

Smart Car Space

André Siqueira Ruela

Origem

2010 Fourth International Conference on Genetic and Evolutionary Computing

Towards A Context-aware Middleware in Smart Car Space

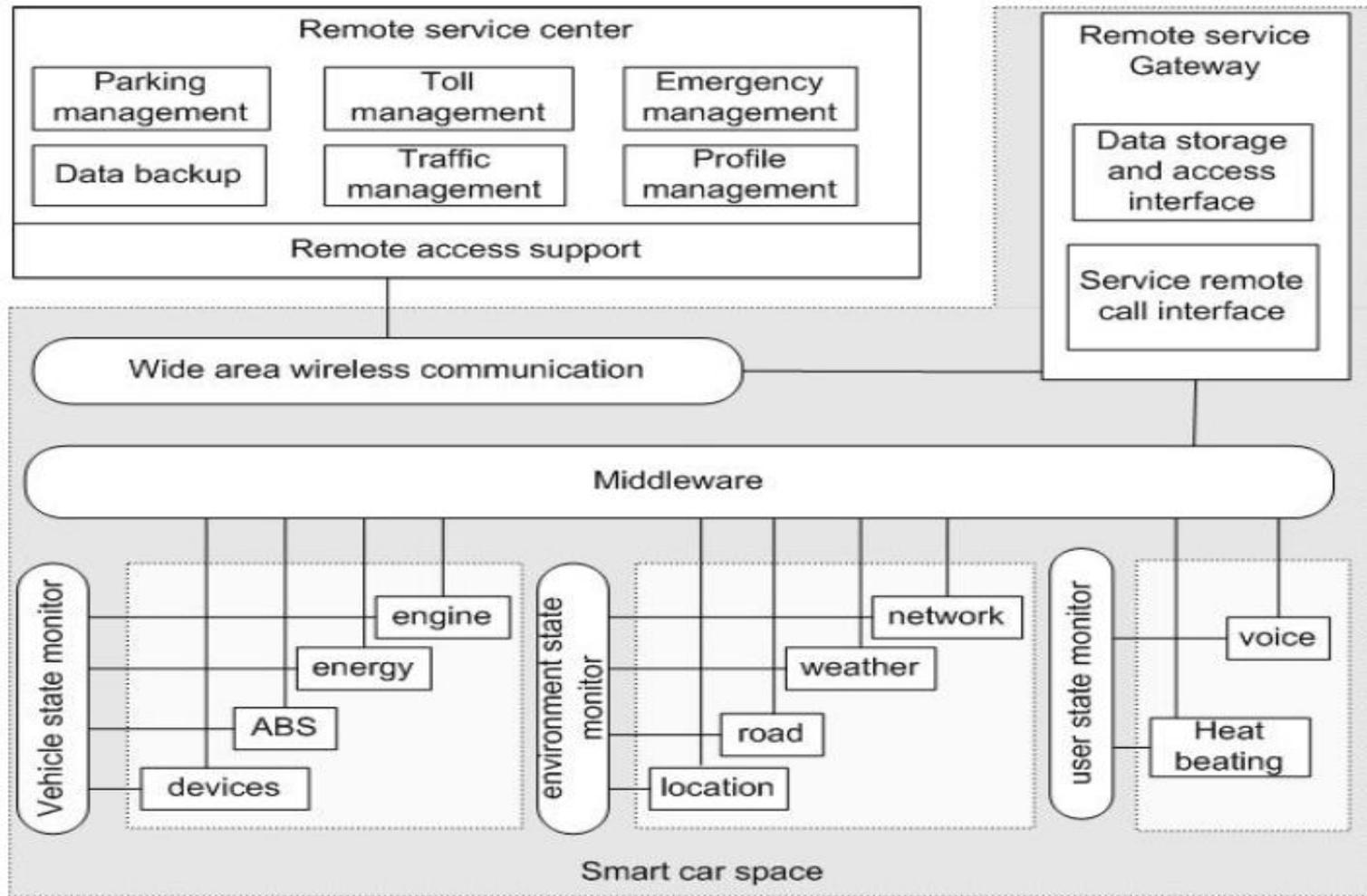
Jie Sun, YongPing Zhang, Jianbo Fan
School of Electronic and Information Engineering
Ningbo University of Technology
Ningbo, China
{sunjie, ypz}@nbut.cn, jbfan@163.com

Introdução

- ▶ Trabalhos recentes em Veículos de Transporte Inteligentes têm focado nos novos dispositivos e serviços que ajudam os motoristas em uma condução mais segura e confortável.
- ▶ Este trabalho se concentra em como construir um espaço de carro inteligente, que irá fornecer informações adequadas para o motorista para uma condução melhor e mais segura e avisar o condutor de forma passiva quando seu carro estiver em uma situação perigosa ou em possíveis situações de risco.

