D))>
<!ATTLIST FenomenoTerrestre
    categoria (SuperficieTopografica | FenomenoTopografico | CorpoHidrico)
#REQUIRED
    tipo NMTOKEN #IMPLIED
    dimensao (0 | 1 | 2 | 3) #REQUIRED
>
<!ELEMENT Atributo EMPTY>
<!ATTLIST Atributo
    nome NMTOKEN #REQUIRED
    valor CDATA #REQUIRED
    tempo CDATA #IMPLIED
>
<!ELEMENT Superficie3D (Ponto+, Face+)>
<!ELEMENT Superficie2D (Ponto+)>
<!ELEMENT Superficie1D (Ponto+)>
<!ELEMENT Superficie0D (Ponto+)>
<!ATTLIST Superficie0D
    modeloVRML NMTOKEN #IMPLIED
>
<!ELEMENT Ponto EMPTY>
<!ATTLIST Ponto
    x NMTOKEN #REQUIRED
    y NMTOKEN #REQUIRED
    z NMTOKEN #REQUIRED
    medido (S | N) #FIXED "S"
    topografico (S | N) #FIXED "S"
>
<!ELEMENT Face EMPTY>
<!ATTLIST Face
    prefs NMTOKENS #REQUIRED
>
<!ELEMENT Cor EMPTY>
<!ATTLIST Cor
    r NMTOKEN #REQUIRED
    g NMTOKEN #REQUIRED
    b NMTOKEN #REQUIRED
>

```

### 5.2 FenomenoTerrestre.dtd

```

<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT FenomenoTerrestre (Cor?, Atributo*, (Superficie3D | Superficie0D |
Superficie1D | Superficie2D))>

```

```

<!ATTLIST FenomenoTerrestre
  categoria (SuperficieTopografica | FenomenoTopografico | CorpoHidrico)
#REQUIRED
  tipo NMTOKEN #IMPLIED
  dimensao (0 | 1 | 2 | 3) #REQUIRED
>
<!ELEMENT Atributo EMPTY>
<!ATTLIST Atributo
  nome NMTOKEN #REQUIRED
  valor CDATA #REQUIRED
  tempo CDATA #IMPLIED
>
<!ELEMENT Superficie3D (Ponto+, Face+)>
<!ELEMENT Superficie2D (Ponto+)>
<!ELEMENT Superficie1D (Ponto+)>
<!ELEMENT Superficie0D (Ponto)>
<!ATTLIST Superficie0D
  modeloVRML NMTOKEN #IMPLIED
>
<!ELEMENT Ponto EMPTY>
<!ATTLIST Ponto
  x NMTOKEN #REQUIRED
  y NMTOKEN #REQUIRED
  z NMTOKEN #REQUIRED
  medido (S | N) #FIXED "S"
  topografico (S | N) #FIXED "S"
>
<!ELEMENT Face EMPTY>
<!ATTLIST Face
  prefs NMTOKENS #REQUIRED
>
<!ELEMENT Cor EMPTY>
<!ATTLIST Cor
  r NMTOKEN #REQUIRED
  g NMTOKEN #REQUIRED
  b NMTOKEN #REQUIRED
>

```

## **APÊNDICE 2 – INSTALAÇÃO DA APLICAÇÃO TESTE VFTapp1**

## SUMÁRIO

<b>1 INSTALAÇÃO DA APLICAÇÃO.....</b>	<b>127</b>
1.1 INSTALAR E COMPILAR A CLASSE VFTAPP1 .....	127
1.2 INSTALAR AS PÁGINAS JSP.....	127
1.3 INSTALAR OS ARQUIVOS DE MODELOS VRML .....	127
1.4 INSTALAR O ARQUIVO DE DISTRIBUIÇÃO DA APLICAÇÃO WEB .....	127
1.5 PREPARAR O SERVIDOR TOMCAT .....	128
1.6 EXECUTAR A APLICAÇÃO .....	128
<b>2 A CLASSE VFTAPP1 .....</b>	<b>129</b>
<b>3 PÁGINAS JSP .....</b>	<b>130</b>
3.1 A PÁGINA DEFAULT.JSP.....	130
3.2 A PÁGINA HEADER.JSP .....	130
3.3 A PÁGINA CADALJSP .....	131
3.4 A PÁGINA CADALFORM.JSP .....	131
3.5 A PÁGINA INCLUIRAL.JSP .....	132
3.6 A PÁGINA UPLOADAL.JSP .....	133
3.7 A PÁGINA MENU.JSP .....	135
3.8 A PÁGINA SELECIONARJV.JSP.....	135
3.9 A PÁGINA SELJVFORM.JSP .....	136
3.10 A PÁGINA AREASLEVANTAMENTO.JSP.....	138
3.11 A PÁGINA AREASLEV.JSP .....	138
3.12 A PÁGINA VISUALIZARAL.JSP.....	141
3.13 A PÁGINA VISUALIZADORVRML.JSP.....	144
<b>4 ARQUIVOS DE MODELOS DE FENÔMENOS EM VRML.....</b>	<b>145</b>
4.1 ARVORE.WRL .....	145
4.2 POSTE.WRL .....	146
<b>5 ARQUIVO DE DISTRIBUIÇÃO DA APLICAÇÃO WEB (WEB.XML).....</b>	<b>148</b>
<b>6 ARQUIVO XML DE ÁREA DE LEVANTAMENTO EXEMPLO.....</b>	<b>150</b>

## 1 INSTALAÇÃO DA APLICAÇÃO

Para instalação da aplicação os seguintes preparativos são necessários:

- Instalar e compilar a classe VftApp1
- Instalar as páginas JSP
- Instalar os arquivos de modelos vrml
- Instalar o arquivo de distribuição da aplicação WEB
- Preparar o servidor Tomcat
- Executar a aplicação

### 1.1 INSTALAR E COMPILAR A CLASSE VFTApp1

Instalar a classe VFTApp1, descrita na seção 2, no diretório **vtf3d\vftserv\src**.  
Compilar a classe gerando o arquivo .class no diretório **vtf3d\vftserv\webapp\WEB-INF\classes**.

### 1.2 INSTALAR AS PÁGINAS JSP

Instalar as páginas, descritas na seção 3, no diretório **vtf3d\vftserv\webapp\jsp**.

### 1.3 INSTALAR OS ARQUIVOS DE MODELOS VRML

Instalar os arquivos, descritos na seção 4, no diretório **vtf3d\vftserv\webapp\wrl**.

### 1.4 INSTALAR O ARQUIVO DE DISTRIBUIÇÃO DA APLICAÇÃO WEB

Instalar o arquivo web.xml, descrito na seção 5, no diretório **vtf3d\vftserv\webapp\WEB-INF**.



## 1.5 PREPARAR O SERVIDOR TOMCAT

Fazer download do contenedor de servlets Tomcat 4.1 do site <http://jakarta.apache.org/tomcat/index.html>. Instalar o Tomcat segundo as instruções disponíveis no site.

Editar o arquivo servlet.xml localizado no diretório %CATALINA\_HOME%/conf criado durante a instalação do Tomcat. Procurar no arquivo por algo como o seguinte

```
<Context path="examples"docBase="examples"debug="0" reloadable="true">  
.  
.  
.  
</Context>
```

Logo após o tag </Context> acrescentar estas linhas:

```
<Context path="/vftserv" docBase="C:\vft3d\vftserv\webapp">  
</Context>
```

## 1.6 EXECUTAR A APLICAÇÃO

Executar o Internet Explorer com o seguinte endereço <http://localhost:8080/vftserv/control>. Para visualizar uma área de levantamento execute a opção Cadastrar Área de Levantamento enviando o arquivo xml da área de levantamento "AL01" cujo conteúdo é dado na seção 6.

## 2 A CLASSE VFTApp1

```
package com.vft3d.vftserv.control;

/* Esra classe implementa o controle da aplicação VFT1
 * extendendo a classe Controle da Estrutura de Aplicação
 */
public class VFTApp1 extends Controle {

    public String getTratador(String acao) {
        String url;
        if (acao.equals("cadastrarAL"))
            return "CadAL.jsp";
        if (acao.equals("selecionarJV"))
            return "SelecionarJV.jsp";
        else if (acao.equals("areasLevantamento"))
            return "AreasLevantamento.jsp";
        else if (acao.equals("incluirAL") )
            return "IncluirAL.jsp";
        else if (acao.equals("visualizarAL") )
            return "VisualizarAL.jsp";
        return null;
    }
}
```

### 3 PÁGINAS JSP

#### 3.1 A PÁGINA Default.jsp

```

<%@page contentType="text/html"%>
<jsp:useBean id="bdBean" scope="application"
class="com.vft3d.vftserv.persist.BdFTBean"/>
<%
    String baseURL = (String) application.getAttribute("baseURL");
%>
<html>
<head>
<title>VFT3D</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body bgcolor="#FFFFFF" text="#000000" leftmargin="0" topmargin="0">
<table width="100%" height="100%" id="vftapp" align="left" valign="top"
    vspace="0" hspace="0" border="1" cellspacing="0" cellpadding="0"
    bgcolor="#FFFCC" name="VFT" >
<tr id="VFTApplet" bgcolor="#CCCCCC" >
    <td id="Header" height="23" valign="top" align="left" COLSPAN="2">
        <jsp:include page="Header.jsp" flush="true"/>
    </td>
</tr>
<tr>
    <td id="Menu" width="20%" height="100%" BGCOLOR="F6F6F6"
        valign="top" align="left" >
        <jsp:include page="Menu.jsp" flush="true"/>
    </td>
    <td id="Conteudo" width="80%" height="100%" bgcolor="#F6F6F6"
        valign="middle" align="CENTER">
        <jsp:include page="VisualizadorVRML.jsp" flush="true" >
            <jsp:param name="cenaURL" value="wrl/cube3scriptless.wrl"/>
        </jsp:include>
    </td>
</tr>
</table>
</body>
</html>

```

#### 3.2 A PÁGINA Header.jsp

```

<%@ page import="java.util.*" %>
<jsp:useBean id="bdBean" scope="application"
    class="com.vft3d.vftserv.persist.BdFTBean"/>
<%
    String baseURL = (String) application.getAttribute("baseURL");
%>
<TABLE WIDTH="100%" HEIGHT="20" CELLSPACING="0" BORDER="0">
<TR BGCOLOR="#CCCCCC">
    <TD ALIGN="left" >
        <FONT FACE="Verdana" SIZE="4">
            <FONT COLOR="#003366" SIZE="3">
                <B>VFT 3D - Visualiza&ccedil;&atilde;o de Fen&ocirc;menos Terrestres</B>
            </FONT>
        </FONT>
    </TD>
</TR>
</TABLE>

```

```
</TABLE>
```

### 3.3 A PÁGINA CadAL.jsp

```
<%@page contentType="text/html"%>
<jsp:useBean id="bdBean" scope="application"
    class="com.vft3d.vftserv.persist.BdFTBean"/>
<%
    String baseURL = (String) application.getAttribute("baseURL");
%>
<html>
<head>
<title>VFT3D</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body bgcolor="#FFFFFF" text="#000000" leftmargin="0" topmargin="0">
<table width="100%" height="100%" id="vftapp"
    align="left" valign="top" vspace="0" hspace="0"
    border="1" cellspacing="0" cellpadding="0" bgcolor="#FFFCC" name="VFT" >
<tr id="VFTApplet" bgcolor="#CCCCCC" >
    <td id="Header" height="23" valign="top" align="left" COLSPAN="2">
        <jsp:include page="Header.jsp" flush="true"/>
    </td>
</tr>
<tr>
    <td id="Menu" width="20%" height="100%" BGCOLOR="F6F6F6"
        valign="top" align="left" >
        <jsp:include page="Menu.jsp" flush="true"/>
    </td>
    <td id="Conteudo" width="80%" height="100%" bgcolor="#F6F6F6"
        valign="middle" align="CENTER">
        <jsp:include page="CadALForm.jsp" flush="true"/>
    </td>
</tr>
</table>
</body>
</html>
```

### 3.4 A PÁGINA CadALForm.jsp

```
<%@page contentType="text/html"%>
<jsp:useBean id="bdBean" scope="application"
    class="com.vft3d.vftserv.persist.BdFTBean"/>
<%
    String baseURL = (String) application.getAttribute("baseURL");
%>
<TABLE WIDTH="518">
    <TR>
        <TD VALIGN="MIDDLE" ALIGN="CENTER">
            <FORM enctype="MULTIPART/FORM-DATA" method="POST"
                action="<%=baseURL%>?acao=incluirAL">
                <TABLE WIDTH="100%" BORDER="1" BORDERCOLOR="#CCCCCC">
                    <TR BGCOLOR="#CCCCCC">
                        <TD COLSPAN="2" VALIGN="MIDDLE" HEIGHT="30" ALIGN="CENTER">
                            <I><B><FONT FACE="Verdana" SIZE="3" COLOR="#003366">
                                Cadastro de &Aacute;rea de Levantamento
                            </FONT></B></I>
                        </TD>
                    </TR>
                </TABLE>
            </FORM>
        </TD>
    </TR>
</TABLE>
```

```

        </TD>
    </TR>
    <TR>
        <TD VALIGN="MIDDLE" WIDTH="135" ALIGN="CENTER">
            <B><FONT FACE="Verdana" SIZE="2">
                Nome da &acute;rea:
            </FONT></B>
        </TD>
        <TD WIDTH="367">
            <INPUT TYPE="TEXT" NAME="nomeArea" SIZE="50">
        </TD>
    </TR>
    <TR>
        <TD VALIGN="MIDDLE" WIDTH="135" ALIGN="CENTER">
            <B><FONT FACE="Verdana" SIZE="2">
                Descri&ccedil;&atilde;o em XML
            </FONT></B>
        </TD>
        <TD WIDTH="367">
            <INPUT TYPE="FILE" NAME="arqXML" SIZE="50">
        </TD>
    </TR>
    <TR>
        <TD VALIGN="MIDDLE" WIDTH="135" ALIGN="CENTER">
            &nbsp;
        </TD>
        <TD WIDTH="367">
            <input type="SUBMIT" value="Enviar" name="SUBMIT">
        </TD>
    </TR>
    <TR BGCOLOR="#CCCCCC">
        <TD VALIGN="MIDDLE" COLSPAN="2" ALIGN="CENTER">
            &nbsp;
        </TD>
    </TR>
</TABLE>
</FORM>
</TD>
</TR>
</TABLE>

```

### 3.5 A PÁGINA IncluirAL.jsp

```

<%@page contentType="text/html"%>
<jsp:useBean id="bdBean" scope="application"
class="com.vft3d.vftserv.persist.BdFTBean"/>
<%
    String baseURL = (String) application.getAttribute("baseURL");
%>
<html>
<head>
<title>VFT3D</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body bgcolor="#FFFFFF" text="#000000" leftmargin="0" topmargin="0">
<table width="100%" height="100%" id="vftapp"
    align="left" valign="top" vspace="0" hspace="0"
    border="1" cellspacing="0" cellpadding="0" bgcolor="#FFFFCC" name="VFT" >
<tr id="VFTApplet" bgcolor="#CCCCCC" >
    <td id="Header" height="23" valign="top" align="left" COLSPAN="2">
        <jsp:include page="Header.jsp" flush="true"/>
    </td>

```

