

ANDERSON RESENDE LAMAS

**UMA ARQUITETURA PARA O DESENVOLVIMENTO DE SISTEMAS  
DE INFORMAÇÃO GEOGRÁFICA MÓVEIS SENSÍVEIS AO  
CONTEXTO**

Dissertação apresentada à  
Universidade Federal de Viçosa, como  
parte das exigências do Programa de  
Pós-Graduação em Ciência da  
Computação, para obtenção do título de  
*Magister Scientiae*.

VIÇOSA  
MINAS GERAIS - BRASIL  
2009

ANDERSON RESENDE LAMAS

**UMA ARQUITETURA PARA O DESENVOLVIMENTO DE SISTEMAS  
DE INFORMAÇÃO GEOGRÁFICA MÓVEIS SENSÍVEIS AO  
CONTEXTO**

Dissertação apresentada à  
Universidade Federal de Viçosa, como  
parte das exigências do Programa de  
Pós-Graduação em Ciência da  
Computação, para obtenção do título de  
*Magister Scientiae*.

APROVADA: 13 de abril de 2009.

---

Alcione de Paiva Oliveira  
(Co-Orientador)

---

Mauro Nacif Rocha  
(Co-Orientador)

---

José Luis Braga

---

Tiago Garcia de Senna Carneiro

---

Jugurta Lisboa Filho  
(Orientador)

*Dedico este trabalho aos meus pais, Inês e Lino. Pelas oportunidades que não tiveram e que me foram proporcionadas, faço deste trecho minha homenagem a eles.*

## **AGRADECIMENTOS**

Agradeço primeiramente aos meus pais pelo exemplo de vida, por minha educação e pela dedicação que sempre tiveram tanto comigo quanto com minha irmã. Tenho certeza que a existência de uma excelente base familiar e os valores que me ensinaram foram fundamentais para que eu me dedicasse e concluísse esta dissertação.

Ao Professor Jugurta Lisboa Filho, meu orientador e muitas vezes um grande amigo, que sempre me incentivou a realizar um trabalho que não fosse o mais óbvio ou o mais fácil, mas sim aquele trouxesse algum desafio e pudesse dar alguma contribuição. Creio que estes objetivos foram alcançados. Obrigado chefe!

Aos Professores Alcione e Mauro que co-orientaram este trabalho e colaboraram diretamente na dissertação ou através das disciplinas que ministraram. Tenho certeza que os tópicos de Ontologias e Computação Móvel foram melhor explorados graças a contribuição deles.

Agradeço também aos ex-alunos de graduação do Departamento de Informática André Marques Lopes, que me auxiliou no tratamento dos dados geográficos, e Ronoel Matos de Almeida Botelho Júnior, que me ajudou a implementar a aplicação de SIG Móvel. Ambos contribuíram muito para minha dissertação e foram inclusive co-autores de artigos que publicamos. Foram dois amigos que fiz e que torço muito pelo sucesso deles.

Aos funcionários da Divisão de Obras e Projetos, Márcio Ivan Pereira Dias e Ricardo Luís Noce, pela disponibilidade em me fornecer um conjunto de mapas do campus da UFV. E também aos funcionários da Divisão de Jornalismo, Álvaro César Sant'Anna e José Paulo Martins, pela gentileza em me fornecerem um conjunto de fotografias sobre o campus.

Aos meus colegas da Central de Processamento de Dados (CPD) pela amizade e apoio principalmente em períodos de indisponibilidade devido a tarefas relacionadas ao mestrado. Um agradecimento especial ao José Mário Braga Filho, que por muito tempo dividiu a sala comigo e à Lícia Félix de Andrade, diretora que apoiou e conseguiu autorização junto à Reitoria para que eu realizasse o mestrado.

Por fim, gostaria de agradecer a todas as outras pessoas que de alguma forma colaboraram para que eu chegasse à conclusão deste mestrado. Seja com incentivo, com carinho ou amizade, sei que cada um teve uma contribuição muito importante para que eu seguisse nesta caminhada que exige muita dedicação e esforço, mas que agora tenho certeza que vale a pena.

# SUMÁRIO

LISTA DE FIGURAS.....	vii
LISTA DE QUADROS.....	viii
RESUMO.....	ix
ABSTRACT.....	x
<b>1 Introdução.....</b>	<b>1</b>
1.1 Objetivos.....	2
1.2 Estrutura de Dissertação.....	3
<b>2 Revisão de Literatura e Trabalhos Relacionados .....</b>	<b>4</b>
2.1 Sistemas de Informação Geográfica.....	4
2.1.1 Sistemas de Informação Geográfica Móveis .....	5
2.1.2 Banco de Dados Geográficos .....	6
2.2 Ontologias.....	7
2.2.1 Especificação de Ontologias.....	8
2.2.2 Uso de Ontologias .....	10
2.3 Aplicações Sensíveis ao Contexto.....	11
2.3.1 Dimensões Semânticas de Contexto .....	13
2.3.2 Modelagem de Contexto.....	14
2.3.2.1 Ontologia <i>Time</i> .....	16
2.3.2.2 Ontologia <i>Temporal Event</i> .....	20
2.3.2.3 Ontologia <i>Spatial</i> .....	22
2.3.2.4 Ontologia <i>Spatial Event</i> .....	25
2.3.2.5 Ontologia <i>Actor</i> .....	26
2.3.2.6 Ontologia <i>Device</i> .....	27
2.3.2.7 Ontologia <i>Activity</i> .....	28
2.4 <i>Web Services</i> .....	31
2.4.1 <i>Web Services</i> na Informação Geográfica (Geo Web Services) .....	33
2.4.1.1 Web Map Service (WMS).....	34
2.4.1.2 Web Feature Service (WFS).....	35

2.5	Trabalhos Relacionados .....	36
<b>3</b>	<b>A Arquitetura Context-Mobile-GIS (CM-GIS).....</b>	<b>39</b>
3.1	A definição do modelo de contexto baseado em ontologias .....	41
3.2	O banco de dados geográficos .....	42
3.3	Os <i>Web Services</i> .....	42
3.4	O Módulo de Consultas Espaciais .....	43
3.5	O Módulo de Gerência de Contexto.....	44
3.6	A aplicação SIG Móvel.....	45
<b>4</b>	<b>UFV-GeoMobile: Um estudo de caso.....</b>	<b>46</b>
4.1	Cenário da Aplicação .....	46
4.2	O modelo de contexto: ontologia de um campus universitário.....	48
4.3	Construção do Banco de Dados Geográficos .....	50
4.4	Definição do <i>Web Service</i> .....	52
4.5	Módulo de Consultas Espaciais .....	53
4.6	Módulo de Gerência de Contexto .....	53
4.7	Aplicação SIG Móvel e Consultas.....	53
4.7.1	Descrição da Interface .....	54
4.7.2	Cenários de Consulta .....	55
4.7.2.1	Consulta a um ponto .....	55
4.7.2.2	Localização de Serviços ou Instalações Diversas .....	57
4.7.2.3	Localização de Pessoas .....	59
4.7.3	Notificação sobre a Localização Geográfica de Eventos .....	61
<b>5</b>	<b>Conclusões e Trabalhos Futuros .....</b>	<b>64</b>
5.1	Conclusões .....	64
5.2	Contribuições .....	65
5.3	Trabalhos Futuros.....	66
<b>6</b>	<b>Referências Bibliográficas .....</b>	<b>67</b>

## LISTA DE FIGURAS

Figura 1 - Modelo SeCoM [Bulcão Neto, 2006].....	16
Figura 2 - Ontologia Time [Bulcão Neto, 2006]. .....	18
Figura 3 - Ontologia Temporal Event [Bulcão Neto, 2006]. .....	21
Figura 4 - Ontologia Spatial [Bulcão Neto, 2006]. .....	23
Figura 5 - Ontologia <i>Spatial Event</i> [Bulcão Neto, 2006]. .....	25
Figura 6 - Ontologia Actor [Bulcão Neto, 2006]. .....	27
Figura 7 - Ontologia Activity [Bulcão Neto, 2006]. .....	30
Figura 8 - Arquitetura dos Web Services (Adaptado de [Araújo, 2005]). .....	32
Figura 9 - Arquitetura proposta para um SIG Móvel sensível ao Contexto. ....	40
Figura 10 - Vista parcial do campus da UFV [Fonte: Divisão de Jornalismo/CCS-UFV]. .....	47
Figura 11 - OntoUFV - Ontologia de um Campus Universitário.....	50
Figura 12 - Modelo do banco de dados geográficos do UFV-GeoMobile .....	52
Figura 13 - Tela inicial da aplicação UFV-GeoMobile. ....	54
Figura 14 - Resultado da consulta a um ponto.....	56
Figura 15 - Telas para consulta e localização geográfica de instalações.....	59
Figura 16 - Telas para consulta e localização geográfica de pessoas. ....	61
Figura 17 - Tela para login na aplicação UFV- GeoMobile.....	62
Figura 18 - Tela para notificação de eventos. ....	63

## LISTA DE QUADROS

Quadro 1 - Operação do Web Service para consulta a um ponto. ....	56
Quadro 2 - Operação e SQL no Módulo de Consultas Espaciais para consulta a um ponto. ....	56
Quadro 3 - Operação do Web Service para consulta a instalação. ....	58
Quadro 4 - Operação e SQL no Módulo de Consultas Espaciais para consulta a uma instalação. ....	58
Quadro 5 - Operação do Web Service para consulta a pessoas. ....	60
Quadro 6 - Operação e SQL no Módulo de Consultas Espaciais para consulta a uma pessoa. ....	60
Quadro 7 - Operação do Web Service para notificar sobre eventos. ....	62
Quadro 8 - Operação e SQL no Módulo de Gerência de Contexto para notificar sobre eventos. ....	62

## RESUMO

LAMAS, Anderson Resende; M.Sc.; Universidade Federal de Viçosa; abril de 2009; **Uma arquitetura para o desenvolvimento de Sistemas de Informação Geográfica Móveis sensíveis ao contexto**; Orientador: Jugurta Lisboa Filho. Co-Orientadores: Alcione de Paiva Oliveira e Mauro Nacif Rocha.

A integração de informações de contexto em um ambiente de SIG Móvel é um fator que pode agregar um grande valor às informações exibidas para o usuário de uma aplicação. Esta afirmação é possível graças à possibilidade de entrega de informações personalizadas ao usuário, combinadas com a exibição de dados geo-espaciais. Diante deste cenário, esta dissertação investiga as características e soluções já apresentadas tanto na área de SIG Móvel quanto de aplicações sensíveis ao contexto e assim propõe uma arquitetura para o desenvolvimento de aplicações de SIG Móveis capazes de gerenciar informações de contexto. Esta arquitetura baseia-se na especificação de um modelo de contexto baseado em ontologias e em um conjunto de *Web Services* para acesso a informações armazenadas remotamente em um banco de dados geográficos. Para validar esta arquitetura, foi desenvolvida uma aplicação tendo como cenário o campus da Universidade Federal de Viçosa.

## ABSTRACT

LAMAS, Anderson Resende; M.Sc.; Universidade Federal de Viçosa; April of 2009; **An architecture for developing of Context-Aware Mobile Geographical Information System**; Adviser: Jugurta Lisboa Filho. Co-Advisers: Alcione de Paiva Oliveira and Mauro Nacif Rocha.

The integration of context information in a Mobile GIS environment can add great value to the information that is displayed to an application user. This is due to the possibility of delivering personalized information to the user in combination with the display of geo-spatial data. In this scenario, this work presents the characteristics and existing solutions for both the field of Mobile GIS and context sensitive applications and proposes an architecture for developing applications of Mobile GIS capable of managing context information. This architecture was established based on the specification of an ontology-based context model and a set of Web Services to access information remotely stored in a geographic database. To validate the architecture proposed in this work, a system was developed targeting the campus of the Federal University of Vicosa (UFV).

# 1 Introdução

Vive-se hoje em um período de avanços sem precedentes em tecnologias que exploram a mobilidade e a comunicação entre usuários e dispositivos. Este fato é evidenciado pela constante expansão das infra-estruturas de rede sem fio e a grande proliferação de dispositivos móveis tais como PDA's (*Personal Digital Assistant*), *Smartphones* e telefones celulares. Este cenário tem causado um grande impacto nas mais diversas áreas do conhecimento, especialmente pela possibilidade do desenvolvimento de novas aplicações que exploram tais características.

Na área de Sistemas de Informação Geográfica (SIG), por exemplo, a interseção de dados geo-espaciais com dispositivos móveis deu origem a um novo tipo de sistemas, denominado SIG Móvel (*Mobile GIS*). Segundo Solyman [2005], SIG Móvel pode ser definido como uma estrutura que integra hardware e software para acesso a dados espaciais através de uma rede sem-fio, utilizando-se dispositivos móveis.

Outra área que foi potencializada graças à mobilidade foi a de aplicações sensíveis ao contexto (*context-aware applications*). Estas aplicações são caracterizadas por utilizarem informações de contexto para fornecerem serviços e informações relevantes a usuários durante a realização de alguma tarefa [Dey, 2000]. Neste cenário, contexto pode ser definido como qualquer informação que possa ser utilizada para caracterizar uma pessoa, um lugar ou um objeto relevantes em uma interação entre um usuário e uma aplicação [Dey, 2000]. Como informações deste tipo tem-se, por exemplo, a identificação e preferências do usuário, sua localização ou características dos dispositivos utilizados.

Considerando-se suas características peculiares, tanto aplicações na área de SIG Móvel quanto na de aplicações móveis sensíveis ao contexto trazem grandes desafios para o seu desenvolvimento. Um desafio comum entre estas áreas constitui em criar um mecanismo que contorne o problema da capacidade limitada dos dispositivos móveis. Entre estas limitações estão o pouco espaço para armazenar dados, baixo poder de processamento, telas pequenas e a baixa largura de banda da rede sem-fio [Caduff e Egenhofer, 2005]. Estes fatores

impedem, por exemplo, que o dispositivo armazene localmente uma grande quantidade de dados ou mesmo que possa processá-los e dar a resposta em um tempo adequado.

Em relação aos SIG Móveis tem-se ainda o desafio de utilizar tecnologias corretas para o armazenamento, consulta e exibição de dados geo-espaciais em dispositivos móveis. Como exemplo destas tecnologias pode-se citar os banco de dados geográficos [Castro et al., 2003] e os Geo Web Services [Araújo, 2005].

Já para as aplicações sensíveis ao contexto um dos desafios está em como definir um modelo de contexto que seja o mais completo e adequado para o domínio da aplicação a ser desenvolvida. Graças ao poder de representação e especificação formal das ontologias, muitos trabalhos têm proposto esta abordagem como forma de modelagem de contexto [Sadeh et al., 2002], [Toivonen et al., 2003] e [Weibenberg, 2006].

A integração de informações de contexto em um ambiente de SIG Móvel é um fator que pode agregar um grande valor às informações exibidas para o usuário de uma aplicação. Esta afirmação é possível graças à possibilidade de entrega de informações personalizadas ao usuário, combinadas com a exibição de dados geo-espaciais. Diante deste cenário, esta dissertação propõe-se a investigar as características e soluções já apresentadas tanto na área de SIG Móvel quanto de aplicações sensíveis ao contexto e assim propor uma arquitetura para o desenvolvimento de aplicações de SIG Móveis capazes de gerenciar informações de contexto.

Os objetivos específicos e estrutura desta dissertação são descritos nas seções que se seguem.

## **1.1 Objetivos**

Este trabalho tem como objetivo principal a especificação de uma arquitetura para o desenvolvimento de aplicações de Sistemas de Informação Geográfica (SIG) Móveis sensíveis ao contexto. Isto implica em definir um modelo que sirva de orientação para a construção de aplicações que gerenciem dados geográficos e informações de contexto em um ambiente com dispositivos móveis.

Para que este objetivo seja alcançado é necessário que esta arquitetura seja capaz de prover mecanismos que são típicos das áreas de SIG e das aplicações sensíveis ao contexto. Deste modo a arquitetura deverá conter as seguintes características:

- Especificar uma forma para modelar o contexto para a aplicação desenvolvida;
- Considerar as tecnologias necessárias para o desenvolvimento de aplicações que manipulem dados geo-espaciais em dispositivos móveis;
- Propor um mecanismo para acesso remoto aos dados armazenados em um banco de dados geográficos.

Além de propor esta arquitetura, este trabalho tem como objetivo adicional desenvolver um estudo de caso que valide a arquitetura proposta.

## **1.2 Estrutura de Dissertação**

Além deste capítulo introdutório, a dissertação está estruturada da seguinte forma:

- O Capítulo 2 trata da Revisão de Literatura e dos trabalhos correlatos que serviram de fundamentação teórica para este trabalho. São descritos itens como Sistemas de Informação Geográfica, Ontologias, Aplicações Sensíveis ao Contexto e *Web Services*;
- O Capítulo 3 apresenta a arquitetura proposta nesta dissertação. São detalhados cada um dos componentes que a formam e o porquê das decisões tomadas para se chegar até este modelo;
- O Capítulo 4 apresenta um estudo de caso desenvolvido neste trabalho para validar a arquitetura proposta;
- Por fim, no Capítulo 5 são feitas as avaliações e listadas as devidas conclusões sobre o trabalho desenvolvido.

## 2 Revisão de Literatura e Trabalhos Relacionados

### 2.1 Sistemas de Informação Geográfica

Os Sistemas de Informação Geográfica (SIG) podem ser definidos como sistemas computacionais capazes de capturar, modelar, armazenar, recuperar, manipular, analisar e apresentar dados geográficos [Worboys e Duckham, 2004]. Seu grande diferencial em relação aos Sistemas de Informação convencionais é a sua capacidade de tratar dados alfanuméricos integrados a dados referenciados geograficamente.

Segundo Câmara [2005], dentre os modelos formais para representação de dados geográficos, distinguem-se dois tipos principais: o modelo de geo-campos e o modelo de geo-objetos. O modelo de geo-campos considera o espaço geográfico como uma superfície contínua, sobre a qual variam os fenômenos a serem observados. Já o modelo de geo-objetos representa o espaço geográfico como uma coleção de entidades individualizadas distintas, onde cada entidade é definida por uma fronteira fechada.

Os modelos geo-campos e geo-objetos podem ser mapeados para estruturas de dados de duas naturezas: matricial (*raster*) ou vetorial. A forma matricial divide o espaço em grades regulares, freqüentemente associado ao pixel, ao qual é atribuído um valor único, que determina o tipo de objeto ou a condição daquele determinado ponto. Já o modelo vetorial usa as entidades ponto, linha e polígono para representar objetos e condições do mundo real [Castro et al., 2003]. As estruturas em formato vetorial ou matricial em um SIG são frequentemente chamadas de *layers* ou mapas temáticos [Vinhas e Ferreira, 2005].

O desenvolvimento de um SIG está fortemente associado ao uso de Sistemas Gerenciadores de Banco de Dados (SGBD). Existem basicamente duas formas de integração entre os SIG e os SGBD, as quais foram denominadas arquitetura dual e arquitetura integrada [Câmara, 2005].

A arquitetura dual caracteriza-se por armazenar as componentes espaciais e os dados alfanuméricos em estruturas diferentes. Enquanto os dados alfanuméricos são armazenados em um SGBD relacional, a componente espacial

é armazenada em arquivos com formato proprietário. Já na arquitetura integrada tanto os dados alfanuméricos quanto os espaciais são armazenados em um único SGBD. O banco de dados utilizado neste caso é chamado de banco de dados geográficos.

### **2.1.1 Sistemas de Informação Geográfica Móveis**

Os Sistemas de Informação Geográfica Móveis (SIG Móveis) podem ser definidos como uma estrutura que integra hardware e software para acesso a dados espaciais através de uma rede sem-fio, utilizando-se dispositivos móveis [Solyman, 2005].

