

# MD-UTS: Um Modelo para Desenvolvimento de Sistemas Ubíquos de Transporte

Luiz Henrique Feltes<sup>1</sup>, Jorge Luis Victória Barbosa<sup>1</sup>

<sup>1</sup>Programa Interdisciplinar de Pós-Graduação em Computação Aplicada  
Universidade do Vale do Rio dos Sinos (UNISINOS)  
Postal 93.022-000 – São Leopoldo – RS – Brasil

luizfeltes@gmail.com, jbarbosa@unisinis.br

**Abstract.** *The use of mobile and ubiquitous computing has been stimulated by the widespread diffusion of mobile devices, wireless networks and location systems. In this context, numerous applications are emerging in different areas such as education, entertainment, commerce and transport. This paper proposes the MD-UTS, a model for development of ubiquitous transport systems. The model allows management of different types of vehicles, making a historical record of their movement. The model provides information through web services that can be used by different applications. The article also describes two applications integrated into an urban transport simulator. The results demonstrate the feasibility of the model, facilitating the construction of applications directed to the transport system.*

**Resumo.** *O uso da computação móvel e ubíqua tem sido estimulada pela grande difusão dos dispositivos móveis, redes sem fio e sistemas de localização. Nesse contexto, diversas aplicações estão surgindo em diferentes áreas, como educação, entretenimento, comércio e transportes. Esse artigo propõe o MD-UTS, um modelo para o desenvolvimento de sistemas ubíquos de transporte. O modelo permite o gerenciamento de diferentes tipos de veículos, criando um registro histórico do seu movimento. O modelo fornece informações através de web services que podem ser utilizadas por diferentes aplicações. O artigo também descreve duas aplicações integradas com um simulador de transporte urbano. Os resultados demonstram a viabilidade do modelo, facilitando a construção de aplicações para o sistema de transporte.*

## 1. Introdução

Em 1991, Mark Weiser introduziu o conceito de computação ubíqua, ao prever um mundo onde dispositivos computacionais estariam presentes em nossas vidas, interagindo naturalmente com os usuários sem que fossem percebidos, em objetos, ambientes e nos próprios seres humanos [Weiser 1991].

A Computação Ubíqua está permitindo o surgimento de oportunidades em diversas áreas como medicina, educação, jogos, comércio, entretenimento, transportes, entre outros. Conceitos como U-Learning [Barbosa et al. 2011], U-Commerce [Franco et al. 2011], U-Health [Kwon et al. 2010] e U-City [Oh and Oh 2011] surgiram nos últimos anos.

Os Sistemas Sensíveis ao Contexto [Dey et al. 2001] podem agregar valor na elaboração dos sistemas de transportes. A popularização crescente de dispositivos móveis e a evolução dos mais variados tipos de sensores são alguns dos fatores que contribuíram para os sistemas sensíveis ao contexto.

Os Sistemas Inteligentes de Transporte e os Sistemas Sensíveis ao Contexto têm uma ligação forte, pois envolvem questões comuns como o deslocamento entre determinadas regiões demográficas [Vieira et al. 2011].

Além dos Sistemas Sensíveis ao Contexto, destaca-se também o uso de registros históricos de uma entidade percorrendo diversos contextos, chamado de Trilha [Silva et al. 2010]. A Trilha é uma coleção de locais visitados, com informações de contexto associadas, e segue uma ordem de cronológica. No caso dos transportes, a Trilha pode representar a rota realizada pelo veículo demarcando pontos importantes durante este percurso.

Os Sistemas Inteligentes de Transporte (ITS - *Intelligent Transport Systems*) são aplicações e tecnologias que envolvem uma integração entre comunicação, gerenciamento e processamento de informações para um sistema de transporte [Miles and Walker 2006].

Sussman [Sussman 2005] indica como principal objetivo de um ITS: “casar o mundo de altas tecnologias e avanços com o mundo convencional da infraestrutura de transporte”. As altas tecnologias são descritas como aquelas envolvendo áreas como: sistemas de informação, comunicação, sensores e métodos matemáticos avançados. Nesse sentido, surge o conceito de *Ubiquitous Transport System* (UTS) [Lee et al. 2008] [Passos and Rossetti 2009], unindo as tecnologias ITS com os conceitos da computação ubíqua.

