



CENTRO PAULA SOUZA

**Faculdade de Tecnologia de Americana
Curso Superior de Bacharelado em Análise de Sistemas e
Tecnologia da Informação**

COMPUTAÇÃO UBÍQUA E SENSIBILIDADE AO CONTEXTO

DIEGO HENRIQUE DIAS

Americana, SP

2013

COMPUTAÇÃO UBÍQUA E SENSIBILIDADE AO CONTEXTO

DIEGO HENRIQUE DIAS

diegohdias@hotmail.com

Trabalho de Conclusão de Curso, prestando à Faculdade de Tecnologia de Americana como parte das exigências do curso de Análise de Sistemas e Tecnologia da Informação para a obtenção do título de Bacharel em Análise de Sistemas.

Orientador: Prof. Dr. José Luís Zem

Área: Computação móvel e ubíqua, computação sensível ao contexto.

Americana, SP

2013

BANCA EXAMINADORA

**Prof. Dr. José Luis Zem
(Orientador)**

Prof. César Augusto Crócomo

Prof. Dra. Maria Victória Guinle Vivacqua

Dedico esse trabalho aos meus pais por me oferecerem todo o apoio e amor do mundo.

AGRADECIMENTOS

Aos Srs. Mário Orlando Pértile Guimarães, Giovane Giacomini e Paulo Eduardo Moro por me oferecerem todo o suporte necessário para que eu pudesse ingressar no mundo acadêmico.

Aos meus pais, por me permitirem ousar e acreditar em um novo caminho a ser traçado.

Aos meus amigos acadêmicos Bruno Campos de Azevedo, Bruno Eduardo Von Ah, Joel Vasconcelos Junior, Tamara Campidelli de Freitas, Natália Galante, Josiane Rosa de Oliveira Gaia, Renan Baldin, Naiade Lameirão, Amaury Borges, Tiago Altomare, Rodrigo Brito, Marshall Nascimento, Lucas Custódio, Gustavo Berno, Mateus Scaramal, Alexander, Pedro Paulo Pinto (PP), Tabata Barbam, Guilherme Sato, Felipe Henrique de Arruda, Marcão, Fernando Dias, Romano e a todos por todo apoio e cumplicidade nesses anos de vida acadêmica.

Aos meus amigos Robson Camargo Bueno (Caborja), Vitor Chavari, Vinicius Chavari, Gustavo Sartori, Bruna Nicartro, Redman Rey Monte, Eduardo Arine, Fran, Júlio Tablas, Mayara Veiga, José Ricardo, Ceci Ortega, Gustavo Muniz, Wagner Mache, Henrique Banov, Thiago Pereira, Hiago da Silva, Valdir Crepaldi, Rodrigo Coxinha, Renan Ogava, Carlos Schnaider, Mateus Tarley, Guto Barbosa, Ademir Pereira, Raul e Henrique Müller, Giovanni Stefano, Thierry Lopes Correia, Vico, Nenê, André, Ricardo e Andrey. Resumindo, a todos com quem convivo, desculpe se esqueci alguém.

Ao professor José Luis Zem que aceitou me orientar nesse trabalho, compartilhando experiências e conhecimento. Sou extremamente grato.

Ao professor Rossano Pablo Pinto que me apresentou de maneira peculiar o mundo científico e a computação ubíqua.

Enfim, agradeço a todos aqueles que acreditaram e duvidaram de mim. Sem mais.

“A simbólica e frágil arte da existência, nada mais é que o som da perseverança.” (Chuck Schuldiner).

RESUMO

A computação móvel e a ubíqua surgiram devido à miniaturização dos dispositivos e da conectividade sem fio. Sucintamente falando, a computação móvel engloba a exploração da conexão de dispositivos portáteis e a computação ubíqua à exploração da integração cada vez mais predominante dos dispositivos de computação no em objetos do cotidiano.

A computação sensível ao contexto, subtipo da computação móvel e ubíqua, contém ferramentas que lhe permitem reconhecer o ambiente em que estão inseridas, inferindo sobre o contexto de execução para realizar adaptações comportamentais perante as informações do espaço inteligente.

Com esses conceitos, observa-se o uso significativo de dispositivos móveis e ubíquos no monitoramento de saúde humana através de sinais fisiológicos provenientes de sensores não-invasivos e/ou invasivos. Os sensores não-invasivos são utilizados para monitoramento pervasivo, onde os recursos computacionais são acoplados ao corpo humano e consegue monitorar dados referentes à pressão arterial, frequência cardíaca, atividade muscular, atividade cerebral, etc. Os sensores invasivos são implantados ao corpo humano e utilizam-se da nanotecnologia para monitorar o funcionamento interno, auxiliando na prevenção de doenças e na detecção de anormalidades.

Palavras Chave: computação pervasiva, computação ubíqua, computação sensível ao contexto.

ABSTRACT

Mobile computing and ubiquitous arose due to device miniaturization and wireless connectivity. Briefly speaking, the mobile computing encompasses the exploration of the connection of portable devices and ubiquitous computing to exploit the integration of increasingly prevalent computing devices in our daily life.

Computing context-sensitive subtype of mobile and ubiquitous computing, contains tools that allow you to recognize the environment in which they operate, inferring on the execution context to perform behavioral adaptations towards intelligent information space.

With these concepts, there is significant use of mobile devices and ubiquitous monitoring of human health through physiological signals from sensors noninvasive and / or invasive. The non-invasive sensors are used for monitoring pervasive, where computing resources are coupled to the human body and can monitor data on blood pressure, heart rate, muscle activity, brain activity, etc. Invasive sensors are implanted in the human body and are used nanotechnology to monitor the inner workings, helping to prevent diseases and to detect abnormalities.

Keywords: pervasive computing, ubiquitous computing, context-aware computing.

LISTA DE FIGURAS

Figura 1 - Sistema distribuído organizado por middleware. A camada de middleware se estende por várias máquinas e oferece a mesma interface a cada aplicação	18
Figura 2 - Cluster de Alto Desempenho	21
Figura 3 - Cluster de Alta Disponibilidade	22
Figura 4 - Cluster de Balanceamento de Carga	22
Figura 5 - Arquitetura em camadas para Sistemas de Computação em grade	23
Figura 6 - Sistema pervasivo doméstico. Distribuição de eletrônicos de consumo gerenciados por um dispositivo central	26
Figura 7 - Monitoração de uma pessoa em um sistema eletrônico pervasivo de tratamento de saúde utilizando (a) um hub local ou (b) uma conexão contínua sem fio.....	27
Figura 8 - Rede de sensores em um ambiente de monitoramento e vigilância	28
Figura 9 - Espaços Inteligentes	31
Figura 10 - Modelo de Dispositivos Móveis e Ubíquos.....	32
Figura 11 - Exemplo de Partículas	33
Figura 12 - Pilha de Localização.	38
Figura 13 - Metamodelo para Aplicações Sensíveis ao Contexto	39
Figura 14 - Modelo para Aplicação de Monitoramento Móvel de Sinais Fisiológicos	45

LISTA DE SIGLAS

<i>BAN</i>	<i>Body Area Network – Rede de Área Corporal</i>
<i>ESSID</i>	<i>Extended Service Set Identifier – Identificação de Conjunto de Serviços Estendidos</i>
<i>IDL</i>	<i>Interface Definition Language – Linguagem de Definição de Interface</i>
<i>LDR</i>	<i>Light Dependent Resistor – Resistor Dependente de Luz</i>
<i>MAC Address</i>	<i>Media Access Control Address – Endereço de Controle de Acesso de Mídia</i>
<i>PDA</i>	<i>Personal Digital Assistant – Assistente Pessoal Digital</i>
<i>RFID</i>	<i>Radio Frequency Identifier – Identificador de Rádio Frequência</i>
<i>UML</i>	<i>Unified Modeling Language – Linguagem de Modelagem Unificada</i>

SUMÁRIO

1	INTRODUÇÃO.....	12
1.1	OBJETIVOS.....	13
1.2	JUSTIFICATIVA.....	13
1.3	METODOLOGIA	14
2	REVISÃO BIBLIOGRÁFICA	17
2.1	SISTEMAS DISTRIBUÍDOS	17
2.1.1	OBJETIVOS DOS SISTEMAS DISTRIBUÍDOS.....	19
2.1.2	TIPOS DE SISTEMAS DISTRIBUÍDOS.....	20
2.1.2.1	SISTEMAS DE COMPUTAÇÃO DISTRIBUÍDOS.....	20
2.1.2.2	SISTEMAS DE INFORMAÇÃO DISTRIBUÍDOS.....	24
2.1.2.3	SISTEMAS DISTRIBUÍDOS PERVASIVOS	25
2.2	COMPUTAÇÃO MÓVEL E UBÍQUA.....	29
2.2.1	SISTEMAS VOLÁTEIS.....	30
2.2.2	ASSOCIAÇÃO	34
2.2.3	SENSIBILIDADE AO CONTEXTO	35
2.2.3.1	SENSORES.....	35
2.2.3.2	ARQUITETURAS DE SENSORIAMENTO	36
2.2.3.3	PERCEPÇÃO DE LOCALIZAÇÃO	37
2.2.4	ADAPTABILIDADE.....	38
2.3	METAMODELO PARA APLICAÇÕES SENSÍVEIS AO CONTEXTO	38
3	DESENVOLVIMENTO.....	42
4	DISCUSSÃO DOS RESULTADOS	47
5	CONSIDERAÇÕES FINAIS.....	49
6	REFERÊNCIAS	51

1 INTRODUÇÃO

A miniaturização dos componentes eletrônicos somado à predominância cada vez maior de redes sem fio proporcionou a crescente integração de dispositivos computacionais portáteis e de pequeno porte aos sistemas distribuídos.

A portabilidade, característica de muitos desses dispositivos, e a capacidade de conexão conveniente em redes de lugares distintos possibilitaram a existência da computação móvel que é uma tecnologia que visa à execução de tarefas computacionais enquanto o usuário encontra-se fora da sua intranet “base”.

A mobilidade oferecida pela computação móvel viabilizou a implementação de sistemas ubíquos ou pervasivos. Esses sistemas consistem na integração da mobilidade com a utilização de vários dispositivos computacionais pequenos e de baixo custo que atuam de maneira transparente e são inseridos no espaço físico em que os usuários se encontram. Assim, a computação se entranha ao ambiente e interage com os recursos existentes, inferindo no contexto em execução, realizando adaptações dinâmicas.

Para possibilitar essa interação entre computação e ambiente, os sistemas ubíquos são dotados de arquiteturas de sensoriamento, onde cada dispositivo ou nó sensor tem a capacidade de efetuar percepções do ambiente, processando e transmitindo informações por comunicação sem fio. Cada um desses dispositivos executa tarefas colaborativas para monitoramento e controle do espaço físico em que estão inseridos.

A partir desse cenário, observa-se o uso significativo de dispositivos móveis e ubíquos no monitoramento de saúde humana através de sinais fisiológicos provenientes de sensores não-invasivos e/ou invasivos. Os sensores não-invasivos são utilizados para monitoramento pervasivo, onde os recursos computacionais são acoplados ao corpo humano e consegue monitorar dados referentes à pressão arterial, frequência cardíaca, atividade muscular, atividade cerebral, etc. Os sensores invasivos são implantados ao corpo humano e utilizam-se da nanotecnologia para

monitorar o funcionamento interno, auxiliando na prevenção de doenças e na detecção de anormalidades.

1.1 OBJETIVOS

O objetivo específico desse estudo consiste em desenvolver um modelo de aplicação para monitoramento móvel de dados fisiológicos através de telefones móveis, PDAs (*Personal Digital Assistant*) ou computadores portáteis, permitindo o acompanhamento contínuo da frequência cardíaca, pressão arterial, atividade muscular e cerebral de maneira não-invasiva.

A modelagem propõe um sistema flexível acoplado ao usuário final, possibilitando o monitoramento em qualquer espaço físico que contenha uma infraestrutura computacional relativamente estável e que permita a inserção de sensores e infraestrutura de rede sem fio, incluindo a conexão com a Internet para o envio de informações para equipe de saúde.

O objetivo geral é de apresentar a computação móvel e ubíqua e a sua capacidade de sensibilidade ao contexto que possibilitam a criação de sistemas pervasivos equipados de sensores invasivos e não-invasivos. Esses sensores permitem captar qualquer grandeza física de relevância ao sistema, inferindo no contexto de execução para realizar adaptações de comportamento do próprio sistema (recursividade).

Ainda como objetivo geral, esse estudo apresenta um metamodelo para aplicações sensíveis ao contexto baseadas em componentes de software, tecnologia empregada na construção de sistemas distribuídos e altamente modulares.

1.2 JUSTIFICATIVA

O envelhecimento da população global, concomitante com o aumento do número de mortes por doenças cardíacas, tornou-se um desafio mundial. A OMS (Organização Mundial de Saúde) projetou no ano de 2010 o número de 20 milhões

de vítimas/ano em 2020 e 24 milhões em 2030. Além disso, existe a elevada incidência de hipertensão arterial, em que a grande maioria dos hipertensos não desconfia que tenha essa disfunção ou não se cuidam adequadamente.

A predominância da conectividade sem fio somada à integração de dispositivos computacionais ao cotidiano permitiu a integração da computação móvel e ubíqua ao monitoramento de saúde através das Redes de Área Corporal (BAN – *Body Area Network*). Essas redes são equipadas com pequenos sensores com capacidade para captar sinais fisiológicos e transmiti-los a um dispositivo computacional o qual consegue processar, armazenar e indicar tais sinais à uma equipe médica que, por sua vez, tem a oportunidade de acompanhar a rotina de um paciente, ajustando seu tratamento de acordo com as necessidades, podendo reduzir o número de visitas ao hospital ou, até mesmo, o período de internação.

A popularização dos dispositivos móveis como PDAs, Smartphones e afins fomentou a ideia de integrar esses dispositivos à BAN permitindo que se consiga fazer o monitoramento e a visualização dos sinais fisiológicos através de aparelhos dos próprios usuários/pacientes.

1.3 METODOLOGIA

A metodologia utilizada nesse trabalho foi à pesquisa bibliográfica, pois a mesma oferece um acervo de ferramentas para auxílio na definição e resolução de problemas já conhecidos nos sistemas distribuídos e na computação sensível ao contexto. A escolha dessa metodologia deve-se ao fato do metamodelo proposto na subseção 2.3 que define sete classes UML (*Unified Modeling Language*) e seus relacionamentos para representar as principais entidades de uma aplicação sensível ao contexto baseada em componentes de *software*.

Através da revisão bibliográfica pode-se contornar problemas relacionados a tempo e reaproveitamento de arquitetura e componentes, uma vez que um estudo dessa fundamentação necessariamente envolveria uma grande quantidade de testes de mesa e mais tempo para coleta e análise de dados.

Como proposta, esse estudo traz o desenvolvimento de um modelo de aplicação para monitoramento móvel de sinais fisiológicos, sensível ao contexto, capaz de monitorar dados de maneira não-invasiva.

Os sistemas de monitoramento móvel de saúde são um novo paradigma na integração entre profissionais de saúde e pacientes. Ferramentas estão sendo desenvolvidas para detectar e gerenciar os riscos de saúde, diagnosticando doenças com antecedência, reduzindo o tempo na tomada de decisão em tratamentos. O monitoramento *fulltime* reduz o tempo de hospitalização, pois otimiza o tempo de resposta no diagnóstico de doenças, no acompanhamento do quadro clínico de pacientes pertencentes a grupos de riscos (por exemplo, hipertensos) e pacientes idosos, que requerem cuidado especial.

Deve-se ressaltar que o estudo é estruturado em 5 (cinco) capítulos, seguidos de Revisão Bibliográfica, constituindo uma base de dados consistente para a elaboração de estudos mais avançados.

O primeiro capítulo fornece uma apresentação sobre o contexto no qual se insere a dissertação, os objetivos, a justificativa e a metodologia implementada.

O segundo capítulo contempla a revisão bibliográfica, abordando, primeiramente, os conceitos e peculiaridades dos sistemas distribuídos. Em seguida explana a computação móvel e ubíqua, descrevendo sua volatilidade, associabilidade e sensibilidade ao contexto. Por último será apresentado um metamodelo para aplicações sensíveis ao contexto baseadas em componentes de *software*.

O terceiro capítulo descreve o desenvolvimento de um modelo de aplicação para monitoramento móvel de sinais fisiológicos como a pressão arterial, frequência cardíaca, atividade cerebral, atividade muscular, além de informações do ambiente em que a aplicação está inserida.

O quarto capítulo foi reservado para discussões dos resultados, relatando os benefícios oriundos do modelo de aplicação proposto no trabalho.

Com base nas informações obtidas a partir dos estudos realizados nos capítulos anteriores, o quinto capítulo foi reservado às Considerações Finais.

2 REVISÃO BIBLIOGRÁFICA

Para atender as necessidades apresentadas nesse trabalho, fez-se necessário o estudo sobre informações básicas referentes ao corpo humano, tais como o coração, pressão arterial e frequência cardíaca, bem como um estudo mais detalhado sobre sistemas distribuídos, computação móvel e ubíqua e metamodelos para implementações sensíveis ao contexto. As subseções 2.1 e 2.2 apresentam, respectivamente, características sobre os sistemas distribuídos e a computação móvel e ubíqua. A subseção 2.3 permite ao leitor o conhecimento sobre um metamodelo sensível ao contexto.

2.1 SISTEMAS DISTRIBUÍDOS

Os sistemas de computação estão em constante evolução e nenhuma área, nos últimos 50 anos, obteve tantas melhorias tecnológicas como a computação. Na década de 1940 até meados da década de 1980, os computadores eram enormes e caros fazendo com que a maioria das organizações tivesse algumas poucas unidades compartilhando com muitos usuários e funcionavam independentemente uns dos outros devido à inexistência de conexão em rede, criando assim, sistemas centralizados ou sistemas monoprocessadores.

Porém, dois avanços tecnológicos começaram a mudar essa situação. O primeiro foi a miniaturização de componentes eletrônicos que possibilitou o desenvolvimento de microprocessadores de grande capacidade (inicialmente 8 bits, depois 16, 32 e 64 bits) e o segundo foi a invenção de redes de computadores de alta velocidade que permitiram centenas de máquinas se conectarem para compartilharem informações e recursos.

O resultado desses avanços permitiu criar sistemas descentralizados compostos por grandes quantidades de computadores conectados em rede de alta velocidade, surgindo assim, as redes de computadores e os sistemas distribuídos (TANENBAUM, 2007).

Coulouris (2007) define um sistema distribuído como um sistema no qual os componentes de *hardware* ou *software*, localizados em computadores interligados em rede, se comunicam e coordenam suas ações apenas enviando mensagens entre si.

Já Tanenbaum (2007) define um sistema distribuído como um conjunto de computadores independentes que se apresenta a seus usuários como um sistema único e coerente.

Através dessas definições é possível compreender que os sistemas distribuídos são formados por componentes (computadores) autônomos e que a sua forma de comunicação e organização interna são imperceptíveis para os usuários que têm a sensação de estar interagindo com sistema único. Para poder suportar computadores e redes heterogêneas e, simultaneamente, oferecer uma visão de sistema único, os sistemas distribuídos são organizados por uma camada de *software* denominada *middleware*, que se situa logicamente entre uma camada de nível mais alto (usuário e aplicações) e uma camada subjacente (sistemas operacionais), ocultando as diferenças em *hardware* e sistemas operacionais permitindo a comunicação entre diferentes aplicações. A Figura 1 ilustra essa organização.

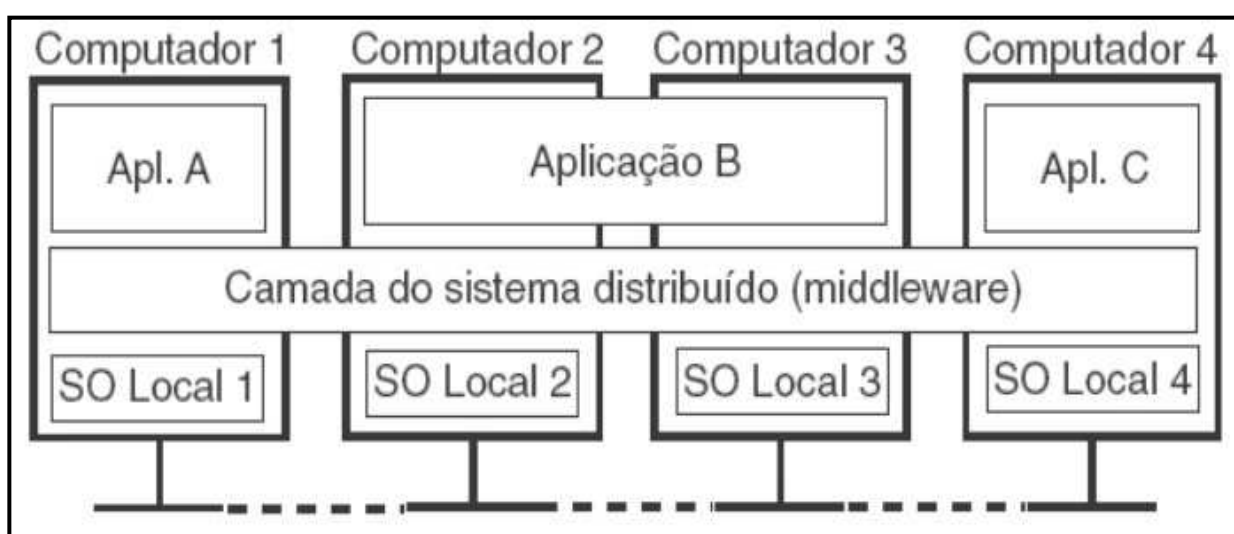


Figura 1 - Sistema distribuído organizado por middleware. A camada de middleware se estende por várias máquinas e oferece a mesma interface a cada aplicação (TANENBAUM, 2007).

2.1.1 OBJETIVOS DOS SISTEMAS DISTRIBUÍDOS

Os sistemas distribuídos têm como objetivo principal aperfeiçoar o acesso a recursos remotos (físicos e/ou lógicos) e o seu compartilhamento de maneira controlada e eficiente. Esses recursos vão de uma simples impressora até supercomputadores, sistemas de armazenamento de alto desempenho, *imagesetters*¹ e outros periféricos caros. O compartilhamento de recursos garante economia em um ambiente distribuído e a colaboração e troca de informações como, por exemplo, a Internet com seus protocolos simples para troca de dados.

Um objetivo importante é a **transparência da distribuição**. Um sistema transparente visa ocultar para os usuários e as aplicações o ambiente fisicamente distribuído, apresentando-se como um único sistema que torna imperceptível o compartilhamento de áreas de memória, trocas de mensagem para sincronização, etc. Esse conceito de transparência pode ser implementado em diversos aspectos de um sistema distribuído como acesso, localização, migração, relocação, replicação, concorrência e falha.

A **transparência de acesso** oculta a diferença na representação de dados e na maneira de acesso aos recursos. A **transparência de localização** oculta o local em que um recurso está inserido, impedindo que usuários consigam dizer qual a localização física desse recurso. A **transparência de migração** consiste na capacidade de recursos serem migrados sem afetar sua maneira de acesso. Já a **transparência de relocação** é a capacidade de dados serem migrados/relocados enquanto estão sendo acessados sem a percepção de usuário ou aplicações. A **transparência de concorrência** oculta o compartilhamento de recursos entre usuários concorrentes de maneira que os mesmos não percebam a utilização simultânea. E, por último, a **transparência à falha** que tem a função de ocultar falhas e a recuperação de recursos.

Outro objetivo importante é a abertura². Sistemas distribuídos abertos oferecem serviços de acordo com as especificações contidas em sua interface. A

¹ Máquina digital com tecnologia laser para produção de fotolitos de páginas. Fotolito é um filme transparente feito a base de acetato, papel vegetal ou laser filme.

² Capacidade de extensão e reimplementação de serviços em um sistema distribuído.

interface descreve a sintaxe e a semântica desses serviços através de IDL (*Interface Definition Language*). Todo sistema distribuído aberto deve ser extensível, ou seja, ter fácil configuração na adição ou substituição de diferentes componentes sem afetar os já existentes ou em funcionamento.

A escalabilidade também é um objetivo importante em um sistema distribuído, pois permite manipular as cargas crescentes de trabalho de forma uniforme, camuflando a latência da comunicação e balanceando a carga de trabalho (TANENBAUM, 2007).

2.1.2 TIPOS DE SISTEMAS DISTRIBUÍDOS

Essa sessão apresenta uma macro-visão sobre os tipos de sistemas distribuídos propostos por Tanenbaum (2007).

2.1.2.1 SISTEMAS DE COMPUTAÇÃO DISTRIBUÍDOS

Classe de sistemas distribuídos utilizada para tarefas de computação de alto desempenho. Existe a distinção de dois subgrupos. O primeiro é a computação em *cluster*, que consiste em conjuntos de estações de trabalhos ou computadores semelhantes conectados em rede local de alta velocidade onde cada nó executa o mesmo sistema operacional. Já o segundo, denominado computação em grade consiste em sistemas distribuídos montados como federação de computadores, onde cada sistema pode ter um domínio diferente, e pode ser muito diferente no sentido de *hardware*, *software* e tecnologia de rede utilizada (TANENBAUM, 2007).

Os **sistemas de computação em cluster** são constituídos muitas vezes por uma série de computadores convencionais conectados a redes de alta velocidade. São empregados em computação paralela na qual um único *software* é executado em paralelo em diversas máquinas (TANENBAUM, 2007).

Um exemplo de *cluster* é o Beowulf³ baseado em Linux. Cada *cluster* é constituído por um conjunto de nós de computação controlados e acessados através de um único nó mestre.

Existem vários tipos de *cluster*, mas os mais conhecidos são:

- **cluster de alto desempenho (alta performance):** permite grandes cargas de processamento com altos volumes de *gigaflops*⁴ em computadores comuns e utilizam sistemas operacionais gratuitos. A Figura 2 ilustra esse tipo de *cluster*.

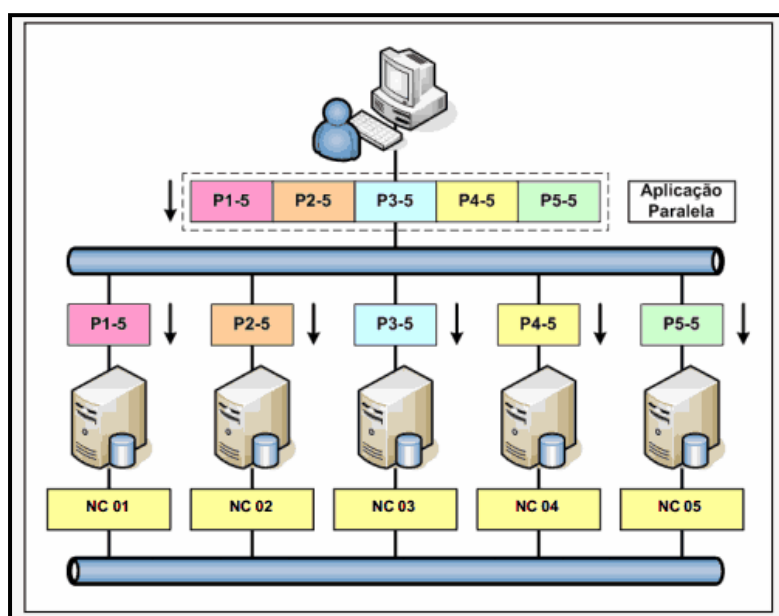


Figura 2 - Cluster de Alto Desempenho (ZEM, 2011).

- **cluster de alta disponibilidade:** possuem sistemas que permanecem ativos e em condições de uso por um longo período de tempo. Tem a capacidade de detectar erros e, assim, protegem-se de possíveis falhas. A Figura 3 ilustra esse tipo de *cluster*.

³ Projeto desenvolvido por Donald Becker, pesquisador da NASA, para computação paralela.

⁴ Equivalente a um bilhão de operações em ponto flutuante por segundo.

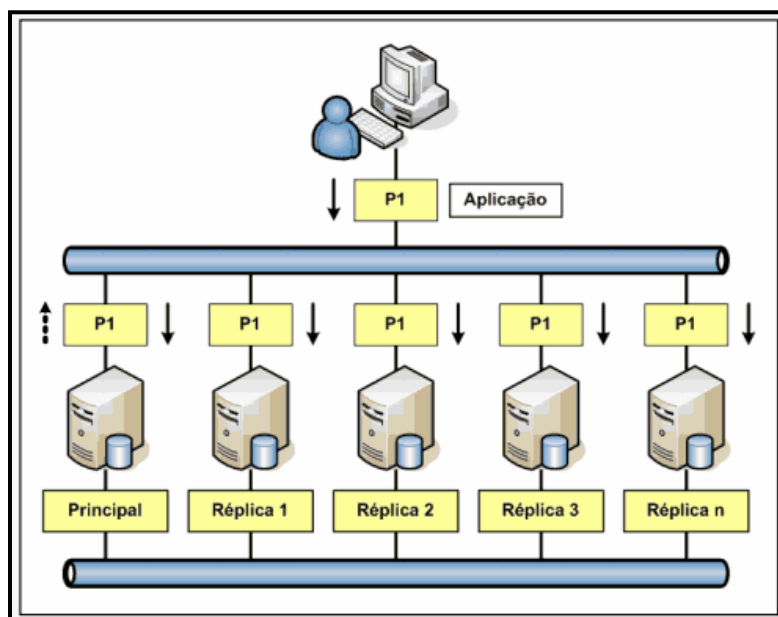


Figura 3 - Cluster de Alta Disponibilidade (ZEM, 2011).

- cluster para balanceamento de carga:** tem a função de controlar a distribuição equilibrada de processamento. Sua comunicação e seus mecanismos de redundância são monitorados constantemente para prevenção de falhas e interrupções de funcionamento. A Figura 4 ilustra esse tipo de *cluster*.

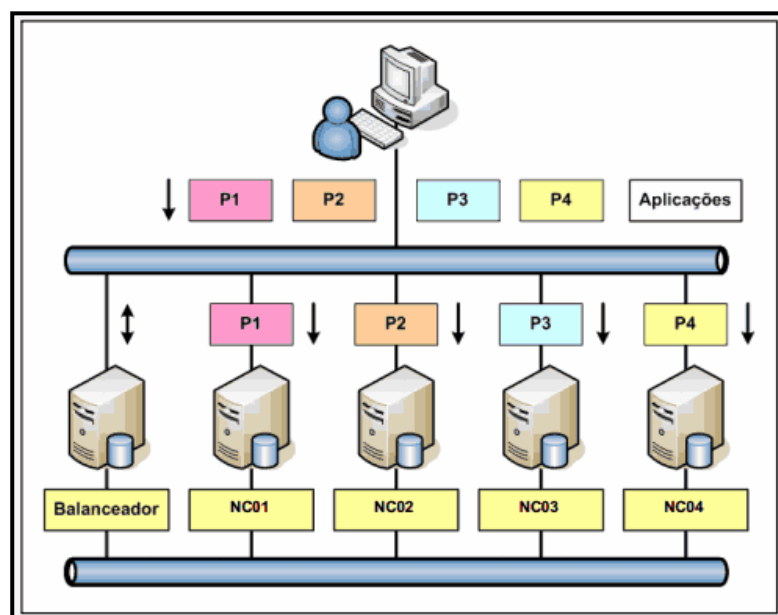


Figura 4 - Cluster de Balanceamento de Carga (ZEM, 2011).

Os **sistemas de computação em grade** são altamente heterogêneos quando comparados aos sistemas de computação em cluster. Assim, não existem premissas de padronização de *hardware*, sistemas operacionais, redes etc.

A computação em grade possui uma organização virtual que possibilita a utilização de recursos de diferentes organizações a um grupo de usuários. Os recursos típicos de uma organização virtual são servidores (supercomputadores e *clusters*), facilidade de armazenamento e banco de dados (TANENBAUM, 2007).

A arquitetura de uma computação em grade consiste em quatro camadas, sendo que a Figura 5 ilustra essa distribuição:

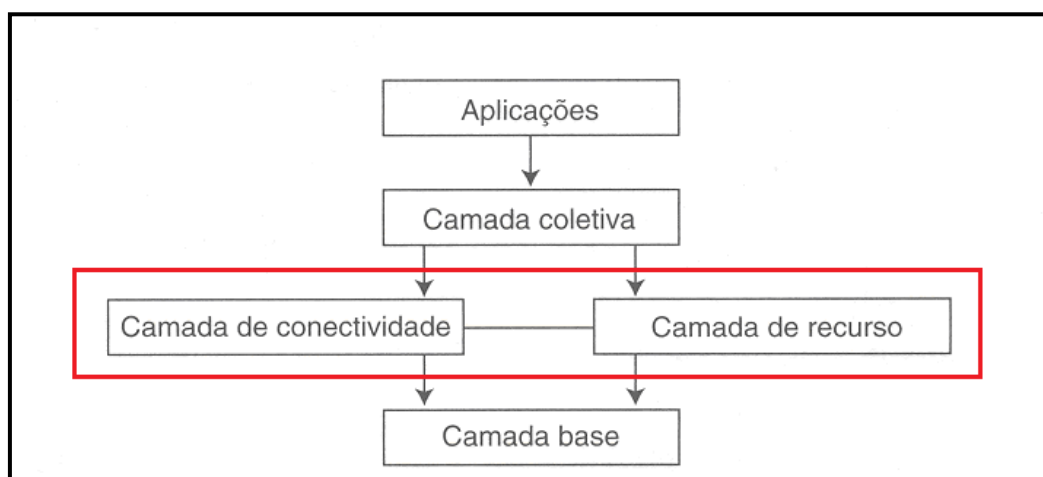


Figura 5 - Arquitetura em camadas para Sistemas de Computação em grade (TANENBAUM, 2007).

- **Camada-base:** camada mais baixa que provê interfaces para recursos locais em um site específico. São projetadas para compartilhar recursos dentro de uma organização virtual.
- **Camada de conectividade:** camada de protocolos de comunicação para suportar transações de grade na utilização de múltiplos recursos. Contém protocolos de segurança para autenticação de usuários e recursos. Delega direitos e hierarquias de uma organização virtual.
- **Camada de recursos:** realiza o controle de acesso e gerenciamento de recursos. Comunica-se com a camada de conectividade para obter

autenticação e funções que permitam interação com a camada-base para chamadas diretas de interfaces.

- **Camada coletiva:** trata a manipulação de acesso a múltiplos recursos. Consiste em serviços de descoberta de recursos, alocação e escalonamento de tarefas para múltiplos recursos, replicação de dados etc.
- **Camada de aplicação:** é constituída pelas aplicações funcionais dentro de uma organização virtual que fazem o uso de um ambiente de computação em grade.

2.1.2.2 SISTEMAS DE INFORMAÇÃO DISTRIBUÍDOS

São sistemas de informação que possuem ambientes distribuídos (aplicações em rede) e são implementados em organizações comerciais e acadêmicas.

Os **sistemas de processamento de transações** são um subtipo de sistemas de informação distribuídos e consistem em fornecer suporte para atividades operacionais cotidianas (controle de estoque, contabilidade, folha de pagamento, atendimento) e o acompanhamento do fluxo transacional de uma organização.

Como benefício, esses sistemas capacitam as organizações a executar suas principais atividades de maneira otimizada. Uma transação consiste na troca de valores que afetam a lucratividade ou o ganho global de uma organização. Assim, os sistemas de processamento de transações podem ser considerados como o centro do sistema da empresa, apoiando a realização e monitorando das negociações.

As atividades de um sistema de processamento de transações são:

- Coleta de dados: consiste na entrada dos dados ou informações.
- Manipulação dos dados: cálculos, classificação, disposição.

- Armazenamento: armazenagem de dados em bancos de dados.
- Produção de documentos: podem ser impressos ou exibidos na tela do computador.

Por serem considerados como o centro do sistema da empresa, os sistemas de processamento de transações têm:

- Necessidade de eficiência de processamento para suportar grandes quantidades de entradas e saídas, além do alto grau de repetição no processamento.
- Grande necessidade de armazenagem.
- Tolerância a falhas, pois o seu não funcionamento ocasiona impacto negativo na organização e afeta inúmeros usuários.

2.1.2.3 SISTEMAS DISTRIBUÍDOS PERVASIVOS

Os sistemas distribuídos pervasivos ou ubíquos surgiram com a introdução de dispositivos de computação móvel e embarcados. O ambiente em que estão inseridos é caracterizado pela instabilidade, pois não há nós fixos, nem conexões permanentes e de alta qualidade com uma rede. Seus dispositivos são caracterizados por seu pequeno tamanho, pela alimentação através de baterias, mobilidade e conexão sem fio. São adaptáveis ao ambiente e buscam serviços para reagir à adaptação com eficiência (TANENBAUM, 2007).

Tanenbaum divide os sistemas distribuídos pervasivos em 3 (três) frentes: os sistemas domésticos, os sistemas eletrônicos para tratamento de saúde e a rede de sensores.

Os **sistemas domésticos** são montados ao redor de redes domésticas compostas por um ou mais computadores pessoais. Integram eletrônicos de

consumo como televisões, aparelhos de áudio e vídeo, vídeo games, Smartphones, PDA's, e outros equipamentos de uso pessoal em um único sistema.

Para esse tipo de sistema se tornar realidade, há vários desafios que precisam ser enfrentados como a capacidade de serem completamente autoconfiguráveis e autogerenciáveis, não precisando da intervenção dos usuários finais para manter-se ligados e em pleno funcionamento.

Outros desafios são as atualizações de *software* e *firmware*⁵ sem a intervenção manual e sem a violação de compatibilidade com outros dispositivos, além do gerenciamento de espaço pessoal (recursos pessoais).

Com crescimento contínuo da capacidade de armazenamento de discos rígidos portáteis, a tendência dos sistemas pervasivos domésticos é adotar uma arquitetura com um único dispositivo central como mestre e os demais dispositivos como fornecedores de interface conveniente para os seres humanos. A Figura 6 ilustra a organização dessa arquitetura.



Figura 6 - Sistema pervasivo doméstico. Distribuição de eletrônicos de consumo gerenciados por um dispositivo central (AVC SERVICE⁶).

⁵ Conjunto de instruções operacionais programadas diretamente no hardware de um equipamento eletrônico.

⁶ Disponível em: <<http://www.avcservice.com.br/website/automacao.asp>>. Acesso em: abr. 2013.

Os **sistemas eletrônicos para tratamento de saúde** são sistemas com o objetivo de monitorar o bem-estar de indivíduos através de sensores organizados em uma rede de área corporal (*Body Area Network - BAN*). A BAN deve ser capaz de funcionar com o indivíduo em movimento sem que o mesmo esteja preso por fios elétricos ou dispositivos imóveis.

Há duas maneiras de organização para o funcionamento de tal sistema. A primeira contém um *hub* central que faz parte da BAN e colhe dados que são descarregados periodicamente em dispositivos de armazenamento. A segunda, a BAN é conectada a uma rede externa (sem fio) a qual envia dados monitorados. A Figura 7 ilustra as duas possíveis maneiras de organização.

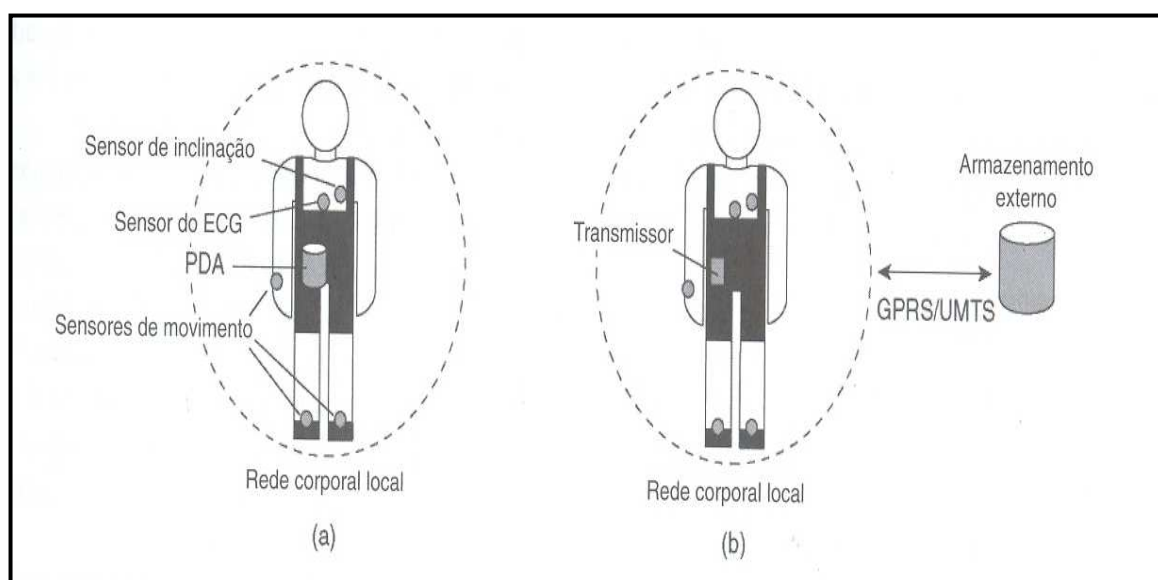


Figura 7 - Monitoração de uma pessoa em um sistema eletrônico pervasivo de tratamento de saúde utilizando (a) um hub local ou (b) uma conexão contínua sem fio (TANENBAUM, 2007).

Diferentemente dos sistemas domésticos, os sistemas pervasivos de tratamento de saúde não podem ter um único servidor e nem dispositivos de monitoramento operando em funcionalidade mínima. Seus dispositivos e redes de área corporal devem suportar processamento de dados na rede para posteriormente armazenar ou enviar informações.

Por fim, a **redes de sensores** têm como características principais o elevado número de dispositivos (nós) que podem chegar a dezenas de milhares, o tamanho

relativamente pequeno de cada elemento e possível mobilidade é que torna a rede dinâmica, mudando frequentemente a sua organização física e lógica.

Grande parte das redes de sensores usa comunicação sem fio e seus dispositivos (nós) são alimentados por bateria, seus recursos são limitados e sua capacidade de armazenamento restrita, o que exige que a eficiência seja um dos itens fundamentais de projeto.

Cada sensor, na perspectiva de sistema, é um elemento autônomo com capacidade de capturar dados do meio em que está inserido. Esses dados podem ser tratados e enviados, possibilitando assim criações de sistemas de monitoramento e vigilância. A Figura 8 ilustra uma rede de sensores para monitoramento e vigilância.

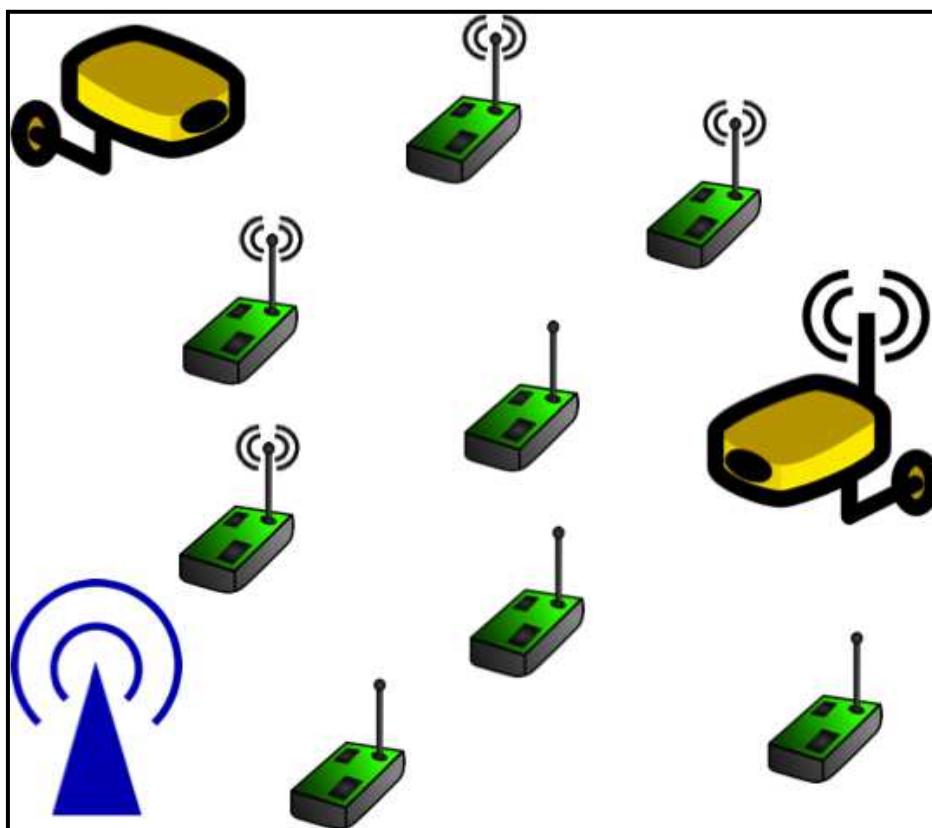


Figura 8 - Rede de sensores em um ambiente de monitoramento e vigilância (UFPR, Núcleo de Redes Sem Fio e Redes Avançadas⁷).

⁷ Disponível em: < http://www.nr2.ufpr.br/?page_id=9 >. Acesso mai. 2013.

2.2 COMPUTAÇÃO MÓVEL E UBÍQUA

A computação móvel e ubíqua surgiu devido à miniaturização dos dispositivos e da conectividade sem fio. A computação móvel engloba a exploração da conexão de dispositivos portáteis e a computação ubíqua a exploração da integração cada vez mais predominante dos dispositivos de computação no cotidiano.

A portabilidade desses dispositivos e a capacidade de se conectar de maneira conveniente em diferentes redes possibilitaram a computação móvel. Coulouris (2007) relata que a computação móvel surgiu como um paradigma no qual os usuários poderiam carregar seus computadores pessoais e manter certa conectividade com outras máquinas.

A computação ubíqua também conhecida como computação pervasiva foi mencionada pelo cientista Mark Weiser em 1988 e publicado em 1991 no seu artigo *The Computer for the 21st Century*. O termo ubíquo significa “em toda parte” e foi cunhado através da percepção da predominância cada vez maior de dispositivos de computação, levando a mudanças revolucionárias na maneira de como as pessoas usariam os computadores.

A primeira mudança, segundo Weiser, é a ideia “de uma pessoa para muitos computadores”, onde esses dispositivos se multiplicariam não apenas em números, mas na sua forma e função para atender a diferentes tarefas. A segunda mudança se refere ao “desaparecimento” dos computadores que se incorporariam em utensílios e objetos do dia a dia, até se tornarem indistinguíveis.

O objetivo da computação ubíqua é tornar onipresente o acesso a serviços computacionais de modo que as pessoas não percebam esse processamento ou até mesmo a existência de dispositivos computacionais (COULOURIS, 2007).

Tanenbaum (2007) denomina os sistemas ubíquos como pervasivos e apresenta os sistemas domésticos, os sistemas eletrônicos para tratamento de saúde e a rede de computadores como subáreas da computação móvel e ubíqua (sessão 2.1.2.3 desse trabalho), enquanto Coulouris (2007) apresenta como

subáreas a computação de mão (*handheld*), a computação acoplada ao corpo (*wearable*) e a computação sensível ao contexto (*context-aware computing*).

A computação de mão consiste em dispositivos móveis de propósito geral com capacidade de executar diferentes tipos de aplicações. Possuem uma dimensão reduzida o que permite a mobilidade, fonte de alimentação limitada por baterias, poder de processamento limitado, telas reduzidas e podem vir equipados com câmeras, leitores de códigos de barra e outros tipos de acessórios especializados.

A computação acoplada ao corpo consiste em equipamentos de computação acoplados ao usuário. Esses equipamentos podem ser presos no tecido de roupas, ou ainda dentro do corpo do próprio usuário, ou transportados de maneira externa como relógios, jóias, óculos etc. Seu funcionamento, na maior parte dos casos, se dá sem intervenção humana e possuem finalidades específicas.

A computação sensível ao contexto faz com que os sistemas computacionais adaptem o seu comportamento automaticamente perante as circunstâncias físicas (contexto). A sessão 2.2.3 relata com mais detalhes essa subárea que é o foco desse estudo.

2.2.1 SISTEMAS VOLÁTEIS

A Computação Móvel e Ubíqua é descrita como um sistema volátil porque, ao contrário dos demais sistemas computacionais, determinados comportamentos inesperados são classificados como normais ao invés de excepcionais. Seu conjunto de usuários, hardware e software são altamente dinâmicos e mudam de maneira imprevisível (COULOURIS, 2007).

A base dos sistemas móveis e ubíquos é constituída por todo e qualquer espaço físico que contenha serviços incorporados, pois são nesses espaços que acontecem a mobilidade e a incorporação ubíqua ou pervasiva. Assim, denominam-se **Espaços Inteligentes** (*smart spaces*) todo e qualquer ambiente com serviços incorporados que normalmente ofereçam uma infraestrutura de computação relativamente estável e que permita a inserção de servidores, impressoras, sensores

e uma infraestrutura de rede sem fio, incluindo a conexão com a Internet. A Figura 9 demonstra uma abstração de espaços inteligentes.

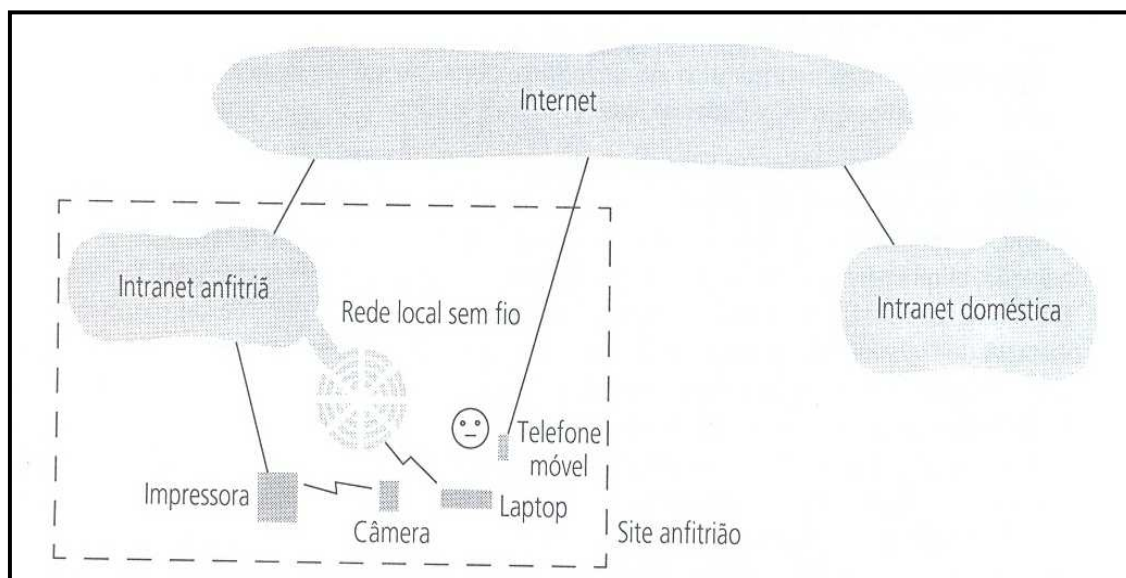


Figura 9 - Espaços Inteligentes (COULOURIS, 2007).

Em um espaço inteligente existe a mobilidade física e a mobilidade lógica. A primeira consiste na entrada e saída de dispositivos de um ambiente inteligente enquanto a segunda se refere a processos ou agentes móveis (processos de software providos de movimentação autônoma entre localizações de rede) que podem entrar ou sair de um ambiente inteligente ou dispositivo pessoal de um usuário. Além dessas movimentações, pode existir a adição de dispositivos estáticos no ambiente (reprodutores de mídia, por exemplo) e a falha que ocasiona o “desaparecimento” de dispositivos.

Para um sistema distribuído, essas movimentações implicam em integrações de componentes de software (módulo de serviços especificados por interface) desde que haja conteúdo representativo.

O **modelo de dispositivo** encontrado em um sistema móvel e ubíquo (Figura 10) é caracterizado por sua pequena dimensão, fonte de energia limitada, mobilidade, conexão sem fio, recursos computacionais limitados e interfaces com o ambiente físico.



Figura 10 - Modelo de Dispositivos Móveis e Ubíquos (Elaborada pelo Autor).

A pequena dimensão proporciona dispositivos portáteis e passíveis de integração com objetos do dia a dia, assim, sua alimentação é realizada por baterias o que garante a inexistência de cabos acoplados. A fonte de energia limitada proporciona recursos computacionais limitados, pois quanto maior o poder de processamento, armazenamento e largura de banda, maior é o consumo energético. Com isso, a eficiência é um requisito primordial em projetos de dispositivos computacionais móveis e ubíquos.

Para interagir com o meio em que estão inseridos, esses dispositivos são equipados com sensores e controladores. Os sensores são providos da capacidade de capturar parâmetros físicos que são fornecidos para *software*. Já os controladores são manipulados pelo software e afetam o ambiente inteligente em que estão inseridos.

Existem ainda os dispositivos denominados partículas (*motes*) que funcionam de maneira autônoma em aplicações como a percepção ambiental. Têm capacidade de incorporação em ambientes inteligentes, descobrimento de dispositivos similares e compartilhamento de dados entre si. A Figura 11 mostra alguns exemplos desse dispositivo.

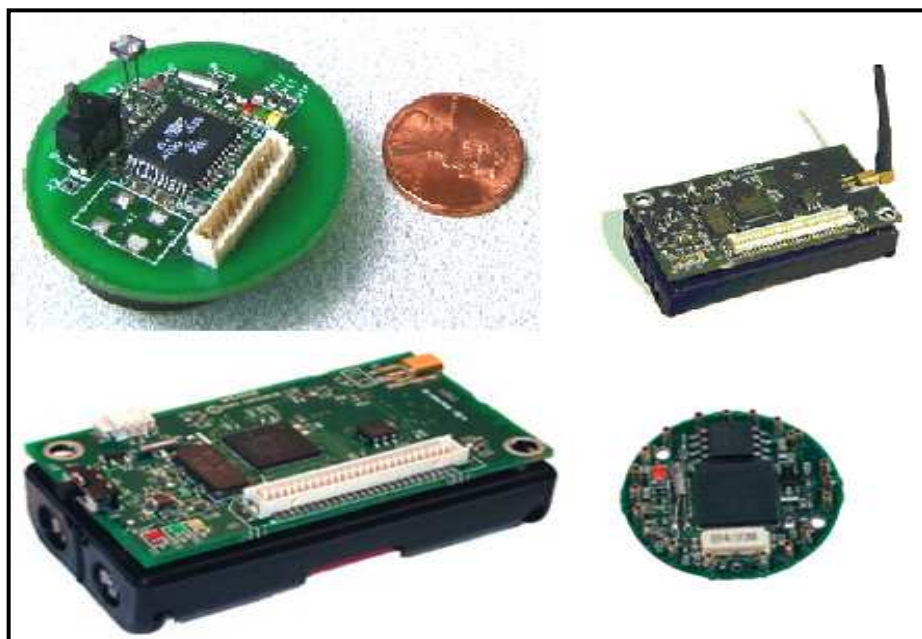


Figura 11 - Exemplo de Partículas (Elaborada pelo Autor).

O modelo de dispositivos móveis e ubíquos possui uma **conectividade volátil**, pois a tecnologia de conexão sem fio varia em sua largura de banda e latência nominal⁸. Essa volatilidade de conexão em tempo de execução, estado e qualidade de serviço impacta diretamente nos projetos de sistemas distribuídos.

A desconexão tem maior probabilidade de existir em redes sem fio (*wireless*) do que em redes com fio, e a largura de banda e latência variável podem ocasionar uma completa desconexão devido fomentarem taxas de erro variáveis e, quanto maior for a taxa de erro, maior será a perda de pacotes trafegados e menor será a taxa de desempenho de saída (*throughput*), além de constantes *timeouts* (tempo limite).

Por serem voláteis, os componentes mudam dinamicamente o conjunto de componentes com que se comunicam à medida que se movem ou que outros componentes aparecem em seu ambiente, constituindo assim, uma **interação espontânea**, o que pode gerar menor privacidade e segurança (COULOURIS, 2007).

⁸ Período de espera para que blocos de informação percorram a rede até seu destino.

2.2.2 ASSOCIAÇÃO

Os dispositivos existentes na computação móvel e ubíqua estão fadados a aparecer e desaparecer de espaços inteligentes de maneira imprevisível. Essa interação espontânea faz com que os componentes dos dispositivos se associem aos servidores ou forneçam serviços para outros componentes em qualquer parte do espaço inteligente.

Os serviços de descoberta englobam mecanismos que permitem essa interação dentro de um espaço inteligente. Eles registram e pesquisam os serviços existentes dentro de um ambiente, mas a sua implementação é realizada com base nas propriedades voláteis do sistema. Os problemas tratados nos serviços de descoberta são:

- Associação apropriada.
- Descrição do serviço e linguagem de consulta.
- Descoberta específica do espaço inteligente.
- Implementação do diretório.
- Volatilidade do serviço.

Os serviços de descoberta não são passíveis de erros e falhas, suas deficiências podem ser solucionadas até certo ponto por meios físicos, mas a intervenção humana é frequentemente necessária. Para amenizar essas inconveniências foram desenvolvidas as seguintes técnicas:

- **Interação Humana na descoberta do escopo** (entrada fornecida por humanos para configuração de abrangência de descoberta).
- **Percepção e canais fisicamente restritos para descoberta de escopo** (utilização de sensores em dispositivos).

- **Associação direta** (intervenção humana por mecanismos físicos que permitem associar dois dispositivos sem o uso de serviços de descoberta).

2.2.3 SENSIBILIDADE AO CONTEXTO

A sensibilidade ao contexto é a subárea da Computação Móvel e Ubíqua que explora as arquiteturas de processamento para coleta de dados (contexto) a partir de sensores e sistemas de reconhecimento que respondam às circunstâncias físicas.

O contexto de uma entidade (pessoa, lugar, objeto) é definido como aspectos de circunstâncias físicas e de importância para o comportamento de um sistema. Contextos podem ser empregados na inclusão de atributos complexos, como atividades do usuário. Um exemplo de atributo complexo é a capacidade de um dispositivo de telefonia móvel reconhecer quando deve entrar em modo silencioso ou simplesmente tocar em volume alto de acordo com o ambiente em que está inserido (COULOURIS, 2007).

2.2.3.1 SENSORES

Sensores são dispositivos eletrônicos (hardware) com capacidade de captar grandezas físicas. Essas grandezas geram sinais analógicos ou digitais que são transmitidos para softwares capazes de interpretá-los.

Acelerômetros, giroscópios, magnetômetros, termômetros, sensores de luminosidade como luxímetro (medidor de fluxo de luminosidade) e LDRs (fotoresistores), dispositivos infravermelhos e RFID (Radio Frequency Identifier) são alguns exemplos de sensores utilizados para reconhecimento de contexto em nível de localização, velocidade e orientação, condições de ambiente e presença respectivamente (COULOURIS, 2007).

2.2.3.2 ARQUITETURAS DE SENSORIAMENTO

Os projetos de sistemas de reconhecimento de contexto enfrentam desafios funcionais como a integração de sensores idiossincráticos, abstração de dados dos sensores, combinação de saídas dos sensores e contexto dinâmico.

A **integração de sensores idiossincráticos** se refere à peculiaridade de implementação e codificação, por exemplo, a aplicação de giroscópios e acelerômetros na captura de movimentos dos dedos ou braços do usuário.

A **abstração de dados dos sensores** engloba aplicações que sejam capazes de abstrair os dados contextuais para evitar preocupação com as individualidades de cada sensor, pois diferentes sensores empregados em garimpar resultados semelhantes fornecem dados brutos diferentes.

A **combinação de saídas dos sensores** consiste na fusão de diferentes sensores para diminuir a taxa de erro no reconhecimento de contexto. Por exemplo, a utilização de dispositivos infravermelhos, sensores de pressão ou células de carga em um piso e câmeras para detectar a presença em determinado ambiente.

Por fim, as aplicações devem responder rapidamente aos **contextos dinâmicos** e não apenas lê-los. Como exemplo, é possível imaginar um dispositivo móvel que projeta imagens em uma tela e que deve parar o fluxo da transmissão quando uma pessoa que não seja funcionário entre na sala ou ainda quando o dono do dispositivo deixe essa mesma sala.

Para atender todos esses requisitos, pesquisadores vêm projetando várias arquiteturas de sensoriamento, como exemplo, existem o sensoriamento dentro de uma infraestrutura e a rede de sensores sem fio.

O **sensoriamento dentro uma infraestrutura** consiste na utilização de hardware/software com a finalidade de reconhecer o contexto do ambiente em que estão inseridos. Os elementos de contexto (geradores, interpretadores e servidores) são obtidos através de componentes distribuídos. Os geradores obtêm dados brutos dos sensores e os transmitem para interpretadores que abstraem valores

contextuais dos dados de baixo nível (sinais analógicos ou digitais) extraindo valores de nível mais alto. Por fim, os servidores, abstraem em um nível mais alto, reúnem, armazenam e interpretam os atributos contextuais.

Já a **rede de sensores sem fio** consiste em um conjunto de sensores provenientes da interação de pequenos dispositivos (nós) de baixo custo. Cada nó possui seus recursos para sensoriamento, computação e comunicação sem fio. A computação (processamento na rede) consiste nos ciclos de execução, agregação de valores entre nós vizinhos para examinar ambientes, filtros de valores relevantes aos sistemas e ativação/desativação de sensores para tráfego de dados. A comunicação é tolerante ao rompimento (*Disruption Tolerant Networking*) e atraso (*Delay Tolerant Networking*), não conta com a conectividade contínua entre dois pontos extremos fixos e os dados são transferidos como e quando puderem ser (COULOURIS, 2007).

2.2.3.3 PERCEPÇÃO DE LOCALIZAÇÃO

Sua arquitetura para percepção de localização exige a generalidade dos tipos de sensores aplicados na percepção de localização, a mudança na escala com relação ao número de objetos a serem localizados e a taxa de eventos de atualização de posicionamento que ocorrem na movimentação de objetos.

Ainda em relação a sua arquitetura, a generalidade dos tipos de sensores é atendida pela **pilha de localização** (HIGHTOWER *et al.* 2002, GRAUMANN *et al.* 2003) que divide o sistema de percepção da localização em camadas individuais. A **camada de sensores** contém *drives* para extrair dados brutos de uma variedade de sensores, a **camada de medidas** transforma os dados brutos em tipos de medidas comuns (distância, ângulo, velocidade), a **camada de fusão** combina as medidas de diferentes sensores, a **camada de arranjos** deduz o relacionamento entre os objetos (fato de estarem no mesmo lugar). Acima dessas camadas existem as camadas para combinação de dados de localização com dados de outros tipos de sensores inseridos no espaço inteligente, determinando atributos contextuais mais complexos (COULOURIS, 2007). A Figura 12 ilustra a pilha de localização.

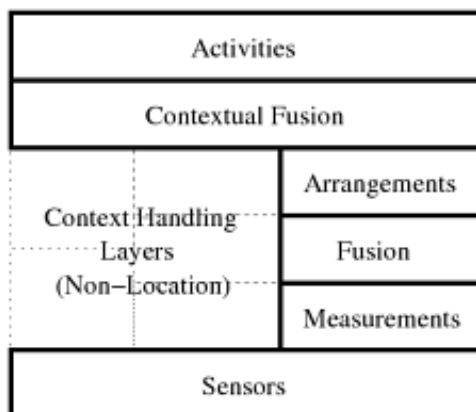


Figura 12 - Pilha de Localização (UBICOMP, 2004).

2.2.4 ADAPTABILIDADE

Os sistemas adaptativos têm o objetivo de atender a heterogeneidade de capacidade de processamento, recursos de entrada e saída, largura de banda da rede, memória e fonte de energia limitada.

Ao acomodar essa heterogeneidade, permite-se a reutilização de software entre contextos que variam, por exemplo, na capacidade do dispositivo e as preferências do usuário, além de acomodar as condições variáveis do recurso em tempo de execução, adaptando o comportamento do aplicativo sem sacrificar suas propriedades fundamentais (COULOURIS, 2007).

2.3 METAMODELO PARA APLICAÇÕES SENSÍVEIS AO CONTEXTO

As aplicações sensíveis ao contexto devem ser capazes de interagir com os recursos presentes no ambiente em que estão inseridas da mesma forma que interagem com seus próprios recursos, tendo a sensibilidade do espaço inteligente para inferir o seu contexto de execução e realizar adaptações dinâmicas.

Os componentes de software⁹, tecnologia empregada para construção de aplicações distribuídas e altamente modulares, não suportam os requisitos atribuídos

⁹ Termo utilizado para descrever o elemento de *software* que encapsula uma série de funcionalidades. Um componente é uma unidade independente, que pode ser utilizado com outros componentes para formar um sistema mais complexo.

a computação sensível ao contexto. Com isso, o reconhecimento e a sensibilidade são incorporados ao aplicativo como um requisito funcional ou não-funcional através de desenvolvedores ou por *containers* com mecanismos de sensibilidade ao contexto (sistemas baseados em componentes).

O metamodelo¹⁰ proposto para aplicações sensíveis ao contexto (Figura 13), define sete classes UML e seus relacionamentos para representar as principais entidades de uma aplicação baseada em componentes e sensível ao contexto (PINTO *et al.* 2008, CARDOZO *et al.* 2008, GUIMARÃES *et al.* 2008).

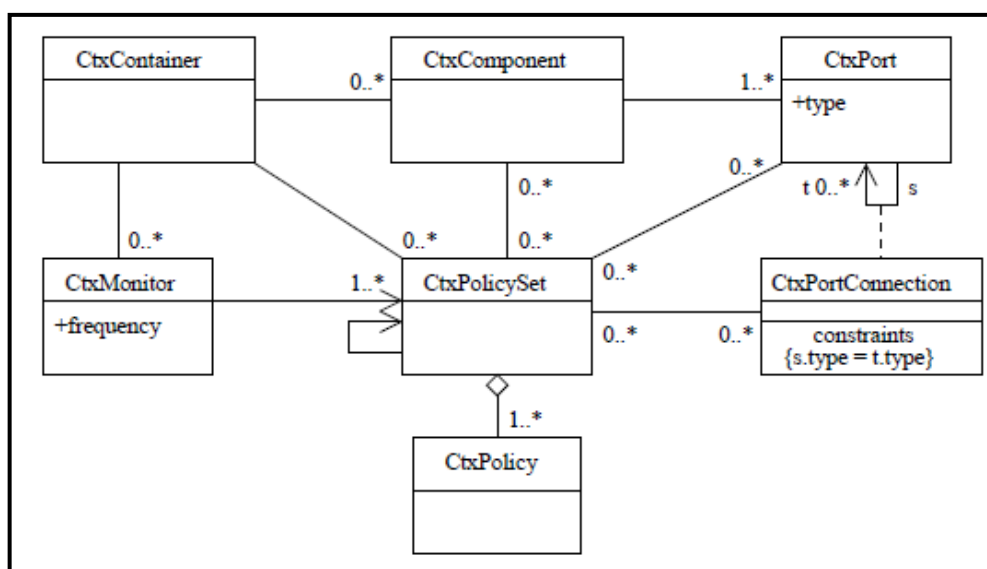


Figura 13 - Metamodelo para Aplicações Sensíveis ao Contexto (PINTO, CARDOZO, GUIMARÃES, 2008).

A entidade **CtxComponent** representa os componentes de software, unidades de implantação e composição que oferecem os requisitos funcionais do sistema, encapsulando uma série de funcionalidades. A entidade **CtxContainer** faz referência aos *containers*, elementos que proporcionam os recursos necessários para a execução dos componentes no ambiente em que estão inseridos, ou seja, os requisitos não-funcionais (mecanismos de sensibilidade ao contexto). Um *container* é uma abstração que define um ambiente de execução protegido, onde implementações de componentes são implantadas, criadas e executadas.

¹⁰ Classe de modelo UML que descreve a própria UML. Permite a definição formal de linguagens de modelagens usando o paradigma de Orientação a Objetos.

A entidade **CtxPort** representa os terminais (portas) dos componentes para interações. Apenas as portas equivalentes¹¹ podem interagir. O **CtxPortConnection** contém as referências das portas equivalentes, representando a ligação existente entre ambas. Tem a função de monitorar e manter os parâmetros de conexão.

O **CtxPolicySet** tem referência com todos os componentes do sistema passíveis de sensibilidade ao contexto, indicando a possibilidade de adaptação dos mesmos. Enxerga o estado geral do sistema e o estado do contexto, realizando as adaptações conforme as informações contextuais adquiridas. As adaptações podem ser realizadas de maneira comportamental (paramétrico) ou estrutural. A primeira é efetuada através da atualização dos componentes e propriedades de configuração de porta de interação ou invocando métodos definidos pelos componentes para realizar as funções de adaptação. A segunda maneira é realizada através da criação, destruição e substituição de componentes ou através de reconfigurações das portas de conexão. A entidade CtxPolicySet é constituída por elementos **CtxPolicy** que representam as regras de adaptação referentes às informações contextuais.

O **CtxMonitor** tem a responsabilidade de adquirir a informação contextual de qualquer fonte de alimentação e repassar para o CtxPolicySet. Essa fonte de alimentação pode ser proveniente da leitura de sensores, invocação de serviços, entrada de usuários e/ou elementos de sistemas como componentes, portas, conexões e outros *containers*. As informações monitoradas pelo CtxMonitor são expostas através dos seus atributos e o próprio elemento pré-processa (abstração para níveis mais altos) essas informações antes de repassá-las ao CtxPolicySet.

O metamodelo apresentado nessa subseção é definido através de um perfil UML que contém esteriótipos¹² e valores etiquetados¹³. Para cada elemento do metamodelo (Figura 13), um esteriótipo com o mesmo nome é definido. Esses esteriótipos e valores etiquetados são empregados para diversas finalidades como dicas para geração automática de códigos.

¹¹ Receptáculos e facetas de mesma funcionalidade.

¹² Mecanismos de extensibilidade UML para classificação de elementos “com algo em comum”.

¹³ Grupo de valores fundamentais que podem ser associados a um elemento modelado, permite etiquetar um valor sobre algum elemento no modelo.

O perfil UML para representação do metamodelo contém uma linguagem de alto nível com sintaxe especial para valores das etiquetas de Regras (*Rules*), Condição (*Condition*) e Ação (*Action*) que permite a definição de políticas de adaptação.

3 DESENVOLVIMENTO

O modelo de aplicação sensível ao contexto proposto nesse estudo tem a sua fundamentação teórica baseada na revisão bibliográfica exposta anteriormente. Seu objetivo é o de abstrair um ambiente de execução onde a aplicação tenha a capacidade de receber, processar, armazenar e enviar dados provenientes dos sensores não-invasivos acoplados ao usuário ou inseridos no espaço inteligente. Sua estrutura relacional consiste em entidades estereotipadas conforme o metamodelo apresentado na subseção 2.3 e apresenta um *container* denominado **HealthContainer**, o qual oferece os requisitos não-funcionais referentes a sensibilidade ao contexto e um componente com o nome de **HealthComponent** que encapsula as funcionalidades do sistemas, contendo todos os requisitos funcionais.

As informações referentes a posição global do paciente são monitoradas pelo **GPSSMonitor**, o qual recebe informações sobre longitude, latitude e horário local. Essas informações são relevantes para o sistema, pois indicam se o paciente encontra-se ou não dentro de um espaço inteligente, possibilitando o monitoramento contínuo.

Dados sobre temperatura e umidade ambiente são monitoradas pelos **TemperaturaAmbienteMonitor** e **UmidadeAmbienteMonitor**, respectivamente. A temperatura pode influenciar na vasodilatação e a umidade do ar pode influenciar em problemas nasais que acarretam no aumento da pressão arterial devido maior esforço para respiração.

Os sinais fisiológicos sobre a pressão arterial, a frequência cardíaca, a atividade cerebral e a atividade muscular são capturados por sensores não-invasivos acoplados ao paciente e transmitidos aos seus respectivos Monitores que os pré-processam e os abstraem a níveis mais altos de compreensão para o sistema. O **PAMonitor** recebe os dados sobre pressão sistólica¹⁴ e diastólica¹⁵ e os pré-

¹⁴ Corresponde a contração do músculo cardíaco, levando o sangue (hemoglobina) para ser oxigenado nos pulmões. Seu valor é denominado como pressão arterial máxima e varia entre 120 e 140 mmHg.

¹⁵ Corresponde ao relaxamento do músculo cardíaco, ou seja, o sangue (hemoglobina) já foi oxigenado pelos pulmões e volta como sangue arterial para suprir as células de oxigênio. Seu valor é denominado como pressão arterial mínima e têm média de 80 mmHg.

processa para obter informações referentes a pressão arterial. O **ECGMonitor** recebe os sinais do sensor de desempenho cardíaco (ECG¹⁶), informando os valores sobre a frequência cardíaca. O **EEGMonitor** monitora os dados do sensor de atividades cerebrais (EEG¹⁷) e o **EMGMonitor** capta informações dos sensores de atividade muscular (EMG¹⁸).

Os níveis de qualidade de conexão são monitorados pelo **WiFiMonitor** e **3GMonitor**. Ambos verificam a velocidade de conexão, o *MAC address*, *ESSID* e Endereço IP dos seus respectivos tipos de conectividade (*Wi-fi* e 3G).

O **BusinessIntelligence** é o componente central do modelo de aplicação e contém a Máquina de Inferência, a Memória de Trabalho e a Base de Regras, além de receber informações de todos os Monitores.

A **Máquina de Inferência** tem o objetivo de comparar os valores provenientes dos Monitores com os padrões existentes em sua Memória de Trabalho gerenciando o processo de aplicação de regras que conta com o auxílio dos seguintes componentes:

- **Pattern Matcher** casa os padrões da Memória de Trabalho, decidindo qual regra será disparada.
- **Agenda** armazena a lista de regras que devem ser disparadas.
- **Máquina de Execução** dispara as regras.

A **Memória de Trabalho** consiste na base de fatos correspondentes a todos os fragmentos de informação existentes na Máquina de Inferência. E a **Base de Regras** é o repositório de todas as regras existentes no sistema e cada regra existente é vinculada a um atributo de um Monitor.

¹⁶ Eletrocardiograma – registro da variação dos potenciais elétricos gerados pela atividade elétrica do coração.

¹⁷ Eletroencefalograma – registro da atividade bioelétrica cerebral natural através de eletrodos fixados no escalpo.

¹⁸ Eletromiografia – registro das atividades elétricas associadas às contrações musculares.

O grupo de regras inicialmente existentes no modelo é formado pelos seguintes componentes:

CtxPolicyGPS: conjunto de premissas, ações e conclusões conforme localização do usuário. Indica se o paciente se encontra em algum espaço inteligente como a sua residência ou consultório médico.

CtxPolicy3G e CtxPolicyWiFi: conjunto de premissas, ações e conclusões conforme nível de qualidade de conexão. Informam o procedimento para procura e estabelecimento de novas conexões, além de orientar a transferência de dados de maneira oportunista.

CtxPolicyTemperaturaAmbiente: conjunto de premissas, ações e conclusões conforme temperatura ambiente. Como exemplo, pode-se listar o acionamento de dispositivos para resfriamento ou aquecimento do ambiente físico em que o paciente está inserido.

CtxPolicyUmidadeAmbiente: conjunto de premissas, ações e conclusões conforme umidade do ar. Como exemplo, pode-se citar o controle de umidade através de umidificadores para amenizar problemas respiratórios.

CtxPolicyECG, CtxPolicyEEG, CtxPolicyEMG, CtxPolicyPA: conjunto de premissas, ações e conclusões conforme dados fisiológicos. Classificam o quadro de saúde em “normal”, “alerta” ou “emergência” conforme análise de dados.

A Figura 14 ilustra o modelo da aplicação baseado no metamodelo proposto na subseção 2.3.

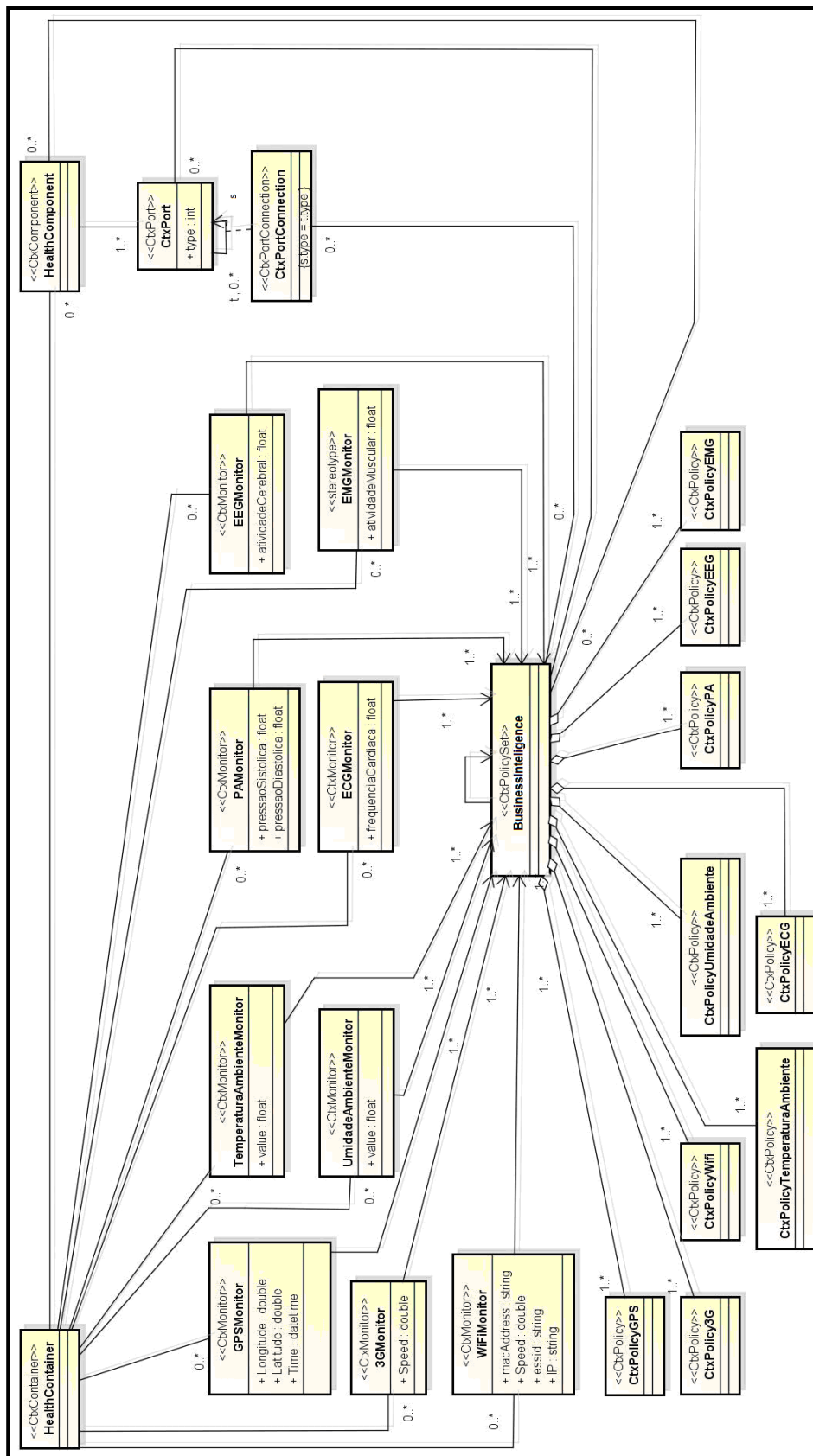


Figura 14 - Modelo para Aplicação de Monitoramento Móvel de Sinais Fisiológicos (Elaborada pelo Autor).

Os sinais fisiológicos oriundos dos sensores não-invasivos são transmitidos para os dispositivos móveis do paciente, onde são reconhecidos, processados, armazenados e enviados por Internet a um dispositivo central pertencente à equipe médica. O sistema permite que o próprio paciente visualize na tela do seu celular as informações referentes ao seu quadro de saúde, que é qualificado como normal, alerta ou emergência.

A todo instante, a equipe médica será informada sobre a rotina do paciente, detectando e gerenciando riscos de saúde. Esse gerenciamento pode diagnosticar enfermidades com antecedência, auxiliar na adequação de tratamento e reduzir o tempo de tomada de decisão.

O monitoramento ininterrupto realizado tanto por celulares, quanto por PDAs e computadores portáteis, reduz consideravelmente o tempo de resposta no diagnóstico de doenças, no acompanhamento do quadro clínico de pacientes pertencentes a grupos de riscos (por exemplo, hipertensos) e pacientes idosos, que requerem cuidado especial.

4 DISCUSSÃO DOS RESULTADOS

O modelo de aplicação apresentado nesse estudo apresenta uma arquitetura com potencial de monitorar dados fisiológicos e potenciais atividades do paciente. Sua arquitetura é desenvolvida sobre filosofia de sistemas baseados em componentes, o que permite a modularidade desde que exista componentes que ofereçam os requisitos funcionais e não-funcionais para tal.

A combinação de sinais fisiológicos e atividades do usuário podem constituir uma situação de risco ao paciente. Essas informações são processadas pela máquina de inferência que compara os valores contextuais adquiridos por sensores com a sua memória de trabalho. Qualquer valor fora dos padrões existentes é imediatamente reportado a equipe de saúde, agilizando o atendimento ao enfermo.

A integração do modelo de aplicação de monitoramento móvel aos sistemas de saúde permite:

- A integração de novos módulos, gerando flexibilidade de arquitetura desde que exista *containers* e componentes para suporte de novas funcionalidades.
- Monitoramento de sensores distintos, combinando informações para promover uma percepção de ambiente (corpo humano) mais precisa.
- Monitoramento *fulltime* de sinais fisiológicos e atividades do paciente, gerando *timeline* de informações, que podem auxiliar no reajuste de tratamentos.
- Monitoramento em qualquer espaço físico desde que exista uma infraestrutura computacional relativamente estável.
- Eliminação de fatores obstrutivos como fios, alimentação externa, tamanho e peso dos componentes envolvidos no sistema.

- Conforto para o paciente, reduzindo o número de visitas ao hospital e possível redução no período de internação.
- Adaptações aos ambientes altamente distribuídos desde que exista um conjunto de premissas, ações e conclusões.

Os sistemas de monitoramento móvel de saúde são um novo paradigma na integração entre profissionais de saúde e pacientes. Para tornar possível a implementação do modelo proposto nesse estudo, a estrutura de redes de sem fio e 3G precisam atender as normas de funcionamento previstas pela ANATEL, pois é conhecida a instabilidade, a qualidade e o não cumprimento dos requisitos mínimos desses serviços. Existe a necessidade da capacitação de programadores devido à peculiaridade dos sensores envolvidos no sistema. É necessária uma reformulação na estrutura da saúde pública, que não está preparada para receber tal ferramental, adequando as instalações hospitalares para oferecer uma infraestrutura computacional relativamente estável. Além disso, há, também, a necessidade de capacitação e treinamento dos profissionais de saúde e pacientes quanto ao funcionamento do sistema de monitoramento móvel.

5 CONSIDERAÇÕES FINAIS

Esse estudo foi desenvolvido com intuito de se ter um modelo de aplicação destinado ao monitoramento de saúde, móvel, ubíquo e sensível ao contexto.

O modelo de aplicação é baseado em sistemas estruturados em componentes de *software* e apresenta um *container* para os requisitos não-funcionais correspondentes aos mecanismos de sensibilidade ao contexto e um componente com os requisitos funcionais que encapsulam as funcionalidades do sistema. As informações fisiológicas são capturadas por sensores não-invasivos e transmitidas aos Monitores que pré-processam os sinais, abstraindo a níveis mais altos de compreensão e, posteriormente, alimentam a Máquina de Inferência. Ao ser alimentada pelos Monitores, a Máquina de Inferência compara as informações contextuais adquiridas com os seus padrões de Memória de Trabalho gerenciando o conjunto de premissas, ações e conclusões. Todo e qualquer valor monitorado é visualizado pelos dispositivos móveis integrados ao sistema e qualquer valor classificado como emergencial dispara regras de alerta sobre o quadro de saúde do paciente.

A estrutura baseada em componentes facilita a flexibilidade de arquitetura permitindo a inserção modular desde que exista componentes com a inteligência e a interface necessária para comunicação.

O monitoramento não-invasivo oferece mobilidade e conforto ao paciente que pode ser acompanhado dentro de sua residência, gerando dados sobre sua saúde e rotina diária. Essas informações possibilitam a redução no número de visitas hospitalares, maior precisão no tratamento e até a diminuição no período de internação.

Embora essa proposta implique em mais um estágio de implementação, o ganho em termos de qualidade de serviço, eficiência e eficácia são relevantes e, portanto, pode ser uma solução para o monitoramento de saúde. Prova disto é a Bélgica que em 2010 implementou a primeira BAN conectada a dispositivos móveis.

Por fim, conclui-se que a modelagem executada nesse estudo, e detalhada por meio de revisão bibliográfica, possibilita um melhor atendimento ao monitoramento móvel de saúde. Logo, entende-se que esse estudo apresenta o início de uma pesquisa que pode, por meio de um processo evolutivo, contribuir de forma significativa para a qualidade de vida de pessoas, em especial aquelas portadoras de anomalias cardíacas e/ou pertencentes a grupos de Risco.

6 REFERÊNCIAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **Citação:** NBR-10520/ago - 2002. Rio de Janeiro: ABNT, 2002.

_____. **Referências:** NBR-6023/ago. 2002. Rio de Janeiro: ABNT, 2002.

BRATSKAS, Pyrros; PASPALLIS, Nearchos; PAPADOPOULOS, George A. *An Evaluation of the State of the Art in Context-aware. Information System Development. 2009.* Disponível em:

< <http://www.cs.ucy.ac.cy/~george/ISD07b.pdf> >. Acesso em: 05 mar. 2013.

COULOURIS, George; DOLLIMORE, Jean; KINDBERG, Tim. **Sistemas distribuídos:** Conceitos e projetos. Tradução de João Tortello. 4. ed. Porto Alegre: Bookman, 2007. p.567 - 619.

DAVIES, Nigel; MYNATT, Elizabeth; SIIO, Itiro. *UbiComp 2004: Ubiquitous Computing. 6th International Conference, 2004.* Disponível em:

< <http://books.google.com.br/books?id=0VLqs1P5XqMC&printsec=frontcover&hl=pt-BR#v=onepage&q&f=false> >. Acesso em: 24 mai. 2013.

NIERSTRASZ, Oscar; DENKER, Marcus; RENGGLI, Lukas. *Model-Centric, Context-Aware Software Adaptation. Software Engineering for Self-Adaptive Systems, 2009.* Disponível em:

< <http://rossano.pro.br/fatec/cursos/topicosati/Nier09aModelCentric.pdf> >. Acesso em: 13 mar. 2013.

PINTO, Rossano Pablo; CARDOZO, Eleri; COELHO, Paulo R.S.L; GUIMARÃES, Eliane G. *A Domain-independent Middleware Framework for Context-Aware Applications. Wireless Communications and Mobile Computing Conference, 2008. IWCMC'08. International.* Disponível em:

< <http://rossano.pro.br/fatec/cursos/topicosati/a5-pinto.pdf> >. Acesso em: 24 abr. 2013.

PINTO, Rossano Pablo; CARDOZO, Eleri; GUIMARÃES, Eliane G. *A Component Framework for Context-Awareness. ARM '07 Proceedings of the 6th international workshop on Adaptive and reflective middleware: held at the ACM/IFIP/USENIX*
Disponível em: < <http://rossano.pro.br/fatec/cursos/topicosati/01-08-04.pdf> >. Acesso em: 24 abr. 2013.

TANENBAUM, Andrew S. **Organização estruturada de computadores**. Tradução de Arlete Simille Marques. 5. ed. São Paulo: Prentice Hall, 2007. p.322 - 382

TANENBAUM, Andrew S; STEEN, Maarten Van. **Sistemas distribuídos: princípios e paradigmas**. Tradução de Arlete Simille Marques. 2. ed. São Paulo: Pearson Prentice Hall, 2007. 1-19.