Ainda de acordo com [Solyman, 2005], existem duas principais áreas de aplicação dos SIG Móveis:

- SIG baseado-em-campo, o qual foca em coleta de dados espaciais e descritivos no campo. A obtenção de dados censitários é um exemplo deste tipo de aplicação;
- Serviço Baseado em Localização (LBS), o qual utiliza dispositivos de orientação espacial, como o GPS (*Global Positioning System*), para prover serviços como determinação de rotas ou monitoramento de veículos. Os trabalhos de [Silva et al., 2004] e [Silva, 2004] são exemplos deste tipo de SIG Móvel.

De acordo com [Tomko, 2003] o desenvolvimento de um SIG Móvel requer atenção aos seguintes elementos:

- Armazenamento e recuperação de dados armazenados em um banco de dados geográficos;
- A correta utilização de tecnologias para manipulação de dados geográficos;
- Prover um mecanismo para realização de consultas e acesso a dados localizados remotamente.

Os trabalhos de [Li et al., 2006] e [Brisaboa et al., 2007] exploram o uso de *Web Services* para permitir o acesso remoto a informações a partir de um SIG

Móvel. O trabalho proposto nesta dissertação também propõe esta abordagem conforme é descrito no Capítulo 3.

### 2.1.2 Banco de Dados Geográficos

Um banco de dados geográficos (BDG) pode ser definido como um banco de dados não-convencional onde os dados tratados possuem, além de atributos descritivos, uma representação geométrica no espaço geográfico [Inpe, 2007].

Para a criação de um BDG é necessário que o SGDB utilizado possua algum mecanismo de extensibilidade, que permita a adição e manipulação de dados espaciais. Os SGBD que possuem esta funcionalidade, em geral, possuem suporte ao modelo Objeto-Relacional (O-R). Segundo [Queiroz, 2004], dentre os itens que deve ser estendidos estão:

- Modelo de dados com suporte a tipos de dados espaciais (Spatial Data Types, SDT), como ponto, linha e polígono;
- Funções e operadores espaciais que permitem que esses tipos de dados sejam manipulados, assim como os tipos alfanuméricos básicos;
- Métodos de indexação espacial;
- Linguagem de consulta estendida para permitir a manipulação dos SDT.

Atualmente existem diversas extensões de SGBD para tratar dados geográficos no formato vetorial, sendo algumas proprietárias e outras disponíveis como software livre. Dentre as comerciais pode-se citar o Oracle Spatial [Oracle, 2008], IBM DB2 Spatial Extender [IBM, 2008a] e Informix Spatial e Geodetic Datablade [IBM, 2008b]. Como alternativa a elas, existem extensões para o PostgreSQL [PostGis, 2008] e MySQL [Mysql, 2008] no universo de software livre.

A fim de estabelecer um conjunto de funcionalidades básicas que todo BDG deveria incorporar, o consórcio Open GIS (OGIS), apoiado pelos principais líderes do mercado mundial de banco de dados (Oracle, IBM e Informix), desenvolveu uma especificação conhecida como *Simple Features Specification For SQL – SFSQL* [OpenGIS, 2007].

Essa especificação apresenta a proposta do modelo dos tipos geométricos, dos operadores topológicos, métricos e dos que geram novas geometrias que todo

banco de dados geográfico deveria apresentar. Além disso, essa especificação apresenta uma proposta de um esquema de metadados para as tabelas que contenham tipos geométricos.

## 2.2 Ontologias

O termo ontologia teve origem na filosofia, significando a representação da existência, através de uma explicação sistemática, de tudo aquilo que pode “existir” ou “ser” [Viegas, 2006]. Na área de computação, pesquisas em ontologias tiveram origem na área de inteligência artificial e representação do conhecimento. Neste contexto, ontologias são definidas como “uma conceitualização formal, explícita e compartilhada de alguma área do conhecimento, também chamado de domínio do discurso” [Uschold e Gruninger, 1996].

Segundo [Studer et al., 1998], a definição de ontologia dada por [Uschold e Gruninger, 1996] pode ser detalhada da seguinte forma: conceitualização refere-se a um modelo abstrato que identifica os conceitos relevantes de algum fenômeno do mundo. Explícita significa que o conjunto de conceitos definidos e as restrições aplicadas devem ser explicitamente definidos. Formal refere-se ao fato de que uma ontologia deve ser definida por uma linguagem que seja processável por computador. Por fim, compartilhada refere-se ao fato de que uma ontologia descreve um conhecimento consensual, que é utilizado por mais de um objeto e aceito por um grupo.

De acordo com [Guarino, 1998] as ontologias podem ser ainda classificadas de acordo com seu nível de abstração ou a sua dependência em relação a uma tarefa específica. Esta classificação é dividida da seguinte forma:

- Ontologias de alto nível: descrevem conceitos gerais como tempo, espaço, objeto, evento, ação, os quais são independentes de um domínio ou problema em particular;
- Ontologias de domínio: descrevem o vocabulário relativo a um domínio específico, tais como medicina ou informática, através da especialização dos termos definidos nas ontologias de alto nível;

- Ontologias de tarefa: descrevem tarefas genéricas ou atividades, tais como diagnosticar ou vender, também através da especialização dos termos definidos nas ontologias de alto nível;
- Ontologias de aplicação: descrevem conceitos dependentes, quer do domínio quer da tarefa, especializando as ontologias correspondentes.

Para descrever de forma explícita a semântica do vocabulário de uma ontologia, são necessárias primitivas de modelagem e relacionamentos semânticos [Lacy, 2005]. Primitivas de modelagem formam a base de qualquer ontologia, representadas por classes, propriedades e indivíduos (ou objetos). Os relacionamentos semânticos expressam os variados tipos de relacionamentos que podem existir entre as primitivas de modelagem de um domínio, tais como especialização, generalização, composição, equivalência, disjunção, simetria, negação, transitividade, entre outros.

Para a especificação formal de uma ontologia foram desenvolvidas diversas linguagens. Entre elas estão a RDF (*Resource Description Framework*), OIL (*Ontology Inference Layer*), DAML (*DARPA Agent Markup Language*), DAML +OIL e a OWL (*Web Ontology Language*). A OWL é inclusive a mais difundida por ser uma recomendação da W3C para o desenvolvimento de aplicações inseridas na Web Semântica [OWL, 2004].

Além das linguagens existem também alguns editores gráficos de ontologias que facilitam o processo de criação e edição das mesmas. Entre eles estão o OilEd [OilEd, 2003], o OntoEdit [OntoEdit, 2007] e o Protégé [Protege, 2005].

### **2.2.1 Especificação de Ontologias**

Existem diversos padrões e linguagens para especificação de ontologias, com graus de formalismo e poder de expressividade diferentes. Para uma especificação adequada de uma ontologia é necessário que ela seja descrita em alguma linguagem formal e implementada para que seja entendida, compartilhada e exportada entre diferentes usuários [Borges, 2006].

A RDF (*Resource Description Framework*) é um modelo desenvolvido pelo W3C para a representação de informação na Web utilizando a sintaxe XML. Auxiliada pelo padrão RDF Schema (RDFS), ela especifica os vocabulários usados para a representação de informação na Web semântica por meio de classes, propriedades, tipos e relacionamentos [RDF, 2004].

A linguagem DAML+OIL é uma combinação de DAML (*DARPA Agent Markup Language*) e OIL (*Ontology Inference Layer*) que estende os padrões RDF/RDFS adicionando maior expressividade. Ela tem uma sintaxe RDF/XML baseada no paradigma de *frames* e descreve a estrutura de um domínio em um estilo orientado a objetos [Borges, 2006].

A OWL (*Web Ontology Language*) é uma linguagem de definição de ontologias projetada para uso por aplicações que necessitam processar o conteúdo das informações descrito nas ontologias, e não apenas exibir estas informações [OWL, 2004]. Esta linguagem consiste em uma revisão da DAML+OIL e o padrão recomendado pelo W3C para o desenvolvimento de aplicações de Web Semântica.

A OWL possui três diferentes sub-linguagens [OWL, 2004]:

- OWL-Lite: Fornece apenas um suporte básico de representatividade, permitindo a definição de uma hierarquia de classes com restrições e propriedades simples;
- OWL-DL: Possui um maior poder de expressividade que a OWL-Lite, permitindo o processamento de informações e raciocínio automatizado com base na Lógica Descritiva;
- OWL-Full: Tem maior expressividade, o que permite projetar e classificar classes com alto grau de complexidade. Seu foco está na representatividade e não no raciocínio automático.

Entre as vantagens apresentadas por essas linguagens destacam-se [Knublauch et al., 2006]:

- são otimizadas para a representação de conhecimento estruturado em um nível elevado de abstração; os modelos de domínio podem ser disponibilizados na Internet e compartilhados entre múltiplas aplicações;

- são baseadas em dialetos não ambíguos da lógica formal, permitindo a utilização de serviços inteligentes baseados no raciocínio (*reasoning*), possibilitando assim, em tempo de execução, a definição dinâmica de classes, a re-classificação de instâncias e a execução de consultas lógicas complexas.
- operam sobre estruturas similares às linguagens orientadas a objeto, e podem ser eficazmente integradas aos componentes de software tradicionais.

### **2.2.2 Uso de Ontologias**

Ontologias são usadas para diversos fins, variando do apoio aos processos de desenvolvimento de software, ou qualquer outra atividade executada por equipes geograficamente distribuídas, ao apoio em tempo de execução aos sistemas de informação [Guarino, 1998]. Uma vantagem obtida com o uso de ontologias é a prevenção de diferentes interpretações a respeito da semântica dos termos de um domínio. A abordagem declarativa utilizada pelas ontologias permite descrever um domínio sem qualquer compromisso com a implementação de um sistema de software, ou seja, o conhecimento modelado é independente de implementação [Novello, 2003].

Um uso freqüente de ontologias é para prover interoperabilidade entre sistemas de software. Sistemas que compartilham um domínio em comum possuem a mesma interpretação da semântica do vocabulário em questão, o que lhes permite integração transparente de serviços, como serviços de armazenamento, classificação, busca e correlação de informações distribuídas na Web [Fonseca et al., 2000].

Um esquema de banco de dados pode ser derivado de uma ontologia pela adição da informação do tipo de dados e traduzindo o formalismo da representação do conhecimento ontológico em um modelo de representação do banco de dados como o relacional ou o modelo orientado a objetos. Da mesma forma, um esquema de banco de dados pode ser utilizado para estabelecer os conceitos para povoar uma ontologia [Viegas, 2006].

As ontologias descritivas e os esquemas conceituais de bancos de dados compartilham o mesmo esforço de modelar algum domínio ou atividade. No entanto, enquanto as ontologias têm sido tradicionalmente consideradas um meio de descrever o mundo, os esquemas de bancos de dados também auxiliam o gerenciamento dos dados que representam o mundo de interesse [Spaccapietra et al., 2004].

O uso de ontologias pode trazer grandes benefícios para o desenvolvimento de SIG, já que elas permitem estabelecer correspondências entre diferentes domínios de entidades espaciais [Fonseca et al., 2000]. Além disso, segundo Smith e Mark [1998], o uso de uma ontologia do mundo geográfico pode:

- Ajudar a entender como diferentes comunidades compartilham informações;
- Ajudar a descobrir certas distorções presentes nos processos cognitivos de apreensão do mundo geográfico;
- Fornecer padrões para o desenvolvimento de SIG.

### **2.3 Aplicações Sensíveis ao Contexto**

O desenvolvimento de aplicações sensíveis ao contexto investiga o emprego de informações que possam fornecer serviços adaptados a usuários e aplicações em uma interação usuário-computador [Bulcão Neto, 2006]. Segundo Dey [2000], uma aplicação é caracterizada como sensível ao contexto “quando utiliza informações de contexto para fornecer serviços e/ou outras informações relevantes a um usuário, onde a relevância está diretamente relacionada à tarefa que o usuário desempenha em um dado momento”.

Segundo Dey et al. [2001], uma aplicação sensível ao contexto possui características que podem ser agrupadas nas três categorias listadas abaixo:

- apresentação de informação e serviços ao usuário - Aplicações que visam o fornecimento de informações e serviços que o usuário solicite. Para uma implementação adequada é conveniente considerar informações como o perfil do usuário, suas preferências e seu estado emocional;

- execução automática dos serviços do usuário - Incluem as aplicações capazes de executar um serviço de forma automática, considerando a situação atual do usuário;
- união de informações de contexto para um uso posterior - Englobam as aplicações que integram informações anteriores de contexto à situação atual do usuário. Isto permite que o ambiente encontre relações entre as informações e forneça novos serviços baseados nos anteriores.

As informações manipuladas pelas aplicações sensíveis ao contexto são chamadas de informações de contexto. Neste cenário, contexto pode ser definido como qualquer informação que possa ser utilizada para caracterizar uma pessoa, um lugar ou um objeto relevantes em uma interação entre um usuário e uma aplicação [Dey, 2000]. Como informações deste tipo tem-se, por exemplo, a identificação e preferências do usuário, sua localização ou características dos dispositivos utilizados.

De acordo com Chen e Kotz [2000] as informações de contexto podem ser obtidas de duas formas: *explícita*, quando a informação é obtida intencionalmente pelo usuário, ou *implícita*, quando a informação é obtida sem a comunicação intencional do usuário. Conforme a fonte em que foram obtidas estas informações podem ainda ser classificadas em: *sensed* (fornecidas por um sensor), *derived* (derivadas de outra fonte por inferência) ou *profiled* (definidas por uma pessoa ou por um serviço, tal como um *Web Service*) [Henricksen et al., 2002].

Aplicações sensíveis ao contexto são uma das bases para o estabelecimento da Computação Ubíqua ou Computação Pervasiva, a qual tem por meta permitir ao usuário o fácil acesso e processamento da informação, utilizando qualquer dispositivo, a qualquer momento e de qualquer lugar [Hansmann et al., 2003]. Esta característica de ciente de contexto permite que um ambiente pervasivo possa interagir com o usuário de uma maneira menos intrusiva, gerenciando adequadamente as informações e os recursos disponíveis [Satyanarayanan, 2001].

### 2.3.1 Dimensões Semânticas de Contexto

Muitos sistemas sensíveis ao contexto limitam o conjunto de informações manipuladas à localização e identificação de pessoas e objetos [Bulcão Neto, 2006]. Porém pela própria definição de contexto apresentada na seção anterior, percebe-se que existe uma grande diversidade de informações que podem ser utilizadas como informações de contexto, tais como informações do ambiente, incluindo tempo, histórico e dados de outros usuários. Esta diversidade de informações depende do domínio da aplicação que esteja sendo desenvolvida.

Abowd e Mynatt [2000] discutem a utilização de cinco dimensões semânticas para auxiliar no processo de especificação, modelagem e estruturação das informações de contexto. Estas dimensões foram divididas da seguinte forma:

- Quem: informações sobre as entidades envolvidas em alguma atividade. Estas entidades podem ser uma pessoa, um dispositivo ou um agente de software que interage com a aplicação;
- Onde: informações sobre a localização das pessoas e os objetos em um determinado espaço ou ambiente. Esta dimensão inclui também informações sobre as relações espaciais entre lugares, objetos e pessoas. Informações de contexto de localização incluem latitude, longitude, altitude e posição relativa a objetos e pessoas;
- Quando: são informações de contexto temporais que podem ser usadas para situar eventos em uma linha do tempo, ou auxiliar na interpretação de atividades humanas e no estabelecimento de padrões de comportamento. Informações de contexto temporais incluem, entre outras, data, hora, intervalos de tempo, dia da semana, mês e ano;
- O quê: informações que identificam as atividades que pessoas ou dispositivos possam estar fazendo em um determinado momento. Esta dimensão contextual é mais complexa que as anteriores, pois há diversas variáveis que devem ser previstas pela aplicação, incluindo a realização de atividades de forma concorrente. Além disso, uma atividade envolve as outras dimensões contextuais, pois uma atividade é realizada em um determinado espaço (Onde), em um determinado momento (Quando) e por

uma pessoa ou dispositivo (Quem). Configura-se, assim, como um dos principais desafios na computação sensível a contexto a obtenção de informações de contexto que possibilitem a interpretação correta da atividade de um usuário;

- Por quê: esta dimensão está relacionada com a determinação do motivo pelo qual determinada ação foi realizada. Em geral, estas informações são determinadas por indicativos como o estado emocional de um usuário. Este estado pode ser determinado por sensores que indicam itens como pressão arterial e expressões faciais;

Para complementar as dimensões semânticas propostas por Abowd e Mynatt [2000], Truong et al. [2001] discutem uma dimensão semântica originada do domínio de aplicações de modo de captura e acesso:

- Como: esta dimensão fornece informações relativas a como recursos de um ambiente físico podem ser capturados e acessados. Ela permite que se tenha informações sobre as características funcionais de cada dispositivo de captura e acesso utilizados. Essas informações podem ser utilizadas, por exemplo, para a personalização de acesso a informações capturadas via dispositivos móveis, como PDA's, os quais possuem características de acesso bastante restritas, como tamanho de tela, quantidade de energia em bateria e suporte à entrada e saída de dados [Bulcão Neto, 2006].

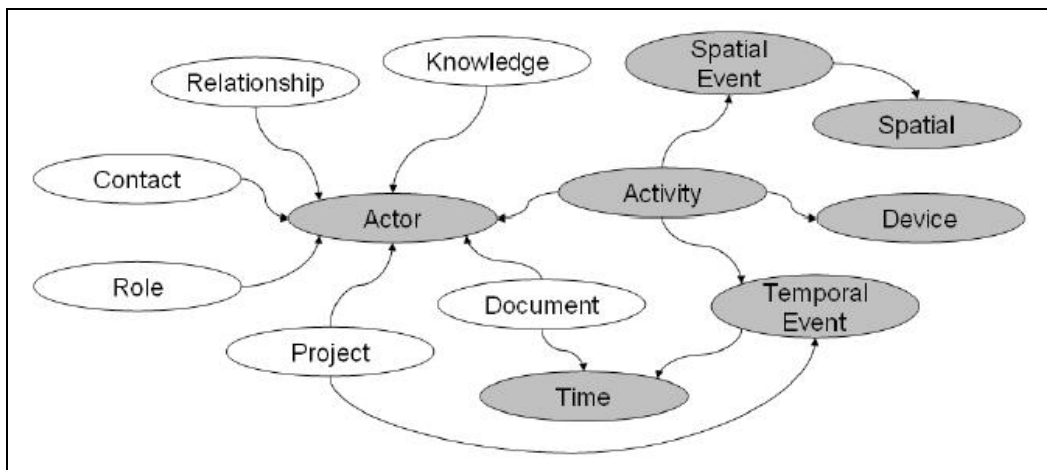
### **2.3.2 Modelagem de Contexto**

Segundo Strang e Linnhoff-Popien [2004] a chave para o desenvolvimento de uma aplicação sensível ao contexto é a definição de um modelo de contexto apropriado, o qual pode ser definido através do uso de ontologias. As características ontológicas de formalidade, semântica explícita e abstração de implementação habilitam sistemas de software não apenas a inferir novas informações a partir de informações modeladas por ontologias, mas também a compartilhar entre si essas informações de maneira a integrar de forma transparente os serviços que as manipulam [Bulcão Neto, 2006].

Baseado nestas características, Bulcão Neto [2006] propõe o modelo SeCoM (Semantic Context Model), um modelo ontológico independente de domínio, descrito na linguagem OWL (Web Ontology Language), que serve como base para a definição de informações de contexto. Este modelo é composto de um conjunto de ontologias inter-relacionadas baseadas nas dimensões semânticas de identidade, localização, tempo, atividade e modo de captura e acesso.

SeCoM foi desenvolvido utilizando conceitos genéricos de tal forma que possa ser reutilizado por uma variedade de aplicações sensíveis ao contexto. Para que isto seja possível, as ontologias seguem uma abordagem em duas camadas: a camada superior de ontologias, apresentada na Figura 1, que representa o modelo em si e a camada inferior de ontologias, que é construída com conceitos específicos da aplicação sensível ao contexto que está sendo desenvolvida. Nesta camada, o modelo SeCoM pode ser reusado ou mesmo estendido com os conhecimentos que são particulares do domínio da aplicação. Um exemplo de reutilização deste modelo será apresentado no Capítulo 4 desta dissertação.