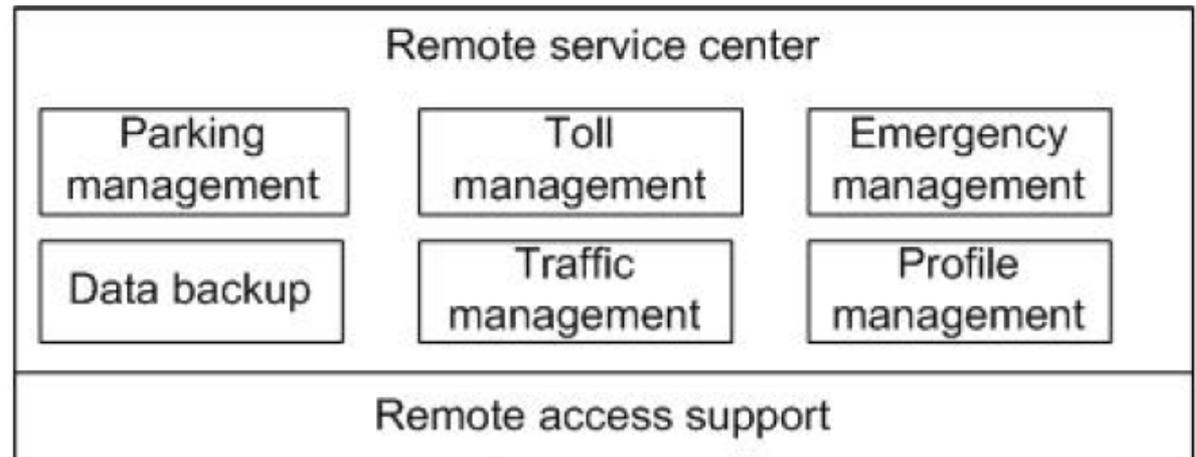


Visão Geral

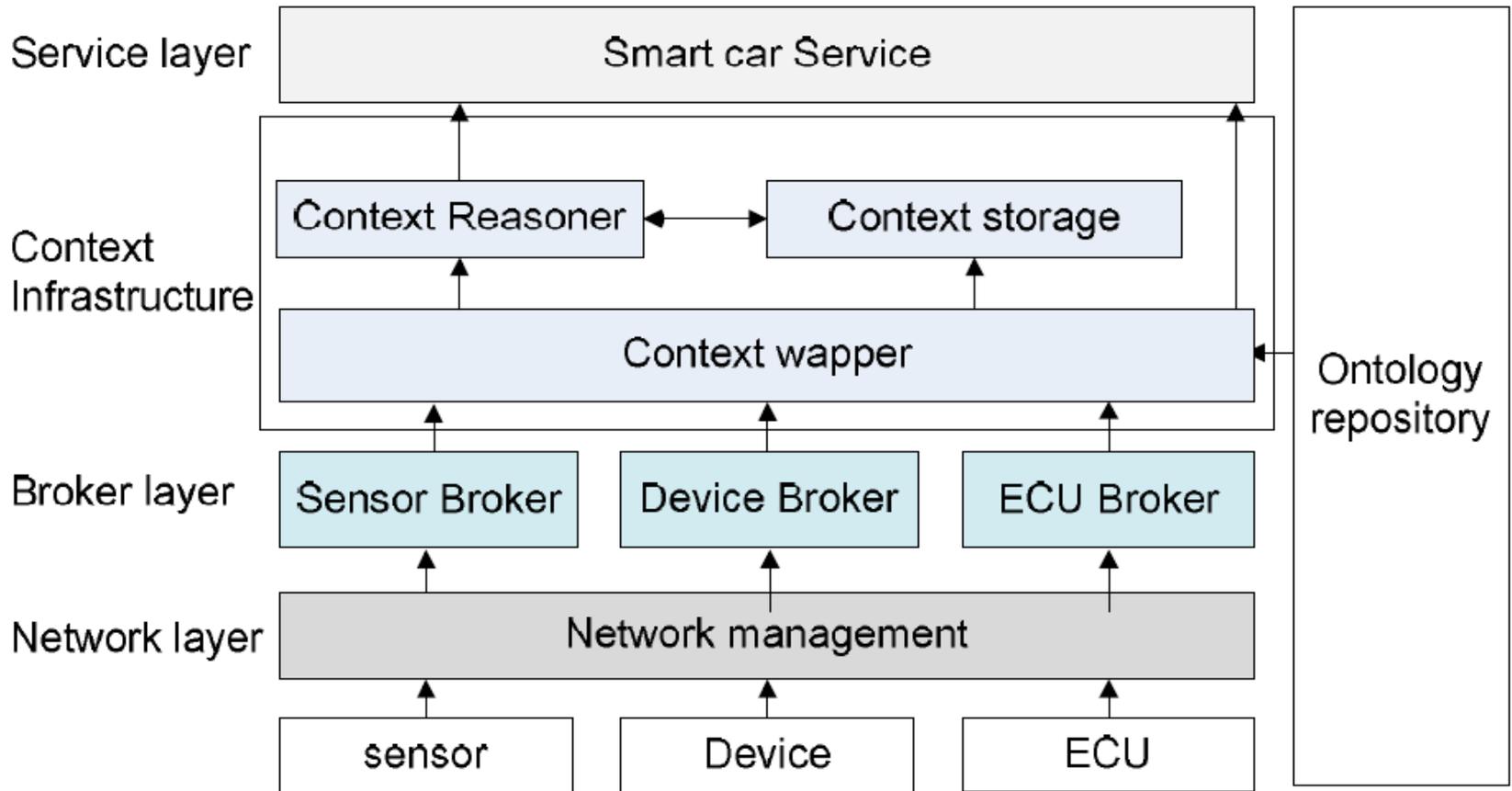


Centro de Serviço Remoto

- ▶ Garante o acesso contínuo ao serviço e implementa serviços fundamentais que podem ser acessados remotamente quando necessários.
 - Gestão de Estacionamento.
 - Gestão de Pedágio.
 - Gestão de Emergência.
 - Gestão do Tráfego.
 - Gestão do Perfil.
 - Backup dos Dados.



O middleware no carro inteligente



O middleware no carro inteligente

▶ Camada de Rede

- Suporta diferentes abordagens de comunicação para interconectar os diferentes sensores, dispositivos e demais hardwares.

▶ Camada Corretora

- Os corretores são responsáveis pela descoberta, configuração e registro de novos sensores, dispositivos ou hardwares adicionados ao carro inteligente.