Esse artigo apresenta um modelo para o desenvolvimento de sistemas ubíquos de transporte denominado MD-UTS, que provê sensibilidade ao contexto, gerenciamento de diferentes tipos de veículos através de um módulo de perfil e o gerenciamento de trilhas.

O artigo está organizado em seis seções. Na seção dois é proposto o modelo MD-UTS. A terceira seção apresenta os aspectos de implementação e na quarta os aspectos de avaliação. Na seção cinco são descritos e comparados os trabalhos relacionados e por fim a seção seis apresenta as considerações finais.

## 2. Modelo MD-UTS

O MD-UTS<sup>1</sup> é um modelo para desenvolvimento de sistemas ubíquos de transporte sendo suas principais características:

- **Suporte a diversos tipos de veículos:** não restringe o tipo de veículo a ser gerenciado;
- **Sensibilidade ao contexto:** possibilita o gerenciamento de contextos, incluindo regiões e pontos de interesse por região;
- **Gerenciamento de trilhas:** permite armazenar e consultar registros históricos sobre os contextos visitados pelos veículos;
- **Gerenciamento de perfil do veículo:** permite a utilização de um perfil para cada veículo, de acordo com suas características;

---

<sup>1</sup>MD-UTS (*Model for Development of Ubiquitous Transport Systems*)

- **Suporte a sensores:** permite o acesso a informações de sensores instalados nos veículos;
- **Portal administrativo:** permite gerenciar as informações dos módulos do modelo;
- **Serviços:** disponibiliza serviços para que aplicações sejam construídas utilizando as informações gerenciadas pelo modelo.

Com base nessas características, aplicações dos mais variados gêneros podem ser desenvolvidas utilizando os serviços fornecidos pelo modelo, permitindo levar as informações para diferentes públicos, como por exemplo, motoristas, usuários de transporte, governo e empresas responsáveis pelo transporte público ou privado.

### 2.1. Arquitetura do MD-UTS

A arquitetura do MD-UTS (Figura 1) possui dois componentes chamados *Transport System* (TS) e *Vehicle System* (VS) e uma ontologia para a representação de conhecimento na área de transporte. A arquitetura prevê ainda o uso de aplicações externas que podem ser conectadas ao modelo.

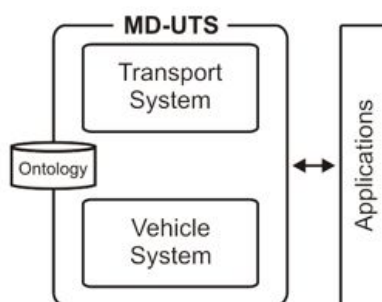


Figura 1. Arquitetura do MD-UTS

O TS é o principal componente do modelo, gerenciando as informações sobre contextos, perfil dos veículos e as trilhas realizadas por eles. O VS suporta a comunicação com os diferentes sensores que podem ser instalados no veículo, os quais são gerenciados pelo TS.

A criação da ontologia para domínio de transportes foi baseada na metodologia proposta por Noy e McGuinness [Noy and McGuinness 2001] que propõe a criação de uma versão inicial que deve ser revisada e refinada gradativamente.

Seguindo essa metodologia, três ontologias relacionadas foram pesquisadas para considerar a reutilização [Wang et al. 2005] [Almeida and Bax 2003] [Houda et al. 2010]. Com base nas ontologias, uma nova ontologia é proposta, reaproveitando aspectos de cada uma delas. A Figura 2 mostra a ontologia proposta, chamada de OntoUTS.

A OntoUTS auxilia na definição dos termos utilizados e seus relacionamentos, facilitando a integração entre os componentes do MD-UTS e também a construção de aplicações. As principais classes da ontologia são as seguintes:

- **Region:** classe responsável por definir uma região demográfica, utilizando um conjunto de coordenadas geográficas, que representam por exemplo um bairro ou uma cidade;



Figura 2. Ontologia para domínio de transporte (OntoUTS)

- **Entity:** classe que representa os objetos físicos que são utilizados no modelo, tais como carros ou rodoviárias;
- **EntityStationary:** classe que representa um objeto físico que não se movimenta, ou seja, sempre está na mesma posição geográfica, tais como uma rodoviária ou um posto de combustível;
- **EntityMobile:** classe que representa um objeto físico que se movimenta, ou seja, sua posição varia de acordo com sua locomoção, tais como carros, ônibus e metrô;
- **Vehicle:** subclasse da classe *EntityMobile* que representa os diferentes tipos de veículos suportados pelo modelo.