Com relação ao papel desempenhado pelas ontologias apresentadas na Figura 1, verifica-se as ontologias de apoio Conhecimento (*Knowledge*), Relacionamento (*Relationship*), Papel (*Role*), Contato (*Contact*), Documento (*Document*) e Projeto (*Project*), as quais modelam diversos aspectos relacionados a um Ator (*Actor*). Neste modelo atores são entidades que executam alguma ação em uma interação usuário-computador. Já as ontologias Espaço (*Spatial*), Tempo (*Time*), Atividade (*Activity*) e Dispositivo (*Device*) modelam, respectivamente, informações de contexto de localização, tempo, atividade e dispositivos computacionais de captura e acesso. As ontologias Evento Espacial (*Spatial Event*) e Evento Temporal (*Temporal Event*) são extensões das ontologias Espaço e Tempo para representar eventos que contenham componentes espaciais e temporais, concomitantemente.



**Figura 1 - Modelo SeCoM [Bulcão Neto, 2006]**

A descrição semântica das principais ontologias que compõem o modelo SeCoM, bem como sua representação gráfica, são descritas a seguir. Esta representação gráfica foi feita utilizando-se o plugin ezOWL, disponível no editor Protégé.

### 2.3.2.1 Ontologia *Time*

A ontologia *Time* representa um conjunto de elementos que apresentam extensão temporal na forma de instantes ou intervalos de tempo, bem como as possíveis relações temporais que podem existir entre esses elementos. Esta ontologia foi construída baseada nas seguintes diretrizes de projeto:

1. Utilização de intervalos de tempo como primitivas do modelo;
2. Representação de relações temporais entre entidades que possuam um componente temporal, como as relações de passado, passado imediato, presente, futuro imediato e futuro;
3. Múltiplas resoluções de representação de informação temporal, como horas, dias da semana, meses e anos;
4. Associação de informação temporal a conceitos do modelo SeCoM, o que permite não apenas inferências sobre os inter-relacionamentos temporais entre entidades de um ambiente de computação sensível a contexto, mas também o registro do histórico de mudanças em informações de contexto relacionadas a uma entidade em particular.

A Figura 2 ilustra a ontologia *Time* e suas principais classes, atributos e relações. A principal classe desta ontologia é a *TemporalThing*, que descreve qualquer tipo de entidade que contenha uma extensão temporal, quer seja uma extensão por instante de tempo (classe *InstantThing*), quer seja uma extensão por intervalo de tempo (classe *IntervalThing*). Daí, a classe *TemporalThing* ser formada pela união (owl:*unionOf*) das classes *InstantThing* e *IntervalThing*. Dessa forma, é possível representar a partir do modelo SeCoM quaisquer entidades que possuam uma extensão temporal, quer seja um instante de tempo, quer seja um intervalo de tempo.

Duas relações temporais básicas são possíveis entre objetos da classe *TemporalThing*: a relação temporal *before* e sua relação inversa (owl:*inverseOf*) *after*. Por meio dessas relações é possível determinar, em sua forma mais básica, a ordem temporal de entidades que contenham uma extensão temporal.

Objetos da classe *TemporalThing* contêm atributos que delimitam seu início e término temporais: os atributos *beginPointOf* e *endPointOf*, respectivamente. Os valores que esses atributos assumem são objetos da classe *InstantThing*. Objetos da classe *IntervalThing* possuem valores diferentes para os atributos *beginPointOf* e *endPointOf*; para objetos da classe *InstantThing*, tais valores são idênticos.

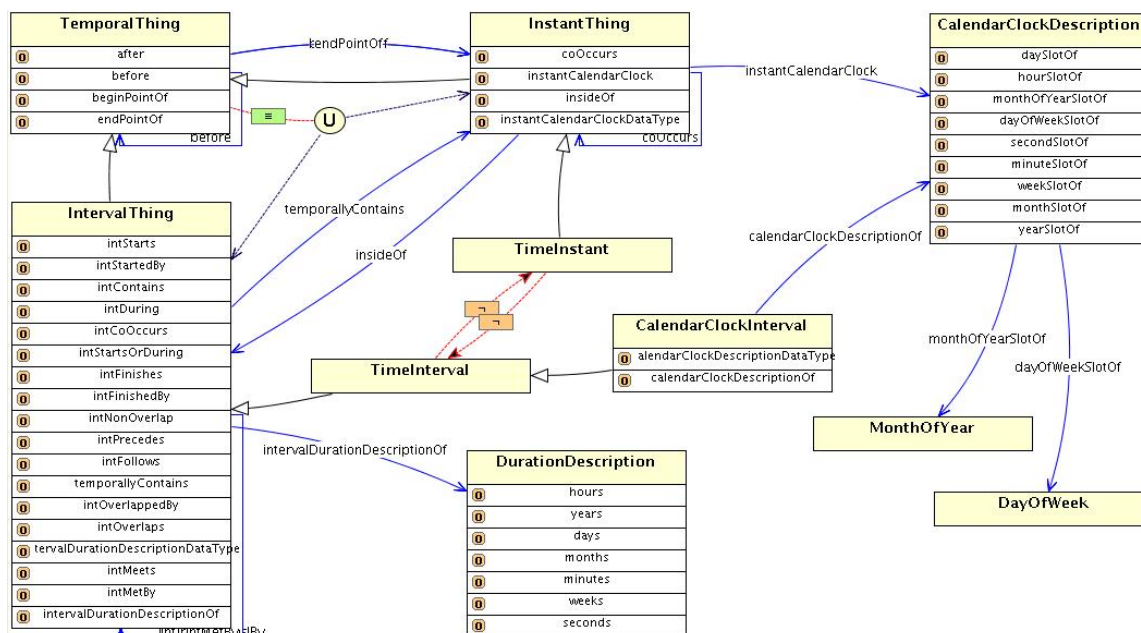


Figura 2 - Ontologia Time [Bulcão Neto, 2006].

Quanto à classe *InstantThing*, a relação temporal *coOccurs* descreve a ocorrência simultânea de objetos dessa classe. A relação temporal *insideOf* descreve que objeto(s) da classe *InstantThing* pode(m) ser parte de um objeto da classe *IntervalThing*. De forma similar, a relação temporal *temporallyContains*, inversa de *insideOf*, descreve que um objeto da classe *IntervalThing* engloba um ou mais objetos da classe *InstantThing*. Dessa forma, é possível descrever relações temporais entre entidades cuja extensão temporal são representadas por instantes e intervalos de tempo.

O conjunto das 13 relações temporais apresentadas atende à diretriz 2 do projeto da ontologia Time. A relação *intPrecedes*, por exemplo, descreve que um intervalo de tempo A ocorreu antes, porém não imediatamente antes, em relação a um intervalo de tempo B (passado); *intMeets* descreve que um intervalo de tempo B ocorreu imediatamente após um intervalo de tempo A (passado imediato); *intCoOccurs* descreve a simultaneidade de dois intervalos de tempo (presente), assim como a relação *intContains*, embora seus respectivos limites temporais (*beginPointOf* e *endPointOf*) sejam diferentes; *intMetBy* descreve a relação inversa de *intMeets*, ou seja, futuro imediato com respeito a dois intervalos de

tempo; por fim, *intFollows* descreve a relação inversa de *intPrecedes*, ou seja, futuro não imediato entre dois intervalos de tempo.

Existem ainda duas relações temporais entre objetos da classe *IntervalThing*. A relação *intNonOverlap* descreve objetos que nunca se sobrepõem no tempo; esta relação tem como sub-propriedades (*rdfs:subPropertyOf*) as relações *intPrecedes* e *intMeets*. A relação *intStartsOrDuring* descreve objetos que não apresentam intersecção temporal nos seus respectivos tempos de término (*endPointOf*). Ambas relações temporais são importantes para verificar se entidades de extensão temporal intervalar, como uma reunião e uma palestra, podem se sobrepor no tempo.

A ontologia *Time* representa instantes de tempo por meio da classe *TimeInstant*, que é subclasse (*rdfs:subClassOf*) da classe *InstantThing*. Um instante de tempo é definido neste trabalho como um ponto inextensível na linha do tempo, ou seja, não faz sentido descrever a duração de um instante de tempo. Por outro lado, intervalos de tempo (diretriz 1 de projeto) são representados na ontologia *Time* por meio da classe *TimeInterval* que, por sua vez, é disjunta (*owl:disjointWith*) da classe *TimeInstant*. Ou seja, não há objetos que representem instantes e intervalos de tempo simultaneamente.

Por serem respectivamente subclasses de *InstantThing* e *IntervalThing*, as classes *TimeInstant* e *TimeInterval* herdam todos os atributos e relações definidos para essas classes. A separação entre elementos que possuem extensão temporal e os próprios instantes e intervalos de tempo permite que outros elementos possam ser definidos a partir da extensão temporal que apresentam, tais como a descrição de eventos temporais realizada na ontologia *Temporal Event*, descrita na seção seguinte, o que permite assim atender à diretriz 4 do projeto da ontologia *Time*.

Atendendo a sua diretriz 3 do projeto, a ontologia *Time* fornece mecanismos que permitem representar instantes e intervalos de tempo sob múltiplas formas. Para descrever um instante de tempo específico, é possível utilizar o atributo *instantCalendarClockDataType*, que reusa a notação do tipo de dado data e hora (*xsd:dateTime*) do padrão XML Schema. É possível também utilizar a relação

*instantCalendarClock*, que descreve data e hora de forma particionada por meio de atributos da classe *CalendarClockDescription*. Esses atributos permitem descrever ano, semestre, mês do ano, mês, dia da semana, semana, dia, hora, minuto e segundo, todos conforme as respectivas notações de tipos de dados do padrão XML Schema. As classes *MonthOfYear* e *DayOfWeek* são enumerações (owl:oneOf) com objetos que representam, respectivamente, cada mês do ano e dia da semana.

Já para descrever a duração de intervalos de tempo, é possível utilizar a notação do tipo de dado duração temporal (xsd:duration) do padrão XML Schema, notação essa aceita pelo atributo *intervalDurationDescriptionDataType*. É possível também descrever a duração de intervalos temporais de maneira particionada, segundo a unidade temporal que se deseja através da relação *intervalDurationDescriptionOf*. Esta relação conecta-se à classe *DurationDescription* que contém atributos para descrever duração temporal em termos de anos, semestres, meses, dias, semanas, horas, minutos e segundos, todos em conformidade com as respectivas notações de tipos de dados do XML Schema.

### **2.3.2.2 Ontologia *Temporal Event***

A ontologia *Temporal Event*, apresentada na Figura 3, representa eventos que possuem alguma extensão temporal. Esta ontologia foi construída para complementar as informações descritas na ontologia *Time*. Desse modo, vários dos conceitos modelados na ontologia *Time* são reutilizados na ontologia *Temporal Event*.

Conforme observado na Figura 3, a principal classe representada é a *TemporalEvent*, a qual modela eventos com extensão temporal, e é subclasse de *time:TemporalThing*, onde “time:” representa o espaço de nomes associado à ontologia *Time*.

A classe *TemporalEvent* é composta pela união de dois tipos de eventos temporais: aqueles descritos por instantes de tempo, caso da classe *InstantEvent*, e os eventos descritos por intervalos de tempo, que são representados pela classe *IntervalEvent*.

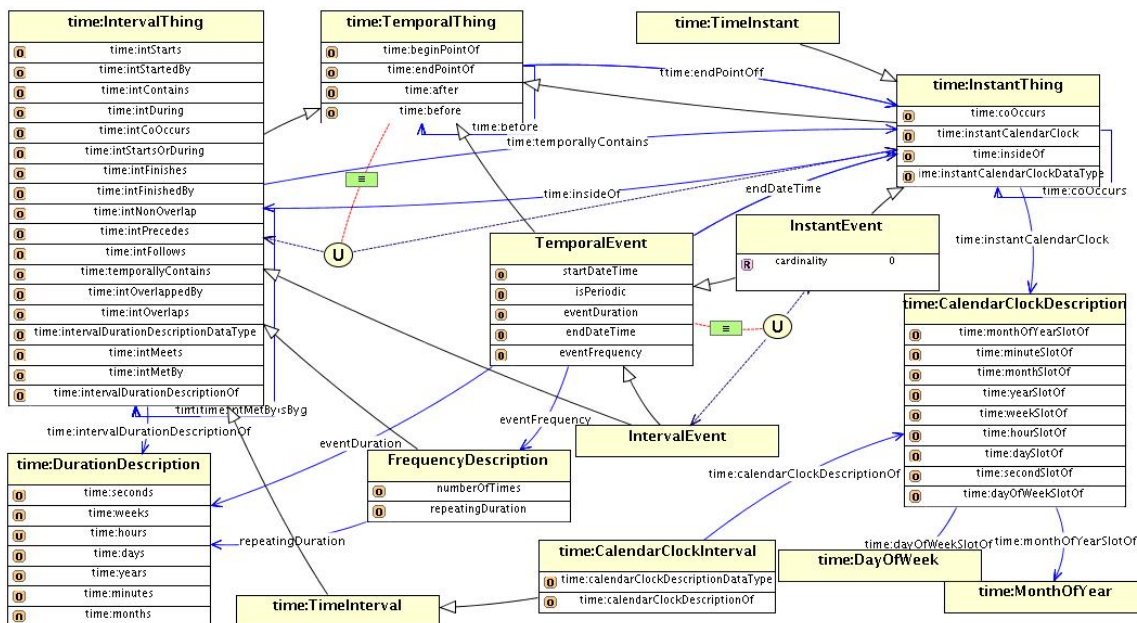


Figura 3 - Ontologia Temporal Event [Bulcão Neto, 2006].

Além de todos os atributos e relações temporais herdados de `time:TemporalThing`, a classe `TemporalEvent` apresenta os seguintes atributos: `startDateTime` e `endDateTime`, que representam, respectivamente, os instantes temporais de início e término de um objeto da classe de eventos temporais; `eventDuration` que modela a duração de um evento temporal com valor oriundo da classe `time:DurationDescription`; `isPeriodic` cujo valor booleano serve para indicar se um evento temporal segue uma periodicidade ou não; e `eventFrequency` que representa a frequência que um evento temporal segue.

A frequência temporal de eventos é expressa por meio da classe `FrequencyDescription`. O atributo `numberOfTimes` expressa o número de vezes que um evento é repetido a cada intervalo de tempo. Seu respectivo valor segue a notação de números inteiros não-negativos (`xsd:nonNegativeInteger`) do XML Schema. Já o atributo `repeatingDuration` descreve a duração do intervalo de tempo no qual um evento é repetido. Por exemplo, combinando esses dois atributos, é possível representar a periodicidade de eventos que se repetem duas vezes por semana ou uma vez por mês.

As classes `InstantEvent` e `IntervalEvent` herdam todos os atributos e relações temporais das classes `time:InstantThing` e `time:IntervalThing`,

respectivamente, uma vez que são subclasses destas. Ou seja, é possível relacionar objetos da classe *InstantEvent* a um objeto de *IntervalEvent* por meio da relação temporal *time:temporallyContains*. O mesmo se aplica a objetos da classe *IntervalEvent* que podem ser ordenados temporalmente entre si por meio das relações temporais descritas na ontologia *Time*.

### 2.3.2.3 Ontologia *Spatial*

A ontologia *Spatial* foi desenvolvida para representar informações de localização. Ela foi desenvolvida baseada nas seguintes diretrizes de projeto:

1. Representação da localização de entidades em um espaço físico ou virtual;
2. Representação de relações mereológicas entre espaços físicos ou virtuais. Vale ressaltar que as relações mereológicas indicam os relacionamentos entre o todo e as partes que o compõe [Novello, 2003];
3. Representação de relações espaciais entre espaços físicos ou virtuais;
4. Múltiplas representações de informação de localização para espaços físicos;
5. Associação de informação de localização a conceitos do modelo SeCoM, no sentido de permitir inferências sobre relacionamentos espaciais entre entidades de um ambiente de computação sensível a contexto.

A Figura 4 ilustra a ontologia *Spatial* e suas principais classes, atributos e relações. A principal classe desta ontologia é a classe *SpatialThing*, que descreve qualquer tipo de entidade que contenha uma extensão espacial, quer seja uma extensão de espaço físico (classe *PhysicalLocation*), quer seja uma extensão de espaço virtual (classe *VirtualLocation*). Daí, a classe *SpatialThing* ser formada pela união das classes *PhysicalLocation* e *VirtualLocation*. Dessa forma, é possível representar a partir do modelo SeCoM quaisquer entidades que possuam uma extensão espacial, seja um espaço físico (ex: sala de aula) ou virtual (ex: sala de *chat*). Assim, atende-se à diretriz 1 de projeto da ontologia *Spatial*.

A separação entre elementos que possuem extensão espacial e as próprias localidades físicas e virtuais permite que outros elementos possam ser definidos a partir desta ontologia, tais como a descrição de eventos espaciais realizada na

ontologia *Spatial Event*, descrita na seção seguinte, o que permite assim atender à diretriz 5 de projeto da ontologia *Spatial*.

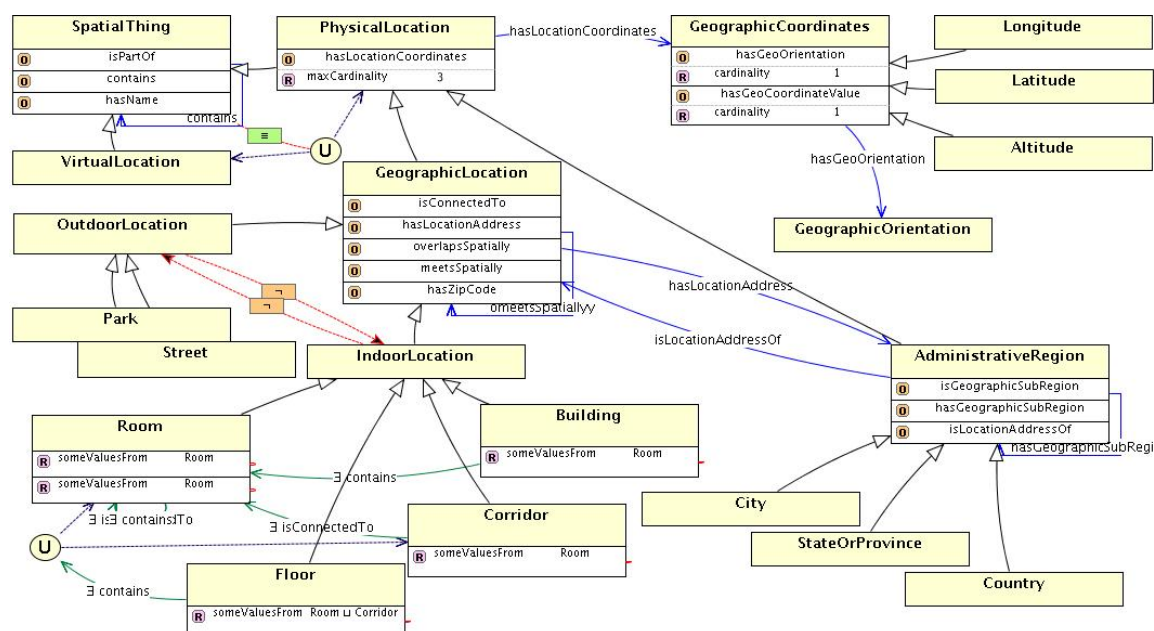


Figura 4 - Ontologia Spatial [Bulcão Neto, 2006].

Duas relações mereológicas são definidas entre objetos da classe *SpatialThing*: a relação *isPartOf* e sua relação inversa *contains*. A relação *isPartOf* é caracterizada como transitiva (*owl:TransitiveProperty*), ou seja, se um objeto A é parte de um objeto B que, por sua vez, é parte de um objeto C, todos objetos da classe *SpatialThing*, então pode-se afirmar que A é parte de C. O mesmo se aplica à relação *contains*. Ou seja, por meio dessas duas relações é possível determinar relações parte-todo entre entidades com extensão espacial, atendendo à diretriz 2 de projeto.