```

</tr>
<tr>
  <td id="Menu" width="20%" height="100%" BGCOLOR="F6F6F6"
    valign="top" align="left" >
    <jsp:include page="Menu.jsp" flush="true"/>
  </td>
  <td id="Conteudo" width="80%" height="100%" bgcolor="#F6F6F6"
    valign="middle" align="CENTER">
    <jsp:include page="UploadAL.jsp" flush="true"/>
  </td>
</tr>
</table>
</body>
</html>

```

### 3.6 A PÁGINA UploadAL.jsp

```

<%@ page import="java.io.*" %>
<%@ page import="java.net.URL" %>
<%@ page import="java.util.Properties" %>
<jsp:useBean id="bdBean" scope="application"
  class="com.vft3d.vftserv.persist.BdFTBean"/>
<%
  String baseUrl = (String) application.getAttribute("baseUrl");

  // Diretorio onde o arquivo sera salvo.
  String xmlPath = (String) application.getAttribute("xmlPath");
  String name="";
  String nameValue="";
  String filename = "";
  Properties campos=new Properties();

  ServletInputStream in = request.getInputStream();

  byte[] line = new byte[255];
  int i = in.readLine(line, 0, 255);
  //-2 despreza o caractere nova-linha
  int boundaryLength = i - 2;
  String boundary = new String(line, 0, boundaryLength);

  while (i != -1) {
    boolean isFile=false;
    String newLine = new String(line, 0, i);
    if (newLine.startsWith("Content-Disposition: form-data; name=\"") {
      String s = new String(line, 0, i-2);
      int nStart=s.indexOf("\n");
      int nEnd=s.indexOf("\n",nStart+1);
      name=s.substring(nStart+1,nEnd);

      int pos = s.indexOf("filename=\"");
      if (pos != -1) {
        isFile = true;
        String filepath = s.substring(pos+10, s.length()-1);
        // Paginadores Windows incluem o caminho completo no cliente.
        // Os paginadores do Linux/Unix e Mac enviam somente o nome do
arquivo
        // Teste se eh caminho completo
        int pos2 = filepath.lastIndexOf("\\");
        if (pos2 != -1)
          filename = filepath.substring(pos2 + 1);

```

```

        else
            filename = filepath;
    }
    if (!isFile) {
        // entao eh campo - trata o conteudo do campo
        i = in.readLine(line, 0, 255);
        // linha em branco
        i = in.readLine(line, 0, 255);
        nameValue=new String(line,0,i);
        campos.put(name,nameValue);
    }
    if (isFile) {
        // trata o conteudo do arquivo
        i = in.readLine(line, 0, 255);
        i = in.readLine(line, 0, 255);
        // linha em branco
        i = in.readLine(line, 0, 255);

        ByteArrayOutputStream buffer = new ByteArrayOutputStream();
        newLine = new String(line, 0, i);

        while (i != -1 && !newLine.startsWith(boundary)) {
            buffer.write(line, 0, i);
            i = in.readLine(line, 0, 255);
            newLine = new String(line, 0, i);
        }
        try {
            // salva o arquivo
            RandomAccessFile f;
            f = new RandomAccessFile(xmlPath + filename, "rw");
            byte[] bytes = buffer.toByteArray();
            f.write(bytes, 0, bytes.length - 2);
            f.close();
        } catch (Exception e) {}
    }
    }
    i = in.readLine(line, 0, 255);

} // end while

String nomeArea=campos.getProperty("nomeArea");
String xmlOK=null;
xmlOK=bdBean.createAreaLevantamentoBD(
    nomeArea,
    new URL("file:/"+xmlPath+filename)
);
if (xmlOK==null){
%>
<B><FONT FACE="Verdana" SIZE="2">
    A área de levantamento<BR>
    -----<BR>
    <%=nomeArea%> <BR>
    -----<BR>
    foi incluida. <BR>
</FONT></B>
<%
    } else {
%>
<B><FONT FACE="Verdana" SIZE="2">
    Area Levantamento com erro.<BR>
    -----<BR>
    <%=xmlOK%> <BR>
    -----<BR>
    Área não incluida. <BR>
</FONT></B>

```

```

<%
}
File o=new File (xmlPath+filename);
if (o.exists())
    o.delete();
%>

```

### 3.7 A PÁGINA Menu.jsp

```

<%@page contentType="text/html"%>
<%
String baseURL = (String) application.getAttribute("baseURL");
%>
<TABLE CELLSPACING="0" CELLPADDING="0" WIDTH="20%" BORDER="0">
  <td id="Menu" width="20%" height="100%" BGCOLOR="F6F6F6"
    valign="top" align="left" >
    <P><BR><FONT FACE="Verdana" SIZE="2">
      <B><FONT COLOR="#003366">Op&cedil;&otilde;es
    </FONT></B><FONT COLOR="#003366">:</FONT></FONT></P>
    <UL>
      <LI>
        <FONT FACE="Verdana" SIZE="1">
          <A HREF="<%=baseURL%>?acao=cadastrarAL">
            Cadastrar Area de Levantamento
          </A>
        </FONT>
      </LI>
      <LI>
        <FONT FACE="Verdana" SIZE="1">
          <A HREF="<%=baseURL%>?acao=selecionarJV">
            Selecionar Janela de Visualiza&cedil;&atilde;o
          </A>
        </FONT>
      </LI>
    </UL>
  </td>
</TABLE>

```

### 3.8 A PÁGINA SelecionarJV.jsp

```

<%@page contentType="text/html"%>
<jsp:useBean id="bdBean" scope="application"
  class="com.vft3d.vftserv.persist.BdFTBean"/>
<%
String baseURL = (String) application.getAttribute("baseURL");
%>
<html>
<head>
<title>VFT3D</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body bgcolor="#FFFFFF" text="#000000" leftmargin="0" topmargin="0">
<table width="100%" height="100%" id="vftapp" align="left" valign="top"
  vspace="0" hspace="0" border="1" cellspacing="0" cellpadding="0"
  bgcolor="#FFFCC" name="VFT" >
<tr id="VFTApplet" bgcolor="#CCCCCC" >
  <td id="Header" height="23" valign="top" align="left" COLSPAN="2">
    <jsp:include page="Header.jsp" flush="true"/>
  </td>

```



```

</tr>
<tr>
  <td id="Menu" width="20%" height="100%" BGCOLOR="F6F6F6"
    valign="top" align="left" >
    <jsp:include page="Menu.jsp" flush="true"/>
  </td>
  <td id="Conteudo" width="80%" height="100%" bgcolor="#F6F6F6"
    valign="middle" align="CENTER">
    <jsp:include page="SelJVForm.jsp" flush="true"/>
  </td>
</tr>
</table>
</body>
</html>

```

### 3.9 A PÁGINA SelJVForm.jsp

```

<%@page contentType="text/html"%>
<%@ page import="com.vft3d.vftserv.model.coord.*" %>
<%@ page import="java.sql.*" %>
<%@ page import="java.util.*" %>
<jsp:useBean id="bdBean" scope="application"
  class="com.vft3d.vftserv.persist.BdFTBean"/>
<%
  String baseURL = (String) application.getAttribute("baseURL");
%>

<TABLE WIDTH="100%">
<TR>
<TD VALIGN="MIDDLE" ALIGN="CENTER">
  <FORM >
    <INPUT TYPE="HIDDEN" NAME="acao" VALUE="areasLevantamento">
    <TABLE WIDTH="300" BORDER="1" BORDERCOLOR="#CCCCCC">
      <TR BGCOLOR="#CCCCCC">
        <TD COLSPAN="5" VALIGN="MIDDLE" HEIGHT="30" ALIGN="CENTER">
          <I><B><FONT FACE="Verdana" SIZE="3" COLOR="#003366">
            Selecionar Janela de Visualiza&ccedil;&atilde;o
          </FONT></B></I>
        </TD>
      </TR>
      <TR>
        <TD VALIGN="MIDDLE" WIDTH="92" ALIGN="CENTER">&nbsp;</TD>
        <TD WIDTH="60">
          <DIV ALIGN="CENTER">
            <B><FONT FACE="Verdana" SIZE="2">Latitude</FONT></B>
          </DIV>
        </TD>
        <TD WIDTH="30">
          <DIV ALIGN="CENTER">
            <B><FONT FACE="Verdana" SIZE="2">N/S</FONT></B>
          </DIV>
        </TD>
        <TD WIDTH="60">
          <DIV ALIGN="CENTER">
            <B><FONT FACE="Verdana" SIZE="2">Longitude</FONT></B>
          </DIV>
        </TD>
        <TD WIDTH="36">
          <DIV ALIGN="CENTER">
            <B><FONT FACE="Verdana" SIZE="2">E/O</FONT></B>
          </DIV>
        </TD>
      </TR>
    </TABLE>
  </FORM>

```

```

<TR>
  <TD VALIGN="MIDDLE" WIDTH="92" ALIGN="CENTER">
    <B><FONT FACE="Verdana" SIZE="2">
      Coordenada m&iacute;nima
    </FONT></B>
  </TD>
  <TD WIDTH="60">
    <INPUT TYPE="TEXT" NAME="latMin" SIZE="10">
  </TD>
  <TD WIDTH="30">
    <SELECT NAME="dirLatMin">
      <OPTION>Sul</OPTION>
      <OPTION>Norte</OPTION>
    </SELECT>
  </TD>
  <TD WIDTH="60">
    <INPUT TYPE="TEXT" NAME="longMin" SIZE="10">
  </TD>
  <TD WIDTH="36">
    <SELECT NAME="dirLongMin">
      <OPTION>Oeste</OPTION>
      <OPTION>Este</OPTION>
    </SELECT>
  </TD>
</TR>
<TR>
  <TD VALIGN="MIDDLE" WIDTH="92" ALIGN="CENTER">
    <B><FONT FACE="Verdana" SIZE="2">
      Coordenada m&aacute;xima
    </FONT></B>
  </TD>
  <TD WIDTH="60"><INPUT TYPE="TEXT" NAME="latMax" SIZE="10">
  </TD>
  <TD WIDTH="30">
    <SELECT NAME="dirLatMax">
      <OPTION>Sul</OPTION>
      <OPTION>Norte</OPTION>
    </SELECT>
  </TD>
  <TD WIDTH="60">
    <INPUT TYPE="TEXT" NAME="longMax" SIZE="10">
  </TD>
  <TD WIDTH="36">
    <SELECT NAME="dirLongMax">
      <OPTION>Oeste</OPTION>
      <OPTION>Este</OPTION>
    </SELECT>
  </TD>
</TR>
<TR>
  <TD VALIGN="MIDDLE" COLSPAN="5" ALIGN="CENTER">
    <INPUT TYPE="SUBMIT" VALUE="Enviar" NAME="SUBMIT">
  </TD>
</TR>
<TR BGCOLOR="#CCCCCC">
  <TD VALIGN="MIDDLE" COLSPAN="5" ALIGN="CENTER">&nbsp; </TD>
</TR>
</TABLE>
</FORM>
</TD>
</TR>
</TABLE>

```

### 3.10 A PÁGINA AreasLevantamento.jsp

```

<%@page contentType="text/html"%>
<jsp:useBean id="bdBean" scope="application"
             class="com.vft3d.vftserv.persist.BdFTBean"/>
<%
String baseURL = (String) application.getAttribute("baseURL");
%>
<html>
<head>
<title>VFT3D</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body bgcolor="#FFFFFF" text="#000000" leftmargin="0" topmargin="0">
<table width="100%" height="100%" id="vftapp" align="left"
        valign="top" vspace="0" hspace="0" border="1"
        cellspacing="0" cellpadding="0" bgcolor="#FFFFCC" name="VFT" >
<tr id="VFTApplet" bgcolor="#CCCCCC" >
  <td id="Header" height="23" valign="top" align="left" COLSPAN="2">
    <jsp:include page="Header.jsp" flush="true"/>
  </td>
</tr>
<tr>
  <td id="Menu" width="20%" height="100%" BGCOLOR="F6F6F6"
        valign="top" align="left" >
    <jsp:include page="Menu.jsp" flush="true"/>
  </td>
  <td id="Conteudo" width="80%" height="100%" bgcolor="#F6F6F6"
        valign="middle" align="CENTER">
    <jsp:include page="AreasLev.jsp" flush="true"/>
  </td>
</tr>
</table>
</body>
</html>

```

### 3.11 A PÁGINA AreasLev.jsp

```

<%@page contentType="text/html"%>
<%@ page import="com.vft3d.vftserv.model.area.AreaLevantamento" %>
<%@ page import="com.vft3d.vftserv.persist.AreaLevantamentoBD" %>
<%@ page import="com.vft3d.vftserv.model.coord.*" %>
<%@ page import="java.text.DecimalFormat"%>

<%@ page import="java.sql.*" %>
<%@ page import="java.util.*" %>
<jsp:useBean id="bdBean" scope="application"
             class="com.vft3d.vftserv.persist.BdFTBean"/>
<%
String baseURL = (String) application.getAttribute("baseURL");
%>

<%
double minX=0;
double minY=0;
double maxX=0;
double maxY=0;
boolean coordValidas=true;

try {

```

```

// converte coordenadas GDC para UTM
// coordenadas minimas
// latitude minima
String latMin=request.getParameter("latMin");
String dirLatMin=request.getParameter("dirLatMin");
double laMin=Double.parseDouble(latMin);
if (dirLatMin.equals("Sul"))
    laMin*=-1;
// longitude minima
String longMin=request.getParameter("longMin");
String dirLongMin=request.getParameter("dirLongMin");
double loMin=Double.parseDouble(longMin);
if (dirLongMin.equals("Oeste"))
    loMin*=-1;

CoordGDC minGDC = new CoordGDC(laMin, loMin, 0.0);
CoordUTM minUTM = minGDC.toUTM();

// coordenadas maximas
String latMax=request.getParameter("latMax");
String dirLatMax=request.getParameter("dirLatMax");
double laMax=Double.parseDouble(latMax);
if (dirLatMax.equals("Sul"))
    laMax*=-1;
String longMax=request.getParameter("longMax");
String dirLongMax=request.getParameter("dirLongMax");
double loMax=Double.parseDouble(longMax);
if (dirLongMax.equals("Oeste"))
    loMax*=-1;

CoordGDC maxGDC = new CoordGDC(laMax, loMax, 0.0);
CoordUTM maxUTM = maxGDC.toUTM();

minX=minUTM.getX();
minY=minUTM.getY();
maxX=maxUTM.getX();
maxY=maxUTM.getY();

%>

LatMin=<%=laMin%>-<%=dirLatMin%>; minY= <%= Double.toString(minY)%><BR>
LongMin=<%=loMin%>-<%=dirLongMin%>; minX= <%= Double.toString(minX)%><BR>
LatMax=<%=laMax%>-<%=dirLatMax%>; maxY= <%= Double.toString(maxY)%><BR>
LongMax=<%=loMax%>-<%=dirLongMax%>; maxX= <%= Double.toString(maxX)%><BR>