## 2.2. Transport System (TS)

O TS gerencia as informações oriundas dos veículos mapeados, fornecendo serviços para que outras aplicações possam usufruir destes dados. O TS é composto por cinco módulos, chamados de *Context*, *Profile*, *Trail*, *Portal* e *Services*, conforme a Figura 3.

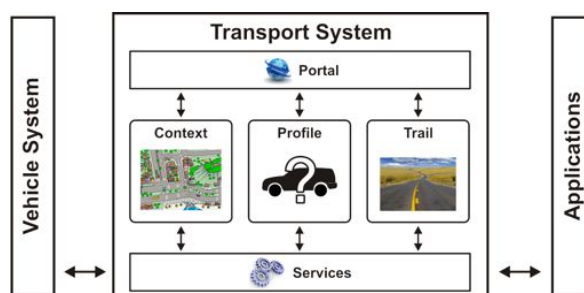


Figura 3. Módulos do *Transport System* (TS)

O módulo *Context* gerencia as informações sobre os contextos. Cada contexto é representado por uma região, conforme descrito pela ontologia. A região é determinada por um conjunto de coordenadas geográficas obtido utilizando um sistema de posicionamento global (GPS).

Além das informações sobre as regiões, o módulo *Context* gerencia as informações

sobre as entidades fixas que pertencem à região. Essas entidades podem representar qualquer ponto de interesse conforme descrito na OntoUTS.

O módulo *Profile* gerencia as informações sobre o perfil de cada veículo. Nele estão os dados informados no momento em que o veículo é adicionado ao modelo, e as informações obtidas durante o seu deslocamento pelas regiões.

O perfil do veículo é composto por propriedades pertinentes a cada veículo. Elas podem ser de dois tipos: estáticas ou dinâmicas. As estáticas são informações cadastrais, diferentemente das dinâmicas que são alteradas conforme os sensores capturam dados do veículo. A Tabela 1 descreve algumas das propriedades disponíveis.

**Tabela 1. Propriedades do Perfil**

Classe	Propriedade	Descrição
Vehicle	Identificador	Identificador único do veículo
Vehicle	PosiçãoAtual	Identifica a posição geográfica atual do veículo, usando a latitude e longitude
Vehicle	RegiãoAtual	Identifica a região onde o veículo se encontra
Vehicle	Velocidade	Identifica a velocidade atual do veículo
Bus	Rota	Identifica a rota do ônibus
Bus	AtualPassageiros	Identifica a quantidade atual de passageiros no ônibus
Bus	Acessibilidade	Identifica se o ônibus possui algum tipo de acessibilidade para PCDs (Pessoas Com Deficiência)

O módulo *Trail* armazena e disponibiliza as informações sobre a trilha do veículo. A trilha é um conjunto de registros, onde cada registro é composto por uma data, uma localização geográfica, o veículo, o evento dessa trilha e uma descrição detalhada.

O *Portal* gerencia os dados armazenados no MD-UTS. Através de um sítio na web é possível realizar a inserção, atualização e remoção dos dados de cada um dos módulos.

A comunicação entre o TS e o VS, assim como entre o TS e as aplicações, passa pelo módulo *Services*. Esse módulo disponibiliza uma camada de comunicação baseada em serviços. O uso dos serviços envolve uma comunicação através do protocolo HTTP. Cada serviço é disponibilizado através de um URI que descreve a sua função.

Os serviços disponibilizados formam uma API para trabalhar com as informações provenientes do modelo. Os serviços foram construídos utilizando os princípios da arquitetura REST [Fielding 2000] e estão agrupados de acordo com os módulos do modelo. A Tabela 2 descreve alguns dos serviços disponíveis.

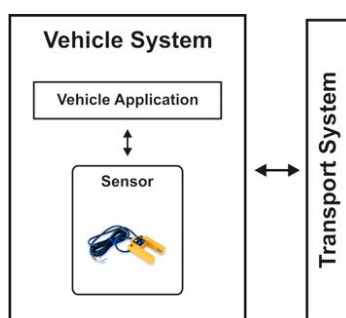
**Tabela 2. Serviços disponíveis no MD-UTS**

URI	Método	Descrição
/context/region	GET	Identifica o nome da região utilizando uma localização
/profile/property	GET	Busca o nome de todas as propriedades de um determinado veículo
/profile/property	POST	Atualiza a informação de uma propriedade específica de um veículo
/trail/trail	POST	Inserir um registro na trilha do veículo
/trail/trail	GET	Busca as trilhas de um determinado veículo

### 2.3. Vehicle System (VS)

O componente VS coleta os dados do veículo utilizando sensores, para envio ao TS. Para realizar essa tarefa, o VS é composto por dois módulos, chamados de *Sensor* e *Vehicle*

Application, conforme a Figura 4.