Objetos da classe *SpatialThing* possuem o atributo *hasName* que lhes associam um nome descritivo. O valor que esse atributo assume segue a notação do tipo de dados cadeia de caracteres (*xsd:string*) do padrão XML Schema. Além de herdarem o atributo *hasName* e as relações mereológicas supracitadas, objetos da classe *PhysicalLocation* possuem o atributo *hasLocationCoordinates*, que descreve que cada localização física pode apresentar, no máximo, três coordenadas de localização geográfica obtidas a partir da classe *GeographicCoordinates* e suas subclasses Latitude, Longitude e Altitude. O atributo *hasGeoCoordinateValue* armazena o valor dessas três coordenadas

geográficas segundo a notação do tipo de dados número real (xsd:float) do XML Schema. Coordenadas geográficas baseadas em latitude e longitude necessitam do atributo *hasGeoOrientation* para associar a direção geográfica correspondente (ex: norte, oeste, etc.). A classe *GeographicOrientation* fornece essa informação no formato de uma enumeração de objetos que representam pontos cardeais e colaterais.

A classe *PhysicalLocation* possui duas subclasses: *GeographicLocation* e *AdministrativeRegion*. A classe *GeographicLocation* representa espaços físicos delimitados ou não por uma construção, caso dos objetos das classes *IndoorLocation* e *OutdoorLocation*, respectivamente. A classe *GeographicLocation* fornece as relações espaciais *isConnectedTo*, *overlapsSpatially* e *meetsSpatially* para descrever espaços físicos quanto as suas adjacências e sobreposições. Todas essas relações espaciais são modeladas como simétricas, ou seja, se um espaço físico A é adjacente a um espaço físico B, então este é também adjacente ao espaço físico A. Dessa forma, atende-se à diretriz 3 de projeto da ontologia em questão.

A classe *OutdoorLocation* possui, basicamente, as subclasses *Street* e *Park*, cujos objetos representam ruas e praças, respectivamente. A classe *IndoorLocation* possui quatro subclasses: a que modela estruturas físicas como prédios (classe *Building*), a que representa andares de um prédio (classe *Floor*), a que modela salas (classe *Room*) e a que representa corredores (classe *Corridor*). Existem entre essas quatro classes mencionadas restrições quanto à relação mereológica *contains* e à relação espacial *isConnectedTo*: em alguns casos (owl:someValuesFrom), objetos da classe *Building* contêm objetos da classe *Room* que, por sua vez, também podem conter e estar conectados a objetos da mesma classe. Em alguns casos, objetos da classe *Corridor* estão conectados a objetos da classe *Room*, enquanto que objetos da classe *Floor* eventualmente contêm objetos das classes *Room* e *Corridor*.

A classe *AdministrativeRegion*, subclasse de *PhysicalLocation*, representa espaços físicos sob a visão de divisões administrativas, como cidade, estado ou província e país. Estes últimos são representados, respectivamente, pelas



A classe *SpatialEvent* é composta pela união de dois tipos de eventos espaciais: aqueles cuja extensão está na forma de espaços físicos, caso da classe *PhysicalEvent*, e os eventos de extensão espacial virtual, representados pela classe *VirtualEvent*. Ambas as classes herdam os atributos e relações das classes *loc:PhysicalLocation* e *loc:VirtualLocation*, respectivamente, uma vez que são subclasses destas. Com as classes *PhysicalEvent* e *VirtualEvent* é possível representar eventos que ocorrem em espaços físicos, como reuniões e aulas, bem como eventos que ocorrem em espaços virtuais, como o envio de mensagens em uma sala de *chat*.

### 2.3.2.5 Ontologia Actor

A ontologia *Actor* modela o conjunto de entidades, chamadas atores, que podem realizar ações em um ambiente de computação sensível a contexto. Como exemplo destes atores estão pessoas, grupos ou organizações. Na Figura 6 é apresentada a descrição básica das informações sobre os atores. As demais informações são descritas através das ontologias de apoio *Knowledge*, *Relationship*, *Role*, *Contact*, *Document* e *Project*.

Conforme descrito na Figura 6, a principal classe da ontologia é a classe *Actor*, que busca representar de maneira genérica os variados tipos de atores em um ambiente de computação sensível a contexto. A classe *Actor* especifica três tipos de atores, podendo ser estendido segundo as necessidades de uma aplicação: pessoas (classe *Person*), grupo (classe *Group*) e organização (classe *Organization*).

Objetos da classe *Actor* possuem um nome que os designa por meio do atributo *hasName*, de valor *xsd:string* do padrão XML Schema, bem como podem ou não participar de algum grupo por meio da relação *isMemberOf*. Essa relação tem *hasGroupMember* como sua relação inversa, de maneira que a cardinalidade mínima (*owl:minCardinality = 1*) de ambas as relações descreve que um grupo pode conter um ou mais atores e um ator pode ser membro de um ou mais grupos.

A classe *Group* tem *PersonGroup* como subclasse, que descreve um grupo cujos membros são todos (*owl:allValuesFrom*) objetos da classe *Person*. A classe

*Person* representa os atores do tipo pessoa que possuem, basicamente, atributos que descrevem dados pessoais (*hasSurname*, *hasFirstName* e *hasBirthday*) e relações com objetos da classe *Group* que descrevem o criador de grupos (*maker*) e sua relação inversa *made*.

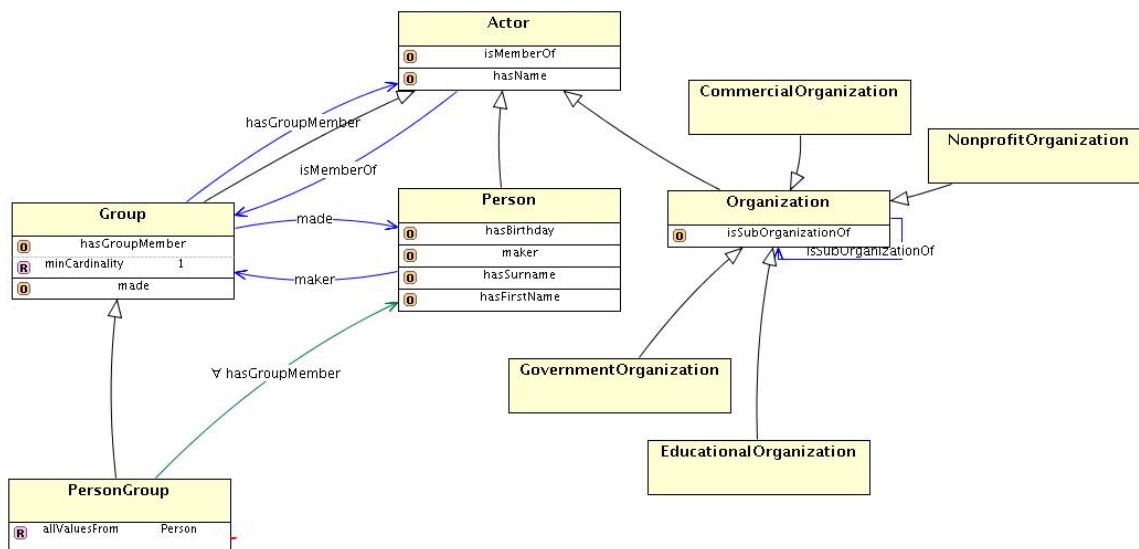


Figura 6 - Ontologia Actor [Bulcão Neto, 2006].

Objetos da classe *Organization* são aqueles atores que representam organizações ou instituições. É possível descrever a hierarquia de organizações entre si por meio da relação transitiva *isSubOrganizationOf*. Sem muitos detalhes, são representados quatro tipos de organizações: comerciais (classe *CommercialOrganization*), governamentais (classe *GovernmentOrganization*), sem fins lucrativos (classe *NonProfitOrganization*) e educacionais (classe *EducationalOrganization*). Definir atributos e/ou relações para esses tipos de organizações, bem como a definição de novos tipos cabe ao desenvolvedor para atender aos seus requisitos específicos de sua aplicação.

### 2.3.2.6 Ontologia *Device*

A ontologia *Device* foi modelada para descrever dispositivos computacionais utilizados para captura e acesso de informações de um ambiente de computação sensível a contexto com o objetivo de promover a adaptação de interfaces com o usuário segundo essas descrições. Para atender a esse objetivo, a ontologia *Device* apresenta as seguintes diretrizes de projeto:

1. Descrição de dispositivos computacionais quanto aos seus componentes de hardware e software;
2. Representação de relações mereológicas entre componentes de dispositivos computacionais;
3. Representação de dispositivos e tecnologias de computação móvel, comuns em cenários de computação sensível a contexto.

As informações modeladas na ontologia *Device* incluem, dentre outras, a descrição dos componentes de hardware, dos componentes de software, do sistema operacional e as características dos dispositivos de rede. Por se tratar de uma ontologia com muitos detalhes semânticos, mas que não será utilizada no estudo de caso apresentado no Capítulo 4, esta ontologia não será detalhada neste trabalho. Informações mais específicas podem ser obtidas no trabalho de Bulcão Neto [2006].

### **2.3.2.7 Ontologia Activity**

A ontologia *Activity* é certamente a mais completa das ontologias descritas no modelo SeCoM. Seu objetivo é descrever ações que atores realizam ou fazem acontecer em um ambiente de computação sensível a contexto. Para atender a esse objetivo, a ontologia *Activity* foi construída baseada nas seguintes diretrizes de projeto:

1. Descrição de atividades quanto ao tempo em que ocorrem;
2. Descrição de atividades quanto ao espaço onde acontecem;
3. Descrição de atividades em função dos atores que as realizam;
4. Descrição de atividades que envolvam atores e dispositivos computacionais;
5. Diferenciação entre atividades que ocorrem espontaneamente daquelas que ocorrem de maneira prevista.

Conforme pode-se observar nas diretrizes de projeto, várias das informações necessárias para a construção da ontologia *Activity* já foram modeladas nas outras ontologias do modelo SeCoM. Diante disso, para o reaproveitamento das informações, a ontologia *Activity*, apresentada na Figura 7, importa as principais ontologias do modelo SeCoM: *Actor*, *Device*, *TemporalEvent*

e *SpatialEvent*. A importação de *TemporalEvent* e *SpatialEvent*, leva indiretamente, a importação das ontologias *Time* e *Event*.

Conforme observado na Figura 7, a principal classe descrita é a classe *Activity*, que descreve atividades como eventos espaço-temporais que, por sua vez, são descritos por meio da classe *SpatioTemporalEvent*. Esta classe representa a união dos objetos pertencentes às classes *tEve:TemporalEvent* e *sEve:SpatialEvent*, onde “tEve:” e “sEve:” são, respectivamente, os espaços de nomes das ontologias *TemporalEvent* e *SpatialEvent* importadas pela ontologia *Activity*. A classe *SpatioTemporalEvent* é subclasse de *SpatioTemporalThing*, que descreve quaisquer entidades que contenham extensões espaciais (*loc:SpatialThing*) e temporais (*time:TemporalThing*) simultaneamente.

Como atividades são descritas como eventos espaço-temporais, estas podem valer-se dos mesmos atributos e relações válidos para eventos espaciais e para eventos temporais. Dessa maneira, é possível, dentre outros, situar atividades quanto a relações espaciais e mereológicas entre espaços físicos virtuais ou reais, ou mesmo situar atividades quanto a instantes e intervalos temporais que descrevam passado, presente e futuro. Assim, as diretrizes 1 e 2 do projeto da ontologia são atendidas.

Todo objeto da classe *Activity* possui uma lista geral de atores (relação *hasParticipant*) que participam da atividade em questão. Como na maioria dos casos os atores de uma atividade são pessoas, foi criada uma restrição de valor (*owl:someValuesFrom*) à relação *hasParticipant* para aceitar como válidos objetos da classe *act:Person*. A relação *isEngagedIn*, inversa de *hasParticipant*, permite descrever que atores estão engajados na realização de uma determinada atividade. Esses conceitos permitem que se atenda à diretriz 3 de projeto da ontologia *Activity*.

Atividades podem ser descritas textualmente por meio dos atributos *hasLongDescription*, para longas descrições, e *hasSummary*, para descrições breves. Para situações em que atividades tenham um prazo de validade associado (ex: uma conferência), foi criado o atributo booleano *validationStatus*. A existência de periodicidade de uma atividade é dada pelo atributo *isPeriodicActivity*, que

quando assume valor booleano verdadeiro faz sentido descrever a freqüência (classe *tEve:FrequencyDescription*) com que essa atividade ocorre por meio da relação *activityFrequency*.

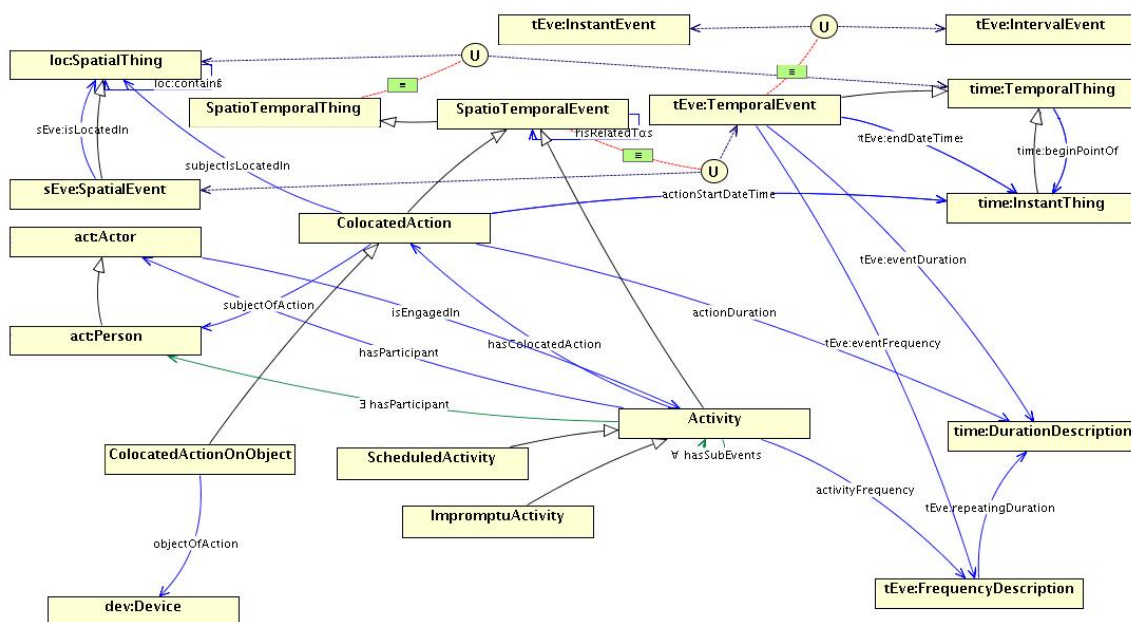


Figura 7 - Ontologia Activity [Bulcão Neto, 2006].

Para descrever atividades que incluam ações específicas de uma pessoa foi criada a classe *ColocatedAction*, que possui os seguintes atributos e relações: *actionStartDateTime*, *actionEndDateTime* e *actionDuration*, para descrever os instantes de tempo em que a ação inicia, termina e sua respectiva duração temporal; e *subjectsIsLocatedIn* que descreve a localização da pessoa (*subjectOfAction*) que realiza a ação em questão. Com esses conceitos é possível descrever, por exemplo, que “Carlos tem estado na sala D-5 por duas horas no contexto de uma reunião”.

De maneira complementar à classe *ColocatedAction*, foi criada sua subclasse *ColocatedActionOnObject* para descrever ações específicas de uma pessoa utilizando um determinado dispositivo computacional. A relação *objectOfAction* é responsável por descrever esse relacionamento entre uma ação e um objeto dessa ação. Dessa maneira é possível descrever, por exemplo, que “Carlos começou a utilizar a lousa eletrônica da sala F-3 às 19:00 de 31/08/2008

no contexto de uma aula”. Com esses conceitos, a diretriz 4 de projeto da ontologia *Activity* é atendida.

No intuito de atender à diretriz 5 de projeto da ontologia *Activity*, foi elaborada o atributo *schedulingStatus*, cujo valor booleano permite classificar uma atividade como espontânea (classe *ImpromptuActivity*) ou agendada (classe *ScheduledActivity*). Essas classes disjuntas permitem descrever atividades que ocorrem de maneira informal, como reuniões em *cocktails*, bem como aquelas atividades planejadas quanto ao espaço e tempo em que devem ocorrer, como palestras de uma conferência.

A complexidade da ontologia *Activity* é devida principalmente ao fato de herdar dados das diversas outras ontologias do modelo SeCoM. Porém, esta característica garante uma modelagem adequada de qualquer atividade que ocorra em um determinado espaço ou intervalo de tempo, considerando-se um ambiente sensível ao contexto.

## **2.4 Web Services**

*Web Services* são aplicações modulares e auto-descritas, distribuídas através da Internet, que permitem a troca de informações em ambientes eminentemente heterogêneos [Araújo, 2005]. Esta troca de informações é possível, pois os *Web Services* são desenvolvidos em uma estrutura que propõe a exposição das transações e das regras de negócios por meio de protocolos que podem ser acessados e traduzidos por qualquer linguagem de programação, em qualquer sistema operacional, rodando em qualquer dispositivo [Costa, 2002].

A arquitetura sobre a qual se baseia a construção e o funcionamento dos *Web Services* é apresentada na Figura 8. Os elementos envolvidos nesta arquitetura são [W3C, 2008a]: (1) Fornecedor do Serviço: publica em um catálogo de serviços a descrição dos *Web Services* que ele oferece; (2) O Catálogo de Serviços é um repositório que armazena as informações sobre os *Web Services* disponíveis, incluindo suas descrições e como acessá-los; (3) Cliente do Serviço: inicialmente consulta algum *Web Service* no catálogo de serviços. Se o *Web Service* for encontrado é feita uma conexão direta com o fornecedor do serviço.



**Figura 8 - Arquitetura dos Web Services (Adaptado de [Araújo, 2005]).**

A comunicação entre os elementos apresentados na Figura 8 é feita por meio de um conjunto de tecnologias e protocolos abertos, todos padronizados pela W3C [Brandão e Ribeiro, 2007]. Entre eles estão: XML (Extensible Markup Language), SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language) e UDDI (Universal Description, Discovery and Integration).

XML é uma linguagem, baseada em padrões definidos pela W3C, que permite a descrição e estruturação de dados [W3C, 2008b]. A XML constitui a base para a comunicação entre os *Web Services*. Como esta linguagem tem como característica o uso de padrões abertos e independente de plataforma, vem daí a interoperabilidade apresentada pelos *Web Services*.

SOAP é um protocolo que permite a troca de mensagens em ambientes descentralizados e distribuídos. É uma das soluções para a invocação e a comunicação entre serviços, realizando o transporte de documentos XML por meio de protocolos de Internet como HTTP (*Hypertext Transfer Protocol*) [W3C, 2008c]. Neste ambiente, as mensagens encapsulam informações que são transmitidas entre *Web Services* e especificam a operação que será invocada. Uma mensagem é um arquivo XML com formato padronizado.

WSDL é uma linguagem baseada em XML que permite a especificação dos serviços disponibilizados por um *Web Service*. Um documento WSDL especifica as funcionalidades do serviço, a localização do serviço na Web e as instruções para acessá-lo [W3C 2008c]. Enquanto o SOAP trata da comunicação entre o

cliente e o fornecedor de serviços, o WSDL serve para descrever os serviços oferecidos.

UDDI é um padrão desenvolvido para permitir o registro e a consulta a *Web Services* existentes em uma rede. As informações sobre os *Web Services* existentes são descritas em WSDL e disponibilizadas em forma de um diretório de serviços, os quais podem ser acessados por outras aplicações [W3C, 2008d]. O catálogo de serviços é o componente responsável por gerenciar as informações sobre os *Web Services* disponíveis.

A evolução das aplicações da Web Semântica fez com que surgisse uma nova linguagem chamada OWL-S, a qual permite adicionar semântica à descrição e composição de um *Web Service* [Martin, 2004]. Esta linguagem baseia-se em três módulos: (1) o *Profile* que descreve as funcionalidades e características do *Web Services*; (2) o *Model* que especifica o protocolo de interação entre o fornecedor e o requisitante; e o (3) *Grounding* que descreve como o *Web Service* pode ser acessado.

