

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA
DA COMPUTAÇÃO**

Cristiano Cortez da Rocha

**Uma Arquitetura para Autenticação Sensível ao Contexto
Baseada em Definições Comportamentais**

Dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Mestre em Ciência da Computação.

**Prof. Mário Antônio Ribeiro Dantas, Dr., UFSC
(Orientador)**

Florianópolis, Agosto de 2010

Uma Arquitetura para Autenticação Sensível ao Contexto Baseada em Definições Comportamentais

Cristiano Cortez da Rocha

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação, área de concentração Computação Paralela e Distribuída e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

Prof. Mário Antônio Ribeiro Dantas, Dr. (Coordenador)

Banca Examinadora

Prof. Mário Antônio Ribeiro Dantas, Dr., UFSC
(Orientador)

Prof. Nelson Francisco Favilla Ebecken, Dr., UFRJ

Profa. Iara Augustin, Dra., UFSM

Prof. João Bosco Manguiera Sobral, Dr., UFSC

"Vim, vi e venci."

Júlio Cesar

Agradecimentos

Primeiramente, gostaria de endereçar os meus agradecimentos ao meu orientador e amigo Mário Dantas por oferecer-me a oportunidade e confiança para desenvolver o trabalho de pesquisa apresentado nesta dissertação. Além disso, gostaria de agradecer-lhe por todo o seu empenho que possibilitou a minha ida ao Canadá para a realização do meu intercâmbio acadêmico na University of Western Ontario (UWO) durante 5 meses.

Através deste intercâmbio, orientado pela professora Miriam Capretz e co-orientado pelo professor Michael Bauer, foi possível utilizar toda a magnífica estrutura oferecida pela UWO e SHARCNET (*Shared Hierarchical Academic Research Computing Network*) para a implementação e validação desta dissertação. Ainda, esta experiência internacional contribuiu muito para o meu aperfeiçoamento profissional e pessoal. Desta forma, gostaria de dizer muito obrigado a estas duas pessoas especiais.

Ainda, gostaria de agradecer o Professor Caio, meu eterno orientador e amigo, que me acompanha desde a minha graduação na UFSM, e que me auxiliou em grande parte das idéias que deram forma e refinamento a este trabalho de pesquisa. Gostaria de agradecer, também, a professora Iara Augustin que me deu apoio técnico e financeiro, através de bolsas de estudo, desde a minha graduação na UFSM e que, durante o meu mestrado na UFSC, continuou com todo esse apoio importantíssimo.

Gostaria de agradecer, também, a doutoranda Heloíse Mânica por oferecer-me a oportunidade de participar do trabalho de pesquisa que fez parte da sua tese. A participação neste trabalho conjunto trouxe uma grande contribuição para esta dissertação, especialmente, para os experimentos envolvendo dispositivos móveis.

Por fim, gostaria de agradecer a minha família: minha mãe Rosana, meu pai Fernando e minha irmã Fernanda. Essas pessoas sempre estiveram do meu lado em todos os momentos importantes e decisivos desta caminhada, assim como os meus tios e avós e minha namorada Priscilla. Amo todos vocês! Obrigado!

Sumário

Sumário	ix
Lista de Figuras	xi
Lista de Tabelas	xiii
Lista de Acrônimos	xv
Resumo	xvii
Abstract	xviii
1 Introdução	1
2 Computação Móvel e Pervasiva	5
2.1 Computação Sensível ao Contexto	6
2.1.1 Abstração de Contexto	7
2.1.2 Requisitos	8
2.1.3 Modelagem Contextual	11
2.2 Computação Centrada no Usuário	17
2.2.1 Reconhecimento de Atividades	18
2.2.2 Modelagem Comportamental	19
2.2.3 Considerações sobre as Abordagens	25
3 Autenticação Sensível ao Contexto	27
3.1 Contexto Relevante à Segurança	28
3.2 Requisitos	28
3.3 Abordagens Propostas	29
3.3.1 Abordagem de Babu e Venkataram	29
3.3.2 Abordagem de Hung et al.	30
3.3.3 Abordagem de Corradi, Montanari e Tibaldi	32
3.3.4 Considerações sobre as abordagens propostas	33

4	Arquitetura para Autenticação Sensível ao Contexto	37
4.1	Modelagem Contextual	37
4.2	Modelagem Comportamental	39
4.3	Modelagem Analítica	39
4.3.1	Modelo de Permutação Espaço-Temporal	39
4.3.2	Modelo de Vetor de Similaridade	41
4.4	Arquitetura	41
4.5	Considerações sobre a Abordagem Proposta	45
5	Resultados Experimentais	47
5.1	Descrição do Ambiente Experimental	47
5.2	Protótipo Implementado	50
5.2.1	Detecção de Eventos	50
5.2.2	Níveis de Autenticação	50
5.2.3	SaTScan	52
5.3	Análise Comparativa da Abordagem Proposta	55
5.3.1	Modelagem Analítica	56
5.3.2	Eficiência Computacional	57
5.3.3	Autonicidade e Dinamicidade	64
6	Conclusões e Trabalhos Futuros	71
	Referências Bibliográficas	73
A	Publicações	83
A.1	Capítulos de Livros Publicados	83
A.2	Trabalhos Completos Publicados em Anais de Congressos	83
B	Configuração do Ambiente de Desenvolvimento	87
B.1	Ambiente de Software	87
B.1.1	Java 2 Micro Edition (J2ME)	87
B.1.2	Framework SuMMIT	87

Lista de Figuras

2.1	Visão geral das diferentes camadas de interpretação e abstração de contextos	8
2.2	Exemplo da modelagem CML [Bettini et al., 2009]	13
2.3	Comparação entre as abordagens de modelagem de contexto	16
2.4	Representação gráfica de um modelo oculto de Markov (HMM) [Kim et al., 2010]	21
2.5	Exemplo de um HMM para a atividade "comer"[Kim et al., 2010]	22
2.6	Um exemplo de estrutura de um campo aleatório condicional com o relacionamento entre a sequência de eventos (Y) e as observações X	23
3.1	TBAS - Análise comportamental do usuário [Babu and Venkataram, 2009]	30
3.2	Arquitetura do sistema de segurança baseado em atividades [Hung et al., 2008]	31
3.3	UbiCOSM - Arquitetura proposta [Corradi et al., 2004]	33
4.1	Modelo contextual	38
4.2	Arquitetura proposta [Rocha et al., 2010a]	43
5.1	Cenário experimental	48
5.2	Funcionamento do mecanismo Push Registry [Knudsen, 2002]	51
5.3	SaTScan - Arquitetura de execução	54
5.4	Comparativo entre os modelos analíticos	58
5.5	Comparativo de consumo energético entre os métodos de posicionamento	60
5.6	Rede Bayesiana em estado de latência	63
5.7	Rede Bayesiana instanciada	63
5.8	Tela de inserção de dados iniciais	66

5.9	(a) Perfil explícito e (b) Perfil de sessão extraídos das informações inseridas pelo usuário	66
5.10	(a) Descrição do evento da segunda interação com o sistema e (b) Perfil de sessão	67
5.11	Desafio apresentado ao usuário	69
5.12	Evolução do sistema conforme o número de interações	70
B.1	Visão geral da plataforma Java	88
B.2	Arquitetura do framework SuMMIT [Rossetto et al., 2007]	89
B.3	Resource Selector - Interação entre os módulos [Rossetto et al., 2007]	90
B.4	Arquitetura estendida [Viera et al., 2010]	91

Lista de Tabelas

3.1	Resumo das características dos trabalhos relacionados . .	35
5.1	Características de hardware e software do ambiente experimental	49
5.2	Relação entre os níveis de autenticação, natureza do usuário e desafios propostos	52
5.3	Descrição dos grupos de localizações	57
5.4	Probabilidades condicionais a priori	62
5.5	Probabilidades para todas as instâncias	64

Lista de Acrônimos

API	<i>Application Programming Interface</i>
ARM	<i>Activity Recognition Manager</i>
ASCII	<i>American Standard Code for Information Interchange</i>
CC/PP	<i>Composite Capability/Preference Profiles</i>
CML	<i>Context Modelling Language</i>
CRF	<i>Conditional Random Field</i>
DBN	<i>Dynamic Bayesian Network</i>
EP	<i>Emerging Pattern</i>
GPS	<i>Global Positioning System</i>
HMM	<i>Hidden Markov Model</i>
HTTP	<i>HyperText Transfer Protocol</i>
HTTPS	<i>HyperText Transfer Protocol Secure</i>
J2ME	<i>Java 2 Platform, Micro Edition</i>
J2SE	<i>Java 2 Platform, Standard Edition</i>
MCA	<i>Mobile Cognitive Agent</i>
MIDP	<i>Mobile Information Device Profile</i>
ORM	<i>Object-Role Modeling</i>
OWL	<i>Web Ontology Language</i>
OWL-DL	<i>Web Ontology Language - Description Logic</i>

PDA	<i>Personal Digital Assistant</i>
SCA	<i>Static Cognitive Agent</i>
SSL	<i>Secure Sockets Layer</i>
SuMMIT	<i>Submission, Monitoring and Management of Interactions of Tasks</i>
TBAS	<i>Transaction-Based Authentication Scheme</i>
UbiCOSM	<i>Ubiquitous Context-Based Security Middleware</i>
VSM	<i>Vector Space Model</i>

Resumo

Dispositivos móveis tornaram-se equipamentos indispensáveis nos tempos modernos, devido à oferta de recursos cada vez mais sofisticados. Esses dispositivos têm sido utilizados em diversos ambientes de computação distribuída, a fim de permitir que os usuários estabeleçam conexões com organizações a qualquer momento e localização.

Entretanto, as conexões estabelecidas são, geralmente, baseadas em processos tradicionais de autenticação que não consideram as características ambientais, restrições dos dispositivos e aplicações e informações provenientes de sensores presentes no espaço pervasivo. Assim, tais mecanismos são ineficazes para um ambiente altamente dinâmico como um ambiente de computação móvel. Logo, uma abordagem baseada em contextos pode representar uma alternativa para contornar esses desafios.

Nesta dissertação é apresentada uma arquitetura que adota a autenticação de usuários baseada em um contexto espaço-temporal. Especificamente, a modelagem contextual proposta visa o aperfeiçoamento do processo de autenticação, onde os eventos capturados pelos diversos sensores encontrados em dispositivos móveis e no próprio ambiente pervasivo passam por um modelo analítico que considera tempo e espaço, simultaneamente e, também, analisa o perfil do usuário a partir desses eventos. O sistema é capaz de determinar a necessidade de autenticação conforme o perfil comportamental e as políticas de segurança das aplicações, refinando, assim, o processo de autenticação conforme as interações do usuário com o sistema.

Nos experimentos realizados, a arquitetura de autenticação sensível ao contexto mostrou ser eficiente, pois aproveita a riqueza de recursos dos dispositivos móveis para obter uma visão mais completa do cenário, aperfeiçoando o processo de tomada de decisão. As melhorias alcançadas através da adoção desta abordagem são: maior eficiência computacional, dinamicidade, autenticidade e flexibilidade.

Palavras-chave: autenticação sensível ao contexto, modelagem espaço-temporal, computação móvel e pervasiva

Abstract

Mobile devices are becoming mandatory equipment in modern times due to the growing processing power and offer of sophisticated resources. These devices are increasingly being used in various distributed computing environments to allow users to be connected to their institutions anytime and anywhere.

However, the connections are usually based on traditional authentication processes, which do not consider the environmental characteristics, restrictions of devices and application, and information provided by sensors within the pervasive space. Thus, these mechanisms are ineffective for a highly dynamic environment such as a mobile computing environment. Then, a context-based approach can represent a useful alternative for circumventing these challenges in a mobile computing scenario.

In this dissertation, it is presented an architecture which adopts an authentication process based on a spatio-temporal context. Specifically, the proposed contextual modeling aim at the improvement of the authentication process where the events descendant from several sensors found in mobile devices are analyzed by a space-time permutation model which considers the time and space simultaneously. Thus, the system is able to determine the need of the authentication process according to the behavioral profile and application security policies. In particular, it provides greater refinement and precision in the behavioral analysis of the user on the basis of his interaction with the system.

In our experiments, the context-aware authentication architecture proved to be efficient because it uses the richness of resources of the mobile devices to obtain a complete view of the user's scenario. Consequently, the authentication system is able to make decisions efficiently by considering the overall situation for the user. Examples of the improvements reached with the adoption of this approach are: greater computing efficiency, dynamism, autonomy, and flexibility.

Keywords: context-aware authentication, space-time modeling, mobile and pervasive computing

Capítulo 1

Introdução

O aumento do poder de processamento de dispositivos móveis tem aproximado tais equipamentos às plataformas computacionais tradicionais, como, desktops e laptops [Black and Edgar, 2009]. Especificamente, a velocidade dos processadores dos smartphones mais recentes aproximam-se de 1 GHz e o tamanho da memória primária estão, agora, situadas entre 128 e 256 MB [Apple, 2010], [HTC, 2010], [RIM, 2010]. Estimativas recentes apontam a existência de 4,1 bilhões de telefones móveis, ao mesmo tempo que os números da indústria indicam que cerca de 20% das vendas de novos dispositivos móveis compreendem em smartphones [Black and Edgar, 2009].

Desta forma, tais dispositivos são capazes de prover recursos cada vez mais sofisticados, como, GPS (*Global Positioning System*), por exemplo. Esses recursos, então, podem ser utilizados para determinar a situação do usuário em um determinado espaço de tempo, pois, pode-se inferir sobre o seu contexto ambiental, que são as entidades que cercam o usuário, contexto operacional, representando as atividades que usuário realiza e o contexto espacial, que determina aonde o usuário está localizado.

Essa situação do usuário, que é determinada através de diversos contextos, consiste em um ambiente típico de computação móvel e pervasiva. Tal ambiente representa um cenário onde usuários e atividades humanas cotidianas compõem a parte central do ambiente. Um ambiente pervasivo é caracterizado pela riqueza de contextos, no qual usuários, dispositivos e agentes se deslocam entre diversos lugares e diversas entidades, como serviços, aplicações e recursos alternando sua disponibilidade sobre o tempo [Saha and Mukherjee, 2003]. Assim, a informação relacionada ao cenário tempo-espaço pode ser utilizada para definir um padrão de comportamento de uma entidade em diversos contextos. Logo, mecanismos que são comumente empregados para coletar informações do contexto do usuário incluem redes de sensores, GPS e agenda do usuário que são oferecidos pelos dispositivos móveis.

Entretanto, o comportamento do usuário é um problema complexo

que necessita ser enfrentado, visto que modelos são desenvolvidos para lidar com diversos comportamentos de usuários. Como resultado, a computação sensível ao contexto é um aspecto importante que pode ser observado em ambientes ubíquos com o objetivo de entender a relação entre usuários, dispositivos e ambientes [Uden, 2007]. Mais especificamente, a quantidade significativa de informação em um ambiente ubíquo pode auxiliar na compreensão do comportamento do usuário em tal ambiente.

Por outro lado, poucas propostas recentes consideram características seguras e transparentes nesses ambientes. Como demonstrado em [Johnson, 2009] e [Babu and Venkataram, 2009], problemas de segurança são comumente observados em diversos projetos. Em particular, desafios como acesso aberto, conexões instáveis, restrições de largura de banda e consumo energético são questões comumente abordadas em diversos esforços [Babu and Venkataram, 2009], e são questões relevantes para aspectos de segurança como autenticação, integridade e confidencialidade.

Com esse propósito, o presente trabalho de pesquisa visa estudar os recursos comumente oferecidos por dispositivos móveis a fim de propor uma abordagem de autenticação sensível ao contexto que considere os diversos contextos que integram o contexto do usuário. Desta forma, o mecanismo de autenticação é capaz de tomar decisões baseadas em um cenário mais completo, obtendo uma maior precisão e refinamento na análise da situação do usuário. Especificamente, esta dissertação tem o foco no estudo e análise de requisitos que devem ser considerados por mecanismos de autenticação sensíveis ao contexto. Em adição, constatou-se que existe uma falta de abordagens que considerem, simultaneamente, aspectos como flexibilidade (diferentes meios de autenticação conforme as diferentes restrições e políticas de segurança), eficiência computacional (utilização de métodos para redução do consumo energético), dinamicidade (capacidade do sistema em agregar o conhecimento e habilidades adquiridas pelo usuário) e autonomicidade (capacidade do sistema em envolver o mínimo de intervenção humana possível).

Portanto, a fim de verificar o comportamento da abordagem proposta, é definido um modelo contextual que visa utilizar os recursos encontrados em dispositivos móveis a fim de estabelecer um perfil comportamental do usuário, que é analisado probabilisticamente através de um modelo de permutação espaço-temporal. Tal modelo provê uma melhora significativa na análise e detecção de anomalias no processo de autenticação, pois considera dados históricos que revelam os hábitos (comportamentos) do usuário. Além disso, analisou-se diversas propriedades que afetam o consumo energético dos dispositivos, adotando as alternativas que oferecem o menor custo computacional e provêm as funcionalidades

exigidas. Ainda, a arquitetura proposta mostrou-se dinâmica, pois é capaz de agregar o conhecimento e habilidades adquiridas pelo usuário durante a sua interação com o sistema.

A dissertação está organizada em seis capítulos. No capítulo 2 encontram-se questões relevantes sobre ambientes de computação móvel e pervasiva, identificando os requisitos que devem ser considerados para tais ambientes. O tópico sobre autenticação sensível ao contexto é abordado no capítulo 3, aonde são levantadas as principais características que devem ser implementadas por mecanismos de autenticação sensíveis ao contexto. No capítulo 4 é apresentada a proposta da arquitetura para autenticação sensível ao contexto, que emprega a análise do perfil comportamental dos usuários. Os resultados experimentais dos testes de campo da dissertação, juntamente com as decisões tomadas durante o processo de implementação e validação da arquitetura são apresentados e discutidos no capítulo 5. Por fim, no capítulo 6, são apresentadas as conclusões e indicações de trabalhos futuros a respeito desta dissertação.

Capítulo 2

Computação Móvel e Pervasiva

Os recentes avanços na indústria de semicondutores e comunicação sem fio têm possibilitado a redução de tamanho e custos de tecnologias computacionais e de sensoriamento, provendo ferramentas eficientes e plataformas de hardware para o monitoramento de indivíduos e seus ambientes [Atallah and Yang, 2009]. Conseqüentemente, isso tem possibilitado a observação das ações, atividades e interações das pessoas de forma pervasiva.

Conforme [Saha and Mukherjee, 2003], computação ubíqua ou pervasiva representa um cenário aonde usuários e atividades humanas cotidianas compõem a parte central do ambiente. Assim, a visão de Weiser [Weiser, 1991], que prevê as tecnologias computacionais presentes de forma ubíqua no dia-a-dia das pessoas, começa a fazer parte dos nossos ambientes. Tal cenário, descrito por [Saha and Mukherjee, 2003] é caracterizado pela riqueza de contextos que cercam as diversas entidades participantes. As entidades, como, usuários, dispositivos e agentes, deslocam-se entre diversas localizações e entidades, como, serviços, aplicações e recursos, alterando sua disponibilidade através do tempo.

Por outro lado, dispositivos móveis, como telefones celulares e PDAs, têm sido amplamente empregados como interfaces flexíveis para acesso *anytime* e *anywhere* a entidades, como, recursos e serviços, em ambientes de computação móvel [Eagle and Pentland, 2006]. Através dessa rápida adoção dos dispositivos móveis, surge uma oportunidade em potencial para coletar um conjunto de dados extremamente maior do comportamento humano [Mannila et al., 2001], [Mantjarvi et al., 2004].

Desta forma, conforme Uden [Uden, 2007], tais dispositivos são adequados para aplicações sensíveis ao contexto, pois são capazes de capturar diversas características que auxiliam na determinação da situação do usuário. Entre essas características, destacam-se as características espaciais, como, a localização do usuário, características ambientais, como, entidades que cercam o usuário e características operacionais, que revelam as tarefas que o usuário desempenha. Além disso, tais características

tornam os dispositivos móveis em veículos ideais para o estudo de indivíduos e organizações: pessoas habitualmente carregam seus dispositivos móveis consigo e os utilizam como meio de comunicação [Eagle and Pentland, 2006].

Apesar da significativa melhora do poder de processamento e armazenamento apresentado por esses equipamentos [Black and Edgar, 2009], o usuário ainda tem que lidar com uma série de limitações impostas pelo meio em que os dispositivos móveis operam. Entre tais limitações, destacam-se:

- autonomia energética;
- instabilidade e qualidade da conexão;
- dinamicidade do ambiente;
- heterogeneidade de dispositivos; e
- interfaces pouco amigáveis para a entrada explícita de dados.

2.1 Computação Sensível ao Contexto

Sensores conectados à rede, câmeras e computadores estão fazendo parte dos nossos ambientes cotidianos em quantidade e velocidade jamais vistas [Malek et al., 2008]. Em muitos lugares, dispositivos computacionais são, comumente, encontrados transmitindo dados provenientes de sensores e/ou câmeras de vídeo sobre a Internet. No paradigma de computação sensível ao contexto (*context-aware computing*), é proposto um ambiente em que diversos sensores, computadores e dispositivos de gravação de som e imagem estão conectados e, ativamente, monitorando mudanças no ambiente.

Tais dispositivos de sensoriamento e computadores colaboram, de forma interativa e pró-ativa, a fim de facilitar a interação homem-computador ou prover informações detalhadas do que ocorre no ambiente. Em tal paradigma, o ambiente é considerado "inteligente", visto que este é equipado com sensores com a finalidade de estar ciente das mudanças nesses ambientes. Um ambiente inteligente é, também, referenciado como um "espaço inteligente" (*smart space*) na literatura. Desta forma, os dados fornecidos por sensores ou outros dispositivos que são capazes de capturar informações do meio em que operam são chamados de dados contextuais, visto que eles contêm informações sobre o contexto em que cada entidade se encontra [Toninelli et al., 2009].

Segundo [Dey, 2001], contexto refere-se a qualquer informação que possa determinar a situação de uma entidade. Portanto, essas informações são relevantes para o processo adaptativo de informações e serviços

oferecidos ao usuário através de aplicações sensíveis ao contexto. Além disso, conforme [Preece et al., 1994], contexto é a disciplina que está preocupada com o projeto, avaliação e implementação de sistemas computacionais interativos para uso humano e com o estudo de importantes fenômenos que cercam os usuários e suas atividades.

Sendo assim, contexto desempenha um papel importante na compreensão e desenvolvimento de aplicações pervasivas, visto que as atividades do usuário não podem ser isoladas do ambiente em que elas ocorrem. Especificamente, as atividades do usuário não podem ser compreendidas sem um contexto [Uden, 2007].