**Figura 4. Módulos do *Vehicle System* (VS)**

O *Sensor* suporta a interação dos sensores instalados nos veículos com o MD-UTS. Podem existir diversos tipos de sensores, como por exemplo, localização [Hightower and Borriello 2001], presença, velocidade e proximidade [Fowler 2009].

Os sensores podem variar de veículo para veículo, de acordo com os dados que o veículo deve disponibilizar. Veículos de transporte público, como por exemplo, ônibus e metrô, podem utilizar um sensor de presença para identificar a quantidade de passageiros dentro do veículo e atualizar essas informações no MD-UTS em tempo real.

O módulo *Vehicle Application* é um aplicativo que trata as informações como velocidade e localização, independente do tipo de veículo e disponibiliza esses dados para o TS. As três funções realizadas são: i) atualizar constantemente a localização do veículo, utilizando um sensor de localização; ii) trabalhar com as propriedades do perfil do veículo, como seu identificador, velocidade e localização; iii) registrar eventos na trilha de acordo com a movimentação do veículo.

Esse aplicativo pode ser estendido e especializado de acordo com o tipo de veículo e sua finalidade. Dessa forma, o VS pode ser adaptado para atender a diversas situações. Veículos como os ônibus, por exemplo, além das informações sobre a sua localização, podem adaptar o aplicativo e utilizar os demais serviços disponíveis no módulo *Services*.

### 3. Implementação e Simulação

O protótipo do MD-UTS foi implementado utilizando a linguagem de programação Java. Para a implementação do módulo *Portal* foi utilizado JSP em conjunto com HTML, CSS e Javascript. Na camada de comunicação do módulo *Services* foi utilizado o *framework* VRaptor para criar os serviços REST. Para a persistência dos dados foi utilizado o banco de dados MySQL.

A geração de dados para a avaliação foi baseada em uma simulação usando o SUMO<sup>2</sup>. A simulação envolveu dados reais oriundos da cidade de São Leopoldo, localizada no estado do Rio Grande do Sul.

A cidade foi inserida no simulador utilizando a exportação de mapas através do Open Street Maps<sup>3</sup>. Pontos de interesse como rodoviárias, estações de metrô e paradas

<sup>2</sup><http://sumo.sourceforge.net>

<sup>3</sup><http://www.openstreetmap.org>

de ônibus também foram adicionadas, assim como 4 das 78 linhas de ônibus disponíveis na cidade. O mapeamento das paradas foi realizado com o Google Street View<sup>4</sup>. Os demais veículos foram inseridos de forma aleatória, utilizando as ferramentas do SUMO. A Figura 5 apresenta a cidade mapeada no simulador.

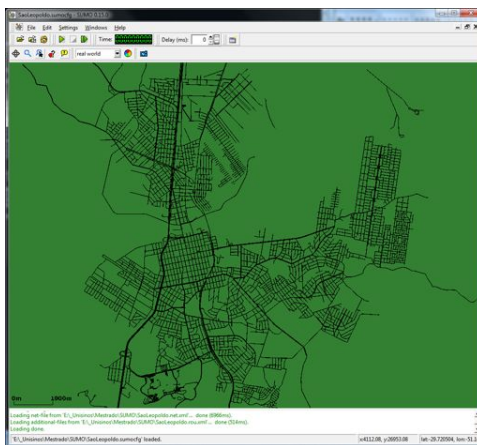


Figura 5. Cidade de São Leopoldo no simulador (SUMO)

## 4. Avaliação

A comunidade científica vem usando em conjunto com os aplicativos a criação de cenários para a validação de sistemas sensíveis ao contexto [Dey et al. 2001] e sistemas ubíquos [Satyanarayanan 2001] [Franco et al. 2011] [Oliveira et al. 2013].

Dessa forma, para avaliar as principais características funcionais do MD-UTS, foram desenvolvidos dois aplicativos baseados em cenários. O primeiro aplicativo avalia as características de sensibilidade ao contexto e o sistema de perfil do veículo, descrevendo um cenário de transporte público. O segundo aplicativo avalia a característica de suportar diferentes tipos de veículos e o módulo de trilhas, permitindo gerenciar os veículos e registrar dados históricos sobre eles.