#### **2.4.1 *Web Services* na Informação Geográfica (Geo *Web Services*)**

A consolidação dos *Web Services* como ferramentas que permitem a distribuição de serviços e dados em ambientes heterogêneos através da Internet trouxe grandes expectativas à comunidade de usuários e desenvolvedores de SIG. Porém, a representação e distribuição de dados geo-espaciais não seguem os mesmos padrões de dados alfanuméricos, os quais podem utilizar a estrutura já definida para os *Web Services*, conforme descrito na seção 2.4.

Visando a interação entre dados geo-espaciais, foi criado o Open GIS Consortium (OGC), que é um consórcio internacional composto por diversas companhias, agências governamentais e universidades cujos membros trabalham num processo de consenso colaborativo para prover e aumentar a interoperabilidade entre tecnologias envolvendo informações e localizações espaciais [OGC, 2008a]. O OGC tem como missão prover uma interface de compartilhamento de informações espaciais e codificar especificações abertas e publicá-las para que fiquem disponíveis para uso global.

Para o desenvolvimento de serviços que promovam a interoperabilidade entre fontes de informação e tecnologias foi criado o documento OpenGIS Reference Model (ORM) [OGC, 2008b]. Este documento apresenta as linhas orientadoras para a criação de uma plataforma de trabalho, sobre a qual empresas ou algum outro órgão, possam desenvolver aplicações que valorizem a interoperabilidade da informação geográfica [Araújo, 2005].

Numa tentativa de criar um modelo conceitual que defina um conjunto de serviços que abranja todas as áreas de SIG, o consórcio OpenGIS sugere a criação de um conjunto de *Web Services*, através dos quais os clientes trocam informação geográfica, de acordo com formatos e protocolos bem definidos [Araújo, 2005]. Dentre os *Web Services* definidos estão [Queiróz, 2007]:

- WMS (Web Map Service): Especifica os serviços para a visualização de mapas através da Web;
- WFS (Web Feature Service): Especifica os serviços para acesso e manipulação de informações associadas aos mapas, que são visualizados através da Web;
- OpenLS (OpenGIS Location Services): Especificação da OGC que define uma plataforma aberta para serviços baseados em localização (LBS). Esta plataforma também é conhecida por GeoMobility Server (GMS);
- WCS (Web Coverage Processing): Web Service cujo objetivo é obtenção e troca de informações espaciais sob a forma de *coverages*. *Coverages* são informações geo-espaciais representando fenômenos que variam no espaço.

Por serem mais difundidos, e certamente mais implementados, serão detalhados nas próximas seções os *Web Services* WMS e WFS.

#### **2.4.1.1 Web Map Service (WMS)**

O Web Map Service (WMS) fornece um serviço para a requisição e visualização de mapas utilizando o protocolo HTTP [OGC, 2008c]. Uma aplicação cliente, que pode ser inclusive um *browser*, faz uma requisição a um WMS indicando as camadas e os parâmetros necessários à visualização de alguma área geográfica.

Dentre estes parâmetros estão a representação do mapa, o tamanho do mesmo e o seu sistema de referência. Como resposta, o WMS retorna o mapa em formato de imagem tais como GIF, JPEG ou PNG.

Na especificação do serviço foram definidas três operações [OGC, 2008c]:

- GetCapabilities: provê metadados sobre o serviço, oferecendo informações sobre os valores de parâmetros aceitáveis para as demais operações;
- GetMap: retorna um mapa para o cliente, com sistema de referência, tamanho, formato e transparências especificados;
- GetFeatureInfo: operação opcional que retorna informações adicionais sobre *pixels* determinados pelo cliente, por meio de um mapa recuperado previamente.

#### **2.4.1.2 Web Feature Service (WFS)**

O Web Feature Service (WFS) fornece um serviço para manipulação das informações associadas aos dados geo-espaciais através da Internet [OGC, 2008d]. Isto inclui funcionalidades para a execução de consultas, inserções, atualizações e exclusões de feições geográficas [Brandão e Ribeiro, 2007].

Uma diferença fundamental no funcionamento do WMS e do WFS é em relação ao formato dos dados transportados. Enquanto no WMS os dados são transportados como uma imagem gráfica, no WFS os dados são transportados no formato GML (Geography Markup Language). GML é uma derivação do XML para a representação de dados geográficos, sendo também uma padronização da OGC [OGC, 2008d].

As operações definidas para o serviço WFS são [Brandão e Ribeiro, 2007]:

- GetCapabilities: retorna um documento que descreve os tipos de feições suportados, bem como as operações permitidas por cada um destes tipos;
- DescribeFeatureType: retorna a descrição de um determinado tipo de feição, suportado pelo WFS;
- GetFeature: é responsável por retornar instâncias de dados oferecidos pelo serviço, representadas por documentos no formato GML.

Além destas operações, as quais definem um WFS básico, existem ainda as seguintes operações, que definem um WFS transacional:

- Transaction: permitem a realização de operações de inserção, atualização e exclusão sobre os dados mantidos sob o WFS;
- LockFeature: é uma operação opcional, utilizada para bloquear um ou mais itens de dados na utilização de um WFS transacional, permitindo a serialização.

## 2.5 Trabalhos Relacionados

O desenvolvimento de aplicações de SIG Móveis sensíveis ao contexto vem sendo pesquisado por diversos grupos. A seguir são descritos alguns dos trabalhos que contribuíram para se chegar à arquitetura proposta nesta dissertação.

Inicialmente pode-se citar o trabalho de Viegas [2006], o qual sugere um mecanismo baseado em ontologias para intermediar consultas a um banco de dados geográficos. Existem também os trabalhos de Araújo [2005] e o de Silva et al. [2004] os quais propõem o uso de *Web Services* para acessar informações geo-espaciais. Estes trabalhos têm em comum o fato de permitirem o acesso para os usuários via *browser* através de um computador pessoal, o que difere da proposta desta dissertação, na qual o acesso é feito via dispositivos móveis. Ainda assim, estes trabalhos serviram como boas referências sobre como acessar informações geo-espaciais armazenadas remotamente.

Em relação à infra-estrutura para aplicações sensíveis ao contexto, Pessoa et al. [2006] define uma arquitetura para o desenvolvimento de um mecanismo baseado em ontologias para a realização de diagnósticos personalizados na área de saúde. Já o trabalho de Bulcão Neto [2006], define desde um modelo de contexto (SeCoM) até um processo de software para o desenvolvimento de aplicações sensíveis ao contexto. A reutilização deste modelo de contexto é inclusive sugerida na arquitetura proposta nesta dissertação.

Li et al. [2006] especificam um mecanismo baseado em *Web Services* para exibir informações contextuais a usuários de dispositivos móveis. As informações

contextuais são definidas através da localização do usuário, via GPS, das informações do hardware e das informações dos softwares utilizados. Esta definição de contexto não considera informações do perfil do usuário ou mesmo trata dos dados geo-espaciais.

Brisaboa et al. [2007] especifica um *middleware* para intermediar as consultas a um banco de dados geográficos a partir de um dispositivo móvel. O acesso às informações é feito por meio de *Geo Web Services* seguindo os padrões recomendados pela OGC. Este trabalho apresenta uma boa referência de como acessar e gerenciar dados geo-espaciais a partir de dispositivos móveis, porém não leva em consideração o contexto do usuário para exibir as informações.

Weibenberg et al. [2006] descreve a plataforma FLAME2008, desenvolvida para ser utilizada nas Olimpíadas de Pequim. Esta plataforma se caracteriza como uma solução para entrega de informações personalizadas a usuários de dispositivos móveis. Este trabalho utiliza um conjunto de ontologias para definir a situação do usuário, que se caracteriza por tentar inferir as atividades que usuários possam realizar durante um intervalo de tempo. A partir da definição da situação do usuário são exibidas, via *Web Services*, as informações que sejam mais adequadas para aquele momento. Apesar de ser um sistema bastante amplo, este trabalho não descreve as implicações de se integrar as informações de contexto em um ambiente de SIG Móvel.

Outro trabalho bastante interessante é o projeto MyCampus [Sadeh et al., 2002], no qual foi desenvolvido uma estrutura baseada em *Web Services* e ontologias para que os estudantes da Universidade de Carnegie Mellon (Estados Unidos), obtenham em seu PDA um conjunto de informações baseadas em seu contexto dentro do campus. Este trabalho, apesar de utilizar algumas informações sobre dados geo-espaciais, não detalha especificamente os desafios nem as ferramentas necessárias para se construir aplicações que manipulem este tipo de dado.

Pode-se citar ainda o desenvolvimento da arquitetura CoMPASS (*Combining Mobile Personalized Applications with Spatial Services*) [Weakliam,

2005] a qual utiliza, além das informações de perfil e localização, a trajetória do usuário para automaticamente entregar informações personalizadas a um cliente de SIG Móvel. A definição e o armazenamento das informações de contexto baseiam-se na utilização de um conjunto de arquivos de *log*. Esta abordagem difere da solução proposta neste trabalho, a qual sugere o uso de ontologias para definição das informações de contexto e o uso de um banco de dados geográficos para armazenamento dos dados.

Considerando as idéias e soluções apresentadas nos trabalhos descritos acima, é que foi proposta uma arquitetura para o desenvolvimento de aplicações de SIG Móvel sensíveis ao contexto, a qual é descrita em detalhes no capítulo a seguir. A arquitetura proposta nesta dissertação difere dos outros trabalhos ao propor a reutilização de um modelo de contexto independente de domínio, integrado ao gerenciamento de dados geo-espaciais. Esta característica permite que se possa desenvolver um SIG Móvel capaz de gerenciar informações de contexto nas mais diversas áreas de aplicação.

### **3 A Arquitetura Context-Mobile-GIS (CM-GIS)**

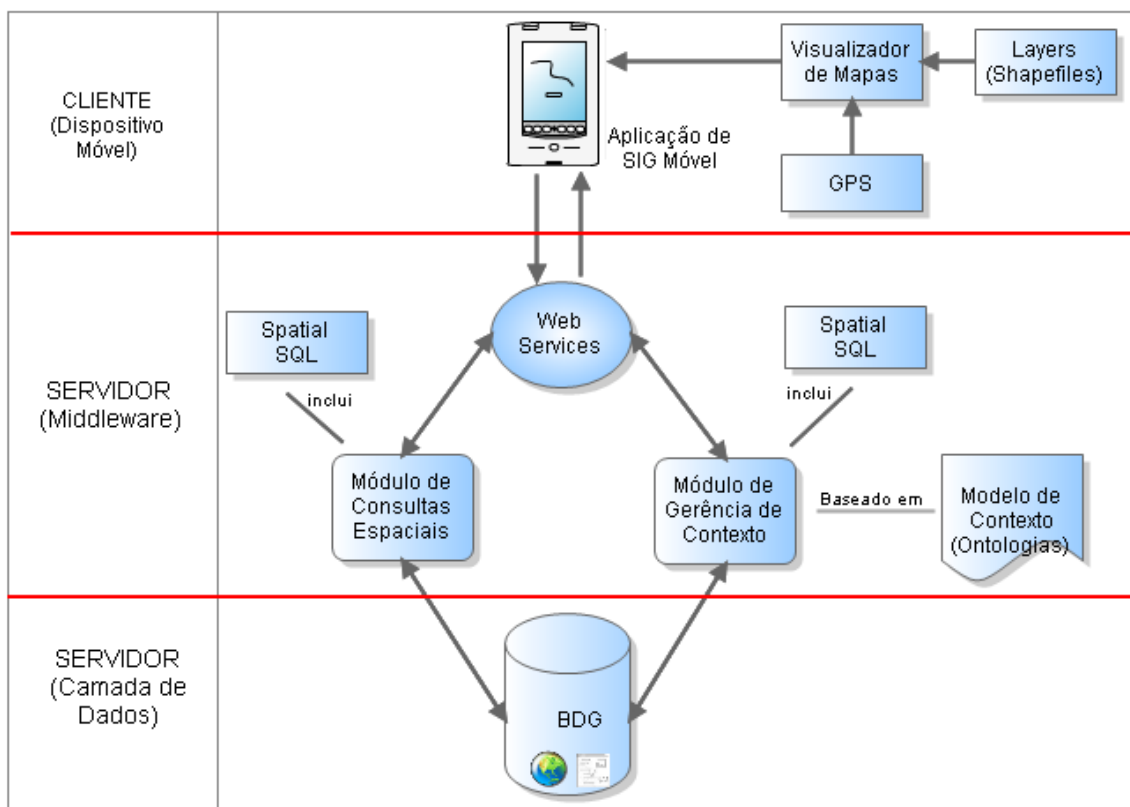
Neste Capítulo é apresentada a arquitetura proposta nesta dissertação, que consiste em um modelo para o desenvolvimento de aplicações de SIG Móveis capazes de gerenciar informações de contexto. A arquitetura, batizada de Context-Mobile-GIS (CM-GIS), tem como objetivo a construção de SIG Móveis integrados a Aplicações Sensíveis ao Contexto. Para que esta arquitetura seja viável, é necessário que se apresentem soluções para estas duas áreas de aplicação.

No caso dos SIG Móveis, conforme definido por Tomko [2003], o desenvolvimento de aplicações requer atenção aos seguintes elementos: (a) armazenamento e consulta a dados em um banco de dados geográficos; (b) prover um mecanismo para realização de consultas e acesso a dados localizados remotamente; (c) a correta utilização de tecnologias.

Com relação ao banco de dados geográficos (a) deve-se utilizar algum Sistema de Gerenciamento de Banco de Dados (SGBD) que tenha capacidade para armazenar e gerenciar dados geo-espaciais. Alguns SGBD disponíveis atualmente disponibilizam esta funcionalidade através de uma extensão para dados espaciais, como é o caso do Oracle, do IBM DB2, do PostGreSQL, dentre outros. Para tratar do acesso remoto às informações (b), uma solução bastante adotada é o uso de *Web Services*, devido à sua capacidade de disponibilizar serviços que podem ser acessados dinamicamente através de uma rede. Por fim, com relação às tecnologias adotadas (c), é importante que se escolha uma ferramenta capaz de manipular e exibir dados geo-espaciais, tanto em sua forma gráfica quanto descritiva, nos dispositivos móveis.

Já no caso das aplicações sensíveis ao contexto é importante que se defina um mecanismo capaz de representar e manipular o contexto envolvido no domínio que está sendo tratado pela aplicação. Conforme descrito no Capítulo 2, esta tarefa pode ser feita utilizando-se ontologias, uma vez que suas características de formalidade, semântica explícita e abstração de implementação possibilitam o compartilhamento e a inferência de novas informações.

Diante das considerações feitas foi proposta a arquitetura CM-GIS para o desenvolvimento de aplicações de SIG Móveis sensíveis ao contexto, conforme apresentado na Figura 9.



**Figura 9 - Arquitetura proposta para um SIG Móvel sensível ao Contexto.**

Esta arquitetura prevê que para o desenvolvimento de aplicações de SIG Móveis capazes de gerenciar informações de contexto são necessários os seguintes elementos:

(1) a aplicação de SIG Móvel, a qual será a interface para o cliente a partir de seu dispositivo móvel. Um requisito importante para esta aplicação é que ela tenha capacidade de manipular e exibir dados geo-espaciais;

(2) um conjunto de *Web Services* para gerenciar a comunicação e o fornecimento de serviços e informações ao usuário. A utilização de *Web Services* tem uma grande importância nesta arquitetura, ao permitir que operações que demandem uma alta carga de processamento possam ser realizadas em máquinas mais robustas e não diretamente no dispositivo móvel;

(3) um módulo (Módulo de Consultas Espaciais) que se comunica com os *Web Services* e realiza consultas espaciais, através de uma versão estendida da linguagem SQL, sobre as informações armazenadas no banco de dados geográficos;

(4) um conjunto de ontologias que servem para modelar o contexto envolvido na aplicação. É sugerido para a modelagem do contexto o uso do modelo SeCoM apresentado no Capítulo 2;

(5) um módulo (Módulo de Gerência de Contexto) definido a partir do modelo de contexto baseado em ontologias que se comunica com os *Web Services* para fornecer informações de contexto. As informações de contexto também devem estar armazenadas no banco de dados e consultadas via SQL; e

(6) um Banco de Dados Geográficos que gerencia todos os dados geo-espaciais e descritivos envolvidos na aplicação.

Cada um destes itens é detalhado nas seções seguintes.

### **3.1 A definição do modelo de contexto baseado em ontologias**

A definição do modelo de contexto nesta arquitetura é importante para que se especifique de uma maneira formal, quais informações serão entregues de forma personalizada ao usuário do SIG Móvel. Existe uma grande variedade de informações que podem ser modeladas, variando de acordo com o domínio que será tratado pela aplicação.

É importante que o modelo de contexto definido para a aplicação represente, da forma mais completa possível, o conjunto de informações que possam ser manipuladas. No processo de especificação deste modelo é sugerida a utilização do modelo SeCoM, uma vez que ele foi definido de forma genérica, e engloba um conjunto bastante amplo de informações representadas através de ontologias.

A utilização de ontologias para modelar o contexto da aplicação desenvolvida permite que se descreva as informações de uma maneira bem mais completa do que utilizando modelos E-R ou diagramas UML. Muitas restrições, relacionamentos ou características podem ser melhor definidas utilizando-se o

poder de expressão definido pelas ontologias descritas através da linguagem OWL. Em termos de documentação e compartilhamento das informações definidas pelo sistema, o uso de ontologias se revela mais eficiente que outros modelos, já que este permite a realização de inferências e uma maior formalização das informações.

Este modelo de contexto tem grande importância no desenvolvimento da aplicação, pois serve de base para a modelagem do banco de dados geográficos e para a implementação das funcionalidades disponíveis no Módulo de Gerência de Contexto.

### **3.2 O banco de dados geográficos**

A utilização de um banco de dados geográficos (BDG) em uma estrutura que envolva Sistemas de Informações Geográficas é algo que enriquece muito este tipo de aplicação. Isto é possível graças à capacidade que o BDG tem para armazenar e realizar operações que envolvam tanto dados geo-espaciais quanto dados descritivos. Diante desta característica ele se torna um componente fundamental na arquitetura proposta neste trabalho.

Para que o banco de dados geográficos utilizado seja capaz de responder tanto a operações espaciais, quanto a operações que envolvam informações de contexto é fundamental que sua estrutura reflita as informações definidas no modelo de contexto da aplicação de SIG Móvel que esteja sendo desenvolvida. A utilização de algum modelo conceitual específico para banco de dados geográficos, tais como o UML-GeoFrame [Lisboa Filho e Lochpe, 2008], poderá ser de grande auxílio nesta tarefa.

Existem diversas extensões de SGBD para tratar dados geográficos no formato vetorial, sendo algumas proprietárias e outras disponíveis como software livre, conforme descrito na Seção 2.1.2. A decisão de qual destes SGBD utilizar fica a cargo do desenvolvedor da estrutura para o SIG Móvel.

### **3.3 Os Web Services**

Os *Web Services* definidos nesta arquitetura têm como função receber os dados enviados pelo usuário, repassar estes dados ao Módulo de Consultas Espaciais

ou ao Módulo de Gerência de Contexto e, por fim, devolver as respostas aos usuários.

Para cada tipo de informação que possa ser repassada pelo usuário deve haver um *Web Service* adequado, o qual receba os parâmetros corretos e retorne os dados solicitados. Um exemplo de informação a ser repassada é uma identificação do usuário, como login e senha, que será útil para identificar informações relacionadas a seu contexto.

Como esta arquitetura trata do desenvolvimento de aplicações sensíveis ao contexto é importante que, além de um mecanismo para identificar o usuário, o *Web Service* também seja capaz gerenciar uma sessão aberta após sua conexão. Este mecanismo será fundamental para gerenciar o contexto do usuário enquanto ele estiver utilizando a aplicação a partir de seu dispositivo móvel.

Vale ressaltar que é devido à capacidade dos *Web Services* de gerenciar ambientes com informações variadas e distribuídas que sua utilização foi proposta nesta arquitetura. Esta característica dos *Web Services* permite que operações que demandem um grande poder de processamento possam ser acessadas a partir de dispositivos móveis e implementadas em servidores adequados.

### **3.4 O Módulo de Consultas Espaciais**

O Módulo de Consultas Espaciais é o componente responsável por armazenar consultas SQL que serão repassadas ao BDG, a fim de responder a alguma pesquisa feita pelo usuário. As opções de pesquisa disponíveis ao usuário dependerão da aplicação SIG Móvel desenvolvida e poderão envolver, por exemplo, a localização de um lugar específico ou de alguma pessoa.

O fato de se utilizar um BDG nesta arquitetura, permitirá a implementação de opções de consulta mais refinadas, as quais incluam, por exemplo, operações espaciais sobre os dados geográficos, sejam eles pontos, linhas ou polígonos. Dentre as operações espaciais possíveis estão cálculos de distância, raio, proximidade, área, ou qualquer outra que envolva operadores disponíveis em uma versão estendida da SQL para tratar dados geo-espaciais.

A intermediação entre a aplicação de SIG Móvel, na qual o usuário escolhe a consulta desejada, e o Módulo de Consultas Espaciais é feita por um *Web Service*. Para cada consulta deverá haver uma operação no *Web Service* que receba os parâmetros escolhidos ou digitados pelo usuário e os repasse ao Módulo de Consultas Espaciais. O resultado da consulta SQL é retornado ao usuário utilizando-se esta mesma operação.

### **3.5 O Módulo de Gerência de Contexto**

O Módulo de Gerência de Contexto será o componente responsável por exibir ao usuário informações que estejam de acordo com seu contexto. A definição do contexto irá depender da aplicação que esteja sendo desenvolvida e será modelado utilizando um conjunto de ontologias, conforme descrito na Seção 3.1. Alguns exemplos de contexto gerenciados por este Módulo seriam apresentar lugares próximos à localização do usuário ou mesmo exibir informações baseadas em atividades que estiverem previamente agendadas.

A partir da definição do modelo de contexto baseado em ontologias, uma solução que poderia ser adotada para a implementação deste Módulo, seria utilizar os conceitos definidos nas ontologias para modelar o BDG. Dessa forma, a obtenção das informações de contexto seriam feitas diretamente a partir do BDG, sem a necessidade de manipulação das ontologias. Para que isto funcione, as funcionalidades implementadas neste Módulo de Gerência de Contexto deverão prever um mecanismo para enviar consultas SQL ao BDG e retornar as informações para um *Web Service*. Este *Web Service* enviará finalmente as informações de contexto para o usuário do dispositivo móvel.

Outra abordagem seria manipular as ontologias antes de enviar as consultas SQL ao BDG. Isto permitiria a realização de inferências sobre as informações, o que garantiria respostas mais amplas para os resultados esperados pelo usuário.

Considerando esta abordagem de utilização de consultas SQL enviadas a um BDG, a principal diferença no funcionamento do Módulo de Consultas Espaciais para o Módulo de Gerência de Contexto é que o primeiro é acionado a

partir de uma solicitação do usuário, já o segundo é acionado automaticamente pela própria aplicação de SIG Móvel sem a participação espontânea do usuário.

### **3.6 A aplicação SIG Móvel**

A aplicação SIG Móvel utilizada nesta arquitetura será a responsável pela interação direta com o usuário. Um componente fundamental para o desenvolvimento desta aplicação será o uso de um componente capaz de manipular dados geográficos em um dispositivo móvel. Como exemplo destes componentes tem-se o ArcPad [ArcPad, 2008], o MapSuite Pocket PC [MapSuite, 2008] e o SharpMap [SharpMap, 2008].

Outra característica importante desta aplicação é que ela tenha funcionalidades que permitam a conexão com *Web Services*, utilizando uma infraestrutura de redes sem fio. Isto permitirá que todo processamento necessário para consultas a dados ou inferências de contexto seja realizado por servidores remotos e mais robustos. Este fato permite que se contornem as limitações inerentes aos dispositivos móveis em termos de memória, armazenamento e processamento de dados.

A aplicação deverá também conter algum mecanismo que permita a identificação do usuário, o que será de grande importância para identificar algumas das informações de contexto. Isto pode ser feito, por exemplo, através de uma identificação por login e senha, como sugerido anteriormente.

## **4 UFV-GeoMobile: Um estudo de caso**

Neste capítulo é descrito um estudo de caso desenvolvido para validar a arquitetura proposta neste trabalho. A estrutura desenvolvida teve como alvo o campus da Universidade Federal de Viçosa (UFV) e foi chamada de UFV-GeoMobile. Todo o processo de desenvolvimento é descrito nas seções que se seguem.

### **4.1 Cenário da Aplicação**

A Universidade Federal de Viçosa (UFV) é uma instituição pública de ensino superior, cujo principal campus, visto parcialmente na Figura 10, está localizado na cidade de Viçosa-MG. A UFV oferece diversos cursos de graduação e pós-graduação, em diversas áreas do conhecimento. Anualmente recebe um grande volume de pessoas que visitam ou que estabelecem algum vínculo com o seu campus. Dentre estas pessoas estão os novos estudantes (~2000 por ano), participantes de congressos, pesquisadores visitantes, ou simplesmente pessoas que desejam visitar suas dependências.

Além destas pessoas existem também os que já possuem alguma ligação com a Universidade, tais como professores, servidores ou ainda os estudantes egressos. Um ponto em comum entre estes diversos indivíduos é que muitas vezes eles têm dificuldades em identificar lugares ou serviços no campus, dada sua grande extensão territorial e à velocidade com que surgem novas construções e instalações diversas.

Considerando este cenário, verifica-se que o mesmo é ideal para a implementação de um sistema que valide a arquitetura CM-GIS, proposta neste trabalho. Seria interessante que houvesse uma fonte prática e automatizada para o usuário buscar informações sobre o campus. Seria ainda mais adequado que as opções de consulta e a obtenção das informações fossem personalizadas de acordo com o contexto em que o usuário se encontra. Como exemplo, pode-se citar que um visitante provavelmente não está interessado no horário da próxima aula, mas gostaria de informações sobre pontos de serviço (banco, farmácia, etc)

ou mesmo de se informar sobre o local de realização de algum evento que ele esteja interessado.



**Figura 10 - Vista parcial do campus da UFV [Fonte: Divisão de Jornalismo/CCS-UFV].**

Diante desta realidade foi implementado o sistema UFV-GeoMobile, a qual permite que usuários utilizando um aplicativo de SIG Móvel instalado em seu PDA possam realizar consultas ou mesmo receber informações personalizadas sobre o campus da UFV. Este aplicativo é composto por um mapa sobre o campus devidamente gerenciado por um componente para manipulação de dados geográficos. Além disso, o aplicativo possui um conjunto de telas com as opções de consulta e um mecanismo para exibição de informações baseadas no contexto do usuário, que neste caso é representado por uma agenda de atividades.

As informações consultadas pelo aplicativo de SIG Móvel estão armazenadas em um banco de dados geográficos e são acessadas via *Web Services*. Já o mecanismo para definição de contexto do usuário foi definido a partir de uma ontologia de domínio, construída a partir do modelo SeCoM, sobre a estrutura e atividades que ocorrem um campus universitário. Mais especificamente a definição de contexto baseia-se em uma agenda de atividades a serem

realizadas pelo usuário para assim exibir a localização geográfica do local onde será realizada esta atividade.

É importante ressaltar que nesta estrutura qualquer informação obtida pode ser representada direta ou indiretamente através de uma representação geográfica no mapa, a qual pode ser ponto, linha ou polígono. Esta funcionalidade, além de servir como um bom referencial de localização para o usuário, é também uma característica importante considerando-se um ambiente de SIG Móvel.

Todo o processo de (1) definição da ontologia de domínio sobre o campus universitário, (2) da definição do banco de dados geográficos, (3) da definição dos *Web Services*, (4) implementação do Módulo de Consultas Espaciais, (5) implementação do Módulo de Gerência de Contexto e (6) da construção da aplicação SIG Móvel são descritos nas seções a seguir.

#### **4.2 O modelo de contexto: ontologia de um campus universitário**

O modelo de contexto para o estudo de caso UFV-GeoMobile deve conter informações sobre a estrutura e as atividades realizadas em um campus universitário. Este modelo foi definido com base no estatuto da UFV e através das informações modeladas na ontologia OntoUFBA, que é uma ontologia de domínio sobre a estrutura da Universidade Federal da Bahia [Abdalla, 2005]. A ontologia de domínio definida como modelo de contexto foi chamada de OntoUFV. Dentre as informações que devem ser modeladas para refletir este domínio estão:

- As pessoas: neste caso envolvem estudantes, funcionários e visitantes. Os funcionários podem ser professores ou servidores técnico-administrativos;
- A estrutura administrativa: reflete a estrutura hierárquica de uma universidade, incluindo reitoria, centros de ensino, departamentos e órgãos;
- A estrutura acadêmica: envolve atividades relacionadas ao ensino, pesquisa e extensão;

- Agenda de atividades: corresponde às tarefas que alguma pessoa possa realizar no campus. Este item será muito importante para o sistema resolver informações de contexto;
- Representações espaciais: será utilizado para representar os itens que possam ter uma representação geo-espacial no campus, seja ela em formato de ponto, linha ou polígono. Dentre estas informações estão as construções e os serviços instalados em cada uma destas construções.

A partir destas informações parte-se para uma representação formal de um modelo de contexto. A partir da reutilização das ontologias *Actor*, *Activity*, *Time* e *Spatial*, apresentadas no modelo SeCoM, chegou-se ao modelo apresentado na Figura 11.

Nesta ontologia foram definidos que as Pessoas, derivadas da ontologia *Actor* do modelo SeCoM, estão divididas em Visitante, Estudante e Funcionário, sendo que este último item está dividido em Técnico e Professor. Estas pessoas estão envolvidas em atividades, herdadas da ontologia *Activity*, que ocorre em um Intervalo de Tempo (*TimeInterval*) e em um determinado local (*PhysicalLocation*).

Os locais podem ser construções ou instalações, as quais estão contidas dentro das construções. Um local pode ser o próprio campus, o qual possui uma Estrutura Administrativa e uma Estrutura Acadêmica. A Estrutura Administrativa é composta por um conjunto de Órgãos, os quais são divididos em Órgãos Técnicos Administrativos, Órgãos da Administração Superior e em Órgãos Representativos e Vinculados. Os Órgãos da Administração Superior são divididos em Reitoria, Pró-Reitorias, Centros de Ensino e Departamentos. A Reitoria controla as Pró-Reitorias e os Centros de Ensino. As Pró-Reitorias controlam os Órgãos Técnicos Administrativos e os Centros de Ensino controlam os Departamentos.

Já a Estrutura Acadêmica é dividida em um conjunto de Áreas de Conhecimento, as quais possuem diversos Cursos. Cada Curso possui um conjunto de Disciplinas, os quais são oferecidas por Departamentos.

Este modelo tem grande importância no desenvolvimento da aplicação, pois serve de base para modelagem do BDG e para a implementação do Módulo de Gerência de Contexto.

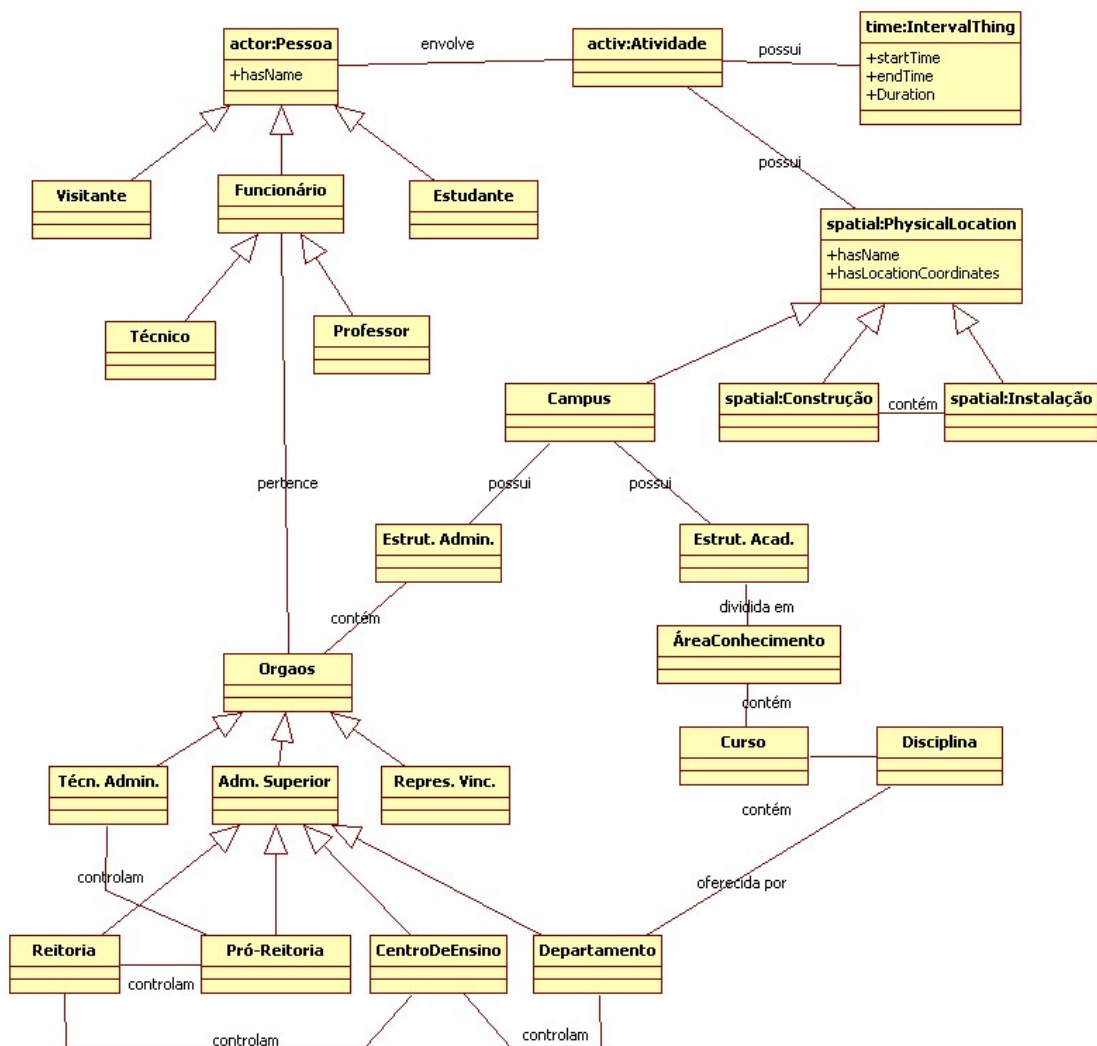


Figura 11 - OntoUFV - Ontologia de um Campus Universitário.

### 4.3 Construção do Banco de Dados Geográficos

O banco de dados geográficos do sistema UFV-GeoMobile foi construído a partir da junção de um conjunto de dados geográficos, mantidos pela administração da UFV, associados a um conjunto de bases relacionais utilizadas por uma variedade de sistemas administrativos, também mantidos pela instituição. Os dados geográficos foram alterados para se adequarem aos formatos de dados manipulados pelo aplicativo de SIG Móvel. Inicialmente os dados estavam armazenados no formato do sistema AutoCad e foram convertidos para o formato

*Shapefile*, utilizando-se o programa *cad2shape* [Cad2Shp, 2006]. As seguintes camadas foram produzidas: logradouros, construções e instalações específicas dentro de uma construção, como órgãos administrativos e laboratórios.

A modelagem conceitual do banco de dados, conforme visualizado na Figura 12, foi definida a partir das informações modeladas na ontologia de domínio de um campus universitário. Para esta tarefa foi utilizado o modelo UML-GeoFrame [Lisboa Filho e lochpe, 2008], que é um modelo específico para bancos de dados geográficos. Posteriormente, o modelo conceitual foi implementado utilizando-se o SGBD PostGreSQL com sua extensão espacial PostGIS.

No diagrama da Figura 12 pode-se observar que o sistema possui classes com e sem representação espacial. De acordo com o modelo UML-GeoFrame, as classes e subclasses Funcionário, Professor, Técnico Administrativo, Pessoa, Estudante, Visitante, Curso, Agenda, Atividade, Órgão e Tipo de Instalação são especializações de ObjetoNãoGeográfico ( $\Delta$ ), ou seja, não possuem representação espacial. As classes Construção, Logradouros e Instalação são percebidas na visão de objetos ( $\triangle$ ) e possuem representação espacial do tipo Área, Linha e Ponto, respectivamente. Por fim, as classes ImagemAérea e CurvaDeNivel são percebidas na visão de campo ( $\triangle$ ). A primeira possui representação espacial do tipo GradeDeCélulas ( $\boxplus$ ), enquanto a segunda possui representação do tipo Isolinhas ( $\boxminus$ ).

As instalações, representadas no diagrama da Figura 12, mapeiam um conjunto de informações que podem ser obtidas sobre a UFV. Elas podem ser de vários tipos como restaurante, supermercado, farmácia, bancos, papelaria, etc. Podem também ser algum laboratório ou ainda a representação de algum órgão da instituição, que por sua vez, pode ser um órgão da administração superior, um órgão técnico-administrativo ou um órgão vinculado.

As representações espaciais das instalações, juntamente com seus dados descritivos, são os itens utilizados para a realização de consultas ao banco de dados geográficos, e para exibir as informações resultantes para os usuários.

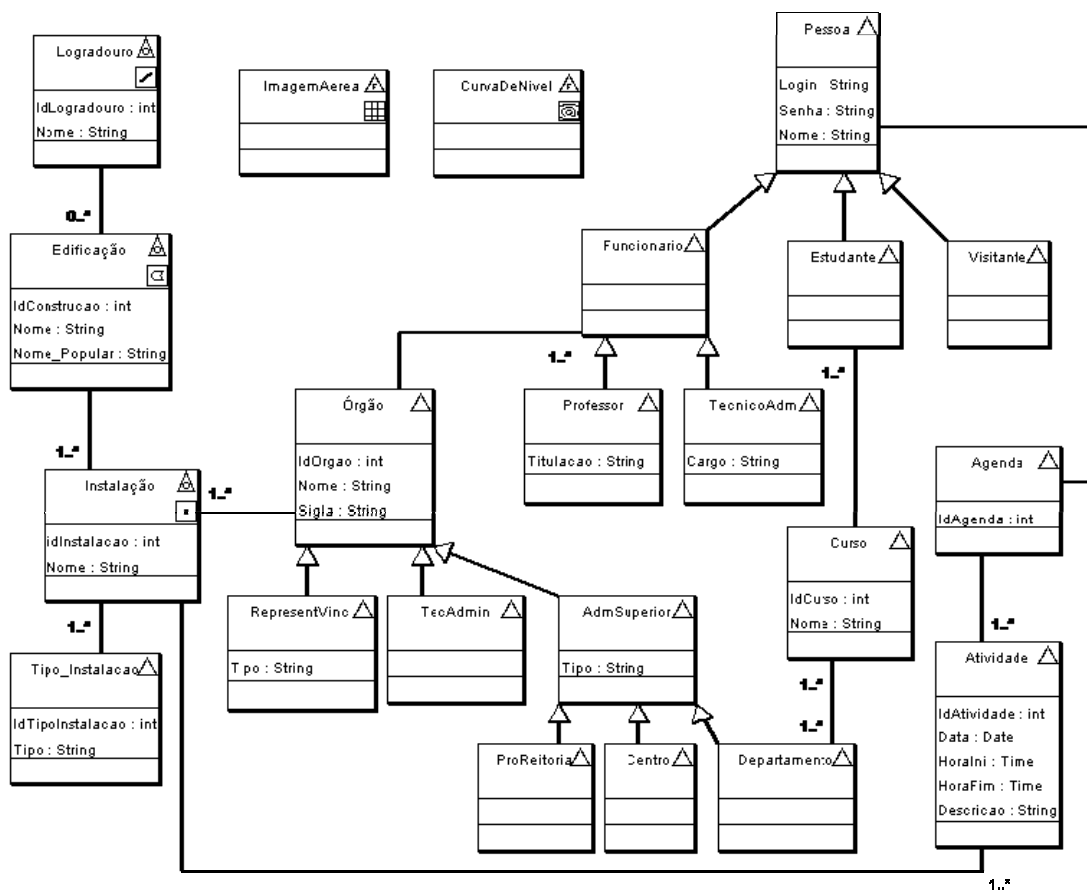


Figura 12 - Modelo do banco de dados geográficos do UFV-GeoMobile

#### 4.4 Definição do Web Service

O *Web Service* desenvolvido terá a função de intermediar a comunicação entre a aplicação no dispositivo móvel e as informações armazenadas no banco de dados geográficos. Foi desenvolvido um *Web Service* na linguagem Java (SOAP), utilizando-se a IDE NetBeans 6.1 e o servidor GlassFish. Para cada funcionalidade disponibilizada no aplicativo de SIG Móvel foi desenvolvida uma operação correspondente no *Web Service*, para receber os parâmetros e repassá-los ao Módulo de Consultas Espaciais ou ao Módulo de Gerência de Contexto.

Dentre as operações implementadas estão as que permitem a localização de pessoas ou lugares e a que indica a localização de atividades agendadas. A descrição detalhada de cada operação implementada no *Web Service* será descrita na Seção 4.7.2, a qual apresenta os cenários de consulta.

#### **4.5 Módulo de Consultas Espaciais**

O Módulo de Consultas Espaciais armazena um conjunto de operações implementadas na linguagem Java. Cada operação armazena consultas SQL capazes de responder as opções de pesquisa disponíveis para o usuário da aplicação de SIG Móvel. Estas consultas SQL incluem, dentre outros itens, cálculos de distância e a localização de pessoas ou lugares.

Cada operação do Módulo de Consultas Espaciais é acessada via uma operação específica do *Web Service*. O Módulo de Consultas Espaciais envia a consulta SQL ao BDG e então retorna o resultado desta consulta para a operação do *Web Service* que a requisitou.

#### **4.6 Módulo de Gerência de Contexto**

O Módulo de Gerência de Contexto é o responsável por fornecer informações de acordo com o contexto do usuário. A principal funcionalidade implementada neste Módulo é a localização geográfica de onde será realizado um evento previamente cadastrado em uma agenda de atividades do usuário.

A abordagem utilizada para a implementação deste Módulo também foi a utilização de consultas SQL, conforme o Módulo de Consultas Espaciais. Não foi utilizado neste estudo de caso a opção de manipulação das ontologias antes de enviar a consulta ao BDG.

#### **4.7 Aplicação SIG Móvel e Consultas**

A aplicação de SIG Móvel do sistema UFV-GeoMobile foi desenvolvida através do *framework* Microsoft Visual Studio 2005, utilizando a linguagem C#.NET. Utilizou-se durante o desenvolvimento o emulador Pocket PC 2003. Para a visualização dos dados geo-espaciais foi utilizado inicialmente o software Map Suite Pocket PC, o qual consiste em um conjunto de controles baseados na plataforma .NET para construção de aplicativos de SIG em dispositivos móveis [MapSuite, 2007]. Uma versão inicial deste protótipo foi publicado no trabalho de Lamas et al. [2008]. Com a evolução do sistema optou-se por utilizar o software SharpMap [SharpMAP,

2008], que possui a mesma finalidade do Map Suíte, porém apresenta um conjunto maior de operações que podem ser implementadas.

Nesta Seção são descritas a interface do sistema e alguns cenários de consulta que o sistema oferece, incluindo operações de consultas a pessoas, lugares ou a exibição de informações baseadas no contexto do usuário.

#### 4.7.1 Descrição da Interface

A interface inicial da aplicação UFV-GeoMobile é apresentada na Figura 13. Nesta interface foram utilizados os *layers* de construções, instalações e logradouros obtidos durante o tratamento dos dados geográficos, conforme apresentado na Seção 4.3, a qual detalha a construção do banco de dados geográficos para este estudo de caso.

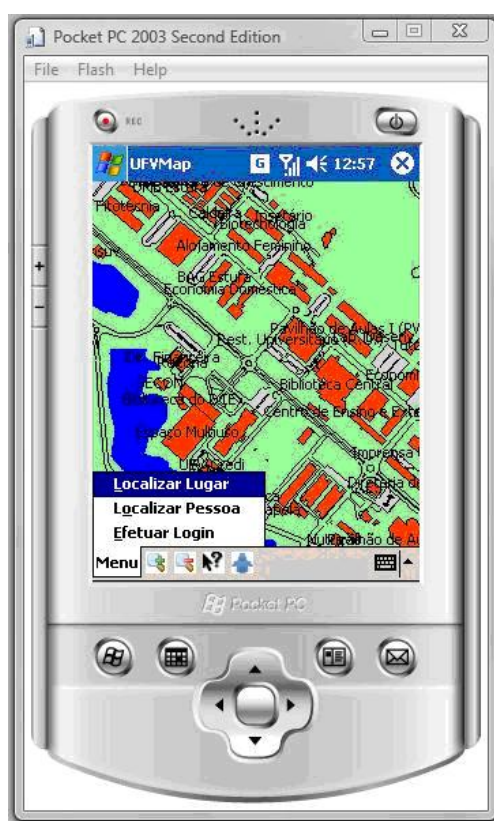


Figura 13 - Tela inicial da aplicação UFV-GeoMobile.

Além da exibição dos dados geográficos, esta interface possui uma barra de ferramentas, composta por um conjunto de ícones, nos quais o usuário tem as opções de aumentar (📏) ou diminuir o zoom (📏). Também existe o ícone [?] que

permite o clique sobre um ponto do mapa, e assim listar todas as instalações que estejam dentro da construção sobre a qual foi clicada. Existe ainda o ícone [📍] que permite ao usuário definir sua localização no mapa. Esta opção foi desenvolvida para que usuários que não possuam um GPS embutido em seu PDA possam também utilizar o sistema.

Foi utilizado ainda um menu que permite ao usuário acessar outras funcionalidades desenvolvidas na aplicação. As funcionalidades incluem “Localizar uma Pessoa”, “Localizar uma Instalação (Lugar)” e “Fazer login no Sistema”. Este último item é necessário para que o usuário possa ser identificado e assim o sistema entregue informações de acordo com o contexto deste usuário.

As funcionalidades disponíveis nesta interface inicial estão detalhadas na próxima Seção na qual são descritos os Cenários de Consulta.

#### **4.7.2 Cenários de Consulta**

São descritas nesta Seção as opções de consulta disponíveis na aplicação UFV-GeoMobile. Para cada operação foi desenvolvida uma operação correspondente no *Web Service*, bem como uma expressão na linguagem SQL para gerar consultas ao BDG. Estas expressões na linguagem SQL estão disponíveis tanto no Módulo de Consultas Espaciais quanto no Módulo de Gerência de Contexto.

##### **4.7.2.1 Consulta a um ponto**

A opção de consulta a um ponto é disponibilizada através do ícone [📍] visualizado na barra de ferramentas da aplicação. Caso o usuário clique sobre qualquer edificação válida, são exibidas em um quadro todas as instalações existentes nesta edificação, conforme visualizado na Figura 14.

O processo de exibição das informações ocorre da seguinte forma: ao clicar sobre um ponto é invocada a operação do *Web Service* “getPonto”, a qual é detalhada no Quadro 1. Esta operação obtém as coordenadas (x, y) do ponto clicado e então repassa estes parâmetros ao Módulo de Consultas Espaciais, através da chamada à operação “getInfPonto”, apresentada no Quadro 2.

A operação “getInfPonto” armazena uma consulta SQL que retorna os nomes das instalações que estão contidas na edificação que possui as

coordenadas (x, y) passadas como parâmetro. Os nomes das instalações são retornados para o *Web Service* na forma de uma lista de Strings (`List<String>` listaInstalacoes). Esta lista é então utilizada para exibir as informações na aplicação instalada no dispositivo móvel do cliente.

Vale ressaltar que as operações implementadas tanto no Módulo de Consultas Espaciais quanto no Módulo de Gerência de Contexto seguem o mesmo modelo da operação apresentada no Quadro 2. Os itens que variam de uma operação para outra são o nome, os parâmetros e a expressão SQL. Diante disso nos próximos exemplos as demais partes do código serão omitidas.



Figura 14 - Resultado da consulta a um ponto.

Quadro 1 - Operação do Web Service para consulta a um ponto.

```
@WebMethod(operationName = "getPonto")
public List<String> getInfPonto(@WebParam(name = "x")
double x, @WebParam(name = "y")
double y) {
    return ModuloCE.getInfPonto(x, y);
}
```

Quadro 2 - Operação e SQL no Módulo de Consultas Espaciais para consulta a um ponto.

```

public static List<String> getInfPonto(double x, double y) {

    Point p = new Point();
    p.setX(x);
    p.setY(y);

    PGgeometry geom = new PGgeometry();
    geom.setGeometry(p);

    // Seleciona as instalações de um ponto
    String sql =
        "SELECT i.nome
        FROM edificaciones e, instalacoes i
        WHERE intersects(e.the_geom,
            GeomFromText ('"+geom.toString()+"',-1))
            and intersects (e.the_geom, i.the_geom)";

    ConnectionBean c = ConnectionBean.getInstance();
    Connection conn = c.getConnection();

    List<String> listaInstalacoes =
        new LinkedList<String>();

    try {
        Statement s = conn.createStatement();
        ResultSet r = s.executeQuery(sql);
        // Adiciona instalações a lista
        while (r.next()) {
            listaInstalacoes.add(r.getString(1));
        }
        s.close();
    } catch (Exception e) {
        e.printStackTrace();
    }
    c.closeConnection(conn);
    return listaInstalacoes;
}

```

#### 4.7.2.2 Localização de Serviços ou Instalações Diversas

A opção para a localização de um serviço ou instalação é acessada através do menu disponível na interface inicial da aplicação UFV-GeoMobile. Ao solicitar esta opção é exibida a tela apresentada na Figura 15(a).

Inicialmente o usuário digita o nome completo ou parcial do item desejado. Ao digitar este nome e clicar no botão “Pesquisar” é invocada a operação “getLocal” do *Web Service*, descrita em detalhes no Quadro 3. Esta operação recebe como parâmetros o nome digitado e as coordenadas (x,y) do usuário, obtidas via GPS ou via um mecanismo implementado no SIG Móvel que permite ao próprio usuário indicar sua localização, inserindo o ícone [📍] no mapa. Esta segunda opção foi implementada para que o usuário possa utilizar o aplicativo UFV-GeoMobile, mesmo que o PDA não possua um GPS embutido.

A operação “getLocal” repassa seus parâmetros ao Módulo de Consultas Espaciais através da chamada a operação “getInfLocal”, apresentada no Quadro 4. A operação “getInfLocal” armazena uma consulta SQL, que retorna os nomes das instalações que correspondem ao nome digitado, a distância de cada instalação em relação a posição do usuário e as coordenadas do ponto central da edificação na qual a instalação esteja alocada. A seleção feita pela consulta SQL é retornada para o dispositivo móvel, que exibe uma lista das instalações disponíveis (Figura 15(a)). Ao selecionar um dos itens desta lista, a aplicação indica, através de um pequeno ícone (📍) desenhado sobre o mapa, a localização geográfica da instalação escolhida, conforme visualizado na Figura 15(b).

**Quadro 3 - Operação do Web Service para consulta a instalação.**

```
@WebMethod(operationName = "getLocal")
public List<String> getLocal(@WebParam(name = "nomeLocal")
String nomeLocal, @WebParam(name = "x")
double x, @WebParam(name = "y")
double y) {
    return ModuloCE.getInfLocal(x, y, nomeLocal);
}
```

**Quadro 4 - Operação e SQL no Módulo de Consultas Espaciais para consulta a uma instalação.**

```
public static List<String> getInfLocal(double x, double y, String
nomeLocal) {
    ...
    String sql =
        "SELECT i.nome,
           distance(e.the_geom, GeomFromText('POINT(x, y)',-1)),
           Centroid(e.the_geom)
        FROM Edificacao e, Instalacao i, Tipo_Instalacao t
        WHERE (i.nome like '%nomeLocal%') or
              ((t.tipo = 'nomeLocal') and (t.tipo = i.tipo)) and
              intersects(e.the_geom, i.the_geom)";
    ...
}
```



(a) Consulta a instalações.

(b) Localização geográfica de uma instalação.

Figura 15 - Telas para consulta e localização geográfica de instalações.

#### 4.7.2.3 Localização de Pessoas

A opção para localização de pessoas permite que se identifique a possível localização geográfica de qualquer pessoa que esteja vinculada à UFV. Isto é feito através da associação do objeto a seu respectivo local de trabalho.

Para acesso a esta opção de consulta utiliza-se o menu da tela inicial da aplicação UFV-GeoMobile. Inicialmente o usuário digita o nome completo ou parcial da pessoa procurada, utilizando a tela apresentada na Figura 16 (a). Ao clicar no botão “Pesquisar” é invocada a operação “getPessoa” do Web Service, detalhada no Quadro 5. Esta operação recebe como parâmetros o nome digitado e as coordenadas (x, y) do usuário, obtida via GPS ou via o ícone da tela inicial. Neste caso a posição do usuário é necessária para calcular a distância da posição do usuário em relação ao local de trabalho da pessoa procurada.

A operação “getPessoa” repassa seus parâmetros ao Módulo de Consultas Espaciais através da chamada a operação “getInfPessoa”, apresentada no Quadro

6. A operação “getInfPessoa” armazena uma consulta SQL, que retorna os nomes das pessoas que correspondem ao nome digitado, o respectivo órgão da UFV ao qual a pessoa está vinculada, a distância de cada órgão em relação a posição do usuário e as coordenadas do ponto central da edificação na qual o órgão esteja alocado. A seleção feita pela consulta SQL é retornada para o dispositivo móvel, que exibe uma lista das possíveis pessoas que atendem ao item digitado (Figura 16 (a)). Ao selecionar um dos itens desta lista, a aplicação indica, através de um pequeno ícone (📍) desenhado sobre o mapa, a localização geográfica do órgão ao qual a pessoa escolhida está vinculada, conforme visualizado na Figura 16 (b).

**Quadro 5 - Operação do Web Service para consulta a pessoas.**

```
@WebMethod(operationName = "getPessoa")
public List<String> getLocal(@WebParam(name = "nomePessoa")
String nomePessoa, @WebParam(name = "x")
double x, @WebParam(name = "y")
double y) {
    return ModuloCE.getInfPessoa(x, y, nomePessoa);
}
```

**Quadro 6 - Operação e SQL no Módulo de Consultas Espaciais para consulta a uma pessoa.**

```
public static List<String> getInfPessoa(double x, double y, String
nomePessoa) {
    ...
    String sql =
        "SELECT f.nome,
           o.nome,
           distance(e.the_geom, GeomFromText('POINT(x, y)', -1)),
           Centroid(e.the_geom)
        FROM Funcionario f, Orgao o, Edificacoes e, Instalacoes i
        WHERE (f.nome like '%nomePessoa%') and
              intersects(e.the_geom, i.the_geom) and
              (o.idOrgao = f.idOrgao) and
              (o.idOrgao = i.idOrgao)";
    ...
}
```



(a) Consulta a pessoa.

(b) Localização geográfica de uma pessoa.

Figura 16 - Telas para consulta e localização geográfica de pessoas.

#### 4.7.3 Notificação sobre a Localização Geográfica de Eventos

O mecanismo para notificar o usuário do SIG Móvel sobre a localização geográfica de alguma atividade que esteja cadastrada em sua agenda, foi desenvolvido como uma opção sensível ao contexto do usuário. Inicialmente para ter acesso a esta funcionalidade o usuário deve se identificar no sistema através de um *login* e uma senha, utilizando a tela apresentada na Figura 17.

Após a identificação do usuário, de minuto em minuto é invocada operação “GetEvento” do *Web Service*, apresentada no Quadro 7. Esta operação recebe como parâmetro, além do *login* e da senha, a hora atual, que é obtida do dispositivo móvel. Esta operação repassa estes parâmetros para o Módulo de Gerência de Contexto através da chamada à operação “getInfEvento”, detalhada no Quadro 8. A operação “getInfEvento” armazena um consulta SQL que retorna todas as atividades cadastradas do usuário que faltem 10 minutos para seu início,

detalhando a descrição da atividade, o horário de início e de término desta atividade, o nome da instalação na qual a atividade será realizada e as coordenadas do ponto central da edificação na qual a instalação está alocada. A partir destas informações, a aplicação de SIG Móvel exibe uma caixa de texto para o usuário e indica através de um ícone (📍) a localização geográfica desta atividade, conforme observado na Figura 18.



Figura 17 - Tela para login na aplicação UFV- GeoMobile.

**Quadro 7 - Operação do Web Service para notificar sobre eventos.**

```
@WebMethod(operationName = "getEvento")
public List<String> getEvento(@WebParam(name = "login")
String login, @WebParam(name = "senha")
String senha, @WebParam(name = "horaAtual")
Object horaAtual) {
    return ModuloGC.getInfEvento(login, senha, horaAtual);
}
```

**Quadro 8 - Operação e SQL no Módulo de Gerência de Contexto para notificar sobre eventos.**

```

public static List<String> getInfEvento(String login, String senha,
Object horaAtual) {
    ...
    String sql =
        "SELECT at.descricao, at.horaIni,
          at.horaFim, i.nome, Centroid(e.the_geom)
        FROM Edificacoes e, Instalacoes i,
          Atividade at, Agenda ag, Pessoa p
        WHERE (i.horaIni - horaAtual) = 10 and
          (p.login = 'login') and
          (p.senha='senha') and
          intersects(e.the_geom, i.the_geom) and
          (p.idpessoa=ag.idpessoa) and
          (ag.idagenda=at.idagenda)";
    ...
}

```



**Figura 18 - Tela para notificação de eventos.**

## 5 Conclusões e Trabalhos Futuros

### 5.1 Conclusões

Neste trabalho foi apresentada a especificação da arquitetura CM-GIS para o desenvolvimento de SIG Móveis capazes de gerenciar informações de contexto. Para validar esta arquitetura foi descrito também o estudo de caso UFV-GeoMobile. Para que se chegasse à solução proposta foram estudadas as características inerentes a este tipo de aplicação, a qual inclui: o uso de *Web Services* para permitir o acesso remoto a informações a partir do dispositivo móvel, ontologias para a modelagem de contexto e o uso de banco de dados geográficos para o armazenamento das informações.

A utilização de *Web Services* revela-se uma solução interessante para resolver problemas da falta de recursos computacionais dos dispositivos móveis. Isto permite a implementação de funcionalidades mais complexas em servidores mais robustos. Outra vantagem do uso de *Web Services* é a interoperabilidade entre diferentes sistemas, descritos em diferentes linguagens. Um exemplo é a aplicação UFV-GeoMobile na qual o sistema foi implementado em C# e o *Web Service* em Java.

Por outro lado o uso de *Web Services* torna a aplicação bastante dependente das conexões de rede. Caso a conexão disponível seja limitada, muitas das funcionalidades disponíveis na aplicação de SIG Móvel não poderão ser exploradas de forma satisfatória pelo usuário.

O uso de ontologias para especificar um modelo de contexto tem como principal vantagem a possibilidade de especificar o correto significado e relacionamento entre os termos, evitando interpretações imprecisas sobre o domínio que está sendo modelado. A adoção do modelo SeCoM pode facilitar bastante este processo de modelagem, visto que ele aborda as diversas dimensões de informação contextual e foi desenvolvido com característica de independência de domínio.

A abordagem utilizada para a implementação do sistema UFV-GeoMobile foi feita através do mapeamento dos termos definidos na ontologia para a

construção de um banco de dados geográficos. O uso de ontologias neste trabalho poderia ser melhor explorado se houvesse um mecanismo que manipulasse as ontologias e inferisse novas informações antes de retorná-las ao usuário do SIG Móvel.

Ainda em relação ao estudo de caso UFV-GeoMobile foram implementadas poucas funcionalidades especialmente em relação à exibição de informações de contexto. As opções de contexto ficaram reduzidas a uma agenda de atividades e ao cálculo de distâncias em relação à posição do usuário. Ainda assim estas funcionalidades permitem classificar a aplicação como sensível ao contexto.

Contudo, a arquitetura CM-GIS apresentada, juntamente com a estrutura UFV-GeoMobile desenvolvida como estudo de caso, mostram como o uso de um aplicativo de SIG Móvel capaz de gerenciar informações de contexto pode ser de grande utilidade para um usuário, especialmente pela capacidade de apresentar informações personalizadas combinadas com a exibição de dados geo-espaciais. Desse modo, este trabalho apresenta uma forma viável e bastante prática para o desenvolvimento de aplicações de SIG Móvel, apresentando soluções para tratar desafios típicos de aplicações que envolvam dados geo-espaciais, contexto e dispositivos móveis.

## **5.2 Contribuições**

Esta dissertação apresenta um conjunto de contribuições, que são parte relacionadas à arquitetura CM-GIS e parte ao sistema UFV-GeoMobile. Elas foram divididas da seguinte forma:

- Em relação à arquitetura CM-GIS:
  - Definição de uma arquitetura para o desenvolvimento de aplicações de SIG Móveis Sensíveis ao Contexto;
  - Proposta da reutilização de um modelo baseado em ontologias (SeCoM) para a modelagem do contexto da aplicação desenvolvida;
  - Mapeamento dos conceitos definidos em uma ontologia para a criação de um banco de dados geográficos;

- Uso de Módulos para armazenar consultas SQL que permitem obter respostas a consultas que envolvam dados espaciais ou a obtenção de informações relacionadas ao contexto do usuário.
- Em relação ao estudo de caso UFV-GeoMobile:
  - Criação de uma ontologia de domínio sobre o campus universitário da UFV (OntoUFV);
  - Tratamento de dados geográficos para criação de um repositório de dados geo-espaciais para a UFV.

### 5.3 Trabalhos Futuros

Vários trabalhos podem ser desenvolvidos tendo como base esta dissertação. Uma sugestão inicial seria a utilização de *Geo Web Services*, os quais são serviços definidos pela OGC (*Open GIS Consortium*) para tratar especificamente de dados geo-espaciais. Este mecanismo evitaria a necessidade da instalação de um aplicativo no PDA, conforme foi feito na aplicação UFV-GeoMobile. Ao invés disso, a aplicação de SIG Móvel poderia ser acessada através de um *browser*.

Outro trabalho poderia ser a criação de um mecanismo que manipulasse as ontologias e inferisse novas informações antes de retorná-las ao usuário do SIG Móvel. Este mecanismo estaria ligado aos módulos de consulta apresentados na arquitetura CM-GIS e poderia ser utilizado antes de enviar consultas ao banco de dados geográficos.

Uma outra sugestão seria incrementar o estudo de caso UFV-GeoMobile com novas funcionalidades, incluindo novas informações em relação ao contexto ou opções de consultas mais elaboradas. Dentre estas opções pode-se utilizar a localização geográfica do usuário para definir melhores rotas para se chegar a alguma instalação desejada. Também é possível explorar as características do hardware para definir a melhor forma de exibição das informações no dispositivo móvel. Pode-se ainda fazer um estudo da usabilidade e desempenho do sistema desenvolvido.

## 6 Referências Bibliográficas

- ABDALLA, D.; MAGALHÃES, C. (2005). Gestão do Conhecimento, Ontologias e Web Semântica. Disponível em: <<http://wiki.dcc.ufba.br/pub/Residencia/MaterialModuloTI5/CursoResidenciaTI5.ppt>>. Acesso em: 07 de maio de 2008.
- ABOWD, G.; MYNATT D. (2000). Charting past, present, and future research in ubiquitous computing. ACM Trans. Comput.-Hum. Interact (ACM Press), v. 7. pp. 29-58.
- ARAÚJO, M.A. (2005). Web Services na Informação Geográfica. Braga, Portugal. Dissertação (Mestrado em Informática) – Universidade do Minho.
- ARCPAD (2008). ArcPad: Mobile GIS Software for Field Mapping . Disponível em: <<http://www.esri.com/software/arcgis/arcpad/index.html>>. Acesso em: 12 de julho de 2008.
- BORGES, k. A. V. (2006). Uso de uma Ontologia de Lugar Urbano para Reconhecimento e Extração de Evidências Geo-espaciais na Web. Belo Horizonte. Tese (Doutorado em Ciência da Computação) – Universidade Federal de Minas Gerais.
- BRANDÃO, F.; RIBEIRO, J. A. (2007). Estudo do XML, GML, SVG e WEBSERVICES (WMS e WFS) para formatação e divulgação de informações geográficas. In: Anais XIII Simpósio Brasileiro de Sensoriamento Remoto. Florianópolis:INPE. pp..5611-5617.
- BRISABOA, N. R.; LUACES, M. R.; PARAMA, J. R.; VIQUEIRA, J. R. (2007). Managing a Geographic Database from Mobile Devices Through OGC Web Services. In: Proceedings of APWeb/WAIM 2007 Ws, LNCS 4537, pp. 174–179.
- BULCÃO NETO, R. F. (2006). Um processo de software e um modelo ontológico para apoio ao desenvolvimento de aplicações sensíveis ao contexto. São Paulo. Tese (Doutorado em Ciência da Computação) – Universidade de São Paulo.
- CAD2SHP (2006). Cad2Shp: AutoCAD DXF/DWG to ArcView Shapefile Converter. Disponível em: <<http://www.brightergraphics.co.uk/guthrie/cad2shp.htm>>. Acesso em: 12 de fevereiro de 2006.
- CADUFF, D.; EGENHOFER, M. J. (2005). *Geo-Mobile Queries: Sketch Based Queries in Mobile Enviroments*. In: Proceedings of W2GIS 2005, LNCS 3833, pp. 143-154.
- CÂMARA, G. (2005). Representação computacional de dados geográficos. In: CASANOVA, et al. Banco de dados geográficos. Curitiba: MundoGeo. 506 p.

- CASTRO, A.F.; SOUTO, M. V. S.; AMARO, V. E.; VITAL H. (2003). Desenvolvimento e Aplicação de um Banco de Dados Geográficos na Elaboração de Mapas da Morfodinâmica Costeira e Sensibilidade Ambiental ao Derramamento de Óleo em Áreas Costeiras Localizadas no Estado do Rio Grande do Norte. *Revista Brasileira de Geociências*. pp. 53-64.
- CHEN, G.; KOTZ, D. (2000). *A Survey of Context-Aware Mobile Computing Research*. Hanover, NH, USA: Dartmouth College, 2000, paginação irregular.
- COSTA, G. (2002). O Modelo de Web Services — Como Desenvolver Aplicações em uma Nova Arquitetura de Software. *Promon Business & Technology Review Series*, n.4.
- DEY, A. K. (2000). *Providing architectural support for building context-aware applications*. Estados Unidos. PhD thesis (Doutorado em Ciência da Computação) - Georgia Institute of Technology.
- DEY, A.; SALBER, D.; ABOWD, G (2001). A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Human-Computer Interaction Journal*, v. 16(2), pp. 97-166.
- FONSECA, F; EGENHOFER, M.; BORGES, K. A. V. (2000). Ontologias e Interoperabilidade Semântica entre SIGs. In: *Anais do II Workshop Brasileiro de Geoinformática (GeoInfo)*.
- GUARINO, N. (1998). *Formal Ontology in Information Systems*. In: *Proceedings of FOIS'98, Trento, Italy*. Amsterdam: IOS Press. pp. 3-15.
- HANSMANN, U. et al. (2003). "Pervasive Computing". Springer-Verlag, Second Edition. 448 p.
- HENRICKSEN, K.; INDULSKA, J.; RAKOTONIRAINY A. (2002). Modeling Context Information in Pervasive Computing Systems. In: *Proceedings of Pervasive '02: Proceedings of the First International Conference*. London, UK: Springer-Verlag. p.167-180.
- IBM (2008a). *Suporte a extensões espaciais IBM – DB2*. Disponível em: <<http://www-306.ibm.com/software/data/spatial>>. Acesso em: 20 de janeiro de 2008.
- IBM (2008b). *Extensões espaciais no Informix*. Disponível em: <[www.ibm.com/software/data/informix/blades/spatial](http://www.ibm.com/software/data/informix/blades/spatial)>. Acesso em: 17 de fevereiro de 2008.
- INPE (2007). Tutorial do Software Spring versão 3.5. Disponível em: <[http://www.epamig.br/geosolos/Apostila\\_PDF/Geo\\_cap2.pdf](http://www.epamig.br/geosolos/Apostila_PDF/Geo_cap2.pdf)>. Acesso em: 28 de janeiro de 2007.

- KNUBLAUCH, H. et al. (2006). A Semantic Web Primer for Object-Oriented Software Developers. Disponível em: <<http://www.w3.org/TR/2006/NOTE-sw-ood-primer-20060309/>>. Acesso em: 12 de março de 2008.
- LACY, L. W. (2005). OWL: Representing Information Using the Web Ontology Language, Canadá: Trafford. 302 p.
- LAMAS, A. R.; LISBOA FILHO, J.; OLIVEIRA, A. P.; LOPES, A. M. (2008). Sistemas de Informação Geográfica Móveis orientados ao contexto: uma abordagem baseada em ontologias de domínios. In: Anais do IV Simpósio Brasileiro de Sistemas de Informação (SBSI 2008). Rio de Janeiro: SBC. pp. 70-81.
- LI, X.; SHIN, W.; LI, L.; YOO, S. B. (2006). GIS Web Service Using Context Information in Mobile Environments. In: Proceedings of ICCSA 2006, LNCS 3980, pp. 895 – 903.
- LISBOA FILHO, J.; IOCHPE, C. (2008). *Modeling with a UML profile*. In: Shashi Shekhar and Hui Xiong. Encyclopedia of Geographical Information Science. New York: Springer. pp. 691-700.
- MAPSUITE (2007). MapSuite Pocket PC .NET GIS Component. Disponível em: <<http://gis.thinkgeo.com/Products/GISComponentsforNETDevelopers/MapSuitePocketPC/tabid/163/Default.aspx>>. Acesso em: 07 de fevereiro de 2007.
- MARTIN, D. (2004). OWL-S: Semantic Markup for Web Services. Disponível em: <<http://www.daml.org/services/owl-s/1.1/>>. Acesso em: 12 de março de 2007.
- MYSQL (2008). *GIS and Spatial Extensions with MySQL*. Disponível em: <<http://dev.mysql.com/tech-resources/articles/4.1/gis-with-mysql.html>>. Acesso em: 20 de janeiro de 2008.
- NOVELLO, T. C. (2003) Ontologias, Sistemas baseados em Conhecimento e Modelos de Banco de Dados. Disponível em: <[http://www.inf.ufrgs.br/~clesio/cmp151/cmp15120021/artigo\\_taisa.pdf](http://www.inf.ufrgs.br/~clesio/cmp151/cmp15120021/artigo_taisa.pdf)>. Acesso em: 20 de maio de 2007.
- OGC (2008a). About OGC. Disponível em: <<http://www.opengeospatial.org/ogc>>. Acesso em: 20 de março de 2008.
- OGC (2008b). OpenGIS Standards and Specifications. Disponível em: <<http://www.opengeospatial.org/standards/>>. Acesso em: 20 de março de 2008.
- OGC (2008c). OpenGIS Web Map Service (WMS) Implementation Specification. Disponível em: <<http://www.opengeospatial.org/standards/wms/>>. Acesso em: 20 de março de 2008.

- OGC (2008d). OpenGIS Web Feature Service (WFS) Implementation Specification. Disponível em: <<http://www.opengeospatial.org/standards/wfs>>. Acesso em: 20 de março de 2008.
- OILED (2003). OilEd – An ontology editor allowing the user to build ontologies using DAML+OIL. Disponível em: <<http://oiled.man.ac.uk/>>. Acesso em: 20 de janeiro de 2007.
- ONTOEDIT (2007). The OTK Tool Repository: OntoEdit. Disponível em: <<http://www.ontoknowledge.org/tools/ontoedit.shtml>>. Acesso em: 20 de janeiro de 2007.
- OPENGIS (2007). Site do Open GeoSpatial Consortium. Disponível em: <<http://www.opengeospatial.org>>. Acesso em: 20 de janeiro de 2007.
- ORACLE (2008). Oracle Spatial. Disponível em: <<http://www.oracle.com/technology/products/spatial>>. Acesso em: 20 de janeiro de 2008.
- OWL (2004). Web Ontology Language. Disponível em: <<http://www.w3.org/TR/owl-guide/>>. Acesso em: 20 de janeiro de 2007.
- PESSOA, R. M.; CALVI, C. Z.; FILHO, J. G. P.; ANDREÃO, R. V. (2006). Aplicação de um middleware sensível ao contexto em um sistema de telemonitoramento de pacientes cardíacos. In: *Anais do XXXIII Seminário Integrado de Software e Hardware (SEMISH'06)*. Disponível em: <<http://www.lprm.inf.ufes.br/~camilo/docs/araq0086.pdf>>. Acesso em: 20 de maio de 2008.
- POSTGIS (2008). Extensão espacial para o PostGreSQL. Disponível em: <<http://postgis.refractor.net/>>. Acesso em: 20 de janeiro de 2008.
- PROTÉGÉ (2005). Protege Ontology Editor. Disponível em: <<http://protege.stanford.edu/>>. Acesso em: 20 de janeiro de 2007.
- QUEIRÓZ, G.R. (2004). Algoritmos Geométricos para Bancos de Dados Geográficos: Da Teoria à Prática na Terralib. São José dos Campos. 147 f. Dissertação (Mestrado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais.
- QUEIRÓZ, G. R., et al. (2007). OGC Web Services. São José dos Campos: INPE. Disponível em: <[http://www.dpi.inpe.br/~gribeiro/apresentacoes/ogc\\_ows\\_2007.pdf](http://www.dpi.inpe.br/~gribeiro/apresentacoes/ogc_ows_2007.pdf)>. Acesso em: 20 de março de 2008.
- RDF (2004). Resource Description Framework. Disponível em: <<http://www.w3.org/TR/rdf-concepts/>>. Acesso em: 20 de janeiro de 2007.

- SADEH, N. M.; CHAN, E.; VAN L. (2002). MyCampus: An agent-based environment for context-aware mobile services. In: Proceedings of Workshop on Ubiquitous Agents on Embedded, Wearable and Mobile Devices (UBIAGENTS 2002). Bologna, Itália. Disponível em: <<http://autonomousagents.org/ubiquitousagents/2002/papers/papers/29.pdf>>. Acesso em: 20 de novembro de 2007.
- SATYANARAYANAN, M. (2001). Pervasive computing: vision and challenges. *Personal Communications, IEEE*, v. 8. p. 10-17.
- SHARPMAP (2008). SharpMap: Geospatial Application Framework for the CLR. Disponível em: <<http://www.codeplex.com/SharpMap>>. Acesso em: 17 de agosto de 2008.
- SILVA, G. K. C.; PEREIRA, P. M.; MAGALHÃES, G. C. (2004). Disponibilização de Serviços Baseados em Localização via Web Services. In: Anais do VI Simpósio Brasileiro de Geoinformática (GeoInfo 2004). Disponível em: <<http://www.geoinfo.info/geoinfo2004/papers/6392.pdf>>. Acesso em: 12 de março de 2007.
- SILVA, P. P. O. (2004). Utilizando a tecnologia SIG para adicionar valor aos serviços baseados em localização para usuários de dispositivos móveis celulares. Rio de Janeiro. Dissertação (Mestrado em Informática) – Universidade Federal do Rio de Janeiro.
- SMITH, B.; MARK, D. (1998) Ontology and Geographic Kinds. In: International Symposium on Spatial Data Handling, Vancouver, Canada, pp. 308-320.
- SOLYMAN, A. A. (2005). Investigating Mobile GIS. Disponível em: <<http://www.directionsmag.com/article.php/>>. Acesso em: 28 de maio de 2007.
- SPACCAPIETRA, S.; CULLOT, N.; PARENT, C.; VANGENOT, C. (2004). On Spatial Ontologies. In: Proceedings of VI Brazilian Symposium on Geoinformatics. Campos do Jordão, SP, Brasil. Disponível em <<http://lbdwww.epfl.ch/~stefano/geoinfolast.pdf>>. Acesso em: Julho de 2007.
- STRANG, T.; LINNHOFF-POPIEN, C. (2004). *A Context Modeling Survey*. In: Proceedings of Workshop on Advanced Context Modelling, Reasoning and Management (UbiComp4). Nottingham-England.
- STUDER, R.; BENJAMINS, V.R.; FENSEL, D. (1998). *Knowledge engineering: principles and methods*. In: *Data & Knowledge Engineering*. Amsterdam, Holanda. pp. 161-197.
- TOIVONEN, S.; KOLARI, J.; LAAKKO, T. (2003). *Facilitating mobile users with contextualized content*. In: Proceedings of Workshop Artificial Intelligence in Mobile System (AIMS 2003). Disponível em: <<http://w5.cs.uni->

- sb.de/~krueger/aims2003/camera-ready /toivonen-5.pdf>. Acesso em 15 de março de 2007.
- TOMKO, M. (2003). Spatial Databases for Mobile GIS Applications. Eslováquia. Dissertação (Mestrado em Engenharia Civil) – Slovak University of Technology.
- TROUNG, K. N.; ABOWD, G. D.; BROTHERTON, J. A. (2001). Who, what, when, where, how: Design issues of capture & access applications. In: Proceedings of the International Conference on Ubiquitous Computing (UbiComp'01), number 2201 in Lecture Notes in Computer Science. Atlanta, Georgia, USA: Springer. , pp. 209–224.
- USCHOLD, M.; GRUNINGER, M. (1996). *Ontologies: Principles, Methods and Applications*. Knowledge Engineering Review, 11(2). pp. 93-155.
- VIEGAS, R. F. (2006). GeOntoQuery: um mecanismo de busca em banco de dados geográficos baseado em ontologias. Natal, RN. Dissertação (Mestrado em Sistemas e Computação) – Universidade Federal do Rio Grande do Norte.
- VINHAS, L.; FERREIRA, K. R. (2005). Descrição da Terralib. In: CASANOVA, et al. Banco de dados geográficos. Curitiba: MundoGeo. 506 p.
- W3C (2008a). Web Services Architecture. Disponível em: <<http://www.w3.org/TR/2002/WD-ws-arch-20021114/>>. Acesso em: 19 de março de 2008.
- W3C (2008b). Extensible Markup Language. Disponível em: <<http://www.w3.org/XML/>>. Acesso em: 19 de março de 2008.
- W3C (2008c). SOAP Version 1.2 Part 1: Messaging Framework (Second Edition). Disponível em: <<http://www.w3.org/TR/soap12-part1/>>. Acesso em: 19 de março de 2008.
- W3C (2008d). WSDL and UDDI. Disponível em: <[http://www.w3schools.com/WSDL/wSDL\\_uddi.asp](http://www.w3schools.com/WSDL/wSDL_uddi.asp)>. Acesso em: 19 de março de 2008.
- WEAKLIAM, J., et al. (2005). Delivering Personalized Context-Aware Spatial Information to Mobile Devices. In: Proceedings of W2GIS 2005, LNCS 3833, pp. 194–205.
- WEIßENBERG, N.; GARTMAN, R.; VOISARD, A. (2006). *An Ontology-based Approach to Personalized Situation-aware Mobile Service Supply*. Geoinformatica, v.10, n.1, pp. 55-90.
- WORBOYS, M.; DUCKHAM, M. (2004). GIS: A Computing Perspective. 2 ed. Boca Raton, Florida, USA: CRC Press. 426 p.