

Uma Contribuição ao Emprego de Ontologias no Sensoriamento de Contexto na Computação Ubíqua

Por
Anderson Medeiros Gomes

Orientador
Adenauer Corrêa Yamin
Prof. G3PD

Co-orientador
Luthiano Rodrigues Venecian
Mestrando G3PD

Dezembro de 2009

1. Objetivos
2. Computação Ubíqua
3. Sensibilidade ao contexto
4. Ontologias: fundamentos e tecnologias
5. EXEHDA – *middleware* para Computação Ubíqua
6. EXEHDA-SS: modelagem e implementação
7. Considerações finais



Objetivos

- Objetivo geral:
 - Contribuir com a modelagem e a implementação do **EXEHDA-SS**, *framework* para suporte à sensibilidade ao contexto na Computação Ubíqua, em tempo de execução.

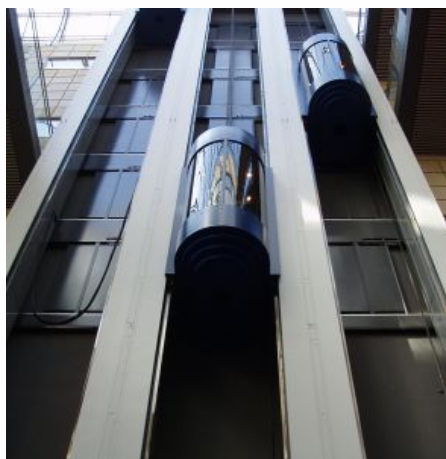
- Objetivos específicos:
 - Mapear características de um *framework* de sensibilidade de contexto;
 - Colaborar com a modelagem e implementação do *framework*, contemporizando flexibilidade e simplicidade;
 - Estreitar compatibilidade com o *middleware* EXEHDA e, também, com padrões de mercado, particularmente com a plataforma Java.



Computação Ubíqua

- Tópicos iniciais:
 - *Mainframes* -> Computação Pessoal -> Computação Ubíqua;
 - Definida por Mark Weiser, na década de 1990;
 - Computação Móvel + Computação Pervasiva;
 - Premissa *sigame*.

- A Computação Ubíqua tem o objetivo de inserir a Computação **no cotidiano da sociedade**, da maneira que estão os motores elétricos hoje.



- Características perseguidas em um cenário ubíquo:
 - Invisibilidade;
 - Pró-atividade;
 - Sensibilidade ao contexto;
 - Interfaces naturais;
 - Descentralização;
 - Heterogeneidade de hardware e software.



Sensibilidade ao contexto

- Consciência do contexto:
 - Ações e comportamentos das pessoas são determinadas pelo conhecimento do contexto;
 - A consciência do contexto torna as aplicações mais amigáveis, flexíveis e fáceis de usar;
 - É tema de pesquisa em várias áreas da computação.

- Principais definições de contexto:
 - Inter-relação de circunstâncias que acompanham um fato ou situação;
 - Qualquer informação que pode ser usada para caracterizar uma situação de uma entidade;
 - Entidade → tudo o que é relevante na interação homem-máquina.
 - Conjunto de estados e características de um ambiente que determina o comportamento de uma aplicação ou no qual um evento de uma aplicação ocorre e interessa ao usuário;
 - Circunstâncias ou situações em que uma tarefa computacional está inserida.

- Categorias de contexto:

- Contexto computacional

- Conectividade de rede;
 - Custos de comunicação;
 - Largura de banda;
 - Recursos disponíveis, como impressoras, processadores e memória.

- Contexto do usuário

- Perfil do usuário;
 - Localização;
 - Pessoas próximas a ele;
 - Humor.

- Contexto físico

- Luminosidade;
 - Níveis de barulhos;
 - Condições do trânsito;
 - Temperatura.

- Contexto temporal

- Tempo (hora do dia, da semana, do mês, estação do ano);
 - Histórico de contexto.

- Sensibilidade e adaptação:
 - Infra-estrutura de software sensível ao contexto: múltiplos perfis e caminhos de execução, sob governança de políticas de adaptação.
 - Transformações de adaptação:
 - Variação de filtros entre cliente e servidor;
 - Alteração da fidelidade de saída;
 - Alterações nas dimensões da interface;
 - Migração de código ou de dados.
 - Processo de adaptação: detecção, escolha e ativação.