### 4.1. Aplicação 1 - *Bus Assistent*

O aplicativo *Bus Assistent* (Figura 6(a)) explora os módulos *Context*, *Profile* e *Services*. Informações sobre contexto (linhas e paradas) e perfil (capacidade e suporte à acessibilidade) de alguns ônibus da cidade foram adicionadas para suporte à simulação.

O cenário é o seguinte: *”Carlos é um estudante da UNISINOS e cadeirante. Ele está utilizando o Bus Assistent no seu smartphone. Todo dia pela manhã, antes de sair de casa para se deslocar até a parada mais próxima, consulta o aplicativo para identificar o ônibus com acessibilidade que faz a linha entre sua casa e a Universidade. Após realizar a busca, Carlos identifica a localização atual do ônibus e a quantidade de passageiros. Como o ônibus estava praticamente lotado e ele ainda tinha tempo para esperar o próximo, optou por aguardar em casa terminando suas atividades escolares”*.

O *Vehicle Application* (VA) foi estendido para trabalhar com as informações de entrada e saída de passageiros do ônibus e enviar esses dados para o *Services*. A Tabela 3 resume a dinâmica do cenário, destacando os atores e as ações.

<sup>4</sup><http://maps.google.com.br/intl/pt-BR/help/maps/streetview/>

**Tabela 3. Dinâmica do primeiro cenário (*Bus Assistant*)**

<b>Ator</b>	<b>Ação</b>
Ônibus	A cada 30 segundos o <i>Vehicle Application</i> (VA) instalado no veículo envia as informações (velocidade, quantidade de passageiros, localização) para o <i>Services</i>
<i>Services</i>	Recebe as informações e registra no módulo de perfil
Usuário	Inicia o aplicativo para consultar sua linha de ônibus
<i>Bus Assistant</i>	Comunica com o <i>Services</i> para localizar a linha de ônibus e a parada desejada
<i>Services</i>	Busca as informações de contexto (linhas e paradas) da cidade
<i>Bus Assistant</i>	Mostra para o usuário os ônibus dessa linha e o tempo aproximado para chegar até parada
Usuário	Seleciona o ônibus para obter as informações
<i>Bus Assistant</i>	Chama novamente o <i>Services</i> para buscar as informações sobre perfil do veículo
Usuário	Usuário visualiza as informações sobre o ônibus e toma sua decisão

A Figura 6(a) mostra o aplicativo com o resultado da busca realizada pelo usuário. O marcador representa a posição atual do ônibus e abaixo são exibidas as informações sobre o perfil do veículo como capacidade, quantidade de passageiros e se o veículo tem suporte à acessibilidade.

#### **4.2. Aplicação 2 - *City Manager***

O aplicativo *City Manager* (Figura 6(b)) explora os módulos *Trail* e *Services*, bem como o suporte a diferentes tipos de veículos. Dessa forma, diferentes veículos foram mapeados como carros, ambulâncias, viaturas da guarda municipal, táxis e ônibus.

O seguinte cenário foi executado: "A Prefeitura precisa controlar os veículos da cidade em tempo real e armazenar um registro histórico dos veículos. Além disso, as viaturas da guarda municipal precisam registrar os locais onde são realizadas as ocorrências. Esses dados vão ser utilizados para identificar as áreas onde surgem mais ocorrências para então organizar as viaturas em pontos estratégicos. No início do mês de Fevereiro a Prefeitura acessa o aplicativo *City Manager* para coletar e analisar os dados disponíveis".

No caso das viaturas da guarda municipal, o VA foi estendido para gravação de um registro da trilha sempre que uma viatura chegar ao local da ocorrência. Para isso foram programadas no simulador paradas aleatórias desses veículos durante o percurso, representando uma ocorrência atendida.

A Tabela 4 resume a dinâmica do cenário, destacando os atores e as ações. O simulador foi executado 31 vezes representando os dias do mês de janeiro, sendo geradas trilhas para os veículos nesse período.

A Figura 6(b) mostra a trilha de uma viatura conforme o período selecionado no aplicativo. Cada marcador no mapa representa uma ocorrência e, caso selecionado, são mostrados dados como a data, hora e uma descrição sobre o evento.