2.1.1 Abstração de Contexto

Informações provenientes de sensores físicos, chamados de *contextos de baixo nível*, e adquiridas sem qualquer interpretação adicional, podem ser inexpressivas, triviais, vulneráveis a pequenas mudanças ou incertas [Ye et al., 2007]. Consequentemente, [Schilit et al., 1994] observou que contexto engloba mais do que apenas propriedades básicas, como a localização do usuário, pois outros objetos de interesse, como a situação social do usuário, também sofrem mudanças.

A limitação dos contextos de baixo nível ao se modelar interações e comportamentos humanos pode reduzir a utilidade de aplicações sensíveis ao contexto. Uma alternativa para contornar este problema consiste na derivação de informações de *contexto de alto nível* a partir dos valores não processados dos sensores. Esse processo é, também, chamado de *raciocínio e interpretação de contexto*. A idéia é abstrair o contexto de baixo nível ao criar uma nova camada que recebe as percepções dos sensores como entrada e gera ou invoca ações do sistema.

Na literatura, diferentes noções têm sido empregadas para referenciar a camada de contexto de alto nível. *Contexto situacional* [Gellersen et al., 2002] e *situação* [Dey, 2001, Dobson and Ye, 2006] são os conceitos mais utilizados. Assim, a noção de situação é empregada como um conceito de alto nível para um estado de representação. Desta forma, em aplicações sensíveis ao contexto, situações compreendem interpretações semânticas externas de contextos de baixo nível [Dobson and Ye, 2006], permitindo uma especificação de alto nível do comportamento humano presente e os serviços correspondentes do sistema. Ainda, situações inserem significado à aplicação e são mais estáveis e fáceis de definir e manter, se comparadas às propriedades contextuais básicas.

Adaptações em aplicações sensíveis ao contexto, então, são causadas pela mudança de situações, ou seja, uma mudança no valor do contexto invoca um processo adaptativo se a atualização desse valor de con-

texto modifica a situação corrente. Portanto, o projeto e implementação de aplicações tornam-se mais simples através de situações, pois o projetista/programador pode operar em alto nível de abstração (situação), ao invés de trabalhar com todas as propriedades que determinam tal situação [Bettini et al., 2009].

A Figura 2.1 ilustra as abstrações de contextos descritas nesta seção. Dados provenientes de contextos de baixo nível (baseados em sensores) são semanticamente interpretados pela camada de contexto de alto nível. Situações abstraem dados da camada de baixo nível e são reusáveis em diferentes ambientes e aplicações. Além disso, relacionamentos definidos entre situações podem prover uma abstração e redução de complexidade ainda maior.



Figura 2.1: Visão geral das diferentes camadas de interpretação e abstração de contextos

2.1.2 Requisitos

Recentemente, um grande número de aplicações sensíveis ao contexto baseadas em diversos modelos contextuais tem sido desenvolvido para uma variedade de domínios. As experiências adquiridas através dessa variedade de aplicações influenciaram o conjunto de requisitos necessários para modelagem e raciocínio sobre contexto. Desta forma, a comunidade de computação móvel e pervasiva tem demonstrado de forma crescente que o desenvolvimento de aplicações sensíveis ao contexto deve ser suportado pela modelagem adequada das informações contextuais e por técnicas de raciocínio e inferência. Assim, faz-se necessária a represen-

tação formal das propriedades dos contextos através de um modelo, tanto para a verificação de consistência, quanto para garantir que o raciocínio é realizado de forma adequada sobre os dados coletados [Bettini et al., 2009]. Em [Bettini et al., 2009] é apresentado o estado da arte de abordagens de modelagem de contexto e raciocínio. Além disso, esse trabalho destaca os principais requisitos que devem ser considerados por sistemas sensíveis ao contexto quanto a aspectos de modelagem, gerenciamento e raciocínio. Tais requisitos são:

- *Heterogeneidade*: modelos projetados para as informações coletadas devem lidar com uma grande variedade de fontes provedoras de informações que diferem em taxa de atualização de dados e nível semântico. Ao mesmo tempo que alguns sensores podem observar determinadas situações do ambiente físico e prover acesso rápido e em tempo real, outros sensores oferecem dados não processados (como, coordenadas de um GPS ou imagens de uma câmera), que necessitam ser interpretados antes de serem utilizados pelas aplicações. Por outro lado, informações fornecidas pelo usuário - como perfis de usuário - são atualizadas raramente e, geralmente, não requerem interpretações adicionais. Além disso, informações contextuais podem ser derivadas de informações contextuais já existentes. Por exemplo, dados obtidos de banco de dados ou bibliotecas digitais - como dados de mapas geográficos - são, frequentemente, estáticos. O modelo de contexto deve ser capaz de expressar esses diferentes tipos de informação e o sistema de gerenciamento de contexto deve prover gerenciamento às informações dependendo do seu tipo.
- *Mobilidade*: muitas aplicações sensíveis ao contexto são móveis (i.e., executadas em dispositivos móveis) ou dependem de fontes móveis para o fornecimento de informações contextuais (por exemplo, sensores móveis). Consequentemente, isso acrescenta problemas à heterogeneidade, visto que o fornecimento de informações contextuais deve ser adaptável à mudança ambiental. Além disso, a localização e o caráter espacial das informações capturadas desempenham um importante papel devido a este requisito.
- *Relacionamentos e dependências*: existem diversos tipos de relações entre informações de contextos que devem ser capturados, a fim de garantir o comportamento correto das aplicações. Um desses relacionamentos é a dependência que pode existir entre informações contextuais de entidades/fatos. Por exemplo, uma mudança no valor

de uma propriedade (como, largura de banda) pode afetar os valores de outras propriedades (como, nível restante de bateria).

- *Histórico*: aplicações sensíveis ao contexto podem necessitar do acesso a estados passados e estados futuros (prognóstico). Portanto, históricos de contextos é outra característica que deve ser capturada pelos modelos de contexto e gerenciada pelo sistema de gerenciamento de contexto. Alguns trabalhos (por exemplo, [Peddemors et al., 2010], [Uden, 2007], e [Cassens and Kofod-Petersen, 2006]) buscam definir o papel do tempo em sistemas sensíveis ao contexto. Em tais esforços, nota-se que o histórico é uma parte importante do contexto como um todo e não apenas de um sub-conjunto deste, pois o comportamento de uma entidade reflete sua cultura e desenvolvimento histórico em circunstâncias específicas [McMichael and Melbourne, 1999]. Sendo assim, a informação temporal é crucial para a interpretação do comportamento das entidades que integram o espaço pervasivo. Por outro lado, o gerenciamento de históricos de contextos é penoso, caso o número de atualizações é muito alto. Assim, pode ser inviável o armazenamento de cada valor para acesso futuro, e, então, técnicas de sumarização, como, a utilização de sinopses históricas dos dados, devem ser aplicadas.
- *Imperfeição*: devido a sua natureza dinâmica e heterogênea, as informações contextuais podem diferir quanto a sua qualidade, ou, até mesmo, essas podem ser incorretas. Grande parte dos sensores apresentam uma imprecisão inerente (como, alguns metros para coordenadas fornecidas por um GPS). Além disso, as informações podem ser incompletas ou conflitantes com outras informações contextuais. Então, uma modelagem contextual adequada deve incluir a modelagem de qualidade da informação contextual, a fim de suportar o raciocínio sobre o contexto.
- *Raciocínio*: aplicações sensíveis ao contexto utilizam as informações para avaliar se existe uma mudança na situação do contexto do usuário e/ou ambiente computacional. A decisão de adaptação a alguma mudança, geralmente, requer capacidades de raciocínio. Portanto, é importante que técnicas de modelagem de contexto sejam capazes de suportar tanto verificação de consistência do modelo, quanto técnicas de raciocínio sobre o contexto. Estas podem ser utilizadas para derivar novos fatos/situações a partir de fatos/situações já existentes, além de avaliar abstrações de contextos de alto nível que modelam as situações do mundo real. Além disso, as técnicas de raciocínio devem ser computacionalmente eficientes.

- *Usabilidade de modelos formais*: modelos contextuais são criados por projetistas de aplicações sensíveis ao contexto. Esses, também, são utilizados por sistemas de gerenciamento de contextos e aplicações sensíveis ao contexto, a fim de manipular as informações contextuais. Consequentemente, as características importantes dos formalismos de modelagem são: a facilidade com que os projetistas podem traduzir os conceitos do mundo real para as estruturas do modelo e a facilidade com que as aplicações conseguem utilizar e manipular as informações contextuais em tempo de execução.
- *Acesso eficiente*: ao mesmo tempo que o acesso eficiente às informações contextuais é necessário, este pode ser um requisito difícil de implementar na presença de modelos grandes e numerosos dados. A fim de selecionar os objetos relevantes, atributos que indicam o caminho para acesso adequado devem ser representados na modelagem contextual. Esses caminhos de acesso representam as dimensões em que as aplicações frequentemente selecionam as informações dos contextos, tipicamente suportadas por índices. Essas dimensões são, geralmente, referenciadas aos contextos primários, em contraste aos contextos secundários, que são acessados utilizando-se o contexto primário. Comumente, os atributos de contextos primários utilizados são a identificação de objetos de contextos, localização, tipo de objeto, tempo ou atividade do usuário.

2.1.3 Modelagem Contextual

Nas últimas décadas, tem-se observado um grande número de abordagens propostas para a modelagem de contexto que variam desde os primeiros modelos propostos (mais simples), até os modelos atuais para modelagem contextual [Bettini et al., 2009]. A pesquisa em modelagem contextual tem sido acompanhada pelo desenvolvimento de sistemas de gerenciamento de contextos que são capazes de coletar, gerenciar, avaliar e disseminar informação contextual. Portanto, o desenvolvimento de tais sistemas, também, tem influenciado a pesquisa de modelos que ofereçam grande capacidade de representação, suporte ao raciocínio sobre contexto e desempenho computacional satisfatório. As seções seguintes apresentam modelos que oferecem boas soluções para alguns dos requisitos identificados na seção 2.1.2.

2.1.3.1 Modelagem Baseada em Fatos

Abordagens de modelagem baseada em fatos foram inicialmente desenvolvidas a partir da tentativa de criar modelos formais de contextos, a fim de suportar o processamento de consultas e raciocínio, assim

como prover construções de modelagem adequadas para a sua utilização em etapas da engenharia de software, como, análise e projeto [Bettini et al., 2009]. Tais abordagens de modelagem possuem seus fundamentos baseados em técnicas de modelagem de banco de dados.

Uma das abordagens mais populares consiste na *Context Modeling Language* (CML), que foi definida primeiramente em [Henricksen et al., 2002] e, então, refinada em trabalhos posteriores [Henricksen and Indulska, 2004, Henricksen and Indulska, 2006]. Tal abordagem é baseada em *Object-Role Modeling* (ORM) [Halpin and Morgan, 2001], que foi desenvolvida para modelagem conceitual de banco de dados.

A CML captura a heterogeneidade das fontes de informações de contexto, histórico de informações contextuais e, ainda, provê um mapeamento claro dos conceitos do mundo real para estruturas de modelagem. Além disso, tal linguagem fornece um bom balanceamento entre capacidade de expressão e eficiência dos procedimentos de raciocínio. Entretanto, a modelagem baseada em fatos através da CML apresenta uma falta de suporte a descrições hierárquicas de contexto. A figura 2.2 apresenta um exemplo de modelagem de contexto utilizando-se CML.

2.1.3.2 Modelagem Espacial

Informação espacial desempenha um importante papel em aplicações sensíveis ao contexto. Consequentemente, grande parte das definições de contexto citam o espaço como uma propriedade vital. Por exemplo, conforme [Dey, 2001], espaço consiste em um aspecto central das entidades: "*Uma entidade é uma pessoa, lugar ou objeto que é considerado relevante na interação entre um usuário e uma aplicação, incluindo-se o usuário e a aplicação*". Desta forma, lugares são entidades espaciais e, ainda, a interação entre entidades, geralmente, envolve uma determinada proximidade. Portanto, algumas abordagens de modelagem de contexto priorizam fatores espaciais e temporais.

Modelos espaciais permitem raciocínio e interpretação sobre a localização e relacionamentos espaciais entre objetos. Tais relacionamentos compreendem a inclusão de objetos em determinada área e a distância entre objetos. Conforme [Becker and Dürr, 2005], existem três classes de consultas típicas que podem ser realizadas sobre informações espaciais: (i) posição: retorna a posição de um objeto; (ii) alcance: retorna os objetos que estão localizados em uma determinada área de alcance; e (iii) vizinho mais próximo: retorna uma lista de um ou mais objetos que estão próximos à posição de um objeto.

Grande parte dos modelos espaciais são modelos baseados em fatos (seção 2.1.3.1) que organizam sua informação contextual através da

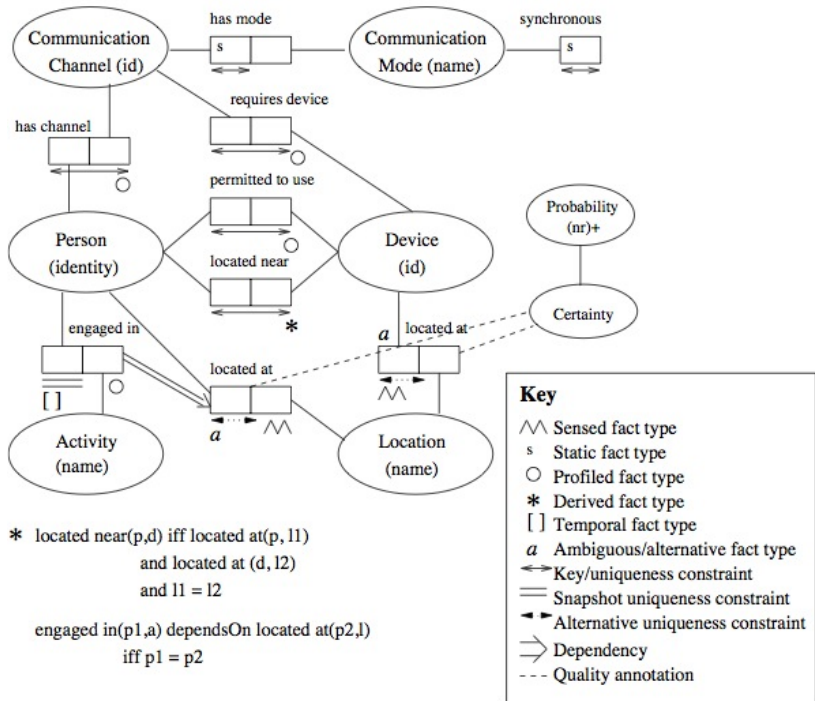


Figura 2.2: Exemplo da modelagem CML [Bettini et al., 2009]

localização física. Tal localização pode ser pré-definida (se as entidades são estáticas), ou pode ser obtida a partir de sistemas de posicionamento que localizam objetos móveis e reportam suas posições para um sistema de gerenciamento de localizações. Basicamente, duas categorias de sistemas de coordenadas são suportadas por tais sistemas de posicionamento:

- *Coordenadas geométricas*: representam pontos ou áreas em um espaço métrico, como as coordenadas de GPS (latitude, longitude e elevação acima do nível do mar). Ao utilizar funções geométricas, é possível a execução de consultas relacionadas a localização de objetos, como, distância entre objetos.
- *Coordenadas simbólicas*: são representadas por um identificador, como, por exemplo, o número de uma sala ou ID de um *access*

point. Em contraste com as coordenadas geométricas, coordenadas simbólicas não oferecem relações espaciais. A fim de permitir interpretação espacial, como, distância entre objetos, deve-se prover informações explícitas sobre as relações espaciais entre pares de coordenadas simbólicas.

A escolha do modelo de coordenadas a ser utilizado consiste em uma das principais considerações para modelos espaciais de contexto [Bettini et al., 2009]. Modelos geométricos e geográficos de localização oferecem um mapeamento simples entre dados e posições, enquanto modelos simbólicos e relacionais de localização são mais fáceis para a construção e representação de uma percepção simples do espaço (através de relacionamentos, como, *parte-de* e *localizado-próximo*). Tal escolha, também, determina como a informação contextual deve ser gerenciada (através de um bando de dados espacial, por exemplo), quais os métodos de raciocínio e interpretação são disponibilizados, assim como as classes de consultas que podem ser utilizadas.

Portanto, modelos espaciais de contexto são adequados para aplicações sensíveis ao contexto que priorizam a definição das situações de entidades a partir do posicionamento destas, utilizando-se um sistema de coordenadas (geométricas ou simbólicas). Porém, deve-se observar, cuidadosamente, as informações espaciais relevantes, a fim de priorizar a eficiência do processo de raciocínio e interpretação através da redução do tamanho da base de conhecimento [Nicklas et al., 2008, Becker and Nicklas, 2004].

2.1.3.3 Modelagem Baseada em Ontologia

Visto que contexto pode ser considerado um tipo específico de conhecimento, é natural investigar se algum *framework* conhecido para representação de conhecimento e raciocínio pode ser apropriado para a manipulação de contexto [Bettini et al., 2009]. O contraste entre a capacidade de expressão e complexidade de raciocínio tem motivado grande parte das pesquisas em modelos simbólicos de representação do conhecimento [Bader et al., 2003].

Segundo [Bettini et al., 2009], visto que ontologias consistem, essencialmente, em descrições de conceitos e seus relacionamentos, modelos contextuais baseados em ontologias exploram o poder de representação e raciocínio por diversos motivos: (a) a expressividade de linguagens, como OWL [Horrocks et al., 2003], é utilizada para descrever dados contextuais complexos que não podem ser representados, por exemplo, por linguagens mais simples como CC/PP [Klyne et al., 2004]; (b) possibilidade de compartilhar e/ou integrar informações contextuais entre diferen-

tes fontes, devido ao formalismo semântico fornecido aos dados contextuais; (c) as ferramentas de raciocínio disponíveis podem ser utilizadas, tanto para a verificação de consistência de um conjunto de relacionamentos que descrevem um cenário, quanto para reconhecer que um conjunto particular de instâncias de dados contextuais e seus relacionamentos revelam a presença de uma caracterização de contexto mais abstrata (por exemplo, a atividade do usuário pode ser automaticamente reconhecida).

Tipicamente, o formalismo escolhido para modelagem contextual baseada em ontologia é a OWL-DL [Horrocks et al., 2003] ou algumas de suas variações, visto que essa está se tornando um padrão entre diversos domínios de aplicações, além de ser suportada por diversos serviços de raciocínio [Bettini et al., 2009]. Através da OWL-DL é possível a modelagem de um domínio específico ao definir classes, indivíduos, características de indivíduos (*datatype properties*) e relações entre indivíduos (*object properties*). Descrições complexas de classes e propriedades podem ser construídas através da composição de descrições elementares utilizando-se os operadores fornecidos pela linguagem.

Além do poder representativo de ontologias, uma vantagem adicional consiste no suporte a procedimentos de raciocínio. Desta forma, através do conhecimento já representado, é possível:

- *Derivar novo conhecimento sobre o contexto atual automaticamente:* o raciocínio ontológico pode ser executado através da inferência de nova informação contextual baseada nas classes e propriedades definidas, e nos objetos provenientes de sensores e outras fontes de contexto. Por exemplo, é possível derivar o conjunto de objetos que são relacionados a um determinado objeto através de uma propriedade particular (tal como, o conjunto de atividades que ocorrem um lugar específico); e
- *Detectar possíveis inconsistências presentes na informação contextual:* crucial para a definição de uma ontologia, assim como para sua população através de novas instâncias. Assim, a verificação automática de consistência pode ser executada, a fim de capturar possíveis inconsistências na definição de classes e propriedades da ontologia (por exemplo, uma classe sendo subclasse de duas classes disjuntas), ou na sua população (por exemplo, uma pessoa estar presente em diferentes lugares simultaneamente).

Além de prover um formalismo para representação de dados contextuais complexos, ontologias, também, desempenham um importante papel no compartilhamento de conhecimento, visto que estas provêm uma

especificação formal da informação contextual. Tal característica é importante para ambientes móveis e pervasivos, pois diferentes entidades heterogêneas e distribuídas interagem, a fim de oferecer/trocar informação dos contextos dos usuários. Conseqüentemente, diversas ontologias descritas em OWL têm sido propostas para representar descrições de dados contextuais. Entre os esforços mais destacados, pode-se citar as ontologias SOUPA [Chen et al., 2004], proposta para modelagem de contexto em ambientes pervasivos, e CONON [Zhang et al., 2005], proposta para modelagem em ambientes de *smart homes*.

Entretanto, experiências com o desenvolvimento de ontologias voltadas para a modelagem contextual têm evidenciado que os operadores providos pela OWL-DL, usualmente, são inadequados para a definição de descrições complexas de contexto [Agostini et al., 2006]. Isto deve-se ao fato de que os construtores inclusos na linguagem OWL-DL foram selecionados a fim de garantir procedimentos de raciocínio. Portanto, OWL-DL não oferece construtores muito expressivos que seriam úteis para a modelagem de domínios complexos, como as atividades do usuário [Bettini et al., 2009].

2.1.3.4 Comparação entre as Abordagens

A fim de explicitar as diferenças entre os três modelos analisados para modelagem de contexto, a Figura 2.3 apresenta um comparativo entre os três modelos, considerando os requisitos identificados na seção 2.1.2.

	Modelagem baseada em fatos	Modelagem espacial	Modelagem baseada em ontologia
Heterogeneidade	+	~	+
Mobilidade	~	+	—
Relacionamentos	~	~	+
Histórico	+	+	—
Imperfeição	~	~	—
Raciocínio	~	—	+
Usabilidade	+	~	~
Eficiência	~	+	—

+ Atende Totalmente
 ~ Atende Parcialmente
 — Não Atende

Figura 2.3: Comparação entre as abordagens de modelagem de contexto

2.2 Computação Centrada no Usuário

Nos últimos anos, pode-se notar um progresso considerável no desenvolvimento de tecnologias de computação móvel [Sousa et al., 2008]. Por exemplo, grupos de pesquisa têm desenvolvido mecanismos para captura e utilização de contextos [Judd and Steenkiste, 2003, Becker and Dürr, 2005, Sheshagiri et al., 2004, Salber et al., 1999, Schilit et al., 1994], plataformas móveis [Augustin et al., 2006, Rhodes et al., 1999] e uma variedade de aplicações sensíveis ao contexto para usuários e domínios específicos, como, gerenciamento de tarefas computacionais [Sousa, 2005, Sousa and Garlan, 2002], guias turísticos [Davies et al., 1998, Yang et al., 1999], experimentos laboratoriais [Arnstein et al., 2001] e ambientes de *smart homes* [Abowd et al., 2002, Intille, 2002, Reyes Alamo and Wong, 2008]. A maioria desses esforços compartilham o mesmo paradigma: eles são orientados aos processos computacionais, visto que os usuários utilizam os computadores para resolverem seus problemas. Especificamente, tais trabalhos são centrados em plataformas computacionais (por exemplo, PDAs, laptops e telefones celulares) que auxiliam os participantes em tarefas específicas, ou na utilização e gerenciamento de tarefas computacionais de forma mais eficiente em diversos ambientes.