- Desafios na concepção de suporte a sensibilidade ao contexto:
 - 1) Identificação de elementos de contexto;
 - 2) Características das informações contextuais: validade, exatidão, representação, dimensão;
 - 3) Aquisição de contexto;
 - 4) Modelagem de contexto;
 - 5) Interpretação do contexto;
 - 6) Processamento e raciocínio;
 - 7) Armazenamento das informações contextuais.



Ontologias: fundamentos e tecnologias

- Ontologia: uma especificação formal e explícita de uma conceituação compartilhada.
 - Conceituação - se refere ao modelo abstrato do mundo real;
 - Explícita - os conceitos e seus requisitos são definidos explicitamente: definições de conceitos, instâncias, relações, restrições e axiomas;
 - Formal - indica que a ontologia é processável por máquina, permite raciocínio automático e possui semântica lógica formal;
 - Compartilhada - significa que uma ontologia captura o conhecimento apresentado não apenas por um único indivíduo, mas por um grupo, sendo uma terminologia comum da área modelada ou acordada entre os desenvolvedores.

- Linguagens ideais para a representação de ontologias possuem:
 - Sintaxe bem definida;
 - Semântica formal;
 - Suficiente potencial de expressividade;
 - Suporte eficiente a raciocínio.

- *OWL - Web Ontology Language:*
 - Recomendação do W3C para a representação de ontologias;
 - Sintaxe baseada em XML;
 - Arquitetura de metadados baseada em RDF, estendida via RDF Schema;

- Tecnologias para o tratamento de ontologias:
 - Protégé – ferramenta extensível e independente para o tratamento de ontologias;
 - Jena API – interface de programação de aplicações desenvolvida para a manipulação de ontologias em tempo de execução;
 - SPARQL – linguagem de consulta e protocolo de acesso a dados em RDF.



EXEHDA - *middleware* para Computação Ubíqua

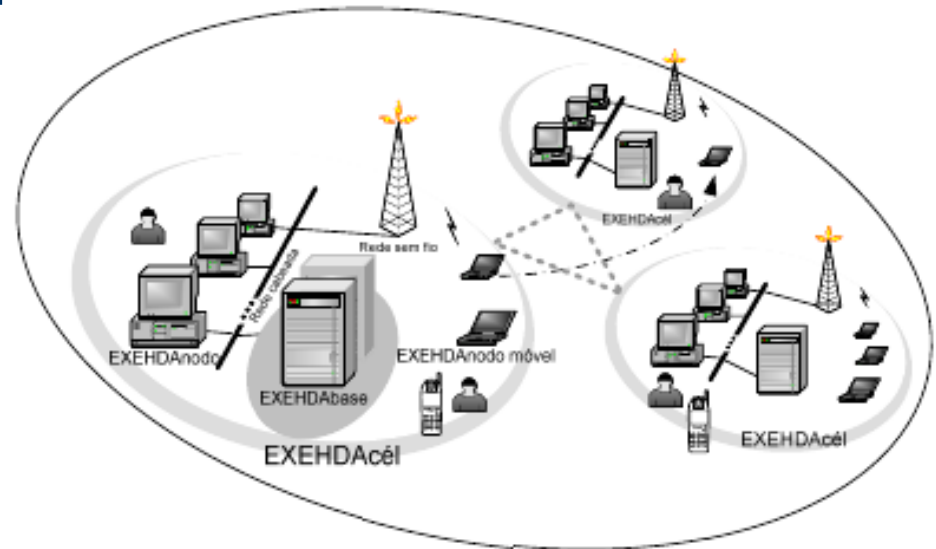
- EXEHDA

- *Execution Environment for Highly Distributed Applications*
(ambiente de execução para aplicações altamente distribuídas)
- *Middleware* direcionado às aplicações distribuídas, móveis e conscientes do contexto da Computação Ubíqua;
- Baseado em serviços, que tem por objetivo criar e gerenciar um ambiente ubíquo;
- Desenvolvido pelo G3PD/UCPel em conjunto com universidades gaúchas.