<%
} catch(Exception e) {
    out.println("Coordenadas inválidas.");
}
%>

<TABLE WIDTH="98%" BORDER="1" BORDERCOLOR="#CCCCCC" ALIGN="LEFT">
<TR BGCOLOR="#CCCCCC">
    <TD COLSPAN="7" VALIGN="MIDDLE" HEIGHT="30" ALIGN="CENTER">
        <I><B><FONT FACE="Verdana" SIZE="3" COLOR="#003366">
            &Aacute;reas de Levantamento
        </FONT></B></I>
    </TD>
</TR>
<TR>
    <TD VALIGN="MIDDLE" ROWSPAN="2" ALIGN="CENTER" WIDTH="181">
        <FONT FACE="Verdana" SIZE="1"><B>Nome</B></FONT>

```

```

</TD>
<TD VALIGN="MIDDLE" ROWSPAN="2" ALIGN="CENTER" WIDTH="92">
  <B><FONT FACE="Verdana" SIZE="1">Data</FONT></B>
</TD>
<TD COLSPAN="2" ALIGN="CENTER">
  <B><FONT FACE="Verdana" SIZE="1">Coordenada<BR>m&iacute;nima</FONT></B>
</TD>
<TD COLSPAN="2" ALIGN="CENTER">
  <B><FONT FACE="Verdana" SIZE="1">Coordenada<BR>m&aacute;xima </FONT></B>
</TD>
<TD ROWSPAN="2" WIDTH="128"><FONT FACE="Verdana" SIZE="1"></FONT>
</TD>
</TR>
<TR>
  <TD WIDTH="51">
    <DIV ALIGN="CENTER"><B><FONT FACE="Verdana" SIZE="1">Lat</FONT></B></DIV>
  </TD>
  <TD WIDTH="37">
    <DIV ALIGN="CENTER"><B><FONT FACE="Verdana" SIZE="1">Long</FONT></B></DIV>
  </TD>
  <TD WIDTH="48">
    <DIV ALIGN="CENTER"><B><FONT FACE="Verdana" SIZE="1">Lat</FONT></B></DIV>
  </TD>
  <TD WIDTH="38">
    <DIV ALIGN="CENTER"><B><FONT FACE="Verdana" SIZE="1">Long</FONT></B></DIV>
  </TD>
</TR>
<%
  ArrayList areasLevantamento = bdBean.findAreasLevantamentoBD
      (
          Double.toString(minX),
          Double.toString(minY),
          Double.toString(maxX),
          Double.toString(maxY)
      );
  Iterator iterator = areasLevantamento.iterator();
%>
%>
<%
while (iterator.hasNext()) {
  AreaLevantamentoBD areaLevantamento;
  areaLevantamento = (AreaLevantamentoBD) iterator.next();
  double aMinX = areaLevantamento.getMinX();
  double aMinY = areaLevantamento.getMinY();
  double aMaxX = areaLevantamento.getMaxX();
  double aMaxY = areaLevantamento.getMaxY();

  CoordUTM cUTMmin = new CoordUTM(aMinX,aMinY,0,22,false);
  CoordGDC cGDCmin = cUTMmin.toGDC();
  CoordUTM cUTMmax = new CoordUTM(aMaxX,aMaxY,0,22,false);
  CoordGDC cGDCmax = cUTMmax.toGDC();
%>
<TR>
  <TD VALIGN="MIDDLE" WIDTH="200" ALIGN="LEFT" HEIGHT="21">
    <FONT FACE="Verdana" SIZE="1"><%=areaLevantamento.getNome()%></FONT>
  </TD>
  <TD VALIGN="MIDDLE" WIDTH="100" ALIGN="CENTER" HEIGHT="21">
    <FONT FACE="Verdana" SIZE="1"><%=areaLevantamento.getTempo()%></FONT>
  </TD>
  <TD WIDTH="65" ALIGN="CENTER" HEIGHT="21">
    <FONT FACE="Verdana" SIZE="1"><%=cGDCmin.getLat()%></FONT>
  </TD>
  <TD WIDTH="65" ALIGN="CENTER" HEIGHT="21">
    <FONT FACE="Verdana" SIZE="1"><%=cGDCmin.getLong()%></FONT>
  </TD>
  <TD WIDTH="65" ALIGN="CENTER" HEIGHT="21">
    <FONT FACE="Verdana" SIZE="1"><%=cGDCmax.getLat()%></FONT>
  </TD>

```



```

    try {
        cenaURL = wrlURL+"vftsplash.wrl";
    } catch (Exception e) {}
%>
<%
try {
    // busca a area solicitada
    AreaLevantamento area= bdBean.loadAreaLevantamento(areaId);
    cenaURL = wrlURL+areaId+".wrl";

    // cria arquivo wrl para conter a cena
    File vrmlFile = new File(wrlPath+areaId+".wrl");
    FileWriter vrmlCena = new FileWriter(vrmlFile);
    vrmlCena.write("#VRML V2.0 utf8\r\n");
%>
<%
    // inclui arquivo com protos padrões para cena
    FileReader from = null;
    try {
        // Cria input stream para o conteudo do arquivo
        from = new FileReader(wrlPath+"protos.wrl");
        char[] buffer = new char[4096];
        int bytesRead;
        // lê até EOF
        while((bytesRead = from.read(buffer,0,4096)) != -1) {
            vrmlCena.write(buffer, 0, bytesRead); // grava
        }
        from.close();
    }
    catch (IOException e) { ; }
%>
<%
    // inclui pontos para visualização da area
    double campoVisao=0.785; // 45 graus
    // obtem dados sobre as dimensões da area
    RetEnvolvente areaRet=area.getRetEnvolvente();
    double midX =
(area.getRetEnvolvente().getMaxX()+area.getRetEnvolvente().getMinX())/2;
    double midY =
(area.getRetEnvolvente().getMaxY()+area.getRetEnvolvente().getMinY())/2;
    double minY = area.getRetEnvolvente().getMinY();
    double minX = area.getRetEnvolvente().getMinX();
    double maxY = area.getRetEnvolvente().getMaxY();
    double maxX = area.getRetEnvolvente().getMaxX();
    double minZ = area.getRetEnvolvente().getMinZ();
    double maxZ = area.getRetEnvolvente().getMaxZ();
    double larg = area.getRetEnvolvente().getLargura();
    double alt = area.getRetEnvolvente().getAltura();
    double diag = area.getRetEnvolvente().getDiagonal();

    // calcula a altura do ponto de visualizaçã
    double h = (((larg>alt)?larg:alt)/Math.tan(campoVisao) );
    vrmlCena.write(
        "#minX="+minX+
        ",minY="+minY+
        "minZ="+minZ+
        "maxX="+maxX+
        "maxY="+maxY+
        "maxZ="+maxZ+
        "\r\n");
    vrmlCena.write("#larg="+larg+",alt="+alt+"h="+h+"\r\n");

    // inclui ponto de visualização perpendicular ao centro da area

```

```

        vrmlCena.write("GeoViewpoint {\r\n");
        vrmlCena.write(
"\tgeoSystem [ \"UTM\", \"Z22\", \"S\", \"WGS84\" \"easting_first\" ]\r\n"
);
        vrmlCena.write("\tposition ");
        vrmlCena.write("\t"+midX);
        vrmlCena.write("\t"+midY);
        vrmlCena.write("\t"+(minZ+h));
        vrmlCena.write("\t\r\n");
        vrmlCena.write("\torientation 1 0 0 -1.57\r\n");
        vrmlCena.write("\tdescription \"Visao Inicial\"\r\n");
        vrmlCena.write(")\r\n");

// inclui ponto de visualização lateral do sul para o norte da area
vrmlCena.write("GeoViewpoint {\r\n");
vrmlCena.write(
"\tgeoSystem [ \"UTM\", \"Z22\", \"S\", \"WGS84\" \"easting_first\" ]\r\n"
);
        vrmlCena.write("\tposition ");
        vrmlCena.write("\t"+midX);
        vrmlCena.write("\t"+(minY-alt));
        vrmlCena.write("\t"+(minZ+h));
        vrmlCena.write("\t\r\n");
        vrmlCena.write("\torientation 1 0 0 -0.85\r\n");
        vrmlCena.write("\tdescription \"Visao de Sul para Norte\"\r\n");
        vrmlCena.write(")\r\n");

//inclui ponto de visualização lateral do oeste para o leste da area
vrmlCena.write("GeoViewpoint {\r\n");
vrmlCena.write(
"\tgeoSystem [ \"UTM\", \"Z22\", \"S\", \"WGS84\" \"easting_first\" ]\r\n"
);
        vrmlCena.write("\tposition ");
        vrmlCena.write("\t"+(minX-(larg/2)));
        vrmlCena.write("\t"+(midY));
        h =alt/Math.tan(campoVisao);
        vrmlCena.write("\t"+(minZ+h));
        vrmlCena.write("\t\r\n");
        vrmlCena.write("\torientation -0.25 -1 -0.25 1.57\r\n");
        vrmlCena.write("\tdescription \"Visao de Oeste para Leste\"\r\n");
        vrmlCena.write(")\r\n");

%>
<%
// inclui vrml da area
vrmlCena.write(area.getVRML());
vrmlCena.close();
%>

<jsp:include page="VisualizadorVRML.jsp" flush="true">
    <jsp:param name="cenaURL" value="<%=cenaURL%"/>
</jsp:include>
<%
}
catch (Exception e) {
    out.print("Erro: "+ e.toString());
    if (DEBUG) {
        e.printStackTrace();
    }
}
%>
</td>
</tr>
</table>
</body>
</html>

```



### 3.13 A PÁGINA VisualizadorVRML.jsp

```
<%@ page contentType="text/html"%>
<%@ page import="java.util.*" %>
<%
    String cenaURL= request.getParameter("cenaURL");
%>
<OBJECT
    CLASSID="CLSID:86A88967-7A20-11d2-8EDA-00600818EDB1"
    CODEBASE="http://www.parallelgraphics.com/bin/cortvrml.cab#Version=4,1,0,90"
    WIDTH="100%"
    HEIGHT="100%">
    <PARAM NAME="SRC" VALUE="<%=cenaURL%>"
    <PARAM NAME="MASK" VALUE="0,0,790,0,790,582,0,582">
    <PARAM NAME="VRML_BACKGROUND_COLOR" VALUE="#CDCDCD">
    <PARAM NAME="VRML_DASHBOARD" VALUE="true">
    <PARAM NAME="VRML_SPLASHSCREEN" VALUE="false">
    <PARAM NAME="CONTEXTMENU" VALUE="true">
    <PARAM NAME="MODE" VALUE="CAD">
</OBJECT>
```

## 4 ARQUIVOS DE MODELOS DE FENÔMENOS EM VRML

### 4.1 Arvore.wrl

```

#VRML V2.0 utf8

#== BEGIN ARVORE =====
Group {
  children [
    Transform { #(2)
      translation 0 0.85 0
      children [
        Shape {
          appearance Appearance {
            material Material {
              emissiveColor 0.8 0.6 0.3
              diffuseColor 0.8 0.6 0.3
            }
          }
          geometry Cylinder {
            radius 0.15
            height 1.70
          }
        }#Shape      #caule da arvore

        Transform { #(3)
          translation 0 3.10 0
          children [
            Shape {
              appearance Appearance {
                material Material {
                  texture ImageTexture {
                    url "folhas.jpg"
                    repeats FALSE
                  }
                  repeatT TRUE
                }
                textureTransform TextureTransform { scale 1 -1 }
              }
              geometry Cone {
                bottomRadius 1.90
                height 4.50
              }
            }#Shape      #copa da arvore
          ]
        }#/transform (3)
      ]#/children
    }#/Transform (2)
  ]
}
#== END ARVORE =====

```

## 4.2 Poste.wrl

```

#VRML V2.0 utf8

#==BEGIN POSTE =====
  Group {
    children [
      Shape {
        appearance Appearance {
          material Material {
            emissiveColor 0.3 0.3 0.3
            diffuseColor 0.4 0.4 0.4
          }
        }
        geometry Cylinder {
          radius 0.12
          height 0.10
        }
      }#Shape    #Base da luminaria

      Transform { #(2)
        translation 0 1.5 0
        children [
          Shape {
            appearance Appearance {
              material Material {
                emissiveColor 0.4 0.4 0.4
                diffuseColor 0.4 0.4 0.4
              }
            }
            geometry Cylinder {
              radius 0.08
              height 3.00
            }
          }#Shape    #Haste da luminaria

          Transform { #(3)
            translation 0 1.75 0
            children [
              Shape {
                appearance Appearance {
                  material Material {
                    emissiveColor 0.9 0.9 0.9
                    #diffuseColor 0.4 0.4 0.4
                  }
                }
                geometry Cylinder {
                  radius 0.20
                  height 0.50
                }
              }#Shape    #Difusor da luminaria

              Transform { #(3)
                translation 0 0.30 0
                children [
                  Shape {
                    appearance Appearance {
                      material Material {
                        emissiveColor 0 0 0.9
                        diffuseColor 0.4 0.4 0.4
                      }
                    }
                    geometry Cone {

```

```
height 0.10                                bottomRadius 0.35
}
}#/Shape
]#/children
}#/transform (3)                            Tampa da luminaria
]#/children
}#/transform (3)
]#/children
}#/Transform (2)
]
}
#==END POSTE =====
```

## 5 ARQUIVO DE DISTRIBUIÇÃO DA APLICAÇÃO WEB (WEB.XML)

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE web-app
PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">

<web-app>
  <!-- Define servlets that are included in the example application -->
  <servlet>
    <servlet-name>control</servlet-name>
    <!--Qual a classe servlet responsável pelo controle da aplicação-->
    <servlet-class>com.vft3d.vftserv.control.VFTApp1</servlet-class>
    <init-param>
      <param-name>baseURL</param-name>
      <!--Qual o URL para a classe servlet de controle da aplicação-->
      <param-value>http://localhost:8080/vftserv/control</param-value>
    </init-param>
    <init-param>
      <param-name>jdbcDriver</param-name>
      <!--Qual a classe driver para conexão JDBC com o banco de dados-->
      <param-value>sun.jdbc.odbc.JdbcOdbcDriver</param-value>
    </init-param>
    <init-param>
      <!--Qual o URL para imagens-->
      <param-name>imageUrl</param-name>
      <param-value>http://localhost:8080/vftserv/images/</param-value>
    </init-param>
    <init-param>
      <param-name>bdURL</param-name>
      <param-value>jdbc:odbc:BDVFT</param-value>
    </init-param>
    <init-param>
      <param-name>bdLinguagem</param-name>
      <!--Qual o locale usado pelo SGBD-->
      <param-value>pt</param-value>
    </init-param>
    <init-param>
      <param-name>dtdURL</param-name>
      <!--Qual o URL para os XML-Document Type Definition -->
      <param-value>http://localhost:8080/vftserv/dtd/</param-value>
    </init-param>
    <init-param>
      <param-name>wrlURL</param-name>
      <param-value>http://localhost:8080/vftserv/wrl/</param-value>
    </init-param>
    <init-param>
      <!--Qual o código de usuário para acessar o banco de dados de fenomenos-->
      <param-name>bdUsuario</param-name>
      <param-value/>
    </init-param>
    <init-param>
      <!--Qual a password para acessar o banco de dados de fenomenos-->
      <param-name>bdPassword</param-name>
      <param-value/>
    </init-param>
    <init-param>
      <!--Onde armazenar os documentos XML recebidos-->
      <param-name>xmlPath</param-name>
      <param-value>c:/vft3d/vftserv/xml/</param-value>
    </init-param>
    <init-param>
      <!--Onde armazenar as visualizações wrl criadas -->