**Tabela 4. Dinâmica do segundo cenário (City Manager)**

<b>Ator</b>	<b>Ação</b>
Prefeitura	Registra os veículos que serão monitorados através do módulo <i>Portal</i>
Veículos	A cada 30 segundos o VA instalado no veículo envia as informações sobre localização
Viatura	Sempre que a viatura parar é acionada a função do VA para criar um registro da trilha informando a ocorrência
Services	Recebe as requisições dos VA e registra os dados no perfil e na trilha
Prefeitura	Abre o aplicativo <i>City Manager</i> e visualiza os veículos em tempo real
City Manager	Faz as requisições no <i>Services</i> para buscar os dados dos veículos
Services	Busca as informações no perfil do veículo
City Manager	Mostra o mapa para usuário visualizar os veículos cadastrados
Prefeitura	Busca a trilha de uma viatura filtrando por uma data e hora inicial e final
City Manager	Chama o serviço do módulo <i>Services</i> para buscar a trilha
Services	Consulta o módulo <i>Trail</i> retornando a trilha do período informado
Prefeitura	Visualiza no mapa a trilha do período informado

## 5. Trabalhos Relacionados

Na área de Sistemas Inteligentes de Transporte destaca-se o i-WAY [Rusconi et al. 2007], um sistema que coleta e utiliza as informações vindas de sensores instalados nos veículos e na rodovia. Com essas informações, o sistema fornece uma ferramenta para gestão do tráfego otimizado, contribuindo para a redução de acidentes e fornecendo informações atualizadas sobre o ambiente.

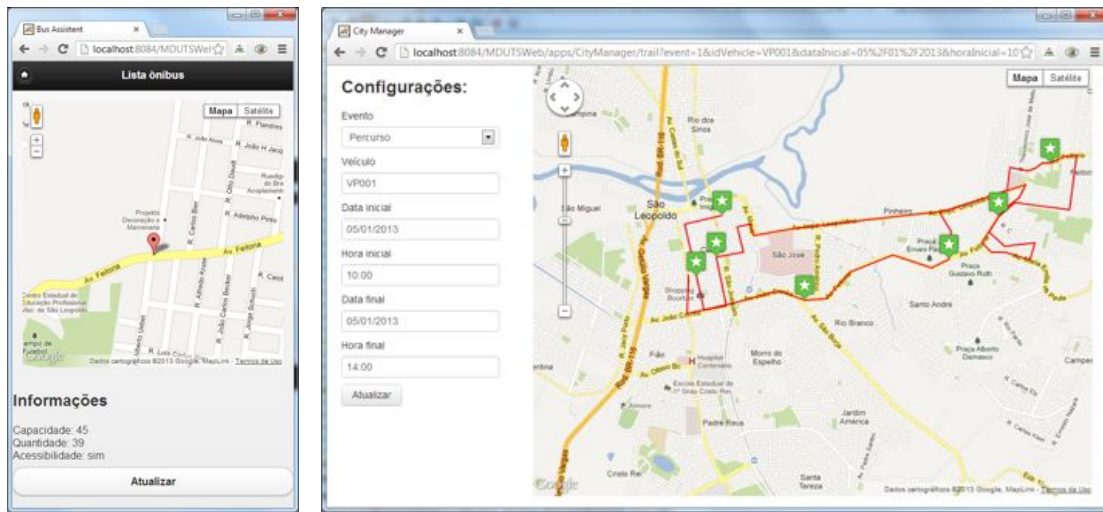
Outro trabalho de destaque é o projeto CVIS [Toulminet et al. 2008] que tem como objeto conceber, desenvolver e testar tecnologias para permitir que veículos se comuniquem de forma segura um com o outro e com a infraestrutura da rodovia. Para alcançar esse objetivo, o CVIS usa o padrão internacional de comunicação chamado de CALM [Ernst et al. 2009].

O OneBusAway [Ferris et al. 2010] é um conjunto de ferramentas desenvolvidas pela Universidade de Washington para fornecer dados sobre as chegadas e partidas de ônibus em Seattle, com o objetivo de ajudar os passageiros que usam frequentemente esse transporte ou que ainda não conhecem o caminho e as paradas do ônibus.

Da mesma forma que o OneBusAway, o Olho Vivo [SPTrans 2009] tem como objetivo visualizar em tempo real a localização dos ônibus na cidade de São Paulo (Brasil), obtendo informações como a quantidade de minutos que o ônibus vai levar até chegar na parada.