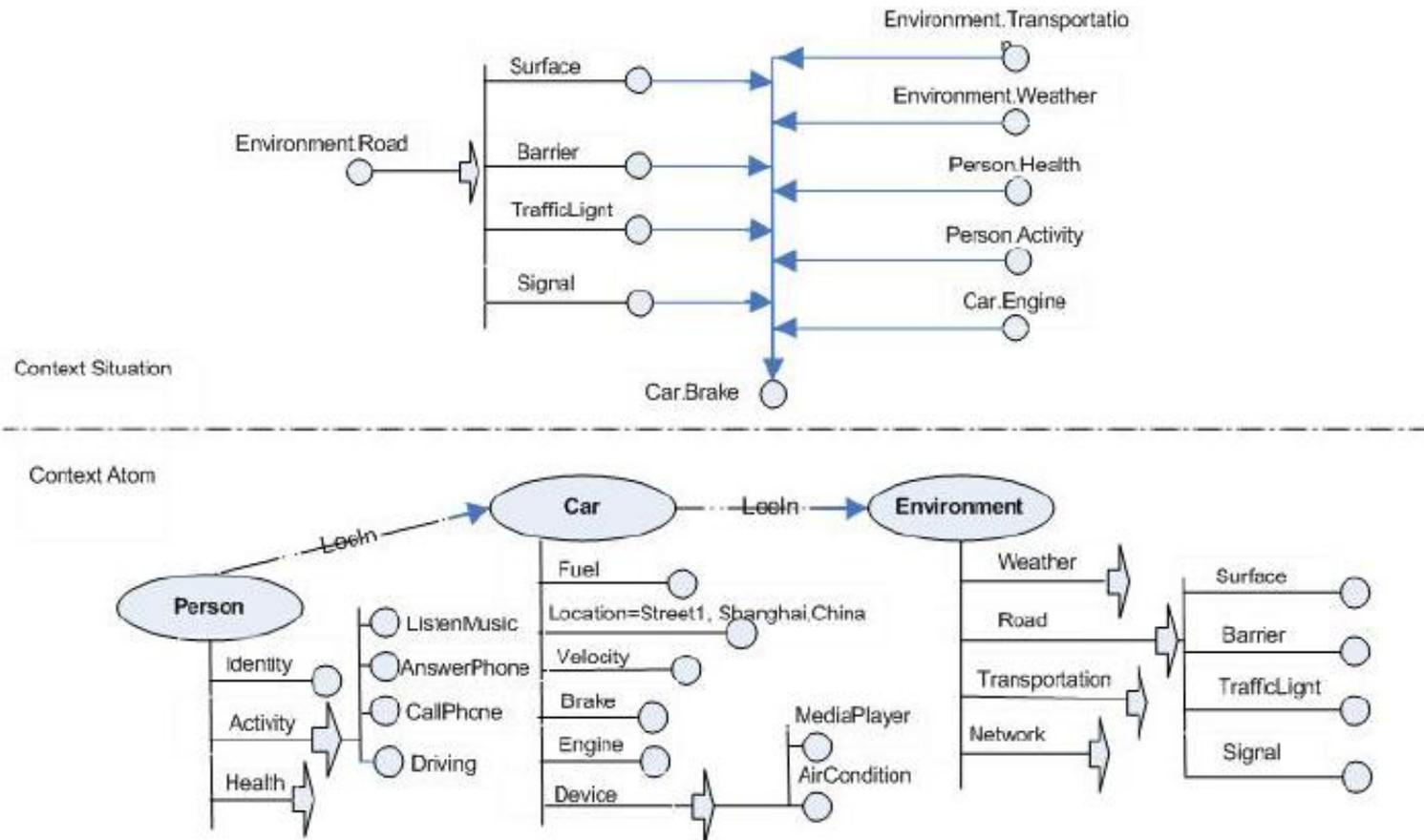
▶ Infraestrutura do Contexto

- Encapsulador: Transforma os dados dos sensores em átomos de contexto semântico.
- Raciocinador: Treina e reconhece situações de contexto pela agregação de vários tipos de átomos de contexto.
- Armazenador: Armazena contextos históricos e fornece serviços para manipulação de queries.

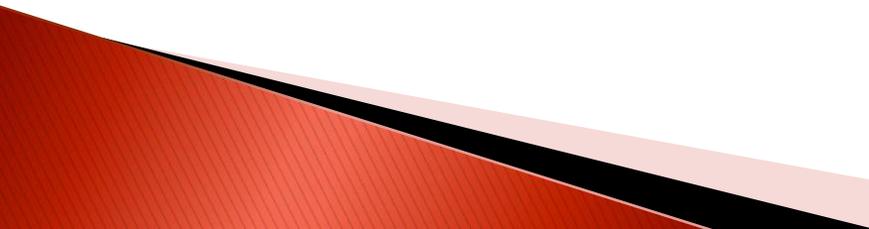
▶ Camada de Serviço:

- Realiza o controle e troca de mensagens sobre os atuadores.