```

```
<param-name>wrlPath</param-name>
<param-value>c:/vft3d/vftserv/webapp/wrl/</param-value>
</init-param>
<init-param>
  <!--Onde estão armazenadas as páginas jsp da aplicação -->
  <param-name>jspDir</param-name>
  <param-value>/jsp</param-value>
</init-param>
</servlet>
<servlet-mapping>
  <servlet-name>control</servlet-name>
  <url-pattern>/control</url-pattern>
</servlet-mapping>
</web-app>
```

## 6 ARQUIVO XML DE ÁREA DE LEVANTAMENTO EXEMPLO

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE AreaLevantamento SYSTEM
"http://vft3d:8081/vftserv/dtd/AreaLevantamento.dtd">
<AreaLevantamento id="AL01" nome="Centro Politecnico" tempo="29/08/03">
  <FenomenoTerrestre categoria="SuperficieTopografica" tipo="Superficie"
dimensao="3">
  <Cor r="0" g="1" b="0"/>
  <Superficie3D>
    <Ponto x="677630.200" y="7183821.720" z="913.160" />
    <Ponto x="677654.520" y="7183820.390" z="913.200" />
    <Ponto x="677607.870" y="7183770.010" z="914.350" />
    <Ponto x="677646.410" y="7183716.550" z="914.350" />
    <Ponto x="677644.710" y="7183755.220" z="914.510" />
    <Ponto x="677680.800" y="7183737.640" z="914.630" />
    <Ponto x="677734.110" y="7183715.770" z="914.630" />
    <Ponto x="677750.250" y="7183752.880" z="913.120" />
    <Ponto x="677766.190" y="7183799.320" z="912.170" />
    <Ponto x="677785.550" y="7183741.400" z="917.130" />
    <Ponto x="677803.360" y="7183779.360" z="917.250" />
    <Ponto x="677811.060" y="7183762.850" z="918.160" />
    <Ponto x="677746.270" y="7183713.760" z="915.000" />
    <Ponto x="677746.070" y="7183715.250" z="915.000" />
    <Ponto x="677746.210" y="7183718.180" z="915.000" />
    <Ponto x="677748.820" y="7183722.660" z="915.000" />
    <Ponto x="677752.410" y="7183727.230" z="915.000" />
    <Ponto x="677754.890" y="7183732.100" z="915.000" />
    <Ponto x="677757.950" y="7183736.840" z="915.000" />
    <Ponto x="677760.510" y="7183741.380" z="915.000" />
    <Ponto x="677763.240" y="7183745.890" z="915.000" />
    <Ponto x="677765.520" y="7183750.380" z="915.000" />
    <Ponto x="677767.880" y="7183755.270" z="915.000" />
    <Ponto x="677770.310" y="7183760.250" z="915.000" />
    <Ponto x="677772.760" y="7183764.980" z="915.000" />
    <Ponto x="677774.830" y="7183770.180" z="915.000" />
    <Ponto x="677777.010" y="7183775.220" z="915.000" />
    <Ponto x="677779.280" y="7183779.970" z="915.000" />
    <Ponto x="677781.840" y="7183784.680" z="915.000" />
    <Ponto x="677784.490" y="7183789.770" z="915.000" />
    <Ponto x="677788.220" y="7183793.290" z="915.000" />
    <Ponto x="677791.720" y="7183796.930" z="915.000" />
    <Ponto x="677794.180" y="7183801.700" z="915.000" />
    <Ponto x="677796.500" y="7183806.400" z="915.000" />
    <Ponto x="677799.090" y="7183810.830" z="915.000" />
    <Ponto x="677801.760" y="7183815.480" z="915.000" />
    <Ponto x="677805.170" y="7183819.930" z="915.000" />
    <Ponto x="677809.190" y="7183824.190" z="915.000" />
    <Ponto x="677812.580" y="7183828.660" z="915.000" />
    <Ponto x="677816.220" y="7183832.090" z="915.000" />
    <Ponto x="677627.520" y="7183713.300" z="910.000" />
    <Ponto x="677627.460" y="7183715.090" z="910.000" />
    <Ponto x="677626.740" y="7183720.650" z="910.000" />
    <Ponto x="677625.490" y="7183726.630" z="910.000" />
    <Ponto x="677623.030" y="7183732.030" z="910.000" />
    <Ponto x="677619.770" y="7183736.580" z="910.000" />
    <Ponto x="677616.680" y="7183740.830" z="910.000" />
    <Ponto x="677612.670" y="7183744.900" z="910.000" />
    <Ponto x="677608.330" y="7183749.020" z="910.000" />
    <Ponto x="677603.270" y="7183751.910" z="910.000" />
    <Ponto x="677600.640" y="7183753.120" z="910.000" />
    <Ponto x="677712.454" y="7183777.395" z="914.000" />
    <Ponto x="677717.146" y="7183775.263" z="914.000" />
  
```

```
<Ponto x="677720.772" y="7183773.343" z="914.000" />
<Ponto x="677722.691" y="7183770.357" z="914.000" />
<Ponto x="677724.611" y="7183767.585" z="914.000" />
<Ponto x="677722.478" y="7183762.253" z="914.000" />
<Ponto x="677718.639" y="7183758.627" z="914.000" />
<Ponto x="677716.506" y="7183756.281" z="914.000" />
<Ponto x="677715.013" y="7183750.096" z="914.000" />
<Ponto x="677722.905" y="7183746.044" z="914.000" />
<Ponto x="677730.530" y="7183743.100" z="914.000" />
<Ponto x="677736.790" y="7183742.020" z="914.000" />
<Ponto x="677741.200" y="7183738.350" z="914.000" />
<Ponto x="677740.630" y="7183732.750" z="914.000" />
<Ponto x="677741.900" y="7183732.130" z="914.000" />
<Ponto x="677746.670" y="7183730.210" z="914.000" />
<Ponto x="677747.460" y="7183731.280" z="914.000" />
<Ponto x="677749.893" y="7183736.859" z="914.000" />
<Ponto x="677751.837" y="7183737.657" z="914.000" />
<Ponto x="677754.070" y="7183738.730" z="914.000" />
<Ponto x="677756.090" y="7183743.880" z="914.000" />
<Ponto x="677758.440" y="7183748.840" z="914.000" />
<Ponto x="677760.242" y="7183752.428" z="914.000" />
<Ponto x="677761.763" y="7183756.395" z="914.000" />
<Ponto x="677765.620" y="7183762.740" z="914.000" />
<Ponto x="677768.490" y="7183767.720" z="914.000" />
<Ponto x="677771.090" y="7183772.370" z="914.000" />
<Ponto x="677773.540" y="7183777.370" z="914.000" />
<Ponto x="677776.750" y="7183782.420" z="914.000" />
<Ponto x="677780.140" y="7183787.870" z="914.000" />
<Ponto x="677782.970" y="7183793.040" z="914.000" />
<Ponto x="677785.830" y="7183797.770" z="914.000" />
<Ponto x="677788.860" y="7183802.240" z="914.000" />
<Ponto x="677791.620" y="7183806.700" z="914.000" />
<Ponto x="677795.150" y="7183811.860" z="914.000" />
<Ponto x="677798.450" y="7183816.700" z="914.000" />
<Ponto x="677801.600" y="7183821.110" z="914.000" />
<Ponto x="677804.530" y="7183825.890" z="914.000" />
<Ponto x="677807.460" y="7183830.380" z="914.000" />
<Ponto x="677808.410" y="7183832.100" z="914.000" />
<Ponto x="677634.180" y="7183713.330" z="914.000" />
<Ponto x="677634.620" y="7183715.100" z="914.000" />
<Ponto x="677637.280" y="7183720.340" z="914.000" />
<Ponto x="677639.110" y="7183725.060" z="914.000" />
<Ponto x="677636.060" y="7183729.610" z="914.000" />
<Ponto x="677632.210" y="7183734.130" z="914.000" />
<Ponto x="677628.640" y="7183738.830" z="914.000" />
<Ponto x="677625.390" y="7183742.980" z="914.000" />
<Ponto x="677621.050" y="7183746.260" z="914.000" />
<Ponto x="677617.240" y="7183750.450" z="914.000" />
<Ponto x="677612.920" y="7183755.040" z="914.000" />
<Ponto x="677607.690" y="7183758.280" z="914.000" />
<Ponto x="677602.700" y="7183760.680" z="914.000" />
<Ponto x="677600.720" y="7183761.530" z="914.000" />
<Ponto x="677759.850" y="7183713.810" z="916.000" />
<Ponto x="677760.240" y="7183715.930" z="916.000" />
<Ponto x="677762.860" y="7183721.530" z="916.000" />
<Ponto x="677760.470" y="7183725.910" z="916.000" />
<Ponto x="677762.250" y="7183731.140" z="916.000" />
<Ponto x="677764.390" y="7183736.700" z="916.000" />
<Ponto x="677766.990" y="7183742.860" z="916.000" />
<Ponto x="677769.570" y="7183747.930" z="916.000" />
<Ponto x="677771.850" y="7183752.580" z="916.000" />
<Ponto x="677773.650" y="7183757.790" z="916.000" />
<Ponto x="677775.510" y="7183762.490" z="916.000" />
<Ponto x="677778.100" y="7183767.340" z="916.000" />
<Ponto x="677780.810" y="7183771.800" z="916.000" />
<Ponto x="677783.360" y="7183776.110" z="916.000" />
```



```
<Ponto x="677786.170" y="7183780.840" z="916.000" />
<Ponto x="677788.980" y="7183785.070" z="916.000" />
<Ponto x="677791.450" y="7183789.420" z="916.000" />
<Ponto x="677795.030" y="7183794.750" z="916.000" />
<Ponto x="677798.450" y="7183800.340" z="916.000" />
<Ponto x="677800.960" y="7183805.940" z="916.000" />
<Ponto x="677803.690" y="7183810.840" z="916.000" />
<Ponto x="677807.390" y="7183814.700" z="916.000" />
<Ponto x="677810.920" y="7183818.960" z="916.000" />
<Ponto x="677816.360" y="7183820.230" z="916.000" />
<Ponto x="677790.230" y="7183713.930" z="917.000" />
<Ponto x="677787.830" y="7183716.130" z="917.000" />
<Ponto x="677785.380" y="7183720.740" z="917.000" />
<Ponto x="677784.620" y="7183726.320" z="917.000" />
<Ponto x="677783.340" y="7183731.360" z="917.000" />
<Ponto x="677779.210" y="7183735.120" z="917.000" />
<Ponto x="677773.665" y="7183740.498" z="917.000" />
<Ponto x="677773.025" y="7183744.764" z="917.000" />
<Ponto x="677776.330" y="7183748.950" z="917.000" />
<Ponto x="677778.190" y="7183754.540" z="917.000" />
<Ponto x="677780.310" y="7183759.960" z="917.000" />
<Ponto x="677782.260" y="7183764.970" z="917.000" />
<Ponto x="677784.190" y="7183769.980" z="917.000" />
<Ponto x="677786.180" y="7183774.820" z="917.000" />
<Ponto x="677788.830" y="7183780.360" z="917.000" />
<Ponto x="677791.480" y="7183784.980" z="917.000" />
<Ponto x="677794.580" y="7183789.150" z="917.000" />
<Ponto x="677798.000" y="7183793.340" z="917.000" />
<Ponto x="677801.010" y="7183798.190" z="917.000" />
<Ponto x="677804.720" y="7183801.760" z="917.000" />
<Ponto x="677809.280" y="7183805.710" z="917.000" />
<Ponto x="677814.020" y="7183804.410" z="917.000" />
<Ponto x="677816.520" y="7183806.540" z="917.000" />
<Ponto x="677816.940" y="7183714.030" z="918.000" />
<Ponto x="677814.300" y="7183717.190" z="918.000" />
<Ponto x="677810.860" y="7183721.050" z="918.000" />
<Ponto x="677810.350" y="7183726.860" z="918.000" />
<Ponto x="677810.360" y="7183732.310" z="918.000" />
<Ponto x="677809.150" y="7183737.070" z="918.000" />
<Ponto x="677807.840" y="7183737.530" z="918.000" />
<Ponto x="677802.660" y="7183739.550" z="918.000" />
<Ponto x="677800.400" y="7183744.590" z="918.000" />
<Ponto x="677803.020" y="7183749.040" z="918.000" />
<Ponto x="677800.070" y="7183753.370" z="918.000" />
<Ponto x="677800.120" y="7183757.590" z="918.000" />
<Ponto x="677801.240" y="7183763.330" z="918.000" />
<Ponto x="677802.760" y="7183768.110" z="918.000" />
<Ponto x="677805.750" y="7183772.630" z="918.000" />
<Ponto x="677806.280" y="7183773.710" z="918.000" />
<Ponto x="677810.080" y="7183777.730" z="918.000" />
<Ponto x="677811.130" y="7183781.590" z="918.000" />
<Ponto x="677812.730" y="7183786.770" z="918.000" />
<Ponto x="677816.700" y="7183791.450" z="918.000" />
<Ponto x="677817.390" y="7183731.450" z="919.000" />
<Ponto x="677817.000" y="7183731.950" z="919.000" />
<Ponto x="677815.020" y="7183737.470" z="919.000" />
<Ponto x="677814.660" y="7183742.490" z="919.000" />
<Ponto x="677814.390" y="7183747.560" z="919.000" />
<Ponto x="677816.720" y="7183752.090" z="919.000" />
<Ponto x="677817.140" y="7183753.620" z="919.000" />
<Ponto x="677703.427" y="7183825.856" z="913.000" />
<Ponto x="677706.220" y="7183824.220" z="913.000" />
<Ponto x="677711.800" y="7183822.170" z="913.000" />
<Ponto x="677716.950" y="7183819.570" z="913.000" />
<Ponto x="677721.660" y="7183817.130" z="913.000" />
<Ponto x="677726.910" y="7183813.940" z="913.000" />
```