O projeto SIMTUR [SIMTUR 2011] visa desenvolver um sistema integrado de soluções para fornecer informações de apoio aos usuários de transporte coletivo e privado. Para os passageiros de transporte coletivo, são disponibilizadas informações relativas aos itinerários e rastreamento de veículos. No caso dos motoristas de veículos, são disponibilizadas informações sobre o tráfego e condições das vias urbanas, além de um serviço de comunicação segura entre veículos.

Através da revisão bibliográfica desse artigo [Rusconi et al. 2007]



(a) Bus Assistent

(b) City Manager

**Figura 6. Telas dos aplicativos**

[Toulminet et al. 2008] [Ferris et al. 2010] [SPTrans 2009] [SIMTUR 2011], constata-se que o MD-UTS é o primeiro modelo que atende a diversos tipos de veículos e que permite gerenciar as características específicas de cada um deles através de seu perfil, com suporte ao uso de Trilhas, conforme a Tabela 5.

**Tabela 5. Comparação entre os trabalhos relacionados**

Quesito	iWay	CVIS	OneBusAway	Olho Vivo	SIMTUR	MD-UTS
Sensível ao Contexto	Sim	Sim	Sim	Sim	Sim	Sim
Utiliza Trilhas	Não	Não	Não	Não	Não	Sim
Utiliza Perfis	Não	Não	Não	Não	Não	Sim
Público Foco	Motoristas e Governo	Motoristas, Pedestres e Governo	Pedestres	Pedestres	Motoristas, Pedestres e Governo	Motoristas, Pedestres, Governo e Empresas
Veículos Suportados	Veículos em geral	Veículos em geral	Ônibus	Ônibus	Veículos em geral	Veículos em geral
Utiliza Ontologia	Não	Não	Não	Não	Não	Sim

## 6. Considerações Finais

Esse artigo propôs um modelo para o desenvolvimento de sistemas inteligentes de transporte, com foco no gerenciamento de diferentes tipos de veículos, suporte a trilhas e perfis. O artigo propôs ainda uma ontologia para o domínio de transportes.

A implementação de um protótipo, sua integração com o SUMO e o desenvolvimento de duas aplicações, permitiu uma avaliação das funcionalidades previstas no modelo. A avaliação mostrou a viabilidade da proposta, estimulando a continuidade do trabalho.

Dentre os trabalhos futuros previstos encontram-se: i) desenvolver serviços adicionais para as aplicações, tornando assim o modelo mais flexível; ii) implementar uma camada de comunicação utilizando sensores reais no módulo Sensor do VS; iii) ampliar

a avaliação, aperfeiçoando os cenários apresentados; iv) disponibilizar o MD-UTS para uso em condições reais na cidade de São Leopoldo.

## 7. Agradecimentos

Os autores agradecem à empresa de transporte público Viação Leopoldense Ltda, pela disponibilização das informações de transporte na cidade de São Leopoldo usadas nesse artigo.

## Referências

- Almeida, M. B. and Bax, M. P. (2003). Uma visão geral sobre ontologias: Pesquisa sobre definições, tipos, aplicações, métodos de avaliação e de construção. *Ciência da Informação*, 32:7 – 20.
- Barbosa, J. L. V., Hahn, R. M., Barbosa, D. N. F., and Saccol, A. I. d. C. Z. (2011). A ubiquitous learning model focused on learner interaction. *Int. J. Learn. Technol.*, 6(1):62–83.
- Dey, A. K., Abowd, G. D., and Salber, D. (2001). A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Hum.-Comput. Interact.*, 16(2):97–166.
- Ernst, T., Nebehaj, V., and Srasen, R. (2009). Cvis: Calm proof of concept preliminary results. In *Intelligent Transport Systems Telecommunications, (ITST), 2009 9th International Conference on*, pages 80 –85.
- Ferris, B., Watkins, K., and Borning, A. (2010). Location-aware tools for improving public transit usability. *Pervasive Computing, IEEE*, 9(1):13 –19.
- Fielding, R. T. (2000). *Architectural Styles and the Design of Network-based Software Architectures*. PhD thesis, University of California, Irvine.
- Fowler, K. (2009). Sensor survey results. *Instrumentation Measurement Magazine, IEEE*, 12(1):39 –44.
- Franco, L. K., Rosa, J. H., Barbosa, J. L., Costa, C. A., and Yamin, A. C. (2011). Mucs: A model for ubiquitous commerce support. *Electronic Commerce Research and Applications*, 10(2):237 – 246. Special Issue on Electronic Auctions: Strategies and Methods.
- Hightower, J. and Borriello, G. (2001). Location systems for ubiquitous computing. *Computer*, 34(8):57 –66.
- Houda, M., Khemaja, M., Oliveira, K., and Abed, M. (2010). A public transportation ontology to support user travel planning. In *Research Challenges in Information Science (RCIS), 2010 Fourth International Conference on*, pages 127 –136.
- Kwon, O., Shin, S., Shin, S., and Kim, W. (2010). Design of u-health system with the use of smart phone and sensor network. In *Ubiquitous Information Technologies and Applications (CUTE), 2010 Proceedings of the 5th International Conference on*, pages 1 –6.
- Lee, E., Ryu, K., and Paik, I. (2008). A concept for ubiquitous transportation systems and related development methodology. In *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*, pages 37 –42.