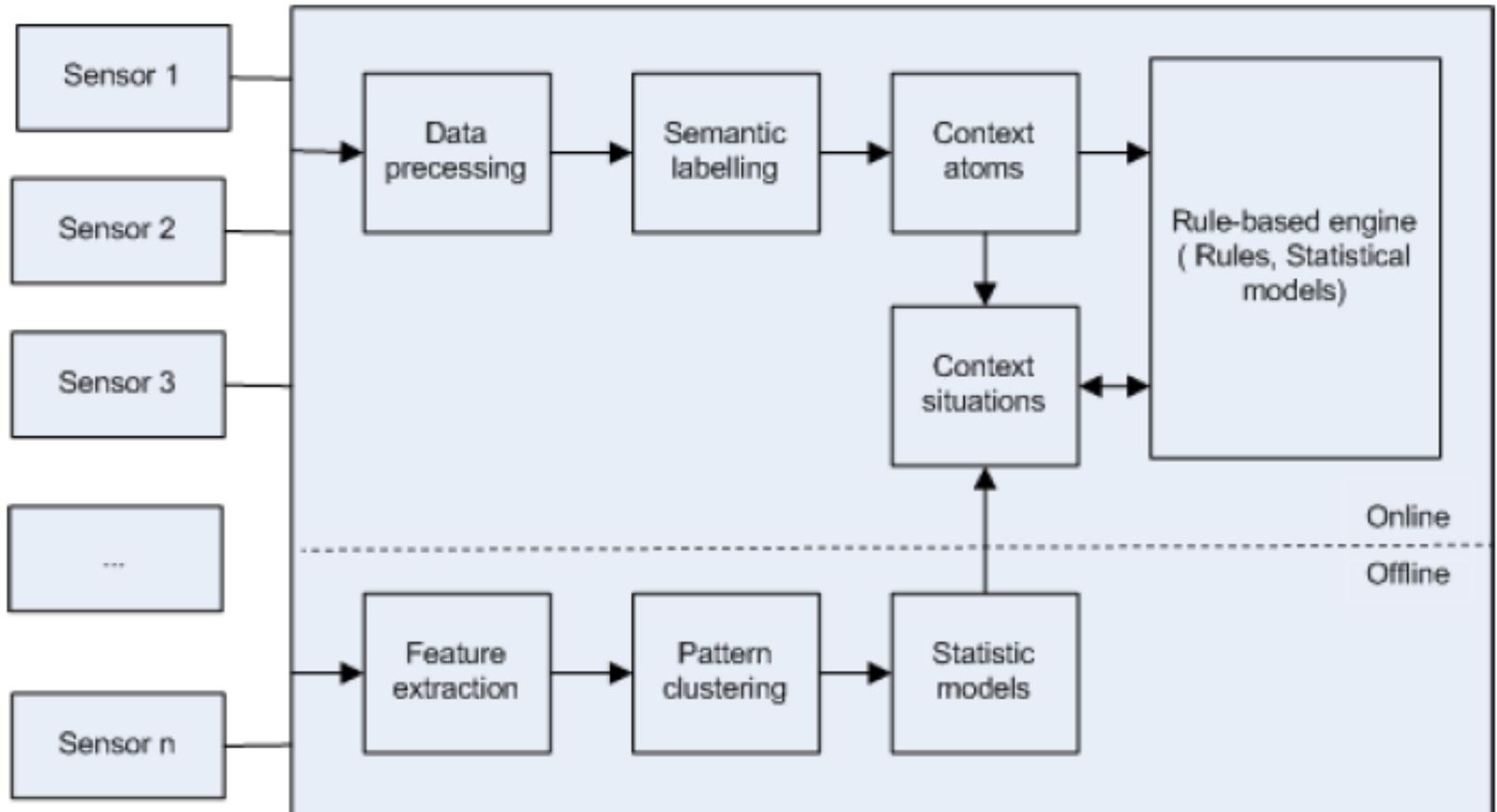
Encapsulador de Contextos



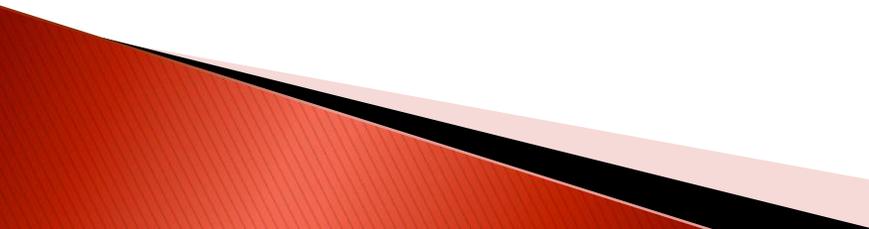
Encapsulador de Contextos

- ▶ O Encapsulador de Contexto é responsável por extrair contexto dos sensores físicos ou semânticos por meio da verificação periódica dos sensores.
 - ▶ Os dados primários do sensor são dados numéricos e sem nenhum sentido para as aplicações, por isso devemos recuperar contextos significativos a partir dos dados do sensor.
 - ▶ Contexto é qualquer informação que pode ser utilizada para caracterizar a situação de uma entidade.
 - ▶ De acordo com o grau de abstração e semântica, podemos dividir o contexto em átomo de contexto e situação de contexto.
- 