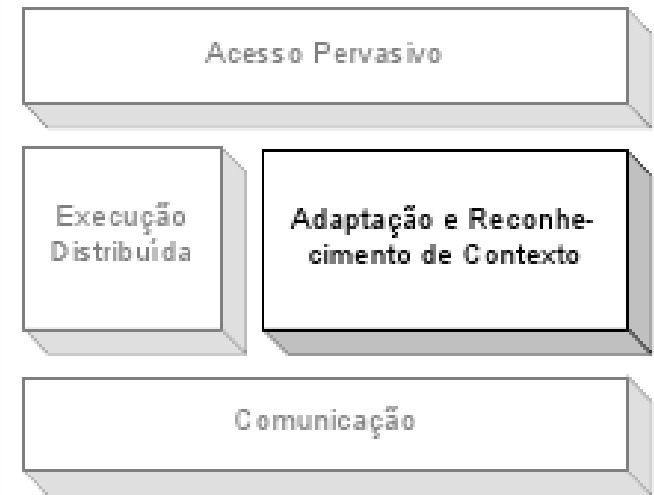
- Premissas perseguidas durante a fase de concepção do EXEHDA:
 - Facultar ao desenvolvedor o modo com que a adaptação é gerenciada: totalmente pelo *middleware*, totalmente pela aplicação ou via uma estratégia colaborativa, com maior ou menor participação da plataforma de execução;
 - Prover suporte às mobilidades lógica e física;
 - Suportar a semântica *sigame*, contemplando adaptação funcional e não-funcional.

• Organização do EXEHDA:

- **EXEHDA_{nodo}** e **EXEHDA_{nodo móvel}** – elemento organizacional básico, onde ocorre a interação com o usuário e a execução de aplicações ubíquas;
- **EXEHDA_{base}** – ponto de contato dos *EXEHDA_{nodos}*;
- **EXEHDA_{acél}** – área de atuação de uma *EXEHDA_{base}*.



- Serviços do ambiente EXEHDA:
 - São organizados em 4 grandes subsistemas;
 - O **Subsistema de Reconhecimento de Contexto e Adaptação** implementa as funcionalidades de reconhecimento e adaptação ao contexto.