Entretanto, essa visão centrada nos processos computacionais tende a limitar o escopo do paradigma de computação móvel a situações em que a plataforma computacional é o objeto primário de atenção do usuário (acesso a informações, comunicação via e-mail, edição de documentos, por exemplo) [Sousa et al., 2008]. Desta forma, essa visão restringe a habilidade de abstração da plataforma computacional utilizada no desenvolvimento de sistemas. Tal abstração é um dos objetivos do paradigma de computação pervasiva, que consiste em, espontaneamente, prover recursos e serviços ao usuário de forma transparente em qualquer lugar e a qualquer momento, a fim de auxiliá-lo em suas atividades cotidianas [Zhang et al., 2009].

Visto que a atividade do usuário consiste na unidade computacional básica do paradigma de computação centrada no usuário, e não mais os arquivos e aplicações deste, novos desafios são inseridos no projeto e desenvolvimento de arquiteturas que adotam tal perspectiva [Bardram, 2005]. Pode-se destacar, como principais desafios, a definição de modelos e mecanismos para o reconhecimento das atividades do usuário, para que o sistema possa inferir sobre o comportamento deste usuário e, por conseguinte, analisar suas intenções, a fim de atuar de forma pró-ativa e colaborativa no auxílio de tais atividades [Sousa et al., 2008]. Portanto, a modelagem comportamental é altamente dependente do processo de detecção de atividades, pois este é considerado uma etapa de pré-processamento

que visa a obtenção e extração de dados e características para uma modelagem subsequente [Atallah and Yang, 2009]. Desta forma, as seções seguintes apresentam abordagens relativas ao reconhecimento de atividades e modelagem comportamental de usuários em ambientes de computação móvel e pervasiva.

2.2.1 Reconhecimento de Atividades

Devido ao avanço das tecnologias de sensoriamento, o reconhecimento das atividades humanas baseado em dados provenientes de sensores tem, recentemente, motivado diversas pesquisas por parte da comunidade de computação pervasiva [Gu et al., 2009]. Tipicamente, esses sensores podem estar presentes em objetos carregados pelo usuário ou, então, incorporados ao ambiente que o usuário integra [Atallah and Yang, 2009]. O objetivo do reconhecimento de atividades reside, portanto, em reconhecer atividades humanas comuns em situações reais, interpretadas a partir dos dados coletados por sensores.

O reconhecimento preciso de atividades, entretanto, é desafiador, pois as atividades humanas são complexas e altamente diversas [Kim et al., 2010]. Especificamente, o processo de reconhecimento de atividades impõe alguns desafios relativos à interpretação devido à natureza das atividades humanas, que podem se apresentar de forma sequencial, intervalada e concorrente [Gu et al., 2009].

Visto que os dados provenientes dos contextos de baixo nível são, usualmente, imprecisos e as atividades são, tipicamente, desempenhadas de forma não-determinística, métodos probabilísticos são apropriados para o processo de reconhecimento de atividades [Gu et al., 2009]. Portanto, diversos pesquisadores têm utilizado algoritmos baseados em métodos probabilísticos a fim de construir modelos de atividades [Kim et al., 2010].

Conforme [Kim et al., 2010], a compreensão das atividades humanas engloba tanto o reconhecimento de atividades, quanto a descoberta de padrões de atividades. O primeiro visa a precisão na detecção de atividades baseado em um modelo pré-definido de atividades. Por outro lado, a descoberta de padrões de atividades tem seu foco na busca por padrões desconhecidos, realizada diretamente sobre os dados provenientes de contextos de baixo nível sem qualquer modelo pré-definido.

Apesar de que essas duas técnicas diferem entre si, ambas visam o aperfeiçoamento da tecnologia de computação centrada no usuário. Portanto, tais técnicas são complementares, ou seja, a descoberta de padrões de atividades é capaz de auxiliar na definição de atividades que podem ser reconhecidas e manipuladas posteriormente. As seções seguintes apre-

sentam os principais mecanismos utilizados para o processo de reconhecimento de atividades.

2.2.1.1 Padrões Emergentes

Um padrão emergente (*Emerging Pattern* - EP) é um tipo de padrão de conhecimento que descreve mudanças significativas entre duas classes de dados [Dong and Li, 1999]. Assim, um EP é um conjunto de itens cuja frequência varia significativamente de um conjunto de dados para outro. Um conjunto de dados consiste em diversas instâncias, em que cada uma possui um conjunto de atributos e valores correspondentes. Entre os atributos disponíveis, alguns representam uma classe mais do que outros atributos. Por exemplo, um conjunto [localização@cozinha, objeto@forno] é um padrão emergente da atividade "cozinhar", e [objeto@pia, objeto@prato, localização@cozinha] é um EP da atividade "limpar pratos". Para encontrar tais atributos, deve-se determinar a representatividade e taxa de crescimento para cada atributo. Portanto, os padrões emergentes são conjuntos de dados com grandes taxas de crescimento de um conjunto de dados para outro.

Conforme [Gu et al., 2009], esses padrões emergentes podem ser analisados a partir de dados provenientes de sensores e, então, aplicados para reconhecimento de atividades mais complexas (intervaladas e concorrentes). Entretanto, tal abordagem não provê suporte à análise espacial das atividades, visto que os valores dos atributos são comparados pelos seus valores absolutos e não através de intervalos.

2.2.1.2 Séries Temporais

Uma forma alternativa para o reconhecimento de atividades consiste na classificação baseada em séries temporais (*Time Series*). Através deste método, uma atividade é modelada como uma sequência de eventos discretos [Hamid et al., 2009], [Peddemors et al., 2010].

As atividades são reconhecidas através da análise e descobrimento de subsequências que possuem comportamento similar e que são percebidas com determinada frequência em sequências temporais. Entretanto, tal abordagem é sensível à ordem dos eventos detectados.

2.2.2 Modelagem Comportamental

O estudo do comportamento humano têm sido o foco de muitas áreas de pesquisa. Por exemplo, comportamento é um assunto extensivamente investigado em molelagem de processos de negócio, modelagem cognitiva, inteligência artificial e psicologia educacional [Atallah and Yang, 2009]. Embora as pesquisas nessas áreas tenham produzido modelos interessantes de comportamento, a validação desses modelos, com

dados obtidos a partir de assuntos reais durante a realização de atividades normais, tem sido o maior obstáculo para a utilização de tais modelos em ocasiões cotidianas. Além disso, o comportamento de uma entidade é altamente dependente de aspectos, como, percepção, contexto, ambiente, conhecimento prévio e interação com outras entidade [Atallah and Yang, 2009]. Desta forma, é necessário considerar o cenário completo das ações, interações e ambientes que cercam o usuário para modelar o seu comportamento.

O desafio agora, portanto, não reside mais na obtenção de dados, mas em como utilizar o grande volume de dados provenientes de diferentes fontes e no reconhecimento de padrões que possam auxiliar na melhor compreensão do comportamento humano [Atallah and Yang, 2009]. Conseqüentemente, o desenvolvimento de modelos realistas do comportamento humano pode ser empregado para a previsão comportamental, assim como para a determinação de anomalias em grupos de comportamentos [Peddemors et al., 2010]. Desta forma, as seções seguintes apresentam as abordagens mais utilizadas para a modelagem do comportamento.

2.2.2.1 Modelos Probabilísticos

Os modelos probabilísticos surgiram como meios eficientes de representação de variáveis aleatórias, dependências e variação temporal, assim, tornando-os em artefatos adequados para a modelagem comportamental. Esta seção apresenta os modelos probabilísticos que são utilizados para a modelagem comportamental, como, os modelos ocultos de Markov (*Hidden Markov Models* - HMMs), campos aleatórios condicionais (*Conditional Random Fields* - CRFs), redes Bayesianas, redes Bayesianas dinâmicas (*Dynamic Bayesian Networks* - DBNs) e detecção de *clusters*.

2.2.2.2 Modelo Oculto de Markov

O modelo oculto de Markov (*Hidden Markov Model* - HMM) consiste em um modelo estatístico no qual o sistema a ser modelado é dito ser um modelo Markoviano. Um HMM representa, então, um conjunto finito de estados ocultos em que cada um destes é associado com uma distribuição probabilística, onde as transações entre os estados são controladas por um conjunto de probabilidades e, assim, uma observação pode ser gerada para cada estado. Especificamente, um HMM é um modelo probabilístico que é utilizado para gerar estados ocultos a partir dos dados observados [Sutton and McCallum, 2006]. O objetivo principal, ao se utilizar tal modelo, consiste em determinar a sequência de estados ocultos (y_1, y_2, \dots, y_n), que corresponde à sequência de saídas observadas (x_1, x_2, \dots, x_n). A Figura 2.4 apresenta a representação gráfica de um HMM composto por

cinco estados ocultos e quatro variáveis observadas.

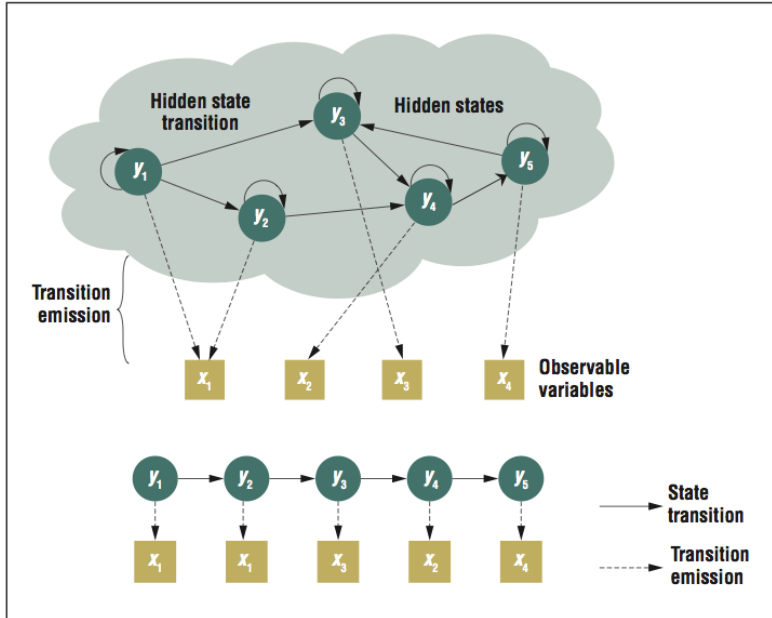


Figura 2.4: Representação gráfica de um modelo oculto de Markov (HMM) [Kim et al., 2010]

Geralmente, a utilização de HMMs para o reconhecimento de atividades consiste em treiná-los, utilizando-se classes pré-definidas. Assim, quando um HMM é utilizado para o reconhecimento de atividades, as atividades representam os estados ocultos e as saídas observadas são os dados provenientes dos sensores. A Figura 2.5 demonstra um HMM para o exemplo da atividade "comer".

Apesar de sua simplicidade e popularidade, o HMM apresenta sérias limitações devido a sua dificuldade em representar múltiplas atividade interativas (concorrentes ou intervaladas) [Gu et al., 2009]. Ainda, um problema nítido presente na utilização de HMMs para reconhecimento de atividades deve-se ao fato de que todas as observações são independentes, assim, tendências de longo prazo e relacionamentos complexos podem, possivelmente, serem perdidos. Tal restrição deve-se ao fato de que um estado futuro depende, somente, do estado atual, desconsiderando os es-

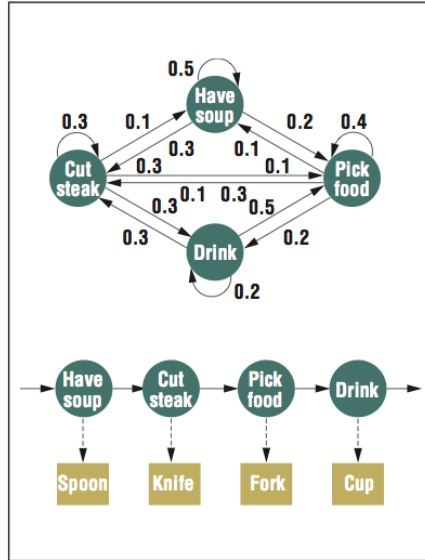


Figura 2.5: Exemplo de um HMM para a atividade "comer"[Kim et al., 2010]

tados passados [Sutton and McCallum, 2006]. Logo, a probabilidade do estado de um evento que é observado durante um certo intervalo de tempo é reduzida exponencialmente com o tamanho deste intervalo. Além disso, a probabilidade de ocorrer uma mudança no estado oculto não depende do tempo decorrido desde a entrada no estado corrente. Tal parâmetro pode ser importante para a modelagem de atividades humanas, pois, as pessoas são criaturas de hábitos e, portanto, relações temporais existem e são importantes para a determinação de eventos sucessivos [Peddemors et al., 2010].

2.2.2.3 Campo Aleatório Condicional

Na prática, muitas atividades e comportamentos podem possuir uma natureza não-determinística em que alguns passos podem ser realizados em qualquer ordem. Um campo aleatório condicional (*Conditional Random Field* - CRF) é uma alternativa mais flexível ao HMM, pois visa estas características comumente encontradas no comportamento humano. Neste contexto, CRF é um modelo gráfico não-direcionado que representa uma probabilidade condicional de uma determinada sequência de eventos,

Y , dado uma sequência de observações X .

CRFs, ilustrados na Figura 2.6, eliminam a independência entre observações que existe quando utilizam-se modelos ocultos de Markov. Campos aleatórios condicionais têm sido comparados aos HMMs quanto a sua utilização para reconhecimento de gestos, atividades, e, consequentemente, comportamentos. Em geral, CRFs oferecem uma melhor classificação para todas essas aplicações [Vail et al., 2007].

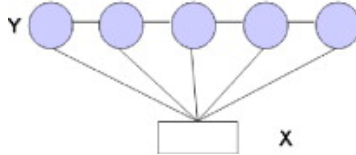


Figura 2.6: Um exemplo de estrutura de um campo aleatório condicional com o relacionamento entre a sequência de eventos (Y) e as observações X

2.2.2.4 Rede Bayesiana

Uma rede Bayesiana consiste em uma técnica de inteligência artificial que representa o conhecimento dentro de um contexto de incerteza, por meio de grafos [Russell and Norvig, 2004]. Segundo [Atallah and Yang, 2009], as redes Bayesianas são esquemas de representação de conhecimentos para desenvolver uma base de conhecimento. A base de conhecimento possui fatos e regras, associados a incertezas que representam o conhecimento do especialista do domínio da aplicação, e estes, por sua vez, explicitam as chances de ocorrência por meio de valores probabilísticos. A partir de dados de entrada e (*input*), o sistema associa as probabilidades com o conjunto de hipóteses H (*output*). A hipótese com maior probabilidade de ocorrência pode ser considerada a conclusão. Para determinar tais probabilidades, é aplicado o teorema de Bayes. O teorema de Bayes é apresentado a seguir:

$$P(H_i|e) = \frac{P(e|H_i) \cdot P(H_i)}{P(e)} \quad (2.1)$$

2.2.2.5 Rede Bayesiana Dinâmica

Em princípio, tanto HMMs e CRFs podem ser interpretados como exemplos de redes Bayesianas dinâmicas. Tais redes são redes Bayesianas que representam uma sequência de variáveis que sofrem alterações

sobre o tempo [Atallah and Yang, 2009]. Desta forma, redes Bayesianas dinâmicas apresentam um *framework* mais geral para o monitoramento de comportamento [Du et al., 2006], [Oliver and Horvitz, 2005].

Além disso, as redes Bayesianas dinâmicas provêm uma ferramenta genérica para manipular tanto dados imprecisos, quanto dados incompletos. Tais redes, ainda, são modulares, capazes de serem treinadas e oferecem a possibilidade de combinação de conhecimento prévio sobre um determinado domínio com dados de exemplo [Atallah and Yang, 2009].

Existem diversos trabalhos que aplicam as redes Bayesianas dinâmicas para modelagem comportamental. Em [Oliver and Horvitz, 2005], por exemplo, é proposto um modelo de inferência do comportamento do usuário e predição de passos futuros através de informações de contextos de baixo nível (coordenadas de GPS).

2.2.2.6 Clusters, Variabilidade e Anomalias Comportamentais

Devido ao grande volume de dados adquiridos a partir de contextos de baixo nível sobre um longo período de tempo, torna-se essencial o desenvolvimento de métodos para detectar similaridades em padrões comportamentais ou desvios a partir de comportamentos padrões [Atallah and Yang, 2009]. Ao utilizar-se de uma extensão da abordagem de conglomeração (*clustering*) para estudos de variabilidade ou anomalia é natural, pois as distâncias para o centro dos conglomerados (*clusters*) podem indicar similaridades entre diferentes assuntos ou entre padrões de comportamento de uma pessoa em diferentes instâncias de tempo [Cook, 2007].

O processo de detecção de anomalias pode ser considerado como uma divergência de um modelo de comportamento instruído (treinado) e classes de distâncias podem ser utilizadas a fim de indicar o nível de anomalia. Apesar de serem úteis para diversas aplicações, abordagens baseadas em modelos (CRFs e redes Bayesianas dinâmicas, por exemplo) podem apresentar dificuldades quanto às generalizações. Tal problema pode ocorrer sempre que um número de atividades incomuns ultrapassa o número de atividades "normais", ou, ainda, a intervenção humana na identificação de atividades e comportamentos (modelos manuais ou semi-automáticos) pode inserir uma tendência ao modelo, deixando de identificar determinadas variações e anomalias [Atallah and Yang, 2009].

A fim de contornar tais restrições, muitos grupos de pesquisa têm procurado analisar atividades e comportamentos sem especificamente ter que identificá-los ou rotulá-los. Por exemplo, sequências de localizações foram utilizadas em [Atallah et al., 2007] para aglomerar comportamen-

tos e detectar variabilidades, utilizando um *framework* de conglomeração baseado em HMM. Sequências de localizações, também, foram utilizadas em [Pentland, 2007], onde entropia foi selecionada como uma estrutura de quantificação do comportamento de uma pessoa. Estruturas em comportamento humano foram identificadas em [Eagle and Pentland, 2006] através da utilização de telefones móveis para fornecimento de informações sobre localizações, proximidade de outras pessoas e comunicações para 100 assuntos durante 9 meses. A estrutura do comportamento de uma pessoa foi representada pelos componentes principais de um conjunto completo de dados comportamentais. Desta forma, o comportamento de um indivíduo durante um dia pode ser eficientemente usado a fim de prever comportamentos, categorizar pessoas e analisar afiliações de grupos. Um modelo de influência foi utilizado para analisar os dados relativos a proximidade a partir desse conjunto completo de dados, que foi capaz de prever amizades e afiliações de grupos de trabalho com grande precisão [Eagle and Pentland, 2006].

2.2.3 Considerações sobre as Abordagens

Conforme as seções anteriores, existem diversas abordagens propostas para reconhecimento de atividades a fim de, posteriormente, determinar o comportamento das entidades participantes do ambiente móvel e pervasivo. Entretanto, nota-se que propriedades fundamentais deste ambiente, como espaço e tempo, não são consideradas simultaneamente, prejudicando uma análise mais completa do cenário de tais entidades. Essas propriedades são responsáveis por caracterizar os hábitos dos usuários, pois, segundo [Peddemors et al., 2010], correlações espaço-temporais são importantes para a determinação de eventos sucessivos. Consequentemente, tais correlações podem indicar o perfil comportamental do usuário.

Em adição, faz-se necessária a capacidade do modelo comportamental em agregar as habilidades e conhecimento adquiridos pelo usuário. Neste contexto, uma abordagem que utilize um modelo probabilístico que adota uma visão baseada em dados históricos, com a possibilidade de inserção de novas variáveis comparativas, pode representar uma alternativa interessante para parametrizar o comportamento do usuário.

Capítulo 3

Autenticação Sensível ao Contexto

O emergente paradigma de computação móvel torna viável o acesso a recursos de qualquer lugar e em qualquer momento. Porém, ao mesmo tempo que esse acesso ubíquo proporciona os seus benefícios, ele cria desafios particulares para prover segurança às entidades participantes. Tais desafios não são tratados de forma apropriada por abordagens tradicionais de segurança [Johnson, 2009].

Especificamente, para prover confidencialidade às informações e recursos do usuário, métodos tradicionais de autenticação necessitam de muita atenção e interação do usuário através de operações *login* e *logout* de forma explícita. Desta forma, os usuários devem lembrar suas senhas, localizar e escolher as informações e recursos desejados [Hung et al., 2008]. Além disso, sistemas convencionais de segurança, geralmente, operam sobre um conjunto fixo de regras e não necessitam identificar e responder a mudanças no ambiente. Em outras palavras, tais sistemas de segurança possuem um caráter estático [Babu and Venkataram, 2009], [Johnson, 2009]. Portanto, mecanismos tradicionais de autenticação são ineficazes para satisfazer as necessidades de ambientes altamente dinâmico como os ambientes móveis e pervasivos [Malek et al., 2008]. Ainda, segundo Babu and Venkataram [Babu and Venkataram, 2009], a eficácia da maioria dos mecanismos de autenticação para computação móvel depende da força dos identificadores utilizados para a autenticação dos usuários.

Consequentemente, autenticação sensível ao contexto, que utiliza a mudança de contexto para permitir a adaptação dos mecanismos de segurança baseada na situação atual, é essencial para prover segurança de forma efetiva para tais ambientes [Johnson, 2009]. Entretanto, conforme Johnson [Johnson, 2009], muitos dos sistemas de segurança sensíveis ao contexto atuais apresentam uma noção vaga sobre contexto ou utilizam contextos limitados. Geralmente, tais sistemas utilizam somente aspectos tradicionais, como, por exemplo, a localização do usuário. Assim, eles provêm uma visão abstrata e fraca de uma situação específica. Então, as

decisões adotadas por estes sistemas são ineficientes, pois são baseadas em um cenário incompleto.

3.1 Contexto Relevante à Segurança

Em [Johnson, 2009], é proposto o conceito de contexto relevante à segurança, que consiste em uma derivação entre uma definição amplamente aceita e utilizada de contexto [Dey, 2001] e a definição oficial de segurança da informação [seg, 2003].

Desta forma, segundo [Johnson, 2009]: "*Contexto relevante à segurança consiste em qualquer informação que pode ser utilizada para caracterizar a situação de uma entidade que poderia afetar uma tentativa do sistema em proteger informações e sistemas de informações contra acesso, uso, modificação ou destruição sem autorização, a fim de prover confidencialidade, integridade e disponibilidade*".