Raciocinador de Contextos

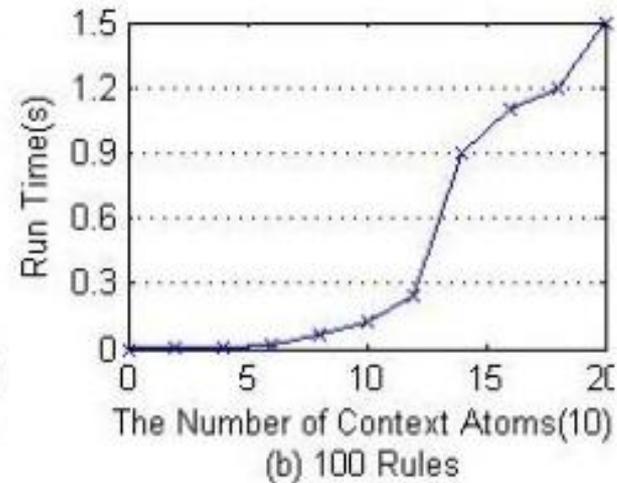
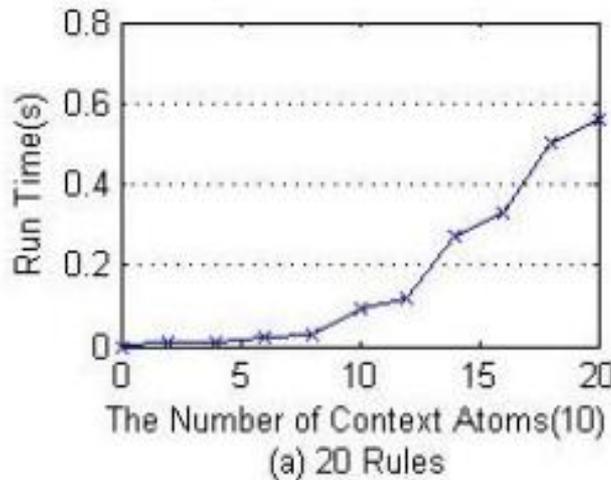


Raciocinador de Contextos

- ▶ O Raciocinador insiste em deduzir situações de contexto de alto nível, a partir de átomos de contexto de baixo nível. O Raciocinador foi implementado utilizando predicados de primeira ordem.
 - ▶ Regras são construídas com duas propriedades: Ação e Condição. A ação é uma declaração que representa o que deve ser feito se a condição for verdadeira. Condição em uma regra é uma coleção de átomos de contexto.
 - ▶ Usamos este Raciocinador para acionar a transição de situação e o serviço adequado, como ligar o ar-condicionado quando está muito quente.
 - ▶ Usamos um algoritmo de aprendizagem de máquina para reconhecer a situação atual.
- 

Avaliação do Desempenho

- ▶ Para determinar o desempenho do middleware, usamos as métricas de desempenho: qualidade de contexto e desempenho do Raciocinador.
- ▶ Qualidade de Contexto: “Frescura” e Precisão.
- ▶ Desempenho do Raciocinador:



Avaliação do Desempenho

- ▶ O desempenho em tempo de execução do Raciocinador depende muito do tamanho do contexto e da complexidade dos conjuntos de regras.
- ▶ O raciocínio é uma tarefa computacional intensiva. No entanto, é ainda viável para aplicações que não sejam críticas em relação ao tempo, de modo que o atraso de Raciocinador (menos de 2 segundos) é aceitável.
- ▶ Em aplicações críticas em relação ao tempo, como segurança e sistemas de navegação, precisamos controlar a escala do conjunto de dados e da complexidade do conjunto de regras.

Conclusões

- ▶ Como uma área de aplicação promissora da computação ubíqua, o carro inteligente está se tornando cada vez mais importante.
- ▶ A contribuição deste artigo é um middleware sensível ao contexto, para o carro inteligente, que é um framework abrangente baseado em componentes com o apoio de alta adaptabilidade.