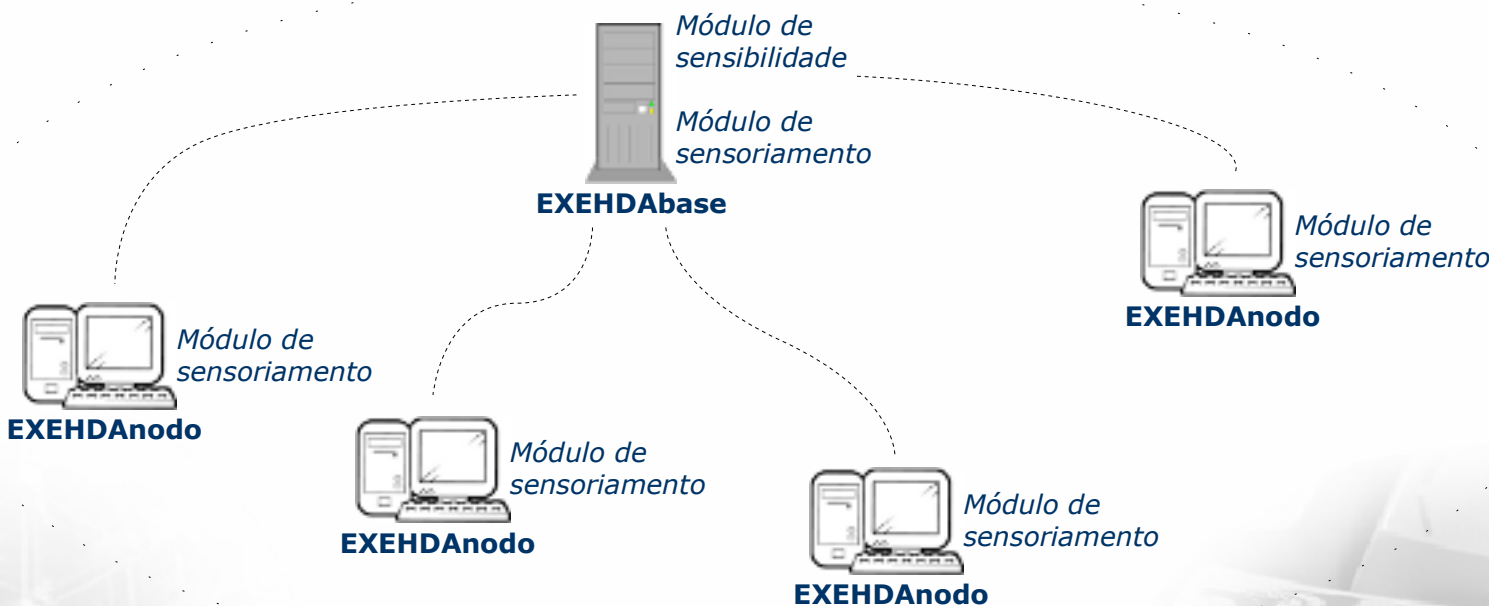




EXEHDA-SS: modelagem e implementação

- EXEHDA-SS:
 - *Framework* direcionado ao *middleware* EXEHDA para suporte à sensibilidade ao contexto na Computação Ubíqua, em tempo de execução;
 - Realiza tarefas relacionadas a captura e processamento de informações contextuais, bem como notificação de aplicações;
 - Implementado na linguagem Java;
 - Contempla o emprego de ontologia na representação de elementos internos, que é manipulada com a JENA API;

- Componentes:
 - Módulo de sensoriamento
 - Módulo de sensibilidade



- Módulo de sensoriamento
 - Fornece ao módulo de sensibilidade uma interface para configuração de sensores e captura de informações contextuais estáticas e dinâmicas.

- Sensores implementados:
 - **CpuSensor** - sensor capaz de capturar o número de núcleos de processamento do nodo, a frequência de *clock* e a porcentagem de utilização de cada núcleo.
 - Implementações avaliadas:
 - **CpuSensorProcImpl** - extrai informações a partir do pseudo-arquivo */proc/stat*, disponível em sistemas GNU/Linux.
 - **CpuSensorSigarImpl** - utiliza a biblioteca externa *Hyperic's System Information Gatherer* (SIGAR) para a captura das informações dos processadores.

- Sensores implementados (2):
 - **MemorySensor** - sensor capaz de capturar a quantidade disponível e total de memória física e *swap* do sistema.
 - Implementações avaliadas:
 - **MemorySensorProcImpl** - extrai informações a partir do pseudo-arquivo */proc/meminfo*, disponível em sistemas GNU/Linux.
 - **MemorySensorSigarImpl** - utiliza a biblioteca externa *Hyperic's System Information Gatherer* (SIGAR) para a captura das informações da memória.