```
<Ponto x="677727.890" y="7183808.670" z="913.000" />
<Ponto x="677730.280" y="7183803.780" z="913.000" />
<Ponto x="677736.768" y="7183800.216" z="913.000" />
<Ponto x="677742.313" y="7183796.164" z="913.000" />
<Ponto x="677744.019" y="7183792.965" z="913.000" />
<Ponto x="677746.152" y="7183789.766" z="913.000" />
<Ponto x="677748.430" y="7183785.270" z="913.000" />
<Ponto x="677750.910" y="7183780.340" z="913.000" />
<Ponto x="677755.060" y="7183776.320" z="913.000" />
<Ponto x="677759.830" y="7183774.020" z="913.000" />
<Ponto x="677764.900" y="7183772.720" z="913.000" />
<Ponto x="677766.510" y="7183774.270" z="913.000" />
<Ponto x="677770.010" y="7183778.260" z="913.000" />
<Ponto x="677772.740" y="7183783.280" z="913.000" />
<Ponto x="677775.430" y="7183788.020" z="913.000" />
<Ponto x="677779.290" y="7183792.720" z="913.000" />
<Ponto x="677782.440" y="7183797.700" z="913.000" />
<Ponto x="677784.790" y="7183802.260" z="913.000" />
<Ponto x="677787.940" y="7183807.160" z="913.000" />
<Ponto x="677791.240" y="7183812.980" z="913.000" />
<Ponto x="677793.940" y="7183818.000" z="913.000" />
<Ponto x="677796.800" y="7183822.280" z="913.000" />
<Ponto x="677799.310" y="7183826.880" z="913.000" />
<Ponto x="677802.580" y="7183832.110" z="913.000" />
<Ponto x="677632.670" y="7183713.320" z="913.000" />
<Ponto x="677633.910" y="7183717.640" z="913.000" />
<Ponto x="677636.200" y="7183722.880" z="913.000" />
<Ponto x="677634.050" y="7183728.020" z="913.000" />
<Ponto x="677630.710" y="7183732.330" z="913.000" />
<Ponto x="677627.750" y="7183736.800" z="913.000" />
<Ponto x="677624.100" y="7183741.630" z="913.000" />
<Ponto x="677619.660" y="7183745.840" z="913.000" />
<Ponto x="677615.850" y="7183749.780" z="913.000" />
<Ponto x="677611.500" y="7183753.750" z="913.000" />
<Ponto x="677606.910" y="7183756.220" z="913.000" />
<Ponto x="677601.610" y="7183759.050" z="913.000" />
<Ponto x="677600.700" y="7183759.430" z="913.000" />
<Ponto x="677754.050" y="7183832.190" z="912.000" />
<Ponto x="677754.770" y="7183831.170" z="912.000" />
<Ponto x="677758.040" y="7183827.020" z="912.000" />
<Ponto x="677762.140" y="7183822.930" z="912.000" />
<Ponto x="677766.970" y="7183820.870" z="912.000" />
<Ponto x="677771.940" y="7183819.290" z="912.000" />
<Ponto x="677773.238" y="7183821.117" z="912.000" />
<Ponto x="677775.584" y="7183820.691" z="912.000" />
<Ponto x="677776.864" y="7183816.639" z="912.000" />
<Ponto x="677776.480" y="7183813.820" z="912.000" />
<Ponto x="677781.850" y="7183812.160" z="912.000" />
<Ponto x="677783.380" y="7183815.760" z="912.000" />
<Ponto x="677787.300" y="7183820.910" z="912.000" />
<Ponto x="677790.540" y="7183824.840" z="912.000" />
<Ponto x="677793.750" y="7183829.650" z="912.000" />
<Ponto x="677795.140" y="7183832.120" z="912.000" />
<Ponto x="677630.830" y="7183713.310" z="912.000" />
<Ponto x="677632.670" y="7183718.940" z="912.000" />
<Ponto x="677633.250" y="7183724.450" z="912.000" />
<Ponto x="677629.390" y="7183728.800" z="912.000" />
<Ponto x="677626.670" y="7183734.570" z="912.000" />
<Ponto x="677623.070" y="7183739.340" z="912.000" />
<Ponto x="677619.980" y="7183743.700" z="912.000" />
<Ponto x="677616.560" y="7183747.570" z="912.000" />
<Ponto x="677612.700" y="7183750.980" z="912.000" />
<Ponto x="677608.510" y="7183753.970" z="912.000" />
<Ponto x="677603.810" y="7183756.710" z="912.000" />
<Ponto x="677600.690" y="7183758.200" z="912.000" />
<Ponto x="677629.120" y="7183713.740" z="911.000" />
```

```
<Ponto x="677629.400" y="7183719.910" z="911.000" />
<Ponto x="677628.590" y="7183725.890" z="911.000" />
<Ponto x="677626.540" y="7183730.630" z="911.000" />
<Ponto x="677623.460" y="7183735.560" z="911.000" />
<Ponto x="677620.440" y="7183740.570" z="911.000" />
<Ponto x="677617.230" y="7183745.370" z="911.000" />
<Ponto x="677612.580" y="7183749.180" z="911.000" />
<Ponto x="677607.750" y="7183752.310" z="911.000" />
<Ponto x="677602.210" y="7183754.570" z="911.000" />
<Ponto x="677600.660" y="7183755.300" z="911.000" />
<Ponto x="677623.580" y="7183713.290" z="909.000" />
<Ponto x="677623.690" y="7183715.140" z="909.000" />
<Ponto x="677623.000" y="7183720.560" z="909.000" />
<Ponto x="677620.320" y="7183725.160" z="909.000" />
<Ponto x="677616.270" y="7183728.240" z="909.000" />
<Ponto x="677611.690" y="7183732.420" z="909.000" />
<Ponto x="677608.040" y="7183736.720" z="909.000" />
<Ponto x="677604.150" y="7183741.230" z="909.000" />
<Ponto x="677600.560" y="7183744.820" z="909.000" />
<Ponto x="677619.530" y="7183713.270" z="908.000" />
<Ponto x="677618.250" y="7183718.220" z="908.000" />
<Ponto x="677615.280" y="7183723.130" z="908.000" />
<Ponto x="677611.970" y="7183727.070" z="908.000" />
<Ponto x="677608.660" y="7183731.170" z="908.000" />
<Ponto x="677603.840" y="7183735.600" z="908.000" />
<Ponto x="677600.510" y="7183738.980" z="908.000" />
<Ponto x="677611.540" y="7183713.240" z="907.000" />
<Ponto x="677609.570" y="7183718.220" z="907.000" />
<Ponto x="677606.840" y="7183722.470" z="907.000" />
<Ponto x="677602.560" y="7183727.130" z="907.000" />
<Ponto x="677600.420" y="7183729.220" z="907.000" />
<Ponto x="677604.740" y="7183713.210" z="906.000" />
<Ponto x="677602.580" y="7183716.780" z="906.000" />
<Ponto x="677600.320" y="7183719.490" z="906.000" />
<Ponto x="677615.839" y="7183831.781" z="914.000" />
<Ponto x="677615.839" y="7183829.862" z="914.000" />
<Ponto x="677613.920" y="7183825.596" z="914.000" />
<Ponto x="677611.574" y="7183820.478" z="914.000" />
<Ponto x="677610.081" y="7183815.572" z="914.000" />
<Ponto x="677607.308" y="7183808.534" z="914.000" />
<Ponto x="677608.190" y="7183806.030" z="914.000" />
<Ponto x="677612.970" y="7183803.890" z="914.000" />
<Ponto x="677618.100" y="7183802.240" z="914.000" />
<Ponto x="677623.990" y="7183800.890" z="914.000" />
<Ponto x="677628.960" y="7183799.370" z="914.000" />
<Ponto x="677634.080" y="7183797.490" z="914.000" />
<Ponto x="677639.080" y="7183796.210" z="914.000" />
<Ponto x="677642.926" y="7183796.590" z="914.000" />
<Ponto x="677647.890" y="7183796.700" z="914.000" />
<Ponto x="677652.840" y="7183795.110" z="914.000" />
<Ponto x="677657.740" y="7183793.430" z="914.000" />
<Ponto x="677662.690" y="7183792.080" z="914.000" />
<Ponto x="677667.810" y="7183790.240" z="914.000" />
<Ponto x="677672.530" y="7183787.940" z="914.000" />
<Ponto x="677677.720" y="7183785.170" z="914.000" />
<Ponto x="677682.500" y="7183782.570" z="914.000" />
<Ponto x="677687.260" y="7183780.060" z="914.000" />
<Ponto x="677690.700" y="7183778.888" z="914.000" />
<Ponto x="677686.560" y="7183832.290" z="913.000" />
<Ponto x="677688.490" y="7183831.470" z="913.000" />
<Ponto x="677694.470" y="7183829.280" z="913.000" />
<Ponto x="677698.863" y="7183827.600" z="913.000" />
<Face prefs="94 3 5" />
<Face prefs="300 298 4" />
<Face prefs="5 3 6" />
<Face prefs="3 6 91" />
```

```
<Face prefs="106 15 108" />
<Face prefs="12 6 13" />
<Face prefs="13 6 14" />
<Face prefs="13 14 106" />
<Face prefs="108 17 109" />
<Face prefs="14 65 15" />
<Face prefs="109 18 110" />
<Face prefs="71 72 20" />
<Face prefs="112 113 137" />
<Face prefs="116 24 25" />
<Face prefs="77 78 26" />
<Face prefs="118 119 27" />
<Face prefs="144 120 121" />
<Face prefs="0 1 285" />
<Face prefs="256 216 244" />
<Face prefs="43 264 44" />
<Face prefs="46 255 45" />
<Face prefs="41 40 262" />
<Face prefs="262 263 42" />
<Face prefs="264 43 263" />
<Face prefs="265 44 45" />
<Face prefs="267 46 47" />
<Face prefs="182 51 185" />
<Face prefs="54 53 52" />
<Face prefs="186 52 187" />
<Face prefs="51 52 186" />
<Face prefs="187 52 53" />
<Face prefs="57 56 54" />
<Face prefs="189 188 187" />
<Face prefs="56 7 55" />
<Face prefs="60 61 56" />
<Face prefs="55 54 56" />
<Face prefs="52 57 54" />
<Face prefs="52 51 57" />
<Face prefs="51 58 57" />
<Face prefs="51 308 58" />
<Face prefs="60 58 59" />
<Face prefs="56 57 60" />
<Face prefs="57 58 60" />
<Face prefs="59 6 60" />
<Face prefs="7 56 61" />
<Face prefs="60 6 61" />
<Face prefs="7 61 62" />
<Face prefs="64 63 62" />
<Face prefs="7 62 63" />
<Face prefs="61 64 62" />
<Face prefs="65 63 64" />
<Face prefs="7 63 68" />
<Face prefs="61 6 64" />
<Face prefs="6 64 65" />
<Face prefs="15 65 66" />
<Face prefs="6 65 14" />
<Face prefs="65 67 66" />
<Face prefs="16 66 67" />
<Face prefs="65 68 67" />
<Face prefs="17 67 68" />
<Face prefs="16 67 17" />
<Face prefs="65 63 68" />
<Face prefs="68 7 69" />
<Face prefs="70 17 69" />
<Face prefs="17 68 69" />
<Face prefs="69 71 70" />
<Face prefs="19 18 70" />
<Face prefs="18 17 70" />
<Face prefs="69 7 71" />
<Face prefs="20 19 71" />
```

```
<Face prefs="19 70 71" />
<Face prefs="21 20 73" />
<Face prefs="71 7 72" />
<Face prefs="72 7 73" />
<Face prefs="21 73 74" />
<Face prefs="194 75 74" />
<Face prefs="22 74 75" />
<Face prefs="21 74 22" />
<Face prefs="196 77 195" />
<Face prefs="24 75 76" />
<Face prefs="23 75 24" />
<Face prefs="78 77 197" />
<Face prefs="24 76 25" />
<Face prefs="193 197 198" />
<Face prefs="78 27 79" />
<Face prefs="26 27 78" />
<Face prefs="199 198 79" />
<Face prefs="79 28 80" />
<Face prefs="28 29 80" />
<Face prefs="29 30 81" />
<Face prefs="201 200 81" />
<Face prefs="231 202 201" />
<Face prefs="83 31 82" />
<Face prefs="201 202 82" />
<Face prefs="31 32 83" />
<Face prefs="203 202 83" />
<Face prefs="83 32 84" />
<Face prefs="32 33 84" />
<Face prefs="34 86 85" />
<Face prefs="35 87 86" />
<Face prefs="37 36 88" />
<Face prefs="37 89 88" />
<Face prefs="207 237 208" />
<Face prefs="39 38 90" />
<Face prefs="6 12 91" />
<Face prefs="92 91 209" />
<Face prefs="238 210 209" />
<Face prefs="93 3 92" />
<Face prefs="240 211 239" />
<Face prefs="252 240 251" />
<Face prefs="94 3 93" />
<Face prefs="2 4 101" />
<Face prefs="4 94 95" />
<Face prefs="96 4 95" />
<Face prefs="95 213 96" />
<Face prefs="4 96 97" />
<Face prefs="4 97 98" />
<Face prefs="4 98 99" />
<Face prefs="4 99 100" />
<Face prefs="44 265 264" />
<Face prefs="246 217 100" />
<Face prefs="46 256 47" />
<Face prefs="247 219 102" />
<Face prefs="103 2 102" />
<Face prefs="2 104 103" />
<Face prefs="248 220 103" />
<Face prefs="294 2 293" />
<Face prefs="250 40 238" />
<Face prefs="12 105 13" />
<Face prefs="131 107 106" />
<Face prefs="109 107 134" />
<Face prefs="105 106 13" />
<Face prefs="106 108 107" />
<Face prefs="14 15 106" />
<Face prefs="135 9 134" />
<Face prefs="15 16 108" />
```

```
<Face prefs="66 16 15" />
<Face prefs="110 134 109" />
<Face prefs="16 17 108" />
<Face prefs="17 18 109" />
<Face prefs="135 134 110" />
<Face prefs="18 19 110" />
<Face prefs="19 20 111" />
<Face prefs="136 135 111" />
<Face prefs="20 21 112" />
<Face prefs="73 20 72" />
<Face prefs="137 136 112" />
<Face prefs="21 22 113" />
<Face prefs="22 114 113" />
<Face prefs="22 23 114" />
<Face prefs="23 115 114" />
<Face prefs="23 24 115" />
<Face prefs="24 116 115" />
<Face prefs="140 139 115" />
<Face prefs="26 25 117" />
<Face prefs="77 25 26" />
<Face prefs="116 141 117" />
<Face prefs="27 26 118" />
<Face prefs="26 117 118" />
<Face prefs="120 119 143" />
<Face prefs="28 27 119" />
<Face prefs="199 80 200" />
<Face prefs="143 144 120" />
<Face prefs="144 145 121" />
<Face prefs="122 30 121" />
<Face prefs="30 29 121" />
<Face prefs="145 146 122" />
<Face prefs="10 146 145" />
<Face prefs="32 31 122" />
<Face prefs="32 33 123" />
<Face prefs="147 122 123" />
<Face prefs="123 148 124" />
<Face prefs="125 34 124" />
<Face prefs="35 34 125" />
<Face prefs="35 126 125" />
<Face prefs="149 124 125" />
<Face prefs="170 146 147" />
<Face prefs="36 127 126" />
<Face prefs="148 170 171" />
<Face prefs="38 37 128" />
<Face prefs="38 39 128" />
<Face prefs="105 129 130" />
<Face prefs="129 131 130" />
<Face prefs="105 130 131" />
<Face prefs="105 131 106" />
<Face prefs="9 133 159" />
<Face prefs="134 132 133" />
<Face prefs="131 132 107" />
<Face prefs="160 9 159" />
<Face prefs="133 9 134" />
<Face prefs="108 107 109" />
<Face prefs="107 132 134" />
<Face prefs="110 111 135" />
<Face prefs="19 111 110" />
<Face prefs="137 9 135" />
<Face prefs="111 112 136" />
<Face prefs="20 112 111" />
<Face prefs="137 135 136" />
<Face prefs="138 9 137" />
<Face prefs="21 113 112" />
<Face prefs="113 138 137" />
<Face prefs="163 162 139" />
```

```
<Face prefs="113 114 138" />
<Face prefs="114 139 138" />
<Face prefs="140 116 115" />
<Face prefs="25 116 117" />
<Face prefs="115 114 139" />
<Face prefs="164 141 140" />
<Face prefs="139 163 140" />
<Face prefs="118 117 141" />
<Face prefs="116 140 141" />
<Face prefs="165 142 141" />
<Face prefs="165 143 142" />
<Face prefs="119 118 142" />
<Face prefs="118 141 142" />
<Face prefs="165 10 143" />
<Face prefs="28 119 120" />
<Face prefs="119 142 143" />
<Face prefs="121 29 120" />
<Face prefs="80 81 200" />
<Face prefs="143 10 144" />
<Face prefs="10 144 145" />
<Face prefs="121 122 145" />
<Face prefs="31 122 30" />
<Face prefs="122 147 146" />
<Face prefs="32 123 122" />
<Face prefs="169 10 170" />
<Face prefs="123 148 147" />
<Face prefs="33 124 123" />
<Face prefs="34 124 33" />
<Face prefs="10 169 168" />
<Face prefs="124 149 148" />
<Face prefs="146 170 10" />
<Face prefs="147 148 170" />
<Face prefs="151 150 149" />
<Face prefs="127 151 126" />
<Face prefs="171 148 150" />
<Face prefs="127 128 151" />
<Face prefs="12 105 238" />
<Face prefs="105 129 152" />
<Face prefs="172 152 153" />
<Face prefs="152 129 153" />
<Face prefs="173 155 156" />
<Face prefs="153 129 154" />
<Face prefs="153 154 172" />
<Face prefs="154 129 155" />
<Face prefs="154 155 172" />
<Face prefs="155 159 156" />
<Face prefs="174 156 157" />
<Face prefs="156 158 157" />
<Face prefs="174 157 175" />
<Face prefs="156 159 158" />
<Face prefs="161 158 159" />
<Face prefs="157 158 175" />
<Face prefs="155 132 159" />
<Face prefs="132 159 133" />
<Face prefs="131 155 132" />
<Face prefs="129 155 131" />
<Face prefs="162 160 161" />
<Face prefs="160 159 161" />
<Face prefs="138 9 162" />
<Face prefs="9 160 162" />
<Face prefs="161 163 162" />
<Face prefs="138 139 162" />
<Face prefs="11 161 163" />
<Face prefs="11 163 164" />
<Face prefs="163 140 164" />
<Face prefs="11 164 165" />
```

```
<Face prefs="164 141 165" />
<Face prefs="167 11 166" />
<Face prefs="11 165 166" />
<Face prefs="165 10 166" />
<Face prefs="10 166 167" />
<Face prefs="11 168 167" />
<Face prefs="168 10 167" />
<Face prefs="36 126 35" />
<Face prefs="126 149 125" />
<Face prefs="178 11 168" />
<Face prefs="150 149 148" />
<Face prefs="37 127 128" />
<Face prefs="37 36 127" />
<Face prefs="168 169 178" />
<Face prefs="151 149 126" />
<Face prefs="161 11 177" />
<Face prefs="39 128 171" />
<Face prefs="128 151 171" />
<Face prefs="177 178 11" />
<Face prefs="174 172 173" />
<Face prefs="177 172 175" />
<Face prefs="172 155 173" />
<Face prefs="161 175 158" />
<Face prefs="177 175 176" />
<Face prefs="161 176 175" />
<Face prefs="171 178 169" />
<Face prefs="174 175 172" />
<Face prefs="161 177 176" />
<Face prefs="151 171 150" />
<Face prefs="156 174 173" />
<Face prefs="170 171 169" />
<Face prefs="177 172 178" />
<Face prefs="181 222 179" />
<Face prefs="224 223 184" />
<Face prefs="306 180 179" />
<Face prefs="312 306 179" />
<Face prefs="223 224 228" />
<Face prefs="222 182 181" />
<Face prefs="227 231 8" />
<Face prefs="181 51 182" />
<Face prefs="227 226 8" />
<Face prefs="232 204 233" />
<Face prefs="232 231 202" />
<Face prefs="308 181 51" />
<Face prefs="232 203 202" />
<Face prefs="84 203 204" />
<Face prefs="187 184 185" />
<Face prefs="231 8 201" />
<Face prefs="187 53 189" />
<Face prefs="185 51 186" />
<Face prefs="185 186 187" />
<Face prefs="189 53 190" />
<Face prefs="225 187 188" />
<Face prefs="190 53 191" />
<Face prefs="8 188 189" />
<Face prefs="191 53 54" />
<Face prefs="8 189 190" />
<Face prefs="55 191 54" />
<Face prefs="8 190 191" />
<Face prefs="55 192 191" />
<Face prefs="8 191 192" />
<Face prefs="55 7 192" />
<Face prefs="198 192 193" />
<Face prefs="8 192 198" />
<Face prefs="192 7 193" />
<Face prefs="7 193 194" />
```



```
<Face prefs="74 7 194" />
<Face prefs="73 7 74" />
<Face prefs="194 196 195" />
<Face prefs="76 75 195" />
<Face prefs="75 194 195" />
<Face prefs="194 197 196" />
<Face prefs="76 195 77" />
<Face prefs="22 75 23" />
<Face prefs="194 193 197" />
<Face prefs="76 77 25" />
<Face prefs="77 196 197" />
<Face prefs="199 8 198" />
<Face prefs="197 78 198" />
<Face prefs="78 79 198" />
<Face prefs="79 80 199" />
<Face prefs="27 28 79" />
<Face prefs="200 8 199" />
<Face prefs="120 29 28" />
<Face prefs="29 81 80" />
<Face prefs="201 8 200" />
<Face prefs="81 82 201" />
<Face prefs="30 82 81" />
<Face prefs="31 82 30" />
<Face prefs="82 83 202" />
<Face prefs="188 8 225" />
<Face prefs="187 224 184" />
<Face prefs="33 34 85" />
<Face prefs="83 84 203" />
<Face prefs="84 33 85" />
<Face prefs="85 205 204" />
<Face prefs="35 36 87" />
<Face prefs="85 86 205" />
<Face prefs="36 87 88" />
<Face prefs="236 235 229" />
<Face prefs="37 38 89" />
<Face prefs="228 229 236" />
<Face prefs="87 88 207" />
<Face prefs="39 90 208" />
<Face prefs="236 237 207" />
<Face prefs="12 238 209" />
<Face prefs="3 91 92" />
<Face prefs="91 12 209" />
<Face prefs="250 239 238" />
<Face prefs="251 239 250" />
<Face prefs="251 240 239" />
<Face prefs="94 93 211" />
<Face prefs="252 241 240" />
<Face prefs="95 94 212" />
<Face prefs="97 96 214" />
<Face prefs="95 212 213" />
<Face prefs="253 242 213" />
<Face prefs="98 97 215" />
<Face prefs="214 243 97" />
<Face prefs="99 98 215" />
<Face prefs="100 99 216" />
<Face prefs="43 44 253" />
<Face prefs="256 244 46" />
<Face prefs="102 101 218" />
<Face prefs="48 49 258" />
<Face prefs="248 102 219" />
<Face prefs="259 248 260" />
<Face prefs="103 221 220" />
<Face prefs="249 260 221" />
<Face prefs="208 39 222" />
<Face prefs="228 237 223" />
<Face prefs="222 184 223" />
```

```
<Face prefs="88 208 89" />
<Face prefs="89 208 90" />
<Face prefs="224 228 225" />
<Face prefs="87 206 207" />
<Face prefs="182 222 183" />
<Face prefs="227 228 229" />
<Face prefs="234 204 205" />
<Face prefs="230 229 227" />
<Face prefs="234 229 233" />
<Face prefs="205 234 235" />
<Face prefs="227 226 228" />
<Face prefs="226 225 228" />
<Face prefs="87 86 206" />
<Face prefs="230 231 227" />
<Face prefs="233 229 230" />
<Face prefs="35 86 34" />
<Face prefs="224 187 225" />
<Face prefs="232 230 231" />
<Face prefs="85 204 84" />
<Face prefs="225 8 226" />
<Face prefs="185 184 183" />
<Face prefs="203 204 232" />
<Face prefs="183 184 222" />
<Face prefs="312 311 305" />
<Face prefs="232 233 230" />
<Face prefs="204 234 233" />
<Face prefs="206 205 235" />
<Face prefs="86 205 206" />
<Face prefs="229 235 234" />
<Face prefs="308 181 180" />
<Face prefs="206 236 235" />
<Face prefs="180 181 179" />
<Face prefs="38 89 90" />
<Face prefs="207 206 236" />
<Face prefs="88 207 208" />
<Face prefs="208 222 237" />
<Face prefs="222 223 237" />
<Face prefs="228 236 237" />
<Face prefs="105 152 238" />
<Face prefs="92 210 93" />
<Face prefs="41 251 250" />
<Face prefs="94 211 212" />
<Face prefs="42 252 251" />
<Face prefs="252 43 241" />
<Face prefs="253 213 241" />
<Face prefs="213 212 241" />
<Face prefs="243 214 254" />
<Face prefs="97 243 215" />
<Face prefs="254 214 242" />
<Face prefs="99 215 244" />
<Face prefs="99 244 216" />
<Face prefs="44 254 242" />
<Face prefs="100 216 245" />
<Face prefs="100 245 217" />
<Face prefs="45 255 243" />
<Face prefs="246 217 257" />
<Face prefs="100 246 101" />
<Face prefs="46 244 255" />
<Face prefs="2 101 102" />
<Face prefs="257 48 246" />
<Face prefs="218 101 246" />
<Face prefs="246 258 48" />
<Face prefs="218 102 247" />
<Face prefs="246 218 258" />
<Face prefs="49 219 259" />
<Face prefs="102 103 248" />
```

```
<Face prefs="104 221 103" />
<Face prefs="248 220 249" />
<Face prefs="220 221 249" />
<Face prefs="40 41 250" />
<Face prefs="210 238 239" />
<Face prefs="209 210 92" />
<Face prefs="93 239 211" />
<Face prefs="210 239 93" />
<Face prefs="41 42 251" />
<Face prefs="212 240 241" />
<Face prefs="211 240 212" />
<Face prefs="42 43 252" />
<Face prefs="241 43 253" />
<Face prefs="214 213 242" />
<Face prefs="96 213 214" />
<Face prefs="253 44 242" />
<Face prefs="215 243 255" />
<Face prefs="215 255 244" />
<Face prefs="216 256 245" />
<Face prefs="4 100 101" />
<Face prefs="254 45 243" />
<Face prefs="217 245 257" />
<Face prefs="44 45 254" />
<Face prefs="42 263 43" />
<Face prefs="256 47 245" />
<Face prefs="47 48 257" />
<Face prefs="245 47 257" />
<Face prefs="247 218 258" />
<Face prefs="49 219 258" />
<Face prefs="219 247 258" />
<Face prefs="50 49 259" />
<Face prefs="219 248 259" />
<Face prefs="260 50 259" />
<Face prefs="248 249 260" />
<Face prefs="221 104 260" />
<Face prefs="272 264 265" />
<Face prefs="261 40 262" />
<Face prefs="265 45 266" />
<Face prefs="262 41 42" />
<Face prefs="266 46 267" />
<Face prefs="271 263 264" />
<Face prefs="266 45 46" />
<Face prefs="267 47 268" />
<Face prefs="261 262 270" />
<Face prefs="262 263 271" />
<Face prefs="50 49 269" />
<Face prefs="265 273 272" />
<Face prefs="48 268 47" />
<Face prefs="266 273 265" />
<Face prefs="49 48 269" />
<Face prefs="50 260 269" />
<Face prefs="274 273 266" />
<Face prefs="268 48 269" />
<Face prefs="238 40 270" />
<Face prefs="261 40 270" />
<Face prefs="270 262 271" />
<Face prefs="274 266 267" />
<Face prefs="270 271 277" />
<Face prefs="271 264 272" />
<Face prefs="278 272 279" />
<Face prefs="271 278 277" />
<Face prefs="275 274 267" />
<Face prefs="269 268 276" />
<Face prefs="273 279 272" />
<Face prefs="267 268 275" />
<Face prefs="275 281 276" />
```

```
<Face prefs="268 275 276" />
<Face prefs="270 282 277" />
<Face prefs="274 279 273" />
<Face prefs="282 277 278" />
<Face prefs="271 272 278" />
<Face prefs="275 274 281" />
<Face prefs="278 283 282" />
<Face prefs="279 274 280" />
<Face prefs="276 284 281" />
<Face prefs="279 283 278" />
<Face prefs="274 280 281" />
<Face prefs="238 270 282" />
<Face prefs="284 283 279" />
<Face prefs="281 280 284" />
<Face prefs="269 276 284" />
<Face prefs="282 283 284" />
<Face prefs="279 280 284" />
<Face prefs="0 289 288" />
<Face prefs="285 0 286" />
<Face prefs="104 2 291" />
<Face prefs="285 287 288" />
<Face prefs="293 289 0" />
<Face prefs="292 293 289" />
<Face prefs="288 289 290" />
<Face prefs="289 292 290" />
<Face prefs="104 290 291" />
<Face prefs="290 292 291" />
<Face prefs="285 286 287" />
<Face prefs="286 0 287" />
<Face prefs="2 292 293" />
<Face prefs="291 292 2" />
<Face prefs="287 0 288" />
<Face prefs="0 295 294" />
<Face prefs="293 0 294" />
<Face prefs="296 2 295" />
<Face prefs="2 294 295" />
<Face prefs="0 296 295" />
<Face prefs="1 0 298" />
<Face prefs="2 297 296" />
<Face prefs="0 296 298" />
<Face prefs="2 4 297" />
<Face prefs="296 297 298" />
<Face prefs="297 4 298" />
<Face prefs="300 1 299" />
<Face prefs="1 298 299" />
<Face prefs="298 300 299" />
<Face prefs="94 4 5" />
<Face prefs="304 4 303" />
<Face prefs="4 300 301" />
<Face prefs="300 1 301" />
<Face prefs="303 4 302" />
<Face prefs="4 301 302" />
<Face prefs="301 1 302" />
<Face prefs="1 302 303" />
<Face prefs="305 304 4" />
<Face prefs="309 303 310" />
<Face prefs="306 305 4" />
<Face prefs="303 304 310" />
<Face prefs="5 306 4" />
<Face prefs="304 305 311" />
<Face prefs="304 311 310" />
<Face prefs="5 307 306" />
<Face prefs="5 308 307" />
<Face prefs="6 59 5" />
<Face prefs="305 306 312" />
<Face prefs="185 183 182" />
```

```

<Face prefs="5 59 308" />
<Face prefs="59 58 308" />
<Face prefs="1 285 309" />
<Face prefs="1 309 303" />
<Face prefs="309 311 310" />
<Face prefs="308 307 180" />
<Face prefs="309 222 311" />
<Face prefs="222 179 312" />
<Face prefs="307 306 180" />
<Face prefs="311 222 312" />
</Superficie3D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Edificio"
dimensao="3">
  <Cor r="1" g="0" b="0"/>
  <Superficie3D>
    <Ponto x="677718.49" y="7183732.13" z="914.280000" />
    <Ponto x="677723.05" y="7183742.78" z="914.280000" />
    <Ponto x="677622.27" y="7183785.96" z="914.280000" />
    <Ponto x="677617.71" y="7183775.31" z="914.280000" />
    <Ponto x="677718.49" y="7183732.13" z="924.280000" />
    <Ponto x="677723.05" y="7183742.78" z="924.280000" />
    <Ponto x="677622.27" y="7183785.96" z="924.280000" />
    <Ponto x="677617.71" y="7183775.31" z="924.280000" />
    <Ponto x="677707.80" y="7183756.45" z="913.640000" />
    <Ponto x="677711.24" y="7183764.43" z="913.640000" />
    <Ponto x="677646.32" y="7183792.43" z="913.640000" />
    <Ponto x="677642.88" y="7183784.45" z="913.640000" />
    <Ponto x="677707.80" y="7183756.45" z="923.640000" />
    <Ponto x="677711.24" y="7183764.43" z="923.640000" />
    <Ponto x="677646.32" y="7183792.43" z="923.640000" />
    <Ponto x="677642.88" y="7183784.45" z="923.640000" />
    <Ponto x="677652.76" y="7183816.69" z="913.050000" />
    <Ponto x="677648.09" y="7183805.78" z="913.050000" />
    <Ponto x="677748.49" y="7183762.77" z="913.050000" />
    <Ponto x="677753.16" y="7183773.68" z="913.050000" />
    <Ponto x="677652.76" y="7183816.69" z="923.050000" />
    <Ponto x="677648.09" y="7183805.78" z="923.050000" />
    <Ponto x="677748.49" y="7183762.77" z="923.050000" />
    <Ponto x="677753.16" y="7183773.68" z="923.050000" />
    <Ponto x="677676.43" y="7183823.02" z="913.090000" />
    <Ponto x="677673.03" y="7183814.99" z="913.090000" />
    <Ponto x="677737.83" y="7183787.56" z="913.090000" />
    <Ponto x="677741.23" y="7183795.60" z="913.090000" />
    <Ponto x="677676.43" y="7183823.02" z="923.090000" />
    <Ponto x="677673.03" y="7183814.99" z="923.090000" />
    <Ponto x="677737.83" y="7183787.56" z="923.090000" />
    <Ponto x="677741.23" y="7183795.60" z="923.090000" />
    <Ponto x="677699.63" y="7183827.50" z="916.030000" />
    <Ponto x="677694.44" y="7183815.43" z="916.030000" />
    <Ponto x="677698.11" y="7183813.85" z="916.030000" />
    <Ponto x="677703.30" y="7183825.92" z="916.030000" />
    <Ponto x="677699.63" y="7183827.50" z="926.030000" />
    <Ponto x="677694.44" y="7183815.43" z="926.030000" />
    <Ponto x="677698.11" y="7183813.85" z="926.030000" />
    <Ponto x="677703.30" y="7183825.92" z="926.030000" />
    <Ponto x="677690.92" y="7183807.42" z="916.310000" />
    <Ponto x="677688.37" y="7183801.40" z="916.310000" />
    <Ponto x="677692.06" y="7183799.85" z="916.310000" />
    <Ponto x="677694.60" y="7183805.86" z="916.310000" />
    <Ponto x="677690.92" y="7183807.42" z="926.310000" />
    <Ponto x="677688.37" y="7183801.40" z="926.310000" />
    <Ponto x="677692.06" y="7183799.85" z="926.310000" />
    <Ponto x="677694.60" y="7183805.86" z="926.310000" />
    <Ponto x="677716.16" y="7183796.73" z="916.230000" />
    <Ponto x="677713.57" y="7183790.61" z="916.230000" />
  
```

```
<Ponto x="677717.43" y="7183788.99" z="916.230000" />
<Ponto x="677720.01" y="7183795.10" z="916.230000" />
<Ponto x="677716.16" y="7183796.73" z="926.230000" />
<Ponto x="677713.57" y="7183790.61" z="926.230000" />
<Ponto x="677717.43" y="7183788.99" z="926.230000" />
<Ponto x="677720.01" y="7183795.10" z="926.230000" />
<Ponto x="677694.72" y="7183785.81" z="916.950000" />
<Ponto x="677689.56" y="7183773.77" z="916.950000" />
<Ponto x="677693.28" y="7183772.18" z="916.950000" />
<Ponto x="677698.44" y="7183784.21" z="916.950000" />
<Ponto x="677694.72" y="7183785.81" z="926.950000" />
<Ponto x="677689.56" y="7183773.77" z="926.950000" />
<Ponto x="677693.28" y="7183772.18" z="926.950000" />
<Ponto x="677698.44" y="7183784.21" z="926.950000" />
<Ponto x="677661.09" y="7183776.60" z="917.340000" />
<Ponto x="677658.44" y="7183770.46" z="917.340000" />
<Ponto x="677661.89" y="7183768.98" z="917.340000" />
<Ponto x="677664.54" y="7183775.11" z="917.340000" />
<Ponto x="677661.09" y="7183776.60" z="927.340000" />
<Ponto x="677658.44" y="7183770.46" z="927.340000" />
<Ponto x="677661.89" y="7183768.98" z="927.340000" />
<Ponto x="677664.54" y="7183775.11" z="927.340000" />
<Ponto x="677686.14" y="7183765.79" z="917.340000" />
<Ponto x="677683.52" y="7183759.71" z="917.340000" />
<Ponto x="677687.19" y="7183758.14" z="917.340000" />
<Ponto x="677689.81" y="7183764.21" z="917.340000" />
<Ponto x="677686.14" y="7183765.79" z="927.340000" />
<Ponto x="677683.52" y="7183759.71" z="927.340000" />
<Ponto x="677687.19" y="7183758.14" z="927.340000" />
<Ponto x="677689.81" y="7183764.21" z="927.340000" />
<Face prefs="0 1 2 3" />
<Face prefs="4 5 6 7" />
<Face prefs="0 1 5 4" />
<Face prefs="1 2 6 5" />
<Face prefs="2 3 7 6" />
<Face prefs="3 0 4 7" />
<Face prefs="8 9 10 11" />
<Face prefs="12 13 14 15" />
<Face prefs="8 9 13 12" />
<Face prefs="9 10 14 13" />
<Face prefs="10 11 15 14" />
<Face prefs="11 8 12 15" />
<Face prefs="16 17 18 19" />
<Face prefs="20 21 22 23" />
<Face prefs="16 17 21 20" />
<Face prefs="17 18 22 21" />
<Face prefs="18 19 23 22" />
<Face prefs="19 16 20 23" />
<Face prefs="24 25 26 27" />
<Face prefs="28 29 30 31" />
<Face prefs="24 25 29 28" />
<Face prefs="25 26 30 29" />
<Face prefs="26 27 31 30" />
<Face prefs="27 24 28 31" />
<Face prefs="32 33 34 35" />
<Face prefs="36 37 38 39" />
<Face prefs="32 33 37 36" />
<Face prefs="33 34 38 37" />
<Face prefs="34 35 39 38" />
<Face prefs="35 32 36 39" />
<Face prefs="40 41 42 43" />
<Face prefs="44 45 46 47" />
<Face prefs="40 41 45 44" />
<Face prefs="41 42 46 45" />
<Face prefs="42 43 47 46" />
<Face prefs="43 40 44 47" />
```

```

<Face prefs="48 49 50 51" />
<Face prefs="52 53 54 55" />
<Face prefs="48 49 53 52" />
<Face prefs="49 50 54 53" />
<Face prefs="50 51 55 54" />
<Face prefs="51 48 52 55" />
<Face prefs="56 57 58 59" />
<Face prefs="60 61 62 63" />
<Face prefs="56 57 61 60" />
<Face prefs="57 58 62 61" />
<Face prefs="58 59 63 62" />
<Face prefs="59 56 60 63" />
<Face prefs="64 65 66 67" />
<Face prefs="68 69 70 71" />
<Face prefs="64 65 69 68" />
<Face prefs="65 66 70 69" />
<Face prefs="66 67 71 70" />
<Face prefs="67 64 68 71" />
<Face prefs="72 73 74 75" />
<Face prefs="76 77 78 79" />
<Face prefs="72 73 77 76" />
<Face prefs="73 74 78 77" />
<Face prefs="74 75 79 78" />
<Face prefs="75 72 76 79" />
</Superficie3D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Via" dimensao="3">
  <Cor_r="0.8" g="0.8" b="0.8"/>
  <Superficie3D>
    <Ponto x="677772.11" y="7183832.16" z="912.820000" />
    <Ponto x="677790.81" y="7183832.13" z="909.440000" />
    <Ponto x="677787.72" y="7183825.05" z="912.300000" />
    <Ponto x="677743.76" y="7183722.32" z="915.330000" />
    <Ponto x="677739.84" y="7183713.22" z="916.120000" />
    <Ponto x="677727.27" y="7183713.68" z="916.120000" />
    <Ponto x="677727.16" y="7183714.01" z="916.120000" />
    <Ponto x="677725.78" y="7183715.07" z="916.160000" />
    <Ponto x="677680.58" y="7183734.33" z="916.160000" />
    <Ponto x="677656.01" y="7183744.99" z="915.880000" />
    <Ponto x="677600.78" y="7183768.43" z="915.880000" />
    <Ponto x="677600.88" y="7183778.66" z="916.240000" />
    <Ponto x="677603.40" y="7183777.59" z="916.240000" />
    <Ponto x="677616.62" y="7183812.45" z="916.120000" />
    <Ponto x="677622.14" y="7183812.90" z="916.080000" />
    <Ponto x="677629.69" y="7183832.38" z="916.080000" />
    <Ponto x="677637.79" y="7183832.37" z="914.450000" />
    <Ponto x="677623.47" y="7183798.22" z="915.920000" />
    <Ponto x="677623.13" y="7183796.92" z="915.920000" />
    <Ponto x="677623.59" y="7183795.70" z="915.920000" />
    <Ponto x="677624.86" y="7183794.56" z="915.920000" />
    <Ponto x="677627.14" y="7183793.83" z="915.920000" />
    <Ponto x="677628.70" y="7183794.09" z="915.920000" />
    <Ponto x="677630.24" y="7183795.62" z="915.920000" />
    <Ponto x="677630.21" y="7183798.17" z="915.920000" />
    <Ponto x="677629.23" y="7183800.04" z="915.920000" />
    <Ponto x="677641.67" y="7183832.36" z="915.920000" />
    <Ponto x="677654.70" y="7183832.34" z="915.130000" />
    <Ponto x="677641.21" y="7183799.44" z="915.130000" />
    <Ponto x="677639.40" y="7183795.68" z="915.130000" />
    <Ponto x="677637.40" y="7183793.18" z="915.130000" />
    <Ponto x="677633.78" y="7183790.59" z="915.330000" />
    <Ponto x="677628.39" y="7183788.46" z="915.370000" />
    <Ponto x="677625.45" y="7183788.37" z="915.410000" />
    <Ponto x="677622.06" y="7183788.79" z="915.450000" />
    <Ponto x="677620.41" y="7183789.01" z="915.450000" />
    <Ponto x="677619.30" y="7183788.82" z="915.450000" />
  </Superficie3D>
</FenomenoTerrestre>

```

```
<Ponto x="677617.70" y="7183787.46" z="915.450000" />
<Ponto x="677613.82" y="7183776.92" z="916.280000" />
<Ponto x="677613.64" y="7183774.40" z="916.520000" />
<Ponto x="677614.92" y="7183772.70" z="916.520000" />
<Ponto x="677730.53" y="7183723.70" z="915.720000" />
<Ponto x="677733.45" y="7183722.71" z="915.720000" />
<Ponto x="677735.58" y="7183723.68" z="915.720000" />
<Ponto x="677737.73" y="7183725.99" z="915.720000" />
<Ponto x="677740.89" y="7183733.78" z="918.190000" />
<Ponto x="677741.57" y="7183736.29" z="915.560000" />
<Ponto x="677741.12" y="7183737.74" z="915.560000" />
<Ponto x="677739.75" y="7183738.58" z="915.560000" />
<Ponto x="677733.76" y="7183740.84" z="915.560000" />
<Ponto x="677717.24" y="7183747.19" z="915.560000" />
<Ponto x="677714.43" y="7183749.02" z="915.560000" />
<Ponto x="677712.76" y="7183750.20" z="915.560000" />
<Ponto x="677711.70" y="7183751.91" z="915.560000" />
<Ponto x="677711.35" y="7183756.50" z="915.560000" />
<Ponto x="677712.25" y="7183759.73" z="915.560000" />
<Ponto x="677713.64" y="7183761.92" z="915.560000" />
<Ponto x="677716.32" y="7183763.97" z="915.560000" />
<Ponto x="677718.32" y="7183765.29" z="915.560000" />
<Ponto x="677718.57" y="7183766.75" z="915.560000" />
<Ponto x="677718.15" y="7183769.09" z="915.560000" />
<Ponto x="677715.76" y="7183769.75" z="915.560000" />
<Ponto x="677708.08" y="7183767.56" z="915.450000" />
<Ponto x="677706.12" y="7183769.87" z="915.290000" />
<Ponto x="677709.55" y="7183779.45" z="914.930000" />
<Ponto x="677745.46" y="7183764.07" z="914.890000" />
<Ponto x="677743.34" y="7183758.18" z="914.890000" />
<Ponto x="677744.29" y="7183756.47" z="914.890000" />
<Ponto x="677745.52" y="7183755.64" z="914.890000" />
<Ponto x="677747.75" y="7183755.51" z="914.890000" />
<Ponto x="677749.48" y="7183756.04" z="914.890000" />
<Ponto x="677750.99" y="7183757.24" z="914.890000" />
<Ponto x="677756.40" y="7183770.93" z="914.410000" />
<Ponto x="677756.30" y="7183773.43" z="914.410000" />
<Ponto x="677754.83" y="7183775.91" z="914.370000" />
<Ponto x="677752.10" y="7183779.60" z="914.130000" />
<Ponto x="677750.20" y="7183782.43" z="914.010000" />
<Ponto x="677748.17" y="7183788.21" z="913.850000" />
<Ponto x="677747.91" y="7183790.54" z="913.850000" />
<Ponto x="677749.89" y="7183794.70" z="913.810000" />
<Ponto x="677762.27" y="7183824.16" z="912.900000" />
<Ponto x="677764.07" y="7183827.50" z="912.860000" />
<Ponto x="677765.09" y="7183828.23" z="912.860000" />
<Ponto x="677767.18" y="7183829.89" z="912.860000" />
<Ponto x="677769.42" y="7183831.15" z="912.820000" />
<Ponto x="677761.32" y="7183782.40" z="914.250000" />
<Ponto x="677779.31" y="7183825.49" z="913.140000" />
<Ponto x="677779.79" y="7183828.04" z="913.140000" />
<Ponto x="677779.39" y="7183829.01" z="913.140000" />
<Ponto x="677777.93" y="7183829.64" z="913.180000" />
<Ponto x="677775.85" y="7183828.79" z="913.380000" />
<Ponto x="677774.15" y="7183827.63" z="913.420000" />
<Ponto x="677773.81" y="7183825.02" z="913.850000" />
<Ponto x="677774.47" y="7183823.46" z="913.930000" />
<Ponto x="677759.54" y="7183789.11" z="914.250000" />
<Ponto x="677757.95" y="7183788.12" z="914.250000" />
<Ponto x="677755.98" y="7183787.30" z="914.250000" />
<Ponto x="677754.75" y="7183786.18" z="914.250000" />
<Ponto x="677754.74" y="7183783.96" z="914.250000" />
<Ponto x="677755.91" y="7183781.44" z="914.250000" />
<Ponto x="677757.82" y="7183780.74" z="914.250000" />
<Ponto x="677759.75" y="7183780.92" z="914.250000" />
<Ponto x="677753.60" y="7183713.78" z="916.040000" />
```



```

<Ponto x="677739.84" y="7183713.22" z="915.330000" />
<Ponto x="677743.76" y="7183722.32" z="915.330000" />
<Ponto x="677743.63" y="7183720.14" z="915.680000" />
<Ponto x="677743.73" y="7183718.55" z="915.680000" />
<Ponto x="677745.00" y="7183716.55" z="915.680000" />
<Ponto x="677746.64" y="7183715.11" z="915.680000" />
<Ponto x="677748.14" y="7183715.07" z="915.680000" />
<Ponto x="677750.38" y="7183715.22" z="915.720000" />
<Ponto x="677773.24" y="7183735.63" z="916.640000" />
<Ponto x="677779.68" y="7183740.78" z="917.040000" />
<Ponto x="677784.81" y="7183746.34" z="917.080000" />
<Ponto x="677786.77" y="7183750.28" z="917.080000" />
<Ponto x="677810.16" y="7183806.82" z="917.120000" />
<Ponto x="677814.16" y="7183814.51" z="916.960000" />
<Ponto x="677816.39" y="7183817.47" z="916.400000" />
<Ponto x="677816.49" y="7183809.23" z="917.950000" />
<Ponto x="677807.40" y="7183783.31" z="917.950000" />
<Ponto x="677808.68" y="7183779.14" z="918.430000" />
<Ponto x="677813.30" y="7183775.12" z="918.710000" />
<Ponto x="677816.91" y="7183772.83" z="918.990000" />
<Ponto x="677817.10" y="7183757.11" z="918.910000" />
<Ponto x="677812.54" y="7183754.28" z="918.910000" />
<Ponto x="677798.28" y="7183745.52" z="917.750000" />
<Ponto x="677786.05" y="7183738.61" z="917.470000" />
<Ponto x="677777.69" y="7183733.67" z="917.200000" />
<Ponto x="677771.33" y="7183728.73" z="916.960000" />
<Ponto x="677761.78" y="7183720.64" z="916.880000" />
<Ponto x="677755.39" y="7183715.04" z="916.880000" />
<Ponto x="677788.32" y="7183743.49" z="917.120000" />
<Ponto x="677797.21" y="7183748.74" z="918.270000" />
<Ponto x="677804.43" y="7183754.68" z="918.310000" />
<Ponto x="677814.06" y="7183762.94" z="918.430000" />
<Ponto x="677815.95" y="7183765.89" z="918.430000" />
<Ponto x="677814.64" y="7183767.34" z="918.430000" />
<Ponto x="677812.20" y="7183768.40" z="918.190000" />
<Ponto x="677807.51" y="7183772.01" z="917.670000" />
<Ponto x="677804.44" y="7183774.53" z="917.670000" />
<Ponto x="677803.20" y="7183774.45" z="917.670000" />
<Ponto x="677802.00" y="7183771.25" z="917.510000" />
<Ponto x="677791.31" y="7183749.23" z="917.320000" />
<Ponto x="677677.53" y="7183735.65" z="906.890000" />
<Ponto x="677682.11" y="7183733.68" z="906.890000" />
<Ponto x="677676.20" y="7183727.61" z="906.890000" />
<Ponto x="677672.97" y="7183722.87" z="906.890000" />
<Ponto x="677672.21" y="7183713.20" z="906.890000" />
<Ponto x="677672.75" y="7183713.47" z="906.890000" />
<Ponto x="677666.66" y="7183713.45" z="906.690000" />
<Ponto x="677666.03" y="7183718.04" z="906.690000" />
<Ponto x="677669.67" y="7183724.89" z="906.690000" />
<Ponto x="677673.07" y="7183729.08" z="906.850000" />
<Ponto x="677745.08" y="7183744.29" z="915.450000" />
<Ponto x="677744.16" y="7183743.52" z="915.450000" />
<Ponto x="677742.87" y="7183743.60" z="915.600000" />
<Ponto x="677740.35" y="7183744.70" z="915.600000" />
<Ponto x="677739.38" y="7183747.90" z="915.600000" />
<Ponto x="677718.58" y="7183755.86" z="915.600000" />
<Ponto x="677718.74" y="7183757.15" z="915.600000" />
<Ponto x="677719.88" y="7183757.61" z="915.600000" />
<Ponto x="677740.73" y="7183749.71" z="915.600000" />
<Ponto x="677741.90" y="7183751.28" z="915.600000" />
<Ponto x="677744.32" y="7183751.09" z="915.450000" />
<Ponto x="677745.68" y="7183751.01" z="915.450000" />
<Ponto x="677746.81" y="7183749.50" z="915.450000" />
<Ponto x="677746.59" y="7183747.29" z="915.450000" />
<Ponto x="677745.38" y="7183745.29" z="915.450000" />
<Face prefs="0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

```

```

20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47
48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75
76 77 78 79 80 81 82 83 84" />
        <Face prefs="85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
101" />
        <Face prefs="102 103 104 105 106 107 108 109 110 111 112 113 114
115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130" />
        <Face prefs="131 132 133 134 135 136 137 138 139 140 141 142" />
        <Face prefs="143 144 145 146 147 148 149 150 151 152" />
        <Face prefs="153 154 155 156 157 158 159 160 161 162 163 164 165
166 167" />
    </Superficie3D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677682.16" y="7183716.27" z="904.620000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677808.91" y="7183742.89" z="917.040000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677815.68" y="7183783.26" z="918.110000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677816.42" y="7183796.41" z="917.640000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677810.37" y="7183813.13" z="916.560000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677804.02" y="7183800.30" z="916.480000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677801.30" y="7183789.52" z="916.360000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">

```

```

        <Ponto x="677793.11" y="7183773.66" z="915.250000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677775.51" y="7183744.14" z="914.290000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677747.57" y="7183716.75" z="914.290000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677748.28" y="7183725.16" z="914.290000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677753.77" y="7183738.69" z="914.090000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677756.27" y="7183745.84" z="914.090000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677762.58" y="7183760.42" z="914.050000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677772.55" y="7183783.72" z="913.500000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677778.59" y="7183800.41" z="912.700000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677785.54" y="7183815.21" z="912.620000" />
    </Superficie0D>
</FenomenoTerrestre>

```

```

<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677607.31" y="7183805.73" z="915.850000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677610.12" y="7183812.84" z="915.850000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677613.35" y="7183819.08" z="915.850000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677617.34" y="7183827.41" z="915.850000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677622.85" y="7183816.09" z="915.850000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677618.91" y="7183808.63" z="915.850000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677615.44" y="7183800.66" z="915.850000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677610.04" y="7183786.81" z="915.930000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677606.41" y="7183779.56" z="915.930000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>

```

```

        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677606.49" y="7183763.63" z="917.000000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677612.47" y="7183762.35" z="917.000000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677618.40" y="7183759.82" z="917.000000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677622.98" y="7183757.38" z="915.770000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677627.19" y="7183755.99" z="915.370000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677633.07" y="7183752.60" z="915.290000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677639.53" y="7183750.05" z="915.050000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677642.53" y="7183748.51" z="914.610000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677651.26" y="7183744.53" z="914.610000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677657.24" y="7183741.26" z="914.610000" />
        </Superficie0D>
    </FenomenoTerrestre>

```

```

</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677661.63" y="7183739.13" z="914.610000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677666.32" y="7183737.02" z="914.610000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677670.43" y="7183735.25" z="914.610000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677674.52" y="7183733.31" z="914.610000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677683.18" y="7183729.82" z="914.610000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677688.21" y="7183727.51" z="914.610000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677694.61" y="7183724.31" z="914.610000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677704.43" y="7183719.63" z="914.610000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677710.04" y="7183716.44" z="914.610000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">

```

```

        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677714.34" y="7183715.03" z="914.610000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677685.74" y="7183714.33" z="914.610000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677742.51" y="7183747.35" z="913.860000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677730.83" y="7183736.68" z="913.860000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677728.12" y="7183740.76" z="913.860000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677722.55" y="7183755.33" z="913.860000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677719.53" y="7183756.68" z="913.860000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677710.77" y="7183773.16" z="913.980000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677707.51" y="7183774.43" z="913.980000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677702.29" y="7183776.76" z="913.980000" />
        </Superficie0D>
    </FenomenoTerrestre>

```

```

        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677697.99" y="7183755.67" z="914.810000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677681.31" y="7183763.76" z="915.890000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677669.35" y="7183768.31" z="916.400000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677655.56" y="7183775.19" z="917.000000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677648.41" y="7183777.89" z="917.000000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677638.16" y="7183787.18" z="913.820000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677644.85" y="7183798.19" z="913.820000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677649.10" y="7183800.22" z="914.020000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677664.83" y="7183795.25" z="914.930000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"

```



```

dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677681.93" y="7183786.19" z="915.730000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677688.53" y="7183784.15" z="915.730000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677691.67" y="7183783.64" z="915.730000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677684.44" y="7183806.11" z="916.400000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677679.25" y="7183809.00" z="916.400000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677669.35" y="7183811.99" z="916.400000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677664.35" y="7183814.65" z="916.400000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677666.65" y="7183821.10" z="916.320000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677673.24" y="7183822.85" z="915.570000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">

```

```

        <Ponto x="677678.17" y="7183831.07" z="915.010000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677684.41" y="7183829.70" z="915.010000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677690.26" y="7183827.76" z="915.010000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677695.47" y="7183824.60" z="915.010000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677688.27" y="7183823.90" z="915.010000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677745.33" y="7183801.67" z="914.690000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677744.09" y="7183793.09" z="913.820000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677755.83" y="7183826.70" z="913.660000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677728.24" y="7183831.32" z="914.730000" />
    </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
    <Cor r="1" g="1" b="0"/>
    <Superficie0D modeloVRML="Arvore.wrl">
        <Ponto x="677617.19" y="7183804.73" z="915.370000" />
    </Superficie0D>
</FenomenoTerrestre>

```

```

    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677620.82" y="7183812.70" z="915.370000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677612.91" y="7183792.58" z="915.370000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677625.34" y="7183795.78" z="915.370000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677627.40" y="7183801.15" z="915.370000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677629.62" y="7183805.49" z="915.370000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677631.62" y="7183809.92" z="915.370000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677633.51" y="7183813.49" z="915.370000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677635.61" y="7183817.76" z="915.370000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677637.00" y="7183821.43" z="915.370000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>

```

```

        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677638.54" y="7183825.37" z="915.370000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677640.58" y="7183830.57" z="915.370000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677626.64" y="7183826.57" z="913.540000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677624.35" y="7183821.39" z="913.540000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677614.65" y="7183824.48" z="914.370000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677743.13" y="7183804.58" z="913.020000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677762.21" y="7183793.26" z="912.940000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677765.21" y="7183798.86" z="912.780000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677767.65" y="7183806.39" z="912.780000" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
        <Cor r="1" g="1" b="0"/>
        <Superficie0D modeloVRML="Arvore.wrl">
            <Ponto x="677771.31" y="7183812.49" z="912.780000" />
        </Superficie0D>
    </FenomenoTerrestre>

```

```

</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677773.25" y="7183817.31" z="912.780000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677777.49" y="7183826.67" z="912.780000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Arvore"
dimensao="0">
  <Cor r="1" g="1" b="0"/>
  <Superficie0D modeloVRML="Arvore.wrl">
    <Ponto x="677729.62" y="7183725.92" z="913.140000" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
  <Cor r="0" g="1" b="1"/>
  <Superficie0D modeloVRML="Poste.wrl">
    <Ponto x="677615.93" y="7183780.18" z="915.22" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
  <Cor r="0" g="1" b="1"/>
  <Superficie0D modeloVRML="Poste.wrl">
    <Ponto x="677647.77" y="7183747.45" z="914.87" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
  <Cor r="0" g="1" b="1"/>
  <Superficie0D modeloVRML="Poste.wrl">
    <Ponto x="677680.68" y="7183716.36" z="914.75" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
  <Cor r="0" g="1" b="1"/>
  <Superficie0D modeloVRML="Poste.wrl">
    <Ponto x="677792.04" y="7183786.97" z="915.90" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
  <Cor r="0" g="1" b="1"/>
  <Superficie0D modeloVRML="Poste.wrl">
    <Ponto x="677773.18" y="7183714.89" z="917.41" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
  <Cor r="0" g="1" b="1"/>
  <Superficie0D modeloVRML="Poste.wrl">
    <Ponto x="677723.50" y="7183734.45" z="914.71" />
  </Superficie0D>
</FenomenoTerrestre>
<FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">

```

```

        <Cor r="0" g="1" b="1"/>
        <Superficie0D modeloVRML="Poste.wrl">
            <Ponto x="677728.01" y="7183724.47" z="914.51" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
        <Cor r="0" g="1" b="1"/>
        <Superficie0D modeloVRML="Poste.wrl">
            <Ponto x="677737.38" y="7183729.41" z="914.51" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
        <Cor r="0" g="1" b="1"/>
        <Superficie0D modeloVRML="Poste.wrl">
            <Ponto x="677710.88" y="7183733.61" z="914.83" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
        <Cor r="0" g="1" b="1"/>
        <Superficie0D modeloVRML="Poste.wrl">
            <Ponto x="677692.66" y="7183741.16" z="914.87" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
        <Cor r="0" g="1" b="1"/>
        <Superficie0D modeloVRML="Poste.wrl">
            <Ponto x="677674.40" y="7183749.29" z="914.79" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
        <Cor r="0" g="1" b="1"/>
        <Superficie0D modeloVRML="Poste.wrl">
            <Ponto x="677656.75" y="7183757.20" z="914.75" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
        <Cor r="0" g="1" b="1"/>
        <Superficie0D modeloVRML="Poste.wrl">
            <Ponto x="677638.54" y="7183764.61" z="914.75" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
        <Cor r="0" g="1" b="1"/>
        <Superficie0D modeloVRML="Poste.wrl">
            <Ponto x="677620.61" y="7183772.41" z="914.75" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
        <Cor r="0" g="1" b="1"/>
        <Superficie0D modeloVRML="Poste.wrl">
            <Ponto x="677641.25" y="7183797.55" z="913.51" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
        <Cor r="0" g="1" b="1"/>
        <Superficie0D modeloVRML="Poste.wrl">
            <Ponto x="677770.17" y="7183803.75" z="912.64" />

```

```
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
        <Cor r="0" g="1" b="1"/>
        <Superficie0D modeloVRML="Poste.wrl">
            <Ponto x="677761.72" y="7183784.18" z="912.64" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
        <Cor r="0" g="1" b="1"/>
        <Superficie0D modeloVRML="Poste.wrl">
            <Ponto x="677761.72" y="7183824.41" z="912.52" />
        </Superficie0D>
    </FenomenoTerrestre>
    <FenomenoTerrestre categoria="FenomenoTopografico" tipo="Poste"
dimensao="0">
        <Cor r="0" g="1" b="1"/>
        <Superficie0D modeloVRML="Poste.wrl">
            <Ponto x="677748.90" y="7183795.29" z="912.52" />
        </Superficie0D>
    </FenomenoTerrestre>
</AreaLevantamento>
```