- Miles, J. and Walker, A. (2006). The potential application of artificial intelligence in transport. *Intelligent Transport Systems, IEE Proceedings*, 153(3):183–198.
- Noy, N. F. and McGuinness, D. L. (2001). *Ontology Development 101: A Guide to Creating Your First Ontology*. Online.
- Oh, J. and Oh, S. (2011). Some aspects of the ubiquitous services on the u-city implementation. In *Mobile IT Convergence (ICMIC), 2011 International Conference on*, pages 78–81.
- Oliveira, R. R., Noguez, F. C., Costa, C. A., Barbosa, J. L., and Prado, M. P. (2013). Sw-track: An intelligent model for cargo tracking based on off-the-shelf mobile devices. *Expert Systems with Applications*, 40(6):2023–2031.
- Passos, L. S. and Rossetti, R. J. F. (2009). Intelligent transportation systems: a ubiquitous perspective. In *14th Portuguese Conference on Artificial Intelligence*, pages 27–38.
- Rusconi, G., Brugnoli, M., Dosso, P., Kretzschmar, K., Bougia, P., Fotiadis, D., Salgado, L., Jaureguizar, F., and De Feo, M. (2007). I-way, intelligent co-operative system for road safety. In *Intelligent Vehicles Symposium, 2007 IEEE*, pages 1056–1061.
- Satyanarayanan, M. (2001). Pervasive computing: vision and challenges. *Personal Communications, IEEE*, 8(4):10–17.
- Silva, J. M., Rosa, J. H., Barbosa, J. L., Barbosa, D. N., and Palazzo, L. A. (2010). Content distribution in trail-aware environments. *Journal of the Brazilian Computer Society*, 16:163–176.
- SIMTUR (2011). Simtur - sistema inteligente de monitoramento de tráfego urbano. Disponível em: <http://projeto.unisinos.br/simtur/>. Acesso em: 19 janeiro 2013.
- SPTrans (2009). Sistemas informatizados para a gestão do transporte coletivo do município de são paulo. Disponível em: [http://www.sptrans.com.br/pdf/biblioteca\\_tecnica/SISTEMAS\\_INFORMATIZADOS\\_PARA\\_A\\_GESTAO\\_DO\\_TRANSPORTE.pdf](http://www.sptrans.com.br/pdf/biblioteca_tecnica/SISTEMAS_INFORMATIZADOS_PARA_A_GESTAO_DO_TRANSPORTE.pdf). Acesso em: 19 janeiro 2013.
- Sussman, J. S. (2005). *Perspectives on Intelligent Transportation Systems (ITS)*. Springer Verlag NY.
- Toulminet, G., Boussuge, J., and Laugeau, C. (2008). Comparative synthesis of the 3 main european projects dealing with cooperative systems (cvis, safespot and coopers) and description of coopers demonstration site 4. In *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*, pages 809–814.
- Vieira, V., Caldas, L., and Salgado, A. (2011). Towards an ubiquitous and context sensitive public transportation system. In *Ubi-Media Computing (U-Media), 2011 4th International Conference on*, pages 174–179.
- Wang, J., Ding, Z., and Jiang, C. (2005). An ontology-based public transport query system. In *Semantics, Knowledge and Grid, 2005. SKG '05. First International Conference on*, page 62.
- Weiser, M. (1991). The Computer for the Twenty-First Century. *Scientific American*, 265(3):94–104.