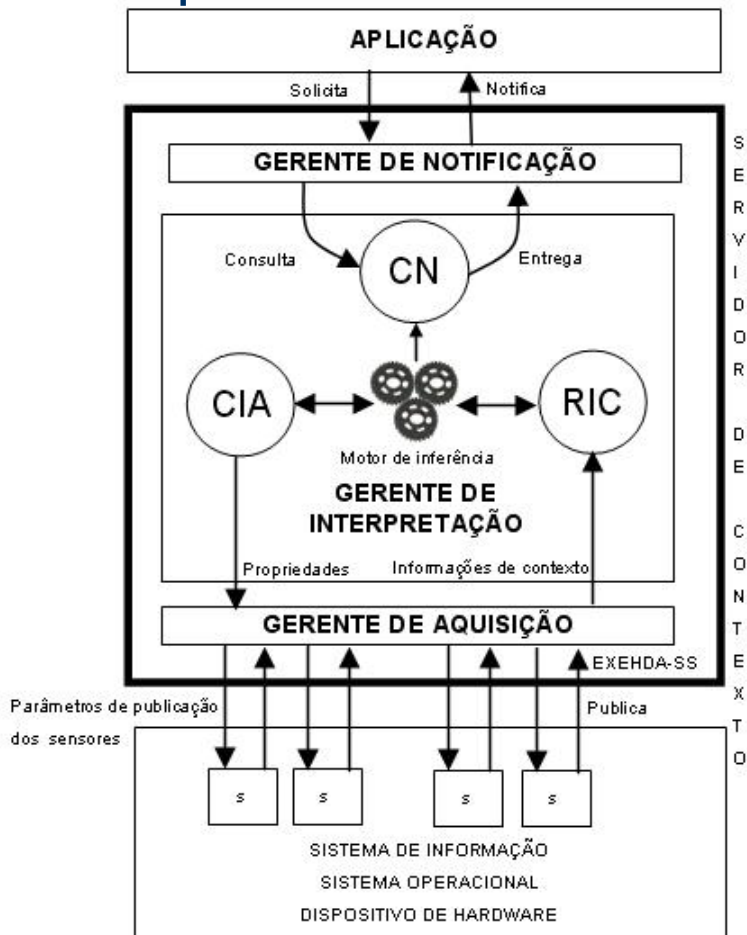
- Sensores implementados (3):
 - **DiskSensor** - sensor capaz de capturar a quantidade disponível e total de espaço de armazenamento existente na partição que hospeda o módulo de sensoriamento.
 - Implementações avaliadas:
 - **DiskSensorCmdImpl** - extrai informações a partir do utilitário *GNU stat*, disponível em sistemas GNU/Linux.
 - **DiskSensorSigarImpl** - utiliza a biblioteca externa *Hyperic's System Information Gatherer* (SIGAR) para a captura das informações de armazenamento.

- Sensores implementados (4):
 - **TemperatureSensor** - sensor capaz de capturar as informações dos sensores de temperatura instalados na *motherboard* do EXEHDA nodo.
 - Implementação avaliada:
 - **TemperatureSensorCmdImpl** - extrai informações a partir do utilitário *lm-sensors*, existente em sistemas GNU/Linux.

- Módulo de sensibilidade

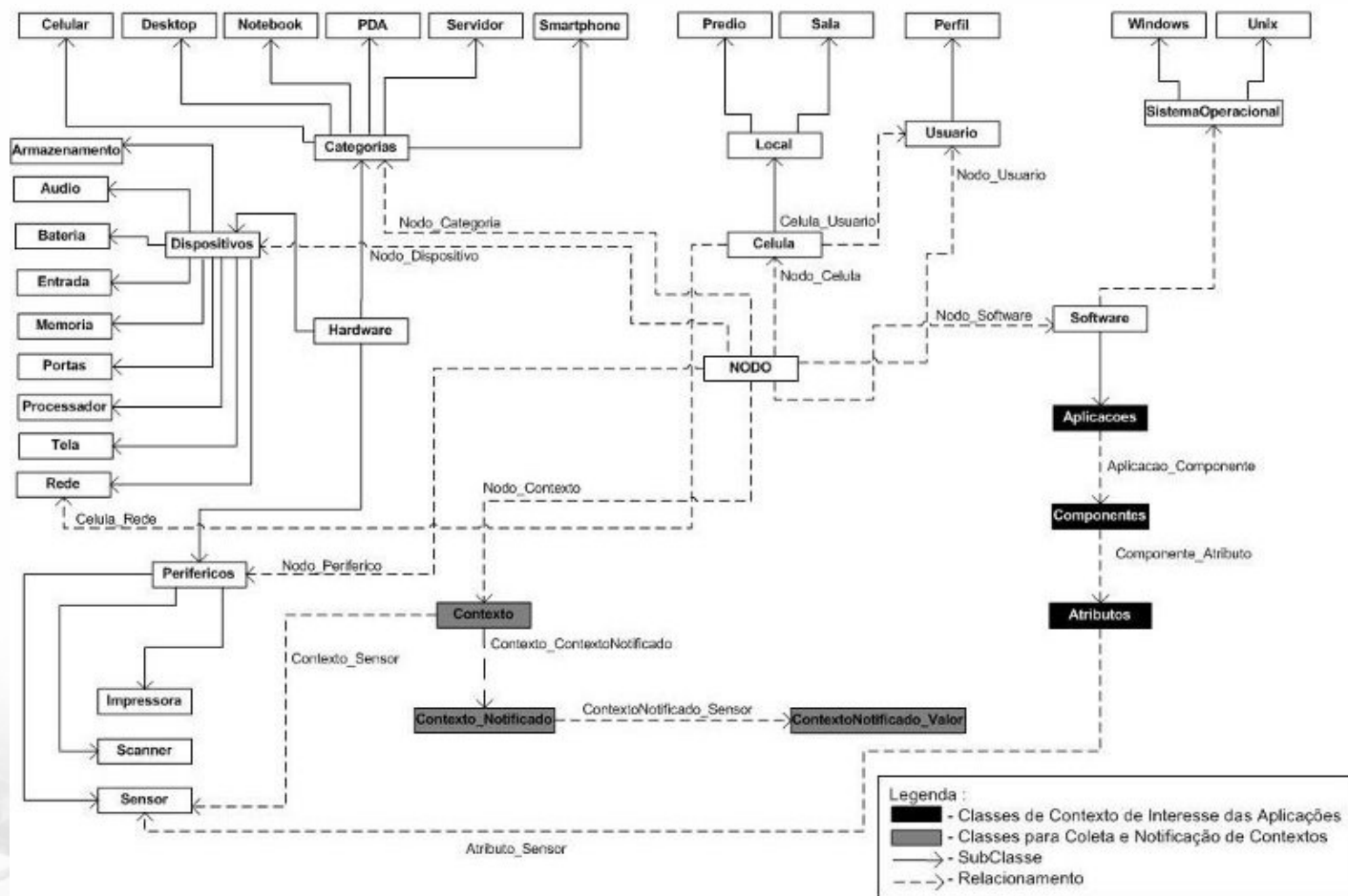
- Instancia e manipula a ontologia base do EXEHDA-SS;
- Gerencia registros de contexto de interesse de aplicações;
- Produz e envia parâmetros de configuração aos sensores - frequência de publicação, flutuação mínima de notificação;
- Obtém informações sensorizadas remetidas pelos módulos de sensoriamento dos EXEHDA nodos;
- Produz novas informações contextuais via um processo contínuo de tradução e inferência;
- Percebe alterações de contexto e remete notificações de alteração às aplicações sensíveis ao contexto modificado.

Arquitetura do módulo de sensibilidade do EXEHDA-SS:

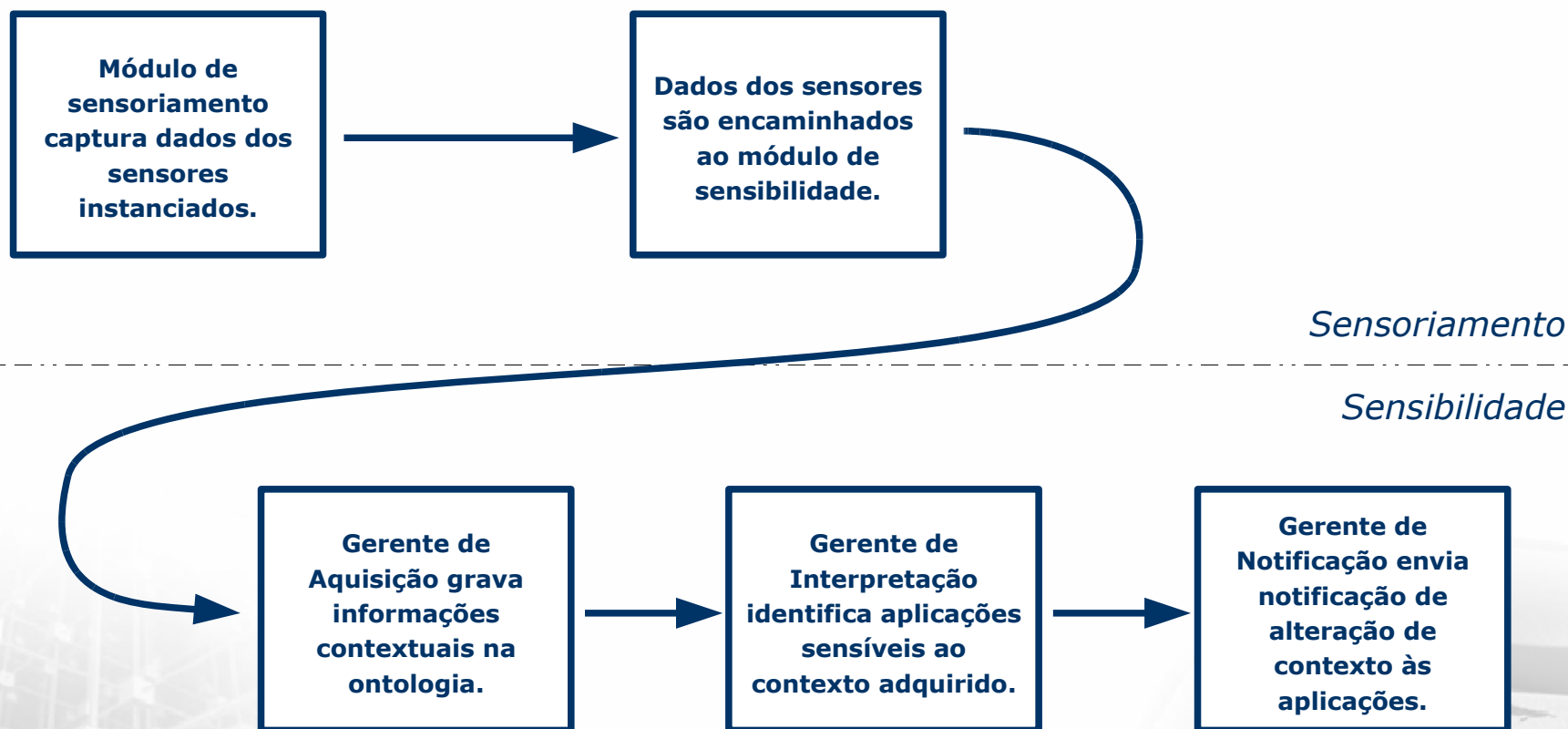


- **CIA** - Contexto de Interesse de Aplicações;
- **RIC** - Repositório de Informações Contextuais;
- **CN** - Contexto Notificado;
- **Gerente de Aquisição** - ativa e configura os componente de sensoriamento de cada EXEHDA nodo e, periodicamente, captura dos mesmos as suas informações contextuais;
- **Gerente de Interpretação** - avalia as informações extraídas, efetua refinamentos, infere novos conhecimentos, percebe alterações de contexto;
- **Gerente de Notificação** - notifica aplicações sensíveis ao contexto sobre alterações ocorridas.

· OntUbi: Ontologia do ambiente ubíquo



· Fluxo de execução do EXEHDA-SS





Considerações finais

- Tópicos estudados:
 - Computação Ubíqua;
 - Adaptação e sensibilidade ao contexto;
 - Ontologias;
 - EXEHDA;
 - Linguagem Java;
 - Bibliotecas: Jena API, SIGAR.

- Concepção do EXEHDA-SS:
 - Módulo de sensoriamento
 - Sensores
 - Módulo de sensibilidade
 - Gerente de Aquisição
 - Gerente de Interpretação
 - Gerente de Notificação
 - Ontologia do ambiente ubíquo

· Trabalhos futuros:

- Concluir a incorporação do EXEHDA-SS à arquitetura do EXEHDA;
- Implementar sensores adicionais, perseguindo portabilidade;
- Modelar interface do Gerente de Notificação com serviços públicos de mensagem, tais como *SMS*, *Google Talk*, *e-mail* e outros;
- Modelar e desenvolver uma API para a gerência dos Contextos de Interesse das Aplicações.

Universidade Católica de Pelotas
Centro Politécnico
Bacharelado em Ciência da Computação

Uma Contribuição ao Emprego de Ontologias no Sensoriamento de Contexto na Computação Ubíqua

Por
Anderson Medeiros Gomes

Orientador
Adenauer Corrêa Yamin
Prof. G3PD

Co-orientador
Luthiano Rodrigues Venecian
Mestrando G3PD

Dezembro de 2009