Consequentemente, um sistema de segurança sensível ao contexto deve basear-se em todas as informações contextuais relevantes que possam determinar ameaças ao uso apropriado do sistema, e, assim, adaptar-se a fim de tratá-las de forma adequada para impedir a intervenção humana de forma explícita.

3.2 Requisitos

No capítulo anterior foram identificadas as propriedades que devem ser satisfeitas por sistemas sensíveis ao contexto de forma geral. Entretanto, visto que o foco deste trabalho está no processo de autenticação sensível ao contexto, novos requisitos são inseridos a este conjunto de propriedades. Tais requisitos são:

- *Sensibilidade ao contexto*: o sistema de autenticação deve adaptar-se conforme o dinamismo da informação contextual. Tal sistema deve ser capaz de ajustar suas ações de acordo com as mudanças na situação das entidades relevantes à interação entre o usuário e o sistema de autenticação. Entre tais entidades, pode-se destacar: o usuário, os dispositivos computacionais do usuário, o ambiente e o mecanismo de comunicação entre o usuário e o sistema [Johnson, 2009].
- *Autonomicidade e dinamicidade*: o sistema de autenticação deve ser capaz de envolver o mínimo de intervenção humana possível. Além disso, o sistema deve ser capaz de observar, identificar e agregar o conhecimento e habilidades adquiridas pelo usuário durante suas interações com o sistema. Desta forma, tal sistema pode improvisar novas políticas baseadas em novas informações contextuais ou/e no histórico de informações.

- *Flexibilidade*: em um ambiente computacional aberto e amplamente distribuído como um sistema móvel e pervasivo, é desejável a utilização de diferentes meios de autenticação. Isto deve-se ao fato de que entidades possuem diferentes requisitos de segurança e políticas. Portanto, o sistema deve apresentar a habilidade de prover um nível satisfatório de customização para as diferentes entidades.
- *Eficiência computacional*: em ambientes sensíveis ao contexto, grande parte dos sensores possuem sua própria fonte energética (bateria), e possuem diversas restrições quando comparados a dispositivos computacionais mais robustos. Desta forma, o sistema de autenticação deve estar ciente de tais restrições de recursos, tanto computacionais, quanto energéticos, para prover o nível desejável de segurança. Esse nível de segurança pode ser determinado a partir de uma categorização das necessidades de segurança dos processos requisitados [Babu and Venkataram, 2009].

3.3 Abordagens Propostas

Esta seção tem como objetivo analisar algumas das abordagens mais relevantes relacionadas à autenticação sensível ao contexto. Por fim, é apresentado um comparativo entre as abordagens analisadas levando em consideração as propriedades identificadas na seção 3.2.

3.3.1 Abordagem de Babu e Venkataram

Na pesquisa [Babu and Venkataram, 2009], é apresentado um esquema de autenticação para transações móveis, chamado de TBAS (*Transaction-Based Authentication Scheme*). Tais esquemas de autenticação baseados em transações são soluções propostas que visam o processo de autenticação a nível de transação, ao invés de depender, apenas, da força dos identificadores durante o processo de autenticação.

Portanto, o objetivo principal deste esforço consiste em classificar as transações operadas pelo usuário ao nível de aplicação em ambientes de computação móvel. Através desta classificação, o sistema é capaz de inferir e analisar o comportamento do usuário através da abordagem de agentes cognitivos (agentes inteligentes). Além disso, através dessa categorização de transações, o sistema pode determinar o nível de segurança necessário, prevendo, então, o custo associado ao atraso do processo de autenticação devido à aplicação de algoritmos de criptografia.

O TBAS utiliza dois tipos de agentes cognitivos: agente cognitivo móvel (*Mobile Cognitive Agent - MCA*) e agente cognitivo estático (*Static Cognitive Agent - SCA*). Assim, o SCA cria o MCA e, então, envia este

agente cognitivo móvel para o dispositivo móvel. Este procedimento é realizado enquanto o cliente é autenticado.

O esquema completo de autenticação é distribuído em dois componentes lógicos: *MCA-component* e *SCA-component*. O MCA gera crenças a partir da observação dos diversos comportamentos do usuário, periodicamente. Por outro lado, o SCA, dinamicamente, cria requisitos de autenticação, utilizando a sensibilidade das transações móveis e das mudanças de comportamento do usuário. Além disso, um protocolo de desafios foi integrado ao sistema com a finalidade de neutralizar alguns ataques comumente encontrados neste tipo de ambiente, como: interrupção, modificação e fabricação de transações. A Figura 3.1 ilustra o processo de análise comportamental do usuário.

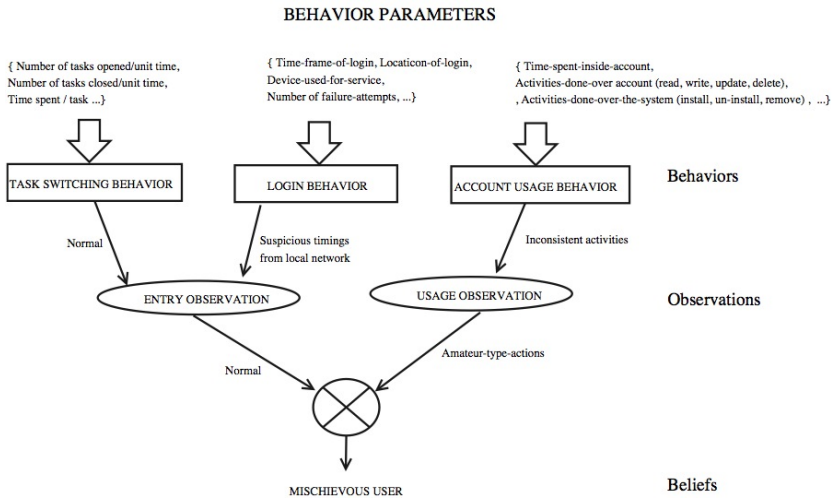


Figura 3.1: TBAS - Análise comportamental do usuário [Babu and Venkataram, 2009]

3.3.2 Abordagem de Hung et al.

O trabalho apresentado em [Hung et al., 2008] propõe um mecanismo de segurança baseado em atividades que visa auxiliar as atividades dos usuários em ambientes ubíquos. Tal mecanismo é composto por um sistema de autenticação baseado na identificação humana de imagens [Jameel et al., 2007] e por um modelo de controle de acesso orientado a

atividades.

O modelo proposto neste trabalho suporta diferentes tipos de dispositivos, incluindo dispositivos móveis (PDAs), laptops e computadores, por exemplo. Com esta finalidade, o gerenciador de reconhecimento de atividades (*Activity Recognition Manager - ARM*) provê informações sobre a atividade do usuário ao serviço de autorização através da coleta de dados contextuais de baixo nível relacionados a tal atividade e, então, produzindo informação contextual de alto nível. Desta forma, o ARM pode realizar o processo de raciocínio sobre as ações do usuário.

O gerenciador de autenticação é dito ser *lightweight*, pois deve ser capaz de ser executado a partir de qualquer dispositivo. Ainda, o gerenciador de autorização baseia-se na atividade atual do usuário a fim de estabelecer as permissões de acesso correspondentes. Então, os recursos consultados são enviados para o usuário. A Figura 3.2 ilustra a arquitetura proposta para o sistema de segurança baseado em atividades.

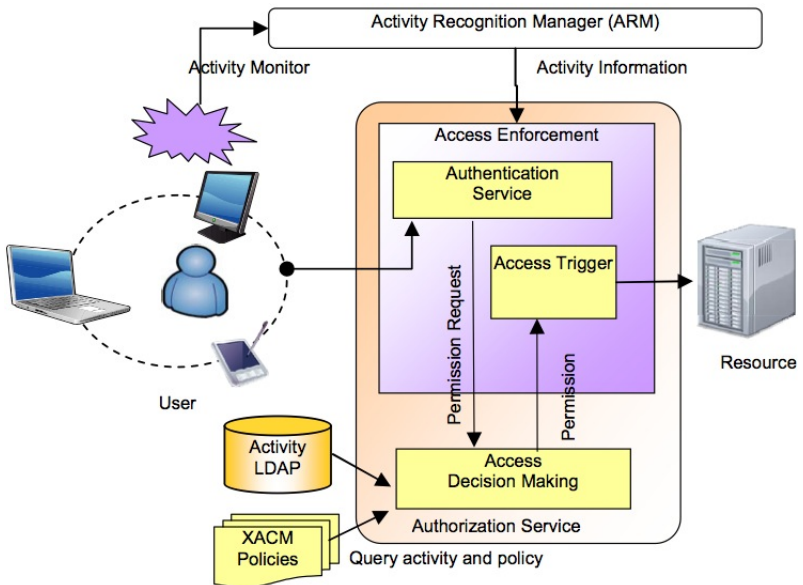


Figura 3.2: Arquitetura do sistema de segurança baseado em atividades [Hung et al., 2008]

3.3.3 Abordagem de Corradi, Montanari e Tibaldi

No trabalho apresentado em [Corradi et al., 2004], os autores propõem um *middleware* de segurança, chamado de UbiCOSM (*Ubiquitous Context-based Security Middleware*). Tal abordagem adota o contexto como conceito básico para especificação e execução de políticas de segurança. Portanto, as permissões são associadas diretamente aos contextos, ao invés das identidades e papéis dos usuários. As informações sobre os contextos e recursos são providas pelo *middleware* CARMEN [Bellavista et al., 2003].

O gerenciador de controle de acesso do UbiCOSM trabalha com duas classificações de contexto: contexto físico e lógico. Contextos físicos identificam espaços físicos delimitados por coordenadas geográficas específicas. Desta forma, um usuário opera em um determinado contexto físico dependendo da sua localização atual. Logo, o usuário só pode pertencer a somente um contexto físico. Além disso, os contextos físicos definem limites específicos para o gerenciador de controle de acesso, pois, cada contexto físico possui as referências para os recursos a serem protegidos.

Por outro lado, os contextos lógicos identificam estados lógicos, tanto dos contextos lógicos, quanto das entidades que compõem um cenário, como, por exemplo, usuários e recursos. Estados lógicos dependem de propriedades lógicas, como, condições temporais, estado e disponibilidade de recursos, atividades do usuário e características do dispositivo do usuário. Além disso, entidades podem estar associadas a diferentes contextos lógicos.

Além disso, o UbiCOSM opera sob um conjunto de 3 diferentes visões (*views*). Logo, ele controla a visibilidade dos recursos físicos/lógicos acessíveis diretamente através da localização do usuário (visões de contexto ativo - *active context views*). Portanto, tais visões contêm os recursos que, tanto o usuário deseja acessar (visão desejada - *desired view*), quanto os recursos que foram classificados como acessíveis pelo serviço de controle de acesso, considerando as políticas de controle de acesso dependentes do contexto ativo atual (visão permitida - *allowed view*).

As informações necessárias para a definição de tais visões são obtidas através de perfis que provêm descrições explícitas das características dos usuários, dispositivos e recursos. Então, os perfis são decompostos em duas subestruturas: propriedades do usuário e visão desejada, que expressa as preferências do usuário sobre as ações desejadas e visibilidade dos recursos.

A Figura 3.3 ilustra a arquitetura proposta para o sistema de segurança sensível ao contexto. O *Policy Manager* (PM) provê ferramentas

de edição de políticas de segurança e controle de acesso para os administradores do sistema e usuários finais. Por outro lado, o *Policy Installation Manager* (PIM) é responsável pela manutenção das associações entre contextos e suas permissões. Desta forma, o PIM instala as políticas de controle de acesso e segurança, armazenando as associações entre os contextos e permissões em tabelas hash. O *Context-Aware Security Manager* (CASM) é responsável por processar o conjunto de políticas de controle de acesso para os clientes móveis e, então, determina a visão de contexto ativo de qualquer usuário. Portanto, o CASM retorna ao usuário uma visão de contexto ativo baseada na situação do contexto ativo, nas políticas de segurança do sistema e nas políticas impostas pelos outros usuários que estão participando do domínio. Por fim, o *Authorization Enforcement Manager* (AEM) é responsável por proibir ou permitir o acesso dos clientes aos recursos, considerando as interações entre usuários e recursos anteriores.

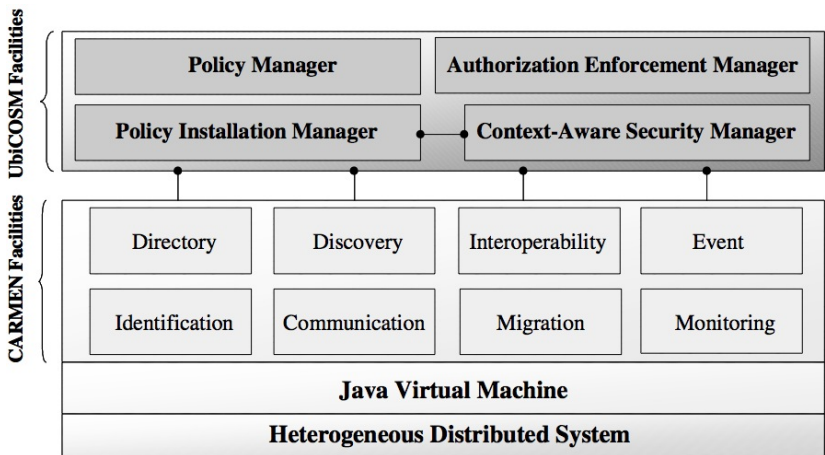


Figura 3.3: UbiCOSM - Arquitetura proposta [Corradi et al., 2004]

3.3.4 Considerações sobre as abordagens propostas

Como pode ser visto nas seções anteriores, todas as abordagens apresentam uma modelagem contextual fraca, pois, consideram apenas aspectos sobre as características dos dispositivos utilizados pelo usuário e seu contexto espacial. Desta forma, tais sistemas possuem uma visão

incompleta do cenário, prejudicando, assim, o processo de tomada de decisão.

Além disso, uma pequena parte das abordagens analisadas apresentam algum mecanismo de análise e modelagem comportamental do usuário de forma dinâmica. As propostas apresentadas em [Hung et al., 2008] e [Corradi et al., 2004] são estáticas, ou seja, não oferecem mecanismos para que o sistema possa, de forma dinâmica, agregar o conhecimento e habilidades adquiridas pelo usuário durante suas interações com o sistema. Especificamente, em [Hung et al., 2008], apesar de ser proposto um mecanismo de reconhecimento de atividades, o usuário deve, explicitamente, informar a atividade que está executando. Além disso, embora seja apresentada uma proposta de utilização de perfis para determinação das permissões dos usuários em [Corradi et al., 2004], o usuário deve, explicitamente, determinar que atividades pretende desempenhar no sistema. Consequentemente, essas duas propostas não estão em conformidade com o requisito de autenticidade e dinamicidade apresentado na seção 3.2.

Visto que o tempo de vida da bateria é a maior preocupação de usabilidade em dispositivos móveis [Rahmati and Zhong, 2009], é desejável que os mecanismos de autenticação levem em consideração aspectos relativos ao consumo de recursos computacionais de forma inteligente, quando estes executam seus procedimentos de segurança. Assim, a validação da proposta apresentada em [Babu and Venkataram, 2009] baseia-se na categorização das transações móveis, considerando o nível de segurança necessário para tais operações. Embora exista essa categorização de custos associados ao processo de autenticação, tal categorização é utilizada, apenas, para determinar o impacto do atraso associado ao algoritmo de criptografia aplicado e definir quais serão os métodos de autenticação (desafios) serão utilizados. Por outro lado, em [Hung et al., 2008], apesar dos autores citarem que a arquitetura proposta é leve, não são apresentados experimentos ou maiores detalhes sobre como tal arquitetura lida com as restrições energéticas dos dispositivos móveis.

A possibilidade de gerenciamento das políticas de segurança, por parte dos usuários, tem se tornado cada vez mais importante no projeto de soluções de segurança que buscam alinhar a usabilidade à capacidade de manter níveis aceitáveis de integridade, confiabilidade e disponibilidade em aplicações móveis [Toninelli et al., 2009]. Entretanto, conforme [Hong et al., 2007], segurança e usabilidade têm sido raramente integradas de forma satisfatória no projeto e desenvolvimento de sistemas móveis. Desta forma, buscou-se analisar as abordagens propostas quanto à privacidade e flexibilidade de meios e políticas de autenticação. Em [Babu

and Venkataram, 2009] e [Corradi et al., 2004], é apresentada mais de uma forma de autenticação. Especificamente, em [Babu and Venkataram, 2009], diferentes desafios são providos ao usuário a fim de comprovar sua identidade, dependendo do nível de segurança necessária para a transação requisitada e do nível de anomalia comportamental do usuário. Por outro lado, em [Corradi et al., 2004], o usuário define suas preferências através de perfis. Portanto, o sistema proposto utiliza esses perfis para o processo de autenticação e autorização do usuário.

Por outro lado, em [Hung et al., 2008], a arquitetura proposta provê apenas um único meio de autenticação ao usuário, que é o processo de identificação de imagens proposto anteriormente pelos autores em [Jameel et al., 2007]. A Tabela 3.1 apresenta as características dos trabalhos relacionados de forma resumida.

Tabela 3.1: Resumo das características dos trabalhos relacionados

Características	[Babu and Venkataram, 2009]	[Hung et al., 2008]	[Corradi et al., 2004]
Modelo Contextual	Espacial	Espacial	Espacial
Modelo Comportamental	Agentes cognitivos	Explícito	Perfil estático
Autonomia e Dinamicidade	Sim	Não	Não
Eficiência	Não	Sim	Não
Flexibilidade	Sim	Não	Sim

Capítulo 4

Arquitetura para Autenticação Sensível ao Contexto

Conforme apresentado no capítulo 3, os esquemas de autenticação propostos consideram, apenas, os aspectos relacionados ao contexto espacial, como localização e velocidade de mobilidade, e aspectos relacionados às propriedades dos dispositivos utilizados. Desta forma, há uma ausência de mecanismos que considerem os aspectos temporais e espaciais simultaneamente, a fim de fornecer um diagnóstico preciso baseado no histórico de decisões. Além disso, os modelos comportamentais propostos não consideram outras situações que estão implícitas na definição do comportamento.

Este capítulo tem como objetivo apresentar a solução desenvolvida para autenticação sensível ao contexto. Os aspectos relacionados à modelagem contextual são apresentados na seção 4.1. Na seção 4.2 são apresentadas as definições da modelagem comportamental proposta. Na seção 4.3 são mostrados os dois modelos propostos para o processo analítico através de uma filtragem híbrida. Por fim, na seção 4.4, é apresentada a arquitetura desenvolvida.

4.1 Modelagem Contextual

O espaço pervasivo existe em um ambiente em que situações ou contextos cercam o usuário. Em particular, tais contextos são relevantes para o processo adaptativo de informações e serviços que são oferecidos ao usuário, através de aplicações sensíveis ao contexto [Dey, 2001]. Entretanto, as situações que o usuário pode experimentar em um ambiente pervasivo são pessoais, dificultando, assim, a representação do contexto deste usuário e seus parâmetros.

A fim de identificar os contextos e propriedades relevantes para os usuários e seus comportamentos em ambientes móveis e pervasivos, analisou-se os recursos comumente providos por dispositivos móveis. Consequentemente, identificou-se os seguintes contextos relevantes à autenticação:

- *Contexto operacional*: descreve os objetivos, tarefas e atividades do usuário;
- *Contexto interpessoal*: descreve os aspectos sociais do usuário, como os relacionamentos e canais de comunicação entre o usuário e sua comunidade;
- *Contexto espacial*: considera os atributos relacionados com a localização do usuário; e
- *Contexto ambiental*: captura as situações que cercam o usuário, tais como os serviços, pessoas e informações acessadas pelo usuário.

A propriedade temporal foi integrada ao modelo contextual com o objetivo de aperfeiçoar o processo de tomada de decisão. Especificamente, dados históricos, como a capacidade de aprendizagem do usuário, incluindo a aquisição de habilidades e conhecimento, e a evolução das atividades e comportamentos foram considerados neste modelo. O modelo proposto é ilustrado na Figura 4.1.

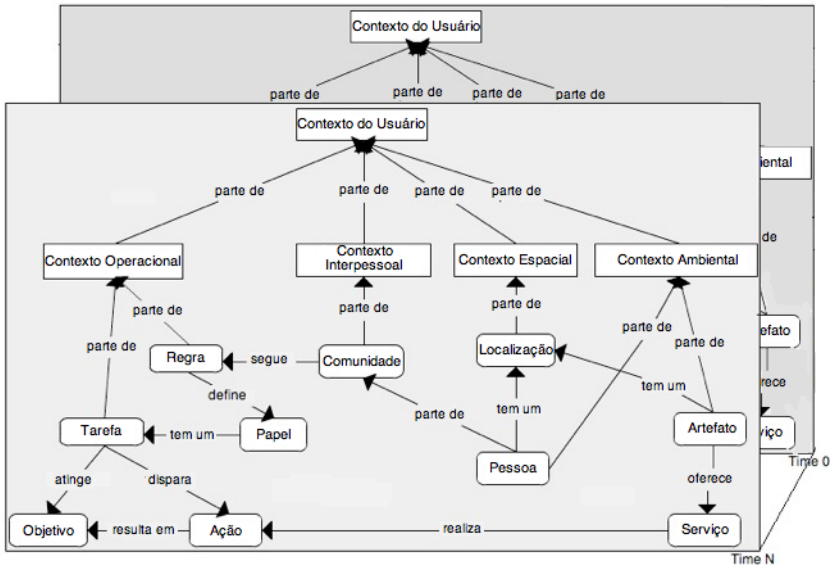


Figura 4.1: Modelo contextual

4.2 Modelagem Comportamental

Visto que os seres humanos são criaturas de hábitos, correlações temporais são importantes para a determinação de eventos sucessivos [Peddemors et al., 2010]. Deste modo, a predição de eventos define que ações e comportamentos o usuário irá adotar. A fim de, formalmente, definir os conceitos de evento e comportamento, este trabalho assume as seguintes definições:

- *Evento*: consiste na situação de uma entidade, em uma determinada localização e em um certo espaço de tempo, determinada por um ou mais contextos. Logo, um evento pode ser determinado, como:

$$E_i = \langle \text{situacao}, \text{tempo}, \text{localizacao} \rangle \quad (4.1)$$

- *Comportamento*: compreende o conjunto de eventos relacionados à execução de uma atividade. Consequentemente, pode ser definido como:

$$C_i = \langle \text{atividade}, \sum_{i=1}^n E_i \rangle \quad (4.2)$$

Portanto, tempo e localização desempenham um importante papel na modelagem comportamental de indivíduos em um ambiente móvel, pois, esses dois aspectos provêm artefatos para o diagnóstico de dados históricos, possibilitando a determinação dos hábitos de uma entidade.

4.3 Modelagem Analítica

O modelo proposto utiliza uma filtragem híbrida que explora a perspectiva espaço-temporal a fim de definir os conglomerados (*clusters*) de eventos e comportamentos do usuário e analisa o perfil do usuário a partir desses eventos e comportamentos. Tal filtragem, que é realizada para determinar a necessidade do processo de autenticação, é composta por dois modelos: modelo de permutação espaço-temporal e modelo de vetor de similaridade. Esses modelos são apresentados nas seções seguintes.

4.3.1 Modelo de Permutação Espaço-Temporal

Os eventos observados na execução de atividades formam uma base de dados para o processo de detecção de conglomerados de informações, que traduzem os hábitos dos usuários. Esses conglomerados podem ser classificados em três categorias:

1. *Puramente espaciais*: quando a ocorrência de eventos é mais alta em determinadas regiões do que em outras;
2. *Puramente temporais*: quando a ocorrência de eventos é mais alta em um período do que os demais; e
3. *Espaço-temporais*: quando a ocorrência dos eventos é temporariamente maior em determinadas regiões.

Dentre os modelos utilizados para predição de eventos em um contexto espaço-temporal, é proposta a utilização da permutação espaço-temporal que permite a incorporação de informação de co-variáveis, oriundas dos demais contextos do espaço pervasivo. Os modelos de Poisson (aplicado para contextos puramente temporais) e Bernoulli (aplicado para contextos preferencialmente espaciais) foram estudados, mas foram descartados devido à necessidade do projeto em determinar os hábitos comportamentais do usuário.

O modelo de permutação espaço-temporal, segundo Kulldorff [Kulldorff, 2006], é baseado em três características; i) a varredura para a detecção de conglomerados de dados é realizada no espaço e no tempo simultaneamente; ii) flexibilidade de trabalhar somente com eventos ou casos; iii) o modelo probabilístico sob hipótese nula, resulta que os casos seguem uma distribuição hipergeométrica¹.

Supondo a contagem de eventos e (em espaço temporal definido em t), localizados em uma região z de características circulares (definida por coordenadas de GPS), seja definido como e_{zt} . O número total de eventos observados E e o número total de eventos condicionados M_{zt} , são expressos pelas fórmulas:

$$E = \sum_z \sum_t e_{zt} \quad (4.3)$$

$$M_{zt} = \frac{1}{E} \left(\sum_z E_{zt} \right) \left(\sum_t E_{zt} \right) \quad (4.4)$$

Para realizar a predição de um evento, toma-se a seguinte pressuposição: a probabilidade condicional de um evento $P(E_a)$ na região z foi observado no tempo t_1 e no tempo t_2 , logo e_a tem uma média M_a e segue distribuição hipergeométrica dadas pela função:

¹Distribuição de probabilidade discreta que descreve a probabilidade de se retirar x elementos do tipo A numa sequência de n extrações de uma população finita de tamanho N , com K elementos do tipo A e $N-K$ elementos do tipo B, sem reposição.

$$M_a = \sum_{(z,t) \in A} M_{zt} \quad (4.5)$$

$$P(E_a) = \frac{\left(\frac{\sum_{t \in (t_1 \vee t_2)} \sum_{z \in A} E_{zt}}{E_a} \right) \left(\frac{E - \sum_{t \in (t_1 \vee t_2)} \sum_{z \in A} E_{zt}}{\sum_{t \in (t_1 \vee t_2)} \sum_{z \in A} E_{zt} - E_a} \right)}{\left(\frac{E}{\sum_{t \in (t_1 \vee t_2)} \sum_{z \in A} E_{zt}} \right)} \quad (4.6)$$

A fim de determinar as regiões dos conglomerados utiliza-se a ferramenta *SaTScan* [Kulldorff, 2006], desenvolvida por Kulldorff. A probabilidade condicional $P(E_a)$ do usuário possibilita estimar que tipo de atividade o usuário esteve executando e qual está executando atualmente. Assim, existem quatro casos que podem ocorrer: i) mesma atividade no mesmo contexto espaço-temporal, define-se como execução normal; ii) mesma atividade em contexto espaço-temporal diferente, define-se com execução suspeita; iii) atividades diferentes no mesmo contexto espaço-temporal, define-se como execução suspeita; iv) atividades diferentes em contexto espaço-temporal diferente, define-se como execução anormal.

4.3.2 Modelo de Vetor de Similaridade

As ações do usuário e o seu comportamento durante a realização de suas atividades definem um perfil que contém características que podem auxiliar a automatização do processo de autenticação do usuário. Inicialmente, o usuário deve, explicitamente, prover a definição do seu perfil. Entretanto, as novas informações adquiridas podem proporcionar uma expansão da base de conhecimento do usuário.

Por outro lado, o modelo de vetor de similaridade provê mecanismos que auxiliam na validação e definição de novos perfis quando ações e comportamentos são detectados. Os pesos associados aos atributos comparativos definem um vetor P , e, os novos eventos e comportamentos do usuário definem um segundo vetor E . Então, o valor do cosseno entre esses dois vetores é calculado através da seguinte função:

$$\text{Similaridade}(E, P_j) = \frac{E \times P_j}{\|E\| \times \|P_j\|} = \frac{\sum_{i=1}^n e_i \cdot p_{(j,i)}}{\sqrt{\sum_{i=1}^n e_i^2 \cdot \sum_{i=1}^n p_{(j,i)}^2}} \quad (4.7)$$

4.4 Arquitetura

A fim de monitorar o comportamento do usuário em diferentes situações e eventos onde o usuário está imerso no espaço pervasivo, a arquitetura proposta para autenticação sensível ao contexto busca utilizar

recursos que são comumente encontrados em dispositivos móveis, como telefones celulares e PDAs. A Figura 4.2 ilustra os componentes da arquitetura proposta. Os dispositivos móveis são considerados artefatos especiais que são utilizados frequentemente pelos usuários com a finalidade de alcançar seus objetivos em ambientes móveis [Uden, 2007]. Tais dispositivos oferecem recursos como:

- *Chamadas do usuário*: provêm informações considerando o contexto interpessoal, que envolve a comunidade na qual o usuário está inserido e o contexto ambiental, que diz respeito às pessoas que cercam o usuário;
- *Agenda do usuário*: é um dos recursos mais ricos em contexto, pois provê informações sobre as relações entre o usuário e os membros de sua comunidade. Ainda, este recurso pode determinar a localização do usuário, as pessoas que o cercam e as atividades que o usuário deseja executar em um determinado intervalo de tempo;
- *GPS*: provê informações relativas à situação espacial do usuário;
- *Nível de bateria do dispositivo*: pode indicar a forma de interação entre o usuário e o ambiente, assim como a intensidade dessa interação;
- *Aplicações do usuário*: provêm informações relacionadas ao contexto operacional e ambiental; em particular, tais aplicações indicam que artefatos o usuário utiliza para alcançar seus objetivos através das atividades desempenhadas;
- *Sensores*: podem prover informações sobre o ambiente, autenticação visual e outras informações que definem o ambiente onde o usuário está interagindo com o sistema de autenticação;
- *Perfil explícito*: é criado durante a primeira interação do sistema com o usuário através de uma interface interativa. Este perfil contém os eventos explicitados pelo usuário e extraídos de seus contatos e da agenda pessoal armazenadas no dispositivo móvel. Este perfil pode ser customizado e/ou sincronizado a qualquer momento;
- *Perfil implícito*: é criado através do processamento dos eventos do usuário e do seu perfil explícito, contém as informações relevantes sobre os eventos que ocorrem com maior frequência, quais as ações tomadas pelo usuário e as suas características espaço-temporais.

Este perfil é determinado por uma estratégia de recomendação baseada no modelo de espaço vetorial conforme definido em [Salton, 1989];

- *Perfil de sessão*: consiste no contexto de execução do usuário, ou seja, indica o estado do usuário, que é determinado pelas últimas ações realizadas por este. Na primeira interação do usuário com o sistema, assume-se o perfil de sessão sendo o seu perfil explícito; e
- *Filtro VSM (Modelo de Espaço Vetorial)*: é um filtro que utiliza o modelo de espaço vetorial para calcular a relevância da informação, utiliza um tratamento formal através de vetores para o cálculo de similaridade entre os perfis em análise.

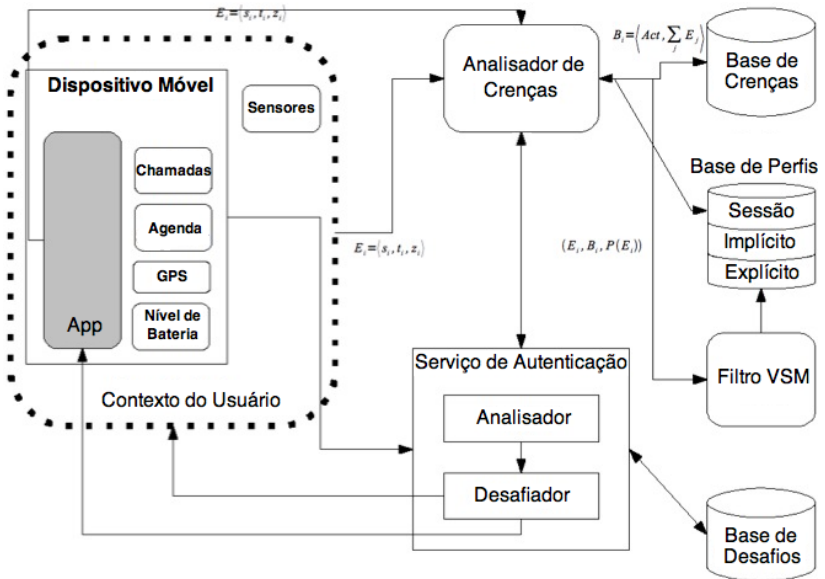


Figura 4.2: Arquitetura proposta [Rocha et al., 2010a]

Portanto, o subsistema de contexto, ou contexto do usuário, é responsável por capturar todas as situações que determinam a ocorrência de um novo evento através dos recursos descritos anteriormente. Assim, este subsistema envia a descrição do evento (e_i) para o subsistema de análise de crenças ou Analisador de Crenças.

O Analisador de Crenças é responsável pela definição de comportamentos, ou crenças, assim como pela classificação de eventos e inferência de comportamentos através das atividades, dos perfis armazenados e eventos que são percebidos e registrados. Esses comportamentos são analisados probabilisticamente para definir novas ocorrências e determinar a atitude a ser adotada pelo sistema, assim como as ações que serão tomadas em uma nova ocorrência. Portanto, a base de dados de crenças trabalha como um repositório de conhecimento (base de dados de crenças e base de dados de perfis).

O Filtro VSM tem como objetivo determinar os novos perfis implícitos através do modelo de vetor de similaridade. A cada combinação de evento com o perfil de sessão é determinado um novo vetor ortogonal. Este vetor é utilizado para o cálculo da similaridade, comparando se o grau de similaridade for superior a um determinado valor, o perfil é relevante e será armazenado no sistema como novo perfil de sessão e o evento será inserido ao perfil implícito do usuário. O funcionamento do Analisador de Crenças é descrito no Algoritmo 4.1.

Algorithm 4.1 Funcionamento do Analisador de Crenças

Início

Buscar na *Base de Perfis* os eventos passados associados ao usuário que gerou o evento E

Extrair as coordenadas geográficas dos eventos

ExecutaPermutacaoEspacoTemporal()

$PValue \leftarrow \min(AnalisaResultadoPermutacao())$

$VetorE_{local} \leftarrow VetorE_{timestamp} \leftarrow PValue$

Definir $VetorE$ baseiando-se no vetor de pesos ($VetorP$)

$GrauSimilaridade \leftarrow CalculaSimilaridade(VetorE, VetorP)$

Fim

O subsistema que analisa probabilidades, conhecido como Analisador de Probabilidades, é responsável pela categorização do usuário baseada nas probabilidades condicionais do seu comportamento. Essa classificação é dividida em três categorias: normal, suspeito e anormal. O funcionamento do Analisador de Probabilidades é mostrado no Algoritmo 4.2.

Finalmente, o subsistema de desafios, ou Desafiador, determina como o usuário será questionado a fim de provar sua identidade no sistema, baseado na categorização feita pelo Analisador de Probabilidades e no nível de autenticação necessária para a operação desejada. A resposta ao desafio proposto ao usuário é, então, armazenada para consultas futuras

Algorithm 4.2 Funcionamento do Analisador de Probabilidades

```

Início
if NaturezaUsuario is USUARIO_NORMAL then
  Realizar a operação requerida pelo usuário
  Definir o evento E como Perfil de Sessao do usuário
  Inserir o evento E ao Perfil Implicito do usuário
else
  Desafiador(NaturezaUsuario, NívelRestriçãoAplicação)
end if
Fim
  
```

realizadas pelo Analisador de Crenças. O funcionamento do Desafiador é detalhado no Algoritmo 4.3.

Algorithm 4.3 Funcionamento do Desafiador

```

Início
if NaturezaUsuario is USUARIO_SUSPEITO then
  Pesquisar na Base de Desafios o desafio referente ao nível da autenticação exigida pela aplicação para comportamentos suspeitos
else if NaturezaUsuario is USUARIO_ANORMAL then
  Pesquisar na Base de Desafios o desafio referente ao nível da autenticação exigida pela aplicação para comportamentos anormais
end if
  Enviar desafio para o usuário
if Resposta do desafio está correta then
  Realizar a operação requerida pelo usuário
  Definir o evento E como Perfil de Sessao do usuário
  Inserir o evento E ao Perfil Implicito do usuário
else
  Fechar a aplicação
end if
Fim
  
```

4.5 Considerações sobre a Abordagem Proposta

A arquitetura proposta para autenticação sensível ao contexto visa lidar com os desafios impostos pela natureza altamente dinâmica dos ambientes de computação móvel. Através da modelagem contextual proposta, é possível tomar decisões baseadas em dados históricos através de uma perspectiva espaço-temporal. Tal análise é provida pelo modelo de

permutação espaço-temporal apresentado que, ainda, possibilita a incorporação de co-variáveis para uma avaliação mais refinada do cenário do usuário. O resultado de tal análise é, então, utilizado para determinar o grau de similaridade entre a atividade executada pelo usuário e o seu perfil comportamental, que é definido e modificado conforme o número de interações do usuário com o sistema. Haja vista que mecanismos de autenticação sensíveis ao contexto devem prover flexibilidade ao usuário, oferecendo diferentes formas de autenticação conforme as políticas de segurança das aplicações, neste trabalho é proposto um componente, chamado de Desafiador, que é responsável por propor diferentes desafios ao usuário, considerando os diferentes níveis de autenticação exigidos pelas diferentes categorias de aplicações.

Capítulo 5

Resultados Experimentais

Este capítulo apresenta o ambiente experimental desenvolvido para validar a abordagem proposta. O objetivo desse ambiente é possibilitar a autenticação do usuário, através da utilização da riqueza de contextos encontrada em ambientes de computação móvel e pervasiva. Especificamente, tais contextos são capturados através de dispositivos móveis, como, smartphones, pois, conforme o estudo apresentado nos capítulos 2 e 3, tais dispositivos são capazes de obter diversas informações ambientais e operacionais dos usuários. Desta forma, foi necessário identificar quais recursos destes dispositivos deveriam ser utilizados como sensores (fontes de informação contextual) para a implementação da abordagem.

A descrição do ambiente experimental, assim como as justificativas de algumas das opções de implementação adotadas neste ambiente, são discutidas na seção 5.1. Na seção 5.2, é apresentado o protótipo implementado. Na seção 5.3, é mostrada uma análise comparativa entre a abordagem proposta nesta dissertação e o tipo de abordagem proposta na maioria das abordagens analisadas.

5.1 Descrição do Ambiente Experimental

A configuração para o desenvolvimento da abordagem consiste em um cenário real de grade computacional móvel. Neste ambiente de grade computacional móvel, o usuário, utilizando um dispositivo móvel como interface, pode interagir com a grade computacional. Por sua vez, o ambiente de grade pode oferecer o compartilhamento de recursos em larga escala aos usuários móveis para processar as aplicações que ajudam a resolver problemas complexos, fornecendo assim maior flexibilidade, desempenho e confiabilidade para o usuário de dispositivos móveis [Black and Edgar, 2009]. Desta forma, os usuários destes aparelhos podem requisitar a execução de vários tipos de aplicações sem se preocupar com limitações de recursos impostas pelos mesmos, já que eles estão utilizando os recursos compartilhados na configuração da grade.

Sob essa ótica, um ambiente de grade computacional móvel con-

siste em um cenário típico de computação móvel, onde clientes em redes sem fio acessam o ambiente dentro de uma rede estruturada através de dispositivos móveis. Este ambiente foi projetado a fim de obter uma avaliação próxima da real da interação dos clientes, através de dispositivos móveis, com o servidor, que gerencia os usuários, recursos e aplicações executadas na grade computacional. O ambiente considerado para o desenvolvimento da abordagem é ilustrado na Figura 5.1.

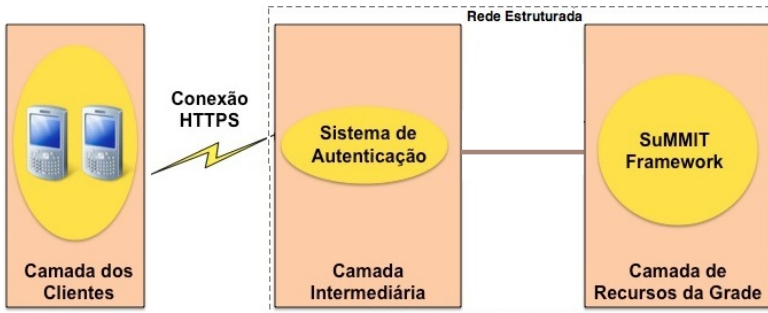


Figura 5.1: Cenário experimental

O servidor foi devidamente configurado com o software de análise estatística SaTScan versão 7.0 [Kulldorff, 2006] (este software é descrito detalhadamente na seção 5.2.2). A escolha foi realizada pelo fato deste ser um software que oferece uma boa interface para análises estatísticas, especialmente, para análise espaço-temporal, possibilitando a detecção de alertas de alterações de casos dentro de perspectivas espaço-temporais. Além disso, tal aplicação possui uma boa documentação.

Por outro lado, para a implementação do ambiente de grade, utilizou-se o *framework* SuMMIT (*Submission, Monitoring and Management of Interactions of Tasks*). Este *framework* foi inicialmente proposto pelo nosso grupo em [Rossetto et al., 2007], e foi, posteriormente, estendido em [Rocha et al., 2010b] e [Viera et al., 2010]. Desta forma, este *framework* possibilita a submissão e monitoramento de *workflows* submetidos, através de dispositivos móveis, à grade computacional para execução. Além disso, este mecanismo possibilita a reserva de recursos da grade computacional para execuções posteriores, assim como, permite o cancelamento de tais reservas. Optou-se pela utilização deste *framework*, pois este já possui diversas aplicações implementadas, em J2ME (*Java 2 Micro Edition*), para gerenciamento e monitoramento de recursos e aplicações

em ambientes de grade. Entretanto, tal ferramenta provê um processo de autenticação simples, ou seja, através de *login* e senha. Consequentemente, implementou-se a arquitetura de autenticação proposta nesta dissertação como uma camada intermediária entre as aplicações implementadas para os dispositivos móveis e o ambiente de grade computacional situado na rede estruturada. O *framework* SuMMIT é detalhadamente descrito no apêndice B.1.2.

Nesta dissertação foram empregadas tecnologias que obtêm mais portabilidade nos dispositivos móveis. Desta forma, os clientes móveis foram desenvolvidos usando a ferramenta J2ME Wireless Toolkit versão 2.5 [SunMicrosystems, 2010], por ser uma ferramenta que fornece facilidades como por exemplo, emuladores, documentação e portabilidade para um grande número de dispositivos móveis. Ainda, optou-se pela tecnologia J2ME para facilitar a integração entre as funcionalidades já implementadas pelo *framework* SuMMIT e as funcionalidades providas pelo mecanismo de autenticação sensível ao contexto. As características do ambiente J2ME são apresentadas no apêndice B.1.1.

A Tabela 5.1 apresenta as características de hardware e pacotes de software do servidor e dos clientes utilizados no protótipo implementado.

Tabela 5.1: Características de hardware e software do ambiente experimental

Nodos	Modelo	Clock	Memória	SO	Java
Celular	Nokia N95	332MHz	64 MB	Symbian 9.2	MIDP2.0
Celular	Nokia N95	332MHz	64 MB	Symbian 9.2	MIDP2.0
Servidor	Intel Core 2 Duo	2,26GHz	4 GB	Ubuntu 9.04	J2SE1.6.0

Conforme apresentado na Tabela 5.1, os dois celulares disponíveis para os experimentos implementam a especificação MIDP (*Mobile Information Device Profile*) 2.0. Portanto, tais equipamentos suportam o protocolo HTTPS (*HyperText Transfer Protocol Secure*), que consiste em uma implementação do protocolo HTTP (*HyperText Transfer Protocol*) sobre a camada SSL (*Secure Sockets Layer*), provendo, então, uma conexão segura, pois os dados trafegados pela rede são criptografados [Knudsen, 2002].

5.2 Protótipo Implementado

A presente seção tem como objetivo a discussão e apresentação das decisões tomadas durante o processo de implementação do protótipo para validação da arquitetura proposta. Na seção 5.2.1, é apresentada a forma utilizada para captura de eventos. Na seção 5.2.2, são apresentados os diferentes níveis de autenticação requeridos para o cenário ambiental proposto. Na seção 5.2.3 são apresentadas as características da ferramenta de análise estatística SaTScan.

5.2.1 Detecção de Eventos

A cada evento registrado é capturada a coordenada geográfica (latitude e longitude) do usuário através do GPS do dispositivo móvel. Para que a aplicação de monitoramento pudesse ter acesso aos eventos modelados, utilizou-se o mecanismo oferecido pela especificação MIDP 2.0, chamado de *Push Registry*. Tal mecanismo possibilita que uma aplicação seja inicializada/ativada automaticamente, sem a intervenção do usuário, ou seja, a aplicação é inicializada/ativada quando ocorrem eventos de rede (por exemplo, recebimento e envio de pacotes) e alarmes baseados em temporizadores. A Figura 5.2 abaixo ilustra a interação do mecanismo *Push Registry* com uma aplicação.

5.2.2 Níveis de Autenticação

A sensibilidade da informação e o tamanho dos segredos (senhas) são utilizados como fatores fundamentais na proposta de níveis de segurança para sistemas computacionais [Bertino et al., 1998], [Rosenthal and Fung, 1999]. Conforme [Babu and Venkataram, 2009], todas as operações executadas em um ambiente móvel não podem ser classificadas em uma única categoria. Portanto, a natureza das operações deve ser envolvida a fim de categorizar os níveis de autenticação.

Com o propósito de estar em conformidade com o requisito de flexibilidade, apresentado na seção 3.2, buscou-se identificar quais os diferentes níveis de autenticação são requeridos pelas diversas aplicações encontradas no ambiente de grade computacional móvel. Em outras palavras, identificou-se o nível de risco para a segurança (integridade, confidencialidade e disponibilidade) das informações do usuário que está utilizando a aplicação. Assim, definiu-se a seguinte categorização:

- *Alto*: representa um risco elevado às operações do usuário realizadas através da aplicação utilizada. Pode-se citar como exemplo o processo de cancelamento de reserva de recursos realizada anteriormente;

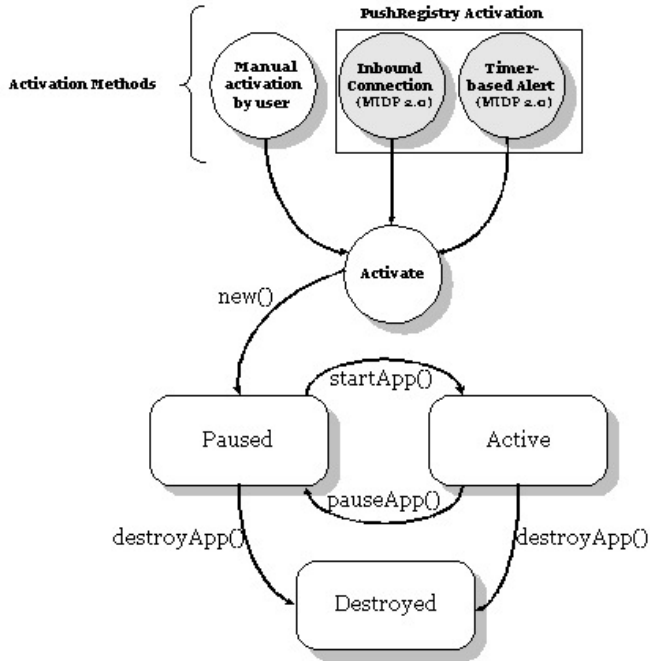


Figura 5.2: Funcionamento do mecanismo Push Registry [Knudsen, 2002]

- *Médio*: apresenta um risco moderado às operações realizadas pelo usuário. Exemplos são: submissão de tarefas para execução na grade computacional e realização de reserva de recursos; e
- *Baixo*: apresenta um risco mínimo às operações do usuário realizadas através da aplicação em questão. Pode-se citar como exemplos: monitoramento da execução de tarefas na grade computacional e monitoramento de reservas de recursos realizadas.

A partir desta categorização, escolheu-se quais serão os desafios propostos ao usuário quando houver a necessidade de um processo de autenticação. Consequentemente, estipulou-se quais os desafios seriam apresentados ao usuário para cada nível de autenticação identificado. Na Tabela 5.2, pode-se observar os desafios que foram cadastrados no sistema.

Tabela 5.2: Relação entre os níveis de autenticação, natureza do usuário e desafios propostos

Nível	Natureza do Usuário	Desafio
Alto	Suspeito Anormal	"Por favor, digite o seu CPF" "Por favor, digite seu login e senha"
Médio	Suspeito Anormal	"Por favor, digite a sua data de nascimento" "Por favor, digite o seu RG"
Baixo	Suspeito Anormal	"Por favor, digite/escolha o seu CEP" "Por favor, digite/escolha a sua cor favorita"

5.2.3 SaTScan

O SaTScan [Kulldorff, 2006] é um software livre que visa a análise de dados espaciais, temporais e espaço-temporais. Tal software foi desenvolvido para atender aos seguintes propósitos:

- Realizar o acompanhamento geográfico de doenças, detectar conglomerados (*clusters*) de doenças sobre uma perspectiva espacial ou espaço-temporal e verificar se tais conglomerados são estatisticamente significantes;
- Testar se uma doença é aleatoriamente distribuída sobre o espaço, tempo, ou espaço e tempo, simultaneamente;
- Avaliar a significância estatística de possíveis alarmes de conglomerados de doenças;
- Realizar o acompanhamento periódico de doenças a fim de detectar previamente surtos de doenças;
- Detecção de aglomerados dos alertas de desmatamento [Balieiro et al., 2009]; e
- Detecção de conglomerados de homicídios e o tráfico de drogas [Beato Filho et al., 2001];

Além disso, o software pode ser aplicado em problemas similares em outros campos de pesquisa, como: arqueologia, astronomia, criminologia, ecologia, economia, engenharia, genética, geografia, etc.

5.2.3.1 Tipos de Dados e Métodos

O SaTScan pode ser utilizado tanto para análise estatística discreta, quanto para análise estatística contínua. Para a primeira, o software utiliza um modelo discreto baseado em Poisson, quando o número de eventos em uma localização geográfica é uma distribuição de Poisson, conforme uma população conhecida em risco; um modelo baseado no modelo de Bernoulli, com dados de eventos como casos e controles; um modelo de permutação espaço-temporal, utilizando somente dados de casos; um modelo multinomial para dados categóricos; um modelo exponencial para dados temporais com ou sem co-variáveis; ou um modelo normal para outros tipos de dados contínuos. Uma característica comum entre todos esses modelos é que as localizações geográficas utilizadas, onde os dados podem ser observados, são não-aleatórias e fixadas pelo usuário do software.

Para a análise estatística discreta, os dados podem estar vinculados a diversos níveis geográficos, como, código postal, município ou setor censitário, ou tais dados podem possuir coordenadas únicas para cada observação. O SaTScan realiza o ajuste de qualquer número de co-variáveis categóricas providas pelo usuário, assim como para tendências temporais, conglomerados espaço-temporais conhecidos e falta de dados. Ainda, é possível analisar múltiplos conjuntos de dados simultaneamente a fim de procurar por aglomerados que ocorrem em um ou mais destes conjuntos de dados.

Por outro lado, para a análise estatística contínua, o SaTScan utiliza um modelo contínuo de Poisson. Nesse modelo, as localizações dos eventos observados são aleatórias dentro de uma área de estudo pré-definida pelo usuário.

5.2.3.2 Análise Estatística Espaço-Temporal

A análise estatística espaço-temporal é definida por uma janela cilíndrica com uma base geográfica circular (ou elíptica) e com uma altura correspondente ao tempo. A base é definida exatamente como na análise estatística puramente espacial, enquanto a altura reflete o período de tempo de conglomerados potenciais. A janela cilíndrica é, então, movida no espaço e tempo, de modo que para cada possível localização geográfica e tamanho, a janela também visita cada período de tempo possível. Na realidade, obtém-se um número infinito de cilindros sobrepostos de diferentes tamanhos e formas, ou seja, um conjunto que abrange toda a região de estudo, onde cada cilindro reflete um possível conglomerado.

A análise estatística espaço-temporal pode ser utilizada tanto para a análise retrospectiva, utilizando-se dados históricos, ou para o acompa-

nhamento prospectivo periódico, onde a análise é repetida, por exemplo, diariamente, semanalmente ou mensalmente.

5.2.3.3 Ambiente de Execução

O SaTScan oferece duas modalidades de execução: modo gráfico (através da interface gráfica do software) e modo *batch* (linha de comando). O aplicativo requer a especificação de parâmetros para definição de opções de análise, dados de entrada e resultados. Tais parâmetros são definidos através de arquivos de entrada e o resultado da análise é apresentada através de arquivos de resultados. A Figura 5.3 ilustra o ambiente de execução do SaTScan.

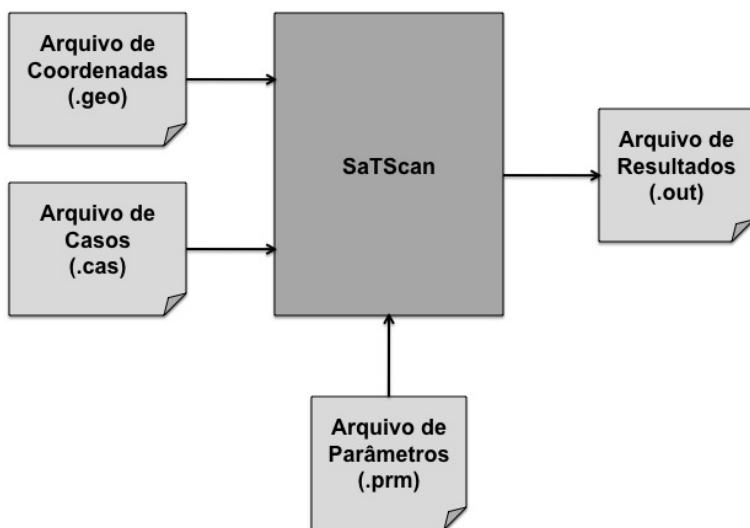


Figura 5.3: SaTScan - Arquitetura de execução

5.2.3.4 Dados de Entrada

- *Arquivo de casos*: provê informações sobre casos. Cada linha deste arquivo informa o identificador da localização do(s) caso(s) em questão, o número de casos ocorridos na localização especificada, o instante de tempo em que ocorreu e os atributos (co-variáveis) relacionados ao(s) caso(s).
- *Arquivo de coordenadas*: provê as coordenadas geográficas para

cada identificador de localização, onde cada linha deste arquivo representa uma localização geográfica. Ainda, informações baseadas em área podem ser agrupadas e, então, representadas por uma única localização geográfica. Além disso, coordenadas podem ser especificadas tanto pela utilização do sistema Cartesiano, quanto através da latitude e longitude.

- *Arquivo de parâmetros*: especifica a localização dos arquivos de casos e coordenadas e a localização onde será salvo o arquivo de resultados da análise. Ainda, este arquivo serve para a definição dos parâmetros necessários para definir o tipo de análise estatística desejada.

5.2.3.5 Resultados da Análise

Como saída, o SaTScan gera um arquivo texto padrão no formato ASCII e até cinco diferentes arquivos de saída opcionais no formato de colunas que podem ser gerados tanto no formato ASCII, quanto no formato dBase. Alguns destes arquivos opcionais, ainda, podem ser exportados para outros softwares, como, planilhas, bases de dados ou sistemas de informações geográficas.

- *Arquivo de resultados*: é gerado logo após os cálculos terem sido completados. Contém informações como:
 - Cluster mais provável
 - Raio: medida do raio do círculo, dado em quilômetros.
 - População: população média na área geográfica do cluster
 - Risco relativo: risco estimado em um cluster, dividido pelo risco estimado fora do cluster.
 - P-value: probabilidade de que a amostra podia ter sido tirada de uma população sendo testada, assumindo que a hipótese nula seja verdadeira.
 - Clusters secundários: informação resumida sobre outros clusters detectados nos dados

5.3 Análise Comparativa da Abordagem Proposta

Esta seção tem como objetivo apresentar os experimentos realizados a fim de validar a abordagem proposta para autenticação sensível ao contexto, considerando os requisitos identificados na seção 3.2. Na seção 5.3.1 é apresentado o experimento que considera a modelagem analítica

utilizando-se o modelo de permutação espaço-temporal. Os resultados que consideram questões de consumo de energia da abordagem proposta são apresentados na seção 5.3.2. Por fim, na seção 5.3.3 são apresentados os resultados experimentais considerando-se a dinamicidade e autonomia da arquitetura proposta, através de perfis comportamentais, que são analisados através do modelo de vetor de similaridade.

5.3.1 Modelagem Analítica

Para facilitar a compreensão dos experimentos realizados, é importante definir alguns conceitos que são utilizados como atributos comparativos. Tais conceitos são:

- *Teste estatístico*: consiste no conglomerado candidato a representar uma falha (anomalia) verdadeira de autenticação, e não apenas uma falha ocasional [Kulldorff, 2006];
- *Co-variável*: consiste em um atributo que é relevante ao modelo analítico. Neste experimento, utiliza-se como co-variáveis relevantes ao processo de autenticação: identificador do dispositivo móvel, identificador da aplicação e o nível de restrição da aplicação;
- *Análise retrospectiva*: tem como objetivo detectar conglomerados ativos até o período de tempo final baseada em conglomerados anteriores; e
- *Análise prospectiva*: tem como objetivo a detecção de conglomerados ativos até o final da simulação.

A fim de analisar e validar o modelo de permutação espaço-temporal, utilizou-se o mesmo conjunto de casos (eventos) como entrada para os outros três modelos analíticos a serem comparados com o modelo proposto nesta pesquisa. São eles:

- Modelo discreto de Poisson - puramente temporal;
- Modelo discreto de Poisson - puramente espacial; e
- Modelo discreto de Poisson - espaço-temporal.

Desta forma, a análise comparativa entre os diferentes modelos analíticos foi realizada a partir de um conjunto composto por 28 eventos provenientes do contexto do usuário. As ações do usuário que originaram os eventos capturados foram agrupadas, através do recurso de agrupamento do software SaTScan, em quatro regiões diferentes durante três

Tabela 5.3: Descrição dos grupos de localizações

Identificador	Latitude	Longitude
88036656	-27.594176	-48.522159
88015380	-27.591456	-48.545739
88040911	-27.600597	-48.518645
88040921	-27.600610	-48.522332

dias. A Tabela 5.3 apresenta detalhadamente a descrição desses quatro grupos de localizações identificadas pelo CEP da localização.

Conforme os resultados apresentados na Figura 5.4 e em [Rocha et al., 2010a], o modelo de permutação espaço-temporal provê uma maior eficiência em comparação com o modelo discreto de Poisson. Especificamente, as melhoras providas pela utilização do modelo de permutação espaço-temporal são: i) suporte à análise de co-variáveis, possibilitando, assim, a análise de outros atributos que também são importantes, além do tempo e localização; e ii) detecção de mais clusters (especificamente neste experimento, apenas o modelo de permutação espaço-temporal foi capaz de detectar o cluster 3).

5.3.2 Eficiência Computacional

A presente seção tem como objetivo apresentar e discutir os principais assuntos que implicam em um impacto mais elevado no consumo de bateria dos dispositivos. Portanto, fez-se necessária a comparação entre diferentes abordagens para escolher qual destas oferece um menor custo computacional (energético) e que mantém um nível aceitável da funcionalidade exigida. Na seção 5.3.2.1 são apresentadas as decisões tomadas considerando-se os métodos de localização estudados e, na seção 5.3.2.2, são apresentados os aspectos relativos à comunicação entre os clientes (smartphones) e o servidor.

5.3.2.1 Localização do Usuário

A determinação da localização do usuário desempenha um papel importante na abordagem proposta, visto que esta localização é característica fundamental dos modelos propostos para a modelagem contextual e comportamental. Entretanto, existem diversos métodos para a consulta da localização geográfica de entidades em tempo real. Ainda, a precisão de tal localização depende do método utilizado. Conforme [Sun Microsystems, 2004], os métodos de posicionamento mais utilizados são os seguintes:

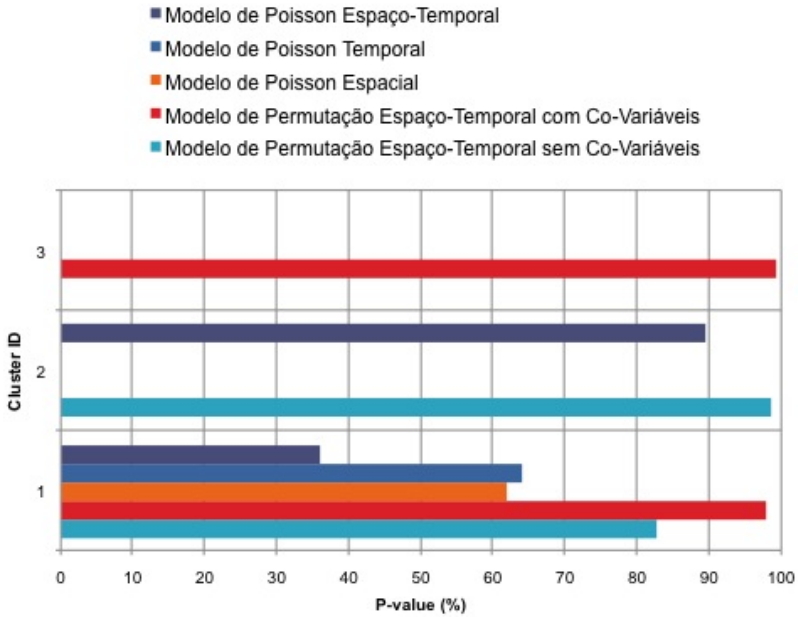


Figura 5.4: Comparativo entre os modelos analíticos

- Rede de telefonia celular:** o identificador da célula atual pode ser utilizado para identificar a estação base com que o dispositivo está se comunicando e a localização dessa estação. Desta forma, a precisão deste método depende do tamanho da célula, que pode ser bastante imprecisa. Por exemplo, uma célula GSM pode estar em qualquer lugar entre 2 a 20 quilômetros de diâmetro. Algumas técnicas que utilizam o identificador da célula podem alcançar uma precisão de até 150 metros.
- Satélites:** O sistema de posicionamento global (GPS), controlado pelo Departamento de Defesa dos Estados Unidos, utiliza um conjunto de 28 satélites, sendo 4 sobressalentes. O receptor capta os sinais de quatro satélites para determinar as suas próprias coordenadas, e ainda o tempo. Então, o receptor calcula a distância de cada um dos quatro satélites pelo intervalo de tempo entre o instante local e o instante em que os sinais foram enviados. Decodificando as

localizações dos satélites a partir dos sinais de microondas (tipo de onda electromagnética) e de uma base de dados interna, e sabendo a velocidade de propagação do sinal, o receptor, pode situar-se. O GPS é considerado o método de localização mais preciso (entre 4 e 40 metros em condições de céu claro). Porém, para prover todo esse processo complexo a fim de obter uma informação precisa, o consumo energético pode ser elevado e, também, a precisão e tempo de resposta dependem das condições de tempo do local.

- *Posicionamento de curto alcance:* Em áreas relativamente pequenas, como, por exemplo, um prédio, uma rede local pode prover informações relativas à localização, juntamente com outros serviços. Por exemplo, dispositivos equipados adequadamente podem utilizar *bluetooth* para o posicionamento de curto alcance.

Desta forma, analisaram-se as características do ambiente experimental (dispositivos móveis e infra-estrutura disponíveis) a fim de escolher qual seria o método de posicionamento a ser utilizado pelo protótipo. Visto que a abordagem de autenticação não deve restringir a mobilidade do usuário, descartou-se o método de posicionamento de curto alcance, pois este só é eficaz em uma pequena área de cobertura. Logo, fez-se necessária a comparação entre os métodos de posicionamento através de satélites e através da rede de telefonia celular.

A métrica escolhida para o processo comparativo foi analisada quanto ao consumo energético na determinação da posição (latitude e longitude) do usuário. Visto que o software SaTScan não considera a altitude em relação ao nível do mar, este atributo foi ignorado no momento da consulta do posicionamento.

O smartphone Nokia N95 oferece suporte à JSR-179, que consiste na API (*Application Programming Interface*) de localização para J2ME. Esta API trabalha com a maioria dos métodos de localização existentes de forma transparente ao usuário, utilizando tanto métodos baseados no dispositivo (GPS e *bluetooth*, por exemplo) quanto baseados na rede de telefonia celular (Cell-ID e triangulação de antenas, por exemplo). Além disso, tal API permite a configuração de alguns parâmetros, como, nível de consumo de potência necessário para realizar o processo de consulta de posicionamento (níveis baixo, médio e alto).

Logo, para a realização do comparativo, primeiramente, foi executada a consulta de localização utilizando-se o GPS oferecido pelo dispositivo. Após a execução da consulta com sucesso, desabilitou-se o GPS para a realização da consulta através da rede de telefonia celular.

Visto que cada fabricante provê a sua própria API para a consulta do nível energético da bateria de seus dispositivos, utilizou-se a aplicação *Nokia Energy Profiler* [Nokia, 2009] para determinar o consumo energético das operações de consulta de posicionamento. Como pode ser visualizado na Figura 5.5, a abordagem que utiliza satélites, através do mecanismo de GPS, apresenta um consumo energético inferior à abordagem que utiliza a rede de telefonia celular, através do identificador da célula e triangulação de antenas. Para este estudo comparativo, definiu-se o nível de consumo de potência como baixo.

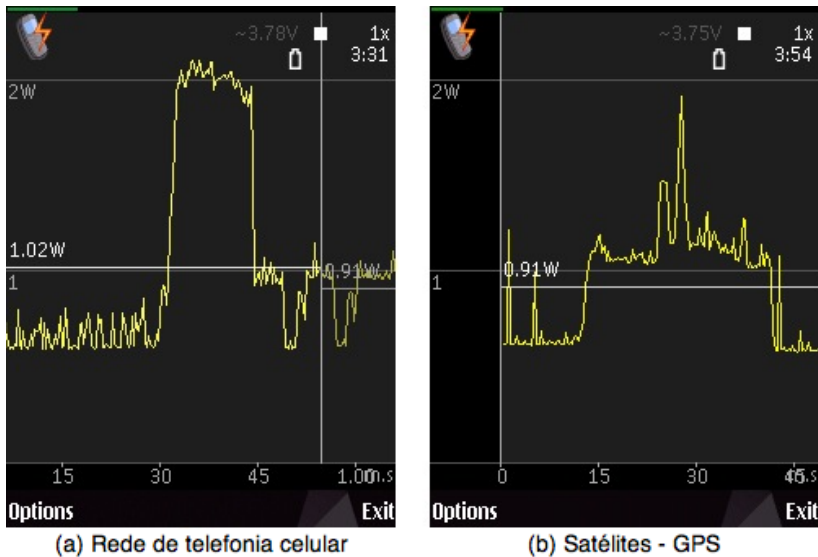


Figura 5.5: Comparativo de consumo energético entre os métodos de posicionamento

5.3.2.2 Intervalo de Comunicação entre Cliente e Servidor

Como mencionado, o módulo Analisador de Crenças fica aguardando por descrições de eventos ocorridos, que são capturados pelo contexto do usuário e enviados pelo dispositivo móvel através da rede sem fio ou rede da operadora de celular. Entretanto, o processo de comunicação através da rede consiste em um procedimento que demanda um custo energético elevado. Consequentemente, tal comunicação deve ser estabelecida de forma inteligente, ou seja, o número de conexões deve ser o

menor possível, mas sem comprometer a funcionalidade do sistema de autenticação sensível ao contexto, que é prover o nível de segurança exigido pela aplicação.

Desta forma, o número de descrições de eventos enviados a cada conexão com o servidor é dinâmico, variando de acordo com alguns parâmetros:

- *Restrição de segurança da aplicação*: conforme apresentado na seção 5.2.2, consiste no nível de risco para a segurança das informações do usuário que está utilizando a aplicação. Assim, definiu-se as categorias alta, média e baixa.
- *Tempo de vida da bateria*: considera o nível de bateria restante no dispositivo móvel. Assumiu-se como possíveis valores:
 - alto: mais de 70% de tempo de vida da bateria;
 - médio: de 30% a 70% de tempo de vida da bateria; e
 - baixo: abaixo de 30% de tempo de vida da bateria.

Este mecanismo foi concebido tendo em vista algumas situações em que não há necessidade de ficar enviando a descrição de eventos no momento em que estes ocorrem, pois a operação executada não representa um risco elevado à segurança oferecida pela aplicação às informações do usuário. Todavia, para aplicações que envolvam um alto risco é necessária uma análise mais cuidadosa das atividades do usuário. Outro aspecto importante é o gasto energético gerado em função do número de mensagens enviados pelas rede, ou seja, quanto maior for o acesso à rede, maior será o consumo de energia.

Para auxiliar na definição da quantidade de eventos que são enviados a cada comunicação estabelecida através da rede, foi utilizada uma rede Bayesiana (seção 2.2.2.4). Na conclusão da rede Bayesiana foram definidas três probabilidades de quantidade de eventos enviados a cada conexão com o servidor: 1 (um), 4 (quatro) e 8 (oito) eventos. Na definição da rede foi considerado que cada quantidade de eventos possui a mesma probabilidade de ocorrência. Também foram definidas as probabilidades condicionais a priori, ou seja, a probabilidade de uma determinada quantidade de eventos diante de uma evidência.

A Tabela 5.4 apresenta as probabilidades condicionais a priori. Por exemplo, se a restrição da aplicação é alta, é indicada uma probabilidade de 85% para o envio de uma mensagem contendo apenas um evento, haja vista que neste caso é necessário analisar as atividades do usuário a medida que estas ocorrem. Porém, se a restrição da aplicação é baixa, pode-

se coletar uma quantidade maior de eventos antes de estabelecer uma comunicação com o servidor para o envio de eventos. Neste caso a probabilidade para o envio de uma mensagem com apenas um evento é de apenas 5%, e 85% para o envio de uma mensagem com 8 eventos. Estas probabilidades foram indicadas pelo autor, analisando-se cada situação. Para cada hipótese, os valores das probabilidades dos itens devem totalizar cem por cento.

Tabela 5.4: Probabilidades condicionais a priori

Parâmetros	Categorias	1 evento	4 eventos	8 eventos
Restrição da Aplicação	Alta	85%	30%	5%
	Média	10%	30%	10%
	Baixa	5%	40%	85%
Status da Bateria	Alto	85%	30%	5%
	Médio	10%	30%	10%
	Baixo	5%	40%	85%

A Figura 5.6 ilustra o diagrama da rede Bayesiana contruída para este estudo de caso. A partir das probabilidades, a priori, são geradas as probabilidades condicionais, ou seja, probabilidade de um evento *A* ocorrer, sabendo que um evento *B* ocorreu. No último passo é aplicado o teorema de *Bayes* para gerar novos valores de probabilidades, estas são as probabilidades posteriores.

Para representar e validar a rede e, assim, melhor compreender o processo, utilizou-se o software *Netica* [Norsys, 2010]. Na Figura 5.6, a rede Bayesiana está em estado de latência, ou seja, nenhuma evidência da rede está selecionada, pois o sistema está aguardando pela entrada de dados para calcular as probabilidades.

A Figura 5.7 apresenta a rede com determinadas evidências instanciadas. Neste exemplo ilustrado, a restrição da aplicação é baixa e bateria está com um nível médio de carga. Então, a partir dessas evidências, a rede Bayesiana determinou as probabilidades para cada hipótese de quantidade de eventos enviados a cada conexão estabelecida. Conforme a Figura 5.7, pode-se observar que a alternativa mais indicada para a quantidade de eventos a serem enviados pela rede para este caso seria uma mensagem contendo quatro eventos.

A fim de obter uma visão completa de todas as possibilidades que podem ocorrer no cenário modelado, utilizou-se a rede Bayesiana para

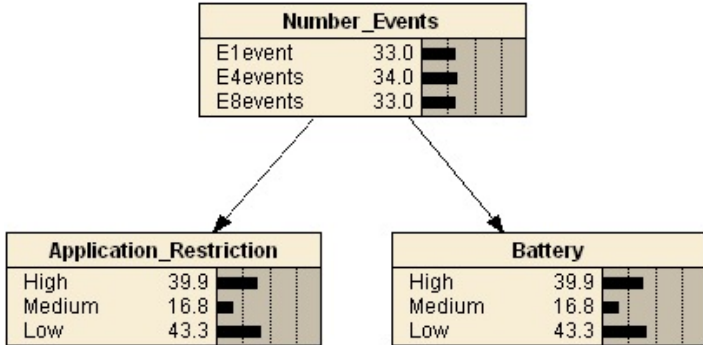


Figura 5.6: Rede Bayesiana em estado de latência

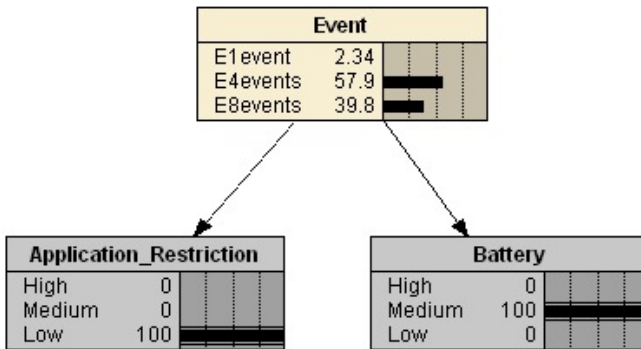


Figura 5.7: Rede Bayesiana instanciada

determinar as probabilidades para tais possibilidades. Como pode ser observado na Tabela 5.5, que apresenta as probabilidades para todas as instâncias, na grande maioria dos casos, a hipótese indicada para quantidade de eventos enviados em cada conexão é a mensagem composta por 4 (quatro) eventos. Por outro lado, a hipótese de enviar uma mensagem contendo apenas um evento deve ser utilizada apenas quando a restrição da aplicação é elevada e o nível restante de bateria é alto. Ainda, a hipótese de enviar uma mensagem contendo 8 eventos deve ser utilizada apenas quando a restrição de segurança da aplicação é baixa e, também, o

nível de bateria é baixo.

Tabela 5.5: Probabilidades para todas as instâncias

Restrição da Aplicação	Nível da Bateria	1 evento	4 eventos	8 eventos
Alta	Alto	88,30%	11,30%	0,30%
Alta	Médio	46,50%	50,70%	2,73%
Alta	Baixo	20,30%	59,20%	20,30%
Média	Alto	46,50%	50,70%	2,73%
Média	Médio	8,87%	82,20%	8,87%
Média	Baixo	2,34%	57,80%	30,70%
Baixa	Alto	20,30%	59,20%	20,30%
Baixa	Médio	2,34%	57,90%	39,80%
Baixa	Baixo	0,28%	18,50%	81,10%

5.3.3 Autonomicidade e Dinamicidade

Após a análise dos módulos auxiliares propostos para possibilitar o processo de autenticação sensível ao contexto, partiu-se para a análise do núcleo da arquitetura (Analisador de Crenças) e a sua interação com os demais módulos. Tal análise é realizada através de ciclos completos de autenticação, ou seja, (i) o evento é capturado no dispositivo móvel, posteriormente, (ii) este evento é analisado sob uma perspectiva espaço-temporal, (iii) depois a probabilidade condicional determinada pelo modelo de permutação espaço-temporal é utilizada para verificar o grau de similaridade deste evento com o perfil comportamental do usuário e, por fim, (iv) analisa-se a necessidade de desafiar o usuário, baseando-se no nível de autenticação exigido pela aplicação.

A partir dessa análise completa, é possível determinar a capacidade do sistema de autenticação sensível ao contexto em agregar novos conhecimentos e habilidades do usuário, possibilitando uma maior autonomicidade do sistema. Com este propósito, primeiramente, definiu-se quais seriam os atributos comportamentais a serem considerados. Então, optou-se pelos seguintes parâmetros: dispositivo móvel utilizado, localização em que o evento ocorre, marca de tempo (*timestamp*) do momento em que o evento ocorre, aplicação executada e restrição da aplicação executada.

Posteriormente, definiu-se quais seriam os pesos associados a cada atributo comportamental, a fim de atribuir diferentes níveis de prioridade

para a análise dos diferentes atributos comparativos. Assim, utilizou-se os seguintes pesos:

- $P_{device} = 0,75$
- $P_{local} = 1,00$
- $P_{timestamp} = 1,00$
- $P_{app_id} = 0,5$
- $P_{app_restriction} = 0,5$
- $P = (P_{device}, P_{local}, P_{timestamp}, P_{app_id}, P_{app_restriction})$

Então, partiu-se para a primeira interação entre o usuário e o sistema de autenticação. Assim, ao executar pela primeira vez a aplicação de submissão de workflow para a grade computacional, o usuário foi requisitado para inserir as suas informações e preferências de autenticação, tais como, nome, login, senha e respostas aos desafios apresentados na Tabela 5.2, opção pela modalidade de autenticação para aplicações que necessitam um baixo nível de autenticação (escrever ou escolher a resposta ao desafio) e dispositivo geralmente utilizado para realizar as operações desejadas. A tela com as informações requisitadas ao usuário é apresentada na Figura 5.8.

A partir dessa entrada de informações, o sistema é capaz de extrair o perfil explícito e o perfil de sessão do usuário (apresentados na Figura 5.9), que na primeira interação são iguais. Conforme mostrado na Figura 5.9, o usuário requisitou a submissão de um workflow para a grade computacional (1- código da aplicação de submissão de workflows). Esta aplicação exige um nível de autenticação médio (2 - código do nível de restrição de autenticação). Tal requisição foi realizada a partir do dispositivo, identificado pelo código 1, e no momento, tal dispositivo possuía carga restante superior a 70% (nível alto de bateria).

Quando o usuário faz seu primeiro acesso ao sistema, cadastrando suas informações pessoais, o usuário está, automaticamente, autenticado no sistema. Assim, o processo de autenticação proposto é realizado a partir da segunda interação do usuário. Assim, para a segunda interação, realizou-se a requisição da mesma aplicação de submissão de workflows, a partir da mesma localização da primeira interação. Entretanto, tal requisição foi realizada a partir do outro dispositivo móvel disponível. A descrição do segundo evento e a descrição do perfil de sessão são apresentadas na Figura 5.10

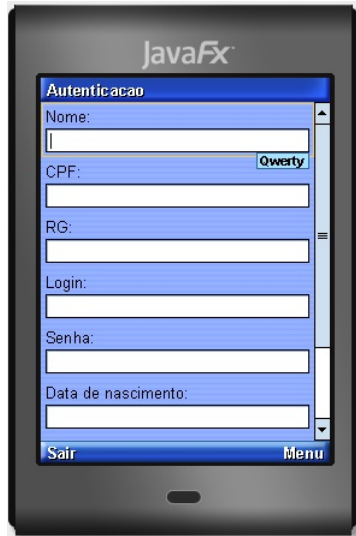


Figura 5.8: Tela de inserção de dados iniciais

<pre> <explicit-profile> <timestamp>1265039001</timestamp> <device> <id>1</id> <battery-level>high</battery-level> </device> <location> <latitude>-27.594176</latitude> <longitude>-48.522159</longitude> </location> <app> <id>1</id> <restriction>2</restriction> </app> </explicit-profile> </pre>	<pre> <session-profile> <timestamp>1265039001</timestamp> <device> <id>1</id> <battery-level>high</battery-level> </device> <location> <latitude>-27.594176</latitude> <longitude>-48.522159</longitude> </location> <app> <id>1</id> <restriction>2</restriction> </app> </session-profile> </pre>
(a)	(b)

Figura 5.9: (a) Perfil explícito e (b) Perfil de sessão extraídos das informações inseridas pelo usuário

A fim de utilizar uma base para uma análise maior sobre os eventos, utilizou-se a base de dados composta pelos 28 eventos utilizados no experimento apresentado na seção 5.3.1. Assim, ao comparar o *p-value* entre as duas execuções do modelo de permutação espaço-temporal (com

<pre> <event> <timestamp>1265040061</timestamp> <device> <id>2</id> <battery-level>high</battery-level> </device> <location> <latitude>-27.594176</latitude> <longitude>-48.522159</longitude> </location> <app> <id>1</id> <restriction>2</restriction> </app> </event> </pre>	<pre> <session-profile> <timestamp>1265039001</timestamp> <device> <id>1</id> <battery-level>high</battery-level> </device> <location> <latitude>-27.594176</latitude> <longitude>-48.522159</longitude> </location> <app> <id>1</id> <restriction>2</restriction> </app> </session-profile> </pre>
(a)	(b)

Figura 5.10: (a) Descrição do evento da segunda interação com o sistema e (b) Perfil de sessão

co-variáveis e sem co-variáveis), o sistema escolhe o valor mínimo entre tais valores, a fim de realizar a análise sobre o pior caso (caso que representa um maior risco ao processo de autenticação). Portanto, o sistema utiliza o *p-value* de 0,826 determinado pela análise espaço-temporal sem co-variáveis. Então, tal valor é utilizado para o cálculo do grau de similaridade entre o evento capturado e o perfil de sessão, que representa o contexto de execução do usuário e, portanto, existe apenas durante a interação do usuário com a aplicação em questão.

Para determinar os vetores necessários para o cálculo do grau de similaridade, faz-se necessária a utilização dos pesos definidos anteriormente para determinar o *VetorP*. Por outro lado, para determinar o *VetorE*, faz-se necessária a comparação dos atributos comparativos entre o evento capturado e o perfil de sessão. O Algoritmo 5.1 descreve o processo de definição do *VetorE*.

Após a definição destes dois vetores, é possível a determinação do grau de similaridade entre o vetor capturado e o perfil de sessão, conforme é apresentado abaixo:

$$\begin{aligned}
 P &= (0.75, 1.0, 1.0, 0.5, 0.5) \\
 E &= (0.0, 0.826, 0.826, 1.0, 1.0) \\
 \text{Grau de Similaridade} &= 0.862119
 \end{aligned}$$

Entretanto, fez-se necessário determinar quais são os intervalos de valores do grau de similaridade que determinam cada categoria do usuário (normal, suspeito e anormal). Então, assumiu-se a seguinte definição:

- *Usuário normal*: grau de similaridade superior a 90%;

Algorithm 5.1 Determinação do VetorE

Início

if Dispositivo do evento is Dispositivo do perfil de sessão **then** $VetorE_{device} \leftarrow 1.0$ **else** $VetorE_{device} \leftarrow 0$ **end if** $VetorE_{local} \leftarrow PValue$ $VetorE_{timestamp} \leftarrow PValue$ **if** Aplicação do evento is Aplicação do perfil de sessão **then** $VetorE_{app_id} \leftarrow 1.0$ **else** $VetorE_{app_id} \leftarrow 0$ **end if****if** Restrição da aplicação do evento is Restrição da aplicação do perfil de sessão **then** $VetorE_{app_restriction} \leftarrow 1.0$ **else** $VetorE_{app_restriction} \leftarrow 0$ **end if**Fim

- *Usuário suspeito*: grau de similaridade entre 70% e 90%; e
- *Usuário anormal*: grau de similaridade inferior a 70%.

Conforme a definição dos intervalos de valores acima, faz-se necessário que o usuário seja desafiado com o desafio referente a um usuário suspeito e referente ao nível de autenticação exigido pela aplicação (nível médio), conforme a Figura 5.11. Quando o usuário responde de forma correta ao desafio, ele é autenticado no sistema, executando a operação desejada e o evento requisitado é inserido no perfil de usuário, que contém o histórico de suas interações com o sistema.



Figura 5.11: Desafio apresentado ao usuário

Então, para analisar a capacidade do sistema em agregar as novas características do contexto de execução do usuário, executou-se uma nova requisição para a mesma aplicação, a partir da mesma localização e, desta vez, utilizando o mesmo dispositivo móvel. Assim, a definição dos vetores utilizados para o cálculo do grau de similaridade são as seguintes para este caso:

$$P = (0.75, 1.0, 1.0, 0.5, 0.5)$$

$$E = (1.0, 0.98, 0.98, 1.0, 1.0)$$

$$\text{Grau de Similaridade} = 0.9972$$

Portanto, conforme ilustrado na Figura 5.12, é perceptível que a arquitetura proposta para autenticação sensível ao contexto possibilita a evolução de seus parâmetros comportamentais, através da incorporação dos eventos (atividades) do usuário aos perfis explícitos, implícitos e de sessão do usuário. Conseqüentemente, o sistema é capaz de ampliar a sua base de conhecimento e, então, agregar as ações do usuário, que refletem o seu conhecimento e habilidades, refinando o processo de autenticação conforme o número de interações com o usuário. Através deste dinamismo oferecido pela arquitetura proposta, é possível, então, prover uma maior autonomicidade ao usuário, ou seja, com o sistema de autenticação cada vez mais preciso reduz-se a necessidade da entrada de informação de forma explícita ao sistema (resposta a desafios).

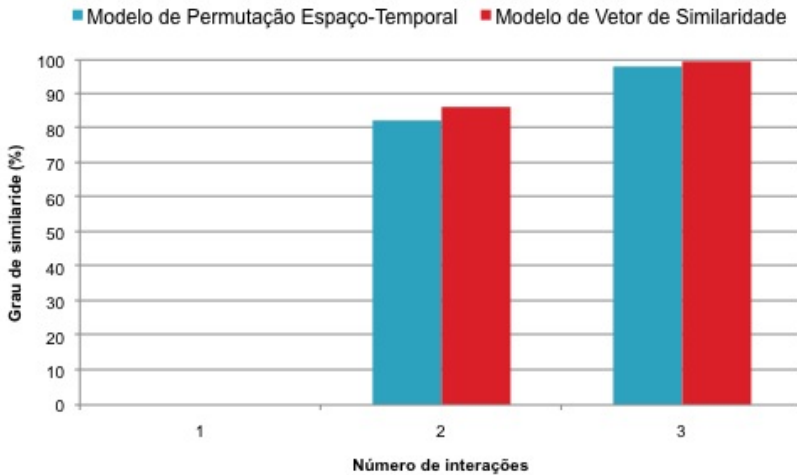


Figura 5.12: Evolução do sistema conforme o número de interações

Capítulo 6

Conclusões e Trabalhos Futuros

Nesta dissertação foi abordado um problema específico encontrado em ambientes de computação móvel que empregam dispositivos móveis, como smartphones e celulares, como interfaces de acesso a serviços e recursos: autenticação de usuários de forma dinâmica e ciente das restrições de tais dispositivos. Como principais restrições, destacam-se a autonomia energética e interfaces pouco amigáveis para entrada explícita de informação, quando comparadas com outras plataformas computacionais comumente utilizadas, como, desktops. Desta forma, a abordagem adotada para solucionar esta dificuldade foi a utilização dos recursos oferecidos por grande parte dos dispositivos móveis presentes no mercado, tais como, sistemas de geoposicionamento, calendário, ligações realizadas e recebidas, nível da bateria e aplicações executadas, por exemplo.

Neste contexto, esta dissertação justificou a sua importância, levantou os requisitos necessários e apresentou uma proposta de uma arquitetura de autenticação sensível ao contexto que permitisse a identificação de usuários de uma forma mais dinâmica. Tal identificação é realizada através da análise comportamental das atividades dos usuários que considera diversos contextos, que foram identificados na modelagem contextual e fazem parte do contexto do usuário. Essa modelagem contextual possibilita uma visão mais completa da situação do usuário e, então, o mecanismo de autenticação é capaz de tomar decisões mais eficazes.

Através da pesquisa realizada com relação à determinação do comportamento de usuários em ambientes de computação móvel, constatou-se a importância de considerar, simultaneamente, dois atributos fundamentais no contexto do usuário: espaço e tempo. Tais propriedades são importantes, visto que os seres humanos possuem hábitos, correlações de tempo são relevantes para determinar eventos sucessivos que definem um perfil comportamental. Desta forma, nesta dissertação, um evento é definido como a situação de uma entidade definida por um ou mais contextos, que compõem o contexto do usuário, em uma determinada localização e em um determinado espaço de tempo.

Portanto, fez-se necessária a pesquisa de modelos que considerassem a análise destes dois atributos simultaneamente, a fim de obter uma avaliação mais precisa do comportamento do usuário, ou seja, tanto identificando uma conformidade no padrão de comportamento, quanto identificando possíveis anomalias de comportamento que caracterizam uma falha no processo de autenticação. Então, foi proposta a utilização de um modelo de permutação espaço-temporal para analisar probabilisticamente a ocorrência de tais anomalias, considerando os eventos capturados através dos recursos disponíveis nos dispositivos móveis. Os resultados experimentais envolvendo o modelo analítico proposto apresentam uma eficiência significativa na detecção e análise de anomalias no processo de autenticação devido à utilização do modelo de permutação espaço-temporal.

Além disso, a arquitetura proposta mostrou que atende ao requisito de autonomicidade e dinamicidade, pois, através dos perfis comportamentais definidos pelo modelo de espaço vetorial, o sistema é capaz de agregar as habilidades e conhecimentos adquiridos pelo usuário durante a sua interação com o sistema. Em adição, a proposta provê flexibilidade por permitir diferentes formas de autenticação, conforme os níveis de segurança exigidos pelas aplicações executadas. Portanto, é perceptível que a abordagem proposta neste trabalho de pesquisa foi capaz de contornar com sucesso a falta de alternativas existente na literatura para atender os requisitos de sensibilidade ao contexto, eficiência computacional, flexibilidade, autonomicidade e dinamicidade simultaneamente.

Como trabalhos futuros, espera-se realizar os seguintes experimentos a fim de obter uma análise mais completa da arquitetura proposta para autenticação sensível ao contexto:

- Análise sobre o impacto do número de usuários sobre o mecanismo de autenticação sensível ao contexto, ou seja, a capacidade do sistema em manter uma taxa aceitável de acerto no processo de autenticação conforme o número de usuários cadastrados no sistema aumenta; e
- Realização dos mesmos experimentos apresentados nesta dissertação utilizando-se outros modelos de smartphones, a fim de obter uma análise mais completa quanto à eficácia energética do mecanismo, independente da arquitetura de hardware dos dispositivos móveis; e
- Avaliação sobre o impacto dos pesos associados aos atributos comparativos para o cálculo do grau de similaridade, a fim de propor uma forma para redefinição dos pesos, conforme o número de interações com o sistema.

Referências Bibliográficas

- [seg, 2003] (2003). Public printing and documents. Title 44 U.S. Code, Pts. 3542.
- [Abowd et al., 2002] Abowd, G., Bobick, A., Essa, I., Mynatt, E., and Rogers, W. (2002). The aware home: Developing technologies for successful aging. In *Proceedings of AAAI Workshop on Automation as a Care Giver*, pages 1–7.
- [Agostini et al., 2006] Agostini, A., Bettini, C., and Riboni, D. (2006). Experience report: ontological reasoning for context-aware internet services. In *Proceedings of the 4th Annual IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2006*, pages 5–12. IEEE Computer Society.
- [Apple, 2010] Apple (2010). Apple iphone technical specifications. <http://support.apple.com/specs/iphone>.
- [Arnstein et al., 2001] Arnstein, L., Sigdursson, S., and Franza, B. (2001). Ubiquitous computing in the biology laboratory. *Journal of the Association for Laboratory Automation*, 6(1):66–70.
- [Atallah et al., 2007] Atallah, L., ElHelw, M., Pansiot, J., Stoyanov, D., Wang, L., Lo, B., and Yang, G. (2007). Behaviour profiling with ambient and wearable sensing. In *Proceedings of the 4th International Workshop on Wearable and Implantable Body Sensor Networks, BSN 2007*, pages 133–138. Springer.
- [Atallah and Yang, 2009] Atallah, L. and Yang, G.-Z. (2009). The use of pervasive sensing for behaviour profiling – a survey. *Pervasive and Mobile Computing*, 5(5):447–464.
- [Augustin et al., 2006] Augustin, I., Yamin, A., da Silva, L., Real, R., Frainger, G., and Geyer, C. (2006). Isamadapt: abstractions and tools for designing general-purpose pervasive applications. *Software: Practice and Experience*, 36(11-12):1231–1256.

- [Baader et al., 2003] Baader, F., Calvanese, D., McGuinness, D., Patel-Schneider, P., and Nardi, D. (2003). *The description logic handbook: theory, implementation, and applications*. Cambridge University Press.
- [Babu and Venkataram, 2009] Babu, B. and Venkataram, P. (2009). A dynamic authentication scheme for mobile transactions. *International Journal of Network Security*, 8:59–74.
- [Balieiro et al., 2009] Balieiro, A., Santos, N., Carneiro, A., and Vieira, C. (2009). Detecção de aglomerados dos alertas de desmatamento no sul do estado do Amazonas usando estatística de varredura espaço-temporal. *Anais do XIV Simpósio Brasileiro de Sensoriamento Remoto*, pages 3519–3526.
- [Bardram, 2005] Bardram, J. (2005). Activity-based computing: support for mobility and collaboration in ubiquitous computing. *Personal and Ubiquitous Computing*, 9(5):312–322.
- [Beato Filho et al., 2001] Beato Filho, C., Assunção, R., Silva, B., Marinho, F., Reis, I., and Almeida, M. (2001). Conglomerados de homicídios eo tráfico de drogas em belo horizonte, minas gerais, brasil, de 1995 a 1999. *Cad Saúde Pública*, 17:1163–71.
- [Becker and Dürr, 2005] Becker, C. and Dürr, F. (2005). On location models for ubiquitous computing. *Personal and Ubiquitous Computing*, 9(1):20–31.
- [Becker and Nicklas, 2004] Becker, C. and Nicklas, D. (2004). Where do spatial context-models end and where do ontologies start? a proposal of a combined approach. In *Proceedings of the 1st International Workshop on Advanced Context Modelling, Reasoning and Management. In Conjunction with UbiComp 2004*. ACM.
- [Bellavista et al., 2003] Bellavista, P., Corradi, A., Montanari, R., and Stefanelli, C. (2003). Context-aware middleware for resource management in the wireless internet. *IEEE Transactions on Software Engineering*, 29(12):1086–1099.
- [Bertino et al., 1998] Bertino, E., Jajodia, S., Mancini, L., and Ray, I. (1998). Advanced transaction processing in multilevel secure file stores. *IEEE Transactions on Knowledge and Data Engineering*, 10(1):120–135.
- [Bettini et al., 2009] Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., and Riboni, D. (2009). A survey

of context modelling and reasoning techniques. *Pervasive and Mobile Computing*, In Press, Corrected Proof.

- [Black and Edgar, 2009] Black, M. and Edgar, W. (2009). Exploring mobile devices as grid resources: Using an x86 virtual machine to run boinc on an iphone. In *Proceedings of the 10th IEEE/ACM International Conference on Grid Computing, GRID 2009*, pages 9–16. IEEE Computer Society.
- [Cassens and Kofod-Petersen, 2006] Cassens, J. and Kofod-Petersen, A. (2006). Using activity theory to model context awareness: a qualitative case study. In *Proceedings of the 19th International Florida Artificial Intelligence Research Society Conference, Florida, USA, AAAI Press*, pages 619–624.
- [Chen et al., 2004] Chen, H., Perich, F., Finin, T., and Joshi, A. (2004). Soupa: standard ontology for ubiquitous and pervasive applications. In *Proceedings of the 1st Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, MobiQuitous 2004*, pages 258–267. IEEE Computer Society.
- [Cook, 2007] Cook, D. (2007). Making sense of sensor data. *IEEE Pervasive Computing*, 6(2):105–108.
- [Corradi et al., 2004] Corradi, A., Montanari, R., and Tibaldi, D. (2004). Context-based access control management in ubiquitous environments. In *Proceedings of the 3rd IEEE International Symposium on Network Computing and Applications, NCA 2004*, pages 253–260. IEEE Computer Society.
- [Davies et al., 1998] Davies, N., Mitchell, K., Cheverst, K., and Blair, G. (1998). Developing a context sensitive tourist guide. In *Proceedings of the 1st Workshop on Human Computer Interaction with Mobile Devices*, pages 64–68.
- [Dey, 2001] Dey, A. (2001). Understanding and using context. *Personal and Ubiquitous Computing*, 5(1):4–7.
- [Dobson and Ye, 2006] Dobson, S. and Ye, J. (2006). Using fibrations for situation identification. In *Proceedings of the 4th International Conference on Pervasive Computing, Pervasive 2006*, pages 645–651. Elsevier.
- [Dong and Li, 1999] Dong, G. and Li, J. (1999). Efficient mining of emerging patterns: Discovering trends and differences. In *Proceedings*

of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'99, pages 43–52. ACM.

- [Du et al., 2006] Du, Y., Chen, F., Xu, W., and Li, Y. (2006). Recognizing interaction activities using dynamic bayesian network. In *Proceedings of the 18th International Conference on Pattern Recognition*, pages 618–621. IEEE Computer Society.
- [Eagle and Pentland, 2006] Eagle, N. and Pentland, A. (2006). Reality mining: sensing complex social systems. *Personal and Ubiquitous Computing*, 10(4):255–268.
- [Gellersen et al., 2002] Gellersen, H., Schmidt, A., and Beigl, M. (2002). Multi-sensor context-awareness in mobile devices and smart artifacts. *Mobile Networks and Applications*, 7(5):341–351.
- [Gu et al., 2009] Gu, T., Wu, Z., Tao, X., Pung, H. K., and Lu, J. (2009). epsicar: An emerging patterns based approach to sequential, interleaved and concurrent activity recognition. In *Proceedings of the 7th Annual IEEE International Conference on Pervasive Computing and Communications, PerCom 2009*, pages 1–9. IEEE Computer Society.
- [Halpin and Morgan, 2001] Halpin, T. and Morgan, T. (2001). *Information modeling and relational databases: from conceptual analysis to logical design*. Morgan Kaufmann, San Francisco.
- [Hamid et al., 2009] Hamid, R., Maddi, S., Johnson, A., Bobick, A., Essa, I., and Isbell, C. (2009). A novel sequence representation for unsupervised analysis of human activities. *Artificial Intelligence*, 173(14):1221–1244.
- [Henricksen and Indulska, 2004] Henricksen, K. and Indulska, J. (2004). Modelling and using imperfect context information. In *Proceedings of the 2nd Annual IEEE Conference on Pervasive Computing and Communications, PerCom 2004*, page 33. IEEE Computer Society.
- [Henricksen and Indulska, 2006] Henricksen, K. and Indulska, J. (2006). Developing context-aware pervasive computing applications: Models and approach. *Pervasive and Mobile Computing*, 2(1):37–64.
- [Henricksen et al., 2002] Henricksen, K., Indulska, J., and Rakotonirainy, A. (2002). Modeling context information in pervasive computing systems. *Lecture Notes in Computer Science*, pages 167–180.

- [Hong et al., 2007] Hong, J., Satyanarayanan, M., and Cybenko, G. (2007). Guest editors' introduction: Security & privacy. *IEEE Pervasive Computing*, 6(4):15–17.
- [Horrocks et al., 2003] Horrocks, I., Patel-Schneider, P., and Van Harmelen, F. (2003). From shiq and rdf to owl: The making of a web ontology language. *Web Semantics: Science, Services and Agents on the World Wide Web*, 1(1):7–26.
- [HTC, 2010] HTC (2010). Htc google nexus one technical specifications. <http://www.htc.com/www/product/nexusone/specification.html>.
- [Hung et al., 2008] Hung, L., Hassan, J., Riaz, A., Raazi, S., Weiwei, Y., Canh, N., Truc, P., Lee, S., Lee, H., Son, Y., et al. (2008). Activity-based security scheme for ubiquitous environments. In *Proceedings of the 27th IEEE International Performance, Computing and Communications Conference, IPCCC 2008*, pages 475–481. IEEE Computer Society.
- [Intille, 2002] Intille, S. (2002). Designing a home of the future. *IEEE Pervasive Computing*, 1(2):76–82.
- [Jameel et al., 2007] Jameel, H., Shaikh, R., Lee, H., and Lee, S. (2007). Human identification through image evaluation using secret predicates. topics in cryptology-ct-rsa 07. *Lecture Notes in Computer Science, Springer-Verlag*, 4377:67–84.
- [Johnson, 2009] Johnson, G. (2009). Towards shrink-wrapped security: A taxonomy of security-relevant context. In *Proceedings of the 7th Annual IEEE International Conference on Pervasive Computing and Communications, PerCom 2009*, pages 1–2. IEEE Computer Society.
- [Judd and Steenkiste, 2003] Judd, G. and Steenkiste, P. (2003). Providing contextual information to pervasive computing applications. In *Proceedings of the 1st Annual IEEE International Conference on Pervasive Computing and Communications, PerCom 2003*, pages 133–142. IEEE Computer Society.
- [Kim et al., 2010] Kim, E., Helal, S., and Cook, D. (2010). Human activity recognition and pattern discovery. *IEEE Pervasive Computing*, 9(1):48–53.
- [Klyne et al., 2004] Klyne, G., Reynolds, F., Woodrow, C., Ohto, H., Hjelm, J., Butler, M., and Tran, L. (2004). Composite capability/preference profiles (cc/pp): Structure and vocabularies 1.0. *W3C Recommendation*, 15.

- [Knudsen, 2002] Knudsen, J. (2002). What's new in midp 2.0. <http://developers.sun.com/mobility/midp/articles/midp20>.
- [Kulldorff, 2006] Kulldorff, M. (2006). SatscanTM v 7.0: Software for the spatial and space-time scan statistics. *Information Management Services Inc., Silver Spring, MD*.
- [Malek et al., 2008] Malek, B., Miri, A., and Karmouch, A. (2008). A framework for context-aware authentication. In *Proceedings of the 4th IET International Conference on Intelligent Environments, 2008*, pages 1–8.
- [Mannila et al., 2001] Mannila, H., Himberg, J., Korpiaho, K., Tikanmaki, J., and Toivonen, H. (2001). Time series segmentation for context recognition in mobile devices. In *Proceedings of the IEEE International Conference on Data Mining, ICDM 2001*, pages 203–210. IEEE Computer Society.
- [Mantjarvi et al., 2004] Mantjarvi, J., Himberg, J., Kangas, P., Tuomela, U., and Huuskonen, P. (2004). Sensor signal data set for exploring context recognition of mobile devices. In *Proceedings of the 2nd International Conference on Pervasive Computing, PERSVASIVE 2004*.
- [McMichael and Melbourne, 1999] McMichael, H. and Melbourne, A. (1999). An activity based perspective for information systems research. In *10th Australian Conference on Information Systems*. Citeseer.
- [Nicklas et al., 2008] Nicklas, D., Grossmann, M., Minguez, J., and Wieland, M. (2008). Adding high-level reasoning to efficient low-level context management: A hybrid approach. In *Proceedings of the 6th Annual IEEE International Conference on Pervasive Computing and Communications, PerCom 2008*, pages 447–452. IEEE Computer Society.
- [Nokia, 2009] Nokia (2009). Nokia energy profiler. http://www.forum.nokia.com/Library/Tools_and_downloads/Other/Nokia_Energy_Profiler/.
- [Norsys, 2010] Norsys (2010). Netica. <http://www.norsys.com/netica.html>.
- [Oliver and Horvitz, 2005] Oliver, N. and Horvitz, E. (2005). A comparison of hmms and dynamic bayesian networks for recognizing office activities. *User Modeling 2005*, pages 199–209.

- [Peddemors et al., 2010] Peddemors, A., Eertink, H., and Niemegeers, I. (2010). Predicting mobility events on personal devices. *To Appear in the Pervasive and Mobile Computing Journal – Special Issue on Human Behaviour in Ubiquitous Environments*.
- [Pentland, 2007] Pentland, A. (2007). Automatic mapping and modeling of human networks. *Physica A: Statistical Mechanics and its Applications*, 378(1):59–67.
- [Preece et al., 1994] Preece, J., Rogers, Y., Sharp, H., Beyon, D., Holland, S., and Carey, T. (1994). *Human-computer interaction*. Addison-Wesley.
- [Rahmati and Zhong, 2009] Rahmati, A. and Zhong, L. (2009). Human-battery interaction on mobile phones. *Pervasive and Mobile Computing*, 5(5):465–477.
- [Reyes Alamo and Wong, 2008] Reyes Alamo, J. and Wong, J. (2008). Service-oriented middleware for smart home applications. In *Proceedings of the IEEE Wireless Hive Networks Conference, WHNC 2008*, pages 1–4. IEEE Computer Society.
- [Rhodes et al., 1999] Rhodes, B., Minar, N., and Weaver, J. (1999). Wearable computing meets ubiquitous computing. In *Proceedings of the 3rd International Symposium on Wearable Computers, ISWC 1999*. IEEE Computer Society.
- [RIM, 2010] RIM (2010). Research in motion blackberry storm technical specifications. <http://www.blackberry.com/blackberrystorm/specifications.shtml>.
- [Rocha et al., 2010a] Rocha, C., Lima, J., Viera, M., Capretz, M., Bauer, M., Augustin, I., and Dantas, M. (2010a). A context-aware authentication approach based on behavioral definitions. *Proceedings of the 2010 International Conference on Information and Knowledge Engineering, IKE'10*.
- [Rocha et al., 2010b] Rocha, C., Viera, M., Capretz, M., Bauer, M., Augustin, I., and Dantas, M. (2010b). A user-centric authentication for advanced resource reservation in mobile grid environments. In *Proceedings of the 2010 International Conference on Grid Computing and Applications, GCA'10*.
- [Rosenthal and Fung, 1999] Rosenthal, D. and Fung, F. (1999). A test for non-disclosure in security level translations. In *Proceedings of the*

1999 IEEE Symposium on Security and Privacy, pages 196–206. IEEE Computer Society.

- [Rossetto et al., 2007] Rossetto, A., Borges, V., Silva, A., and Dantas, M. (2007). Summit - a framework for coordinating applications execution in mobile grid environments. In *Proceedings of the 8th IEEE/ACM International Conference on Grid Computing, GRID 2007*, pages 128–136. IEEE Computer Society.
- [Russell and Norvig, 2004] Russell, S. and Norvig, P. (2004). *Inteligência artificial*. Elsevier.
- [Saha and Mukherjee, 2003] Saha, D. and Mukherjee, A. (2003). Pervasive computing: A paradigm for the 21st century. *IEEE Computer*, 36(3):25–31.
- [Salber et al., 1999] Salber, D., Dey, A., and Abowd, G. (1999). The context toolkit: Aiding the development of context-enabled applications. In *Proceedings of the 17th ACM Conference on Human Factors in Computing Systems, CHI 1999*, page 441. ACM.
- [Salton, 1989] Salton, G. (1989). Automatic text processing: The transformation, analysis, and retrieval of information by computer. *Addison-Wesley Longman Publishing Co.*
- [Schilit et al., 1994] Schilit, B., Adams, N., Want, R., et al. (1994). Context-aware computing applications. In *Proceedings of the Workshop on Mobile Computing Systems and Applications*, pages 85–90.
- [Sheshagiri et al., 2004] Sheshagiri, M., Sadeh, N., and Gandon, F. (2004). Using semantic web services for context-aware mobile applications. In *Workshop on Context Awareness. In Conjunction with the 2nd International Conference on Mobile Systems, Applications and Services, MobiSys 2004*. ACM New York, NY, USA.
- [Sousa, 2005] Sousa, J. (2005). Scaling task management in space and time: Reducing user overhead in ubiquitous-computing environments. Ph.D. Thesis, Department of Computer Science, Carnegie Mellon University.
- [Sousa and Garlan, 2002] Sousa, J. and Garlan, D. (2002). Aura: an architectural framework for user mobility in ubiquitous computing environments. In *Proceedings of the 3rd Working IEEE/IFIP Conference*

- on Software Architecture, WICSA 2002*, pages 29–43. IEEE Computer Society.
- [Sousa et al., 2008] Sousa, J., Poladian, V., Garlan, D., Schmerl, B., and Steenkiste, P. (2008). Steps toward activity-oriented computing. In *Proceedings of the 22nd IEEE International Symposium on Parallel and Distributed Processing, IPDPS 2008*, pages 1–5. IEEE Computer Society.
- [SunMicrosystems, 2004] SunMicrosystems (2004). J2me and location-based services. <http://developers.sun.com/mobility/apis/articles/location>.
- [SunMicrosystems, 2010] SunMicrosystems (2010). Java 2 platform, micro edition (j2me). <http://java.sun.com/javame/index.jsp>.
- [Sutton and McCallum, 2006] Sutton, C. and McCallum, A. (2006). Introduction to statistical relational learning, chapter an introduction to conditional random fields for relational learning.
- [Toninelli et al., 2009] Toninelli, A., Montanari, R., Lassila, O., and Khushraj, D. (2009). What’s on users’ minds? toward a usable smart phone security model. *IEEE Pervasive Computing*, 8(2):32–39.
- [Uden, 2007] Uden, L. (2007). Activity theory for designing mobile learning. *International Journal of Mobile Learning and Organisation*, 1(1):81–102.
- [Vail et al., 2007] Vail, D. L., Veloso, M. M., and Lafferty, J. D. (2007). Conditional random fields for activity recognition. In *AAMAS '07: Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, pages 1–8, New York, NY, USA. ACM.
- [Viera et al., 2010] Viera, M., Rocha, C., Capretz, M., Bauer, M., and Dantas, M. (2010). Toward advance resource reservation in mobile grid configurations based on user-centric authentication. In *Anais do VIII Workshop em Clouds, Grids e Aplicações, WCGA'2010*. SBC.
- [Weiser, 1991] Weiser, M. (1991). The computer for the 21st century. *Scientific American*, 265(3):66–75.
- [Yang et al., 1999] Yang, J., Yang, W., Denecke, M., and Waibel, A. (1999). Smart sight: a tourist assistant system. In *Proceedings of 3rd International Symposium on Wearable Computers*, pages 73–78. IEEE Computer Society.

- [Ye et al., 2007] Ye, J., Coyle, L., Dobson, S., and Nixon, P. (2007). Using situation lattices to model and reason about context. In *Proceedings of the 4th International Workshop on Modeling and Reasoning in Context, MRC 2007*, pages 1–12. Citeseer.
- [Zhang et al., 2005] Zhang, D., Gu, T., and Wang, X. (2005). Enabling context-aware smart home with semantic technology. *International Journal of Human-friendly Welfare Robotic Systems*, 6(4):12–20.
- [Zhang et al., 2009] Zhang, L., Pan, G., Wu, Z., Li, S., and li Wang, C. (2009). Smartshadow: Modeling a user-centric mobile virtual space. In *Proceedings of the 7th Annual IEEE International Conference on Pervasive Computing and Communications, PerCom 2009*, pages 1–4. IEEE Computer Society.

Apêndice A

Publicações

Este apêndice visa ilustrar as publicações realizadas ao longo da pesquisa desse trabalho de dissertação.

A.1 Capítulos de Livros Publicados

- **Título:** Developing Knowledge Representation in Emergency Medical Assistance By Using Semantic Web Techniques
- **Livro:** Canadian Semantic Web: Technologies and Applications
- **Editora:** Springer
- **Data:** Segundo semestre de 2010
- **Autores:** Heloíse Manica, Cristiano C. Rocha, José Leomar Tosdesco e Mário A. R. Dantas

A.2 Trabalhos Completos Publicados em Anais de Congressos

- **Título:** A User-Centric Authentication for Advanced Resource Reservation in Mobile Grid Environments
- **Evento:** GCA 2010 – The 2010 International Conference on Grid Computing and Applications
- **Local:** Las Vegas, Nevada, EUA
- **Data:** Julho de 2010
- **Autores:** Cristiano C. Rocha, Matheus A. Viera, Miriam Capretz, Michael A. Bauer, Iara Augustin e Mário A. R. Dantas
- **Estrato Qualis/CAPES:** B3

- **Título:** Uma Arquitetura de Autenticação Baseada em Perfis Comportamentais
 - **Evento:** WISA - CISTI 2010 – 2nd Workshop on Intelligent Systems and Applications, held in conjunction with the 5th Iberian Conference on Information Systems and Technologies
 - **Local:** Santiago de Compostela, Espanha
 - **Data:** Junho de 2010
 - **Autores:** Cristiano C. Rocha, João Carlos D. Lima e Mário A. R. Dantas
-
- **Título:** Toward Advance Resource Reservation in Mobile Grid Configurations Based on User-Centric Authentication
 - **Evento:** WCGA 2010 – 8th Workshop on Clouds, Grids and Applications
 - **Local:** Gramado, RS, Brasil
 - **Data:** Maio de 2010
 - **Autores:** Cristiano C. Rocha, Matheus A. Viera, Miriam Capretz, Michael A. Bauer e Mário A. R. Dantas
 - **Estrato Qualis/CAPES:** B5
-
- **Título:** A Context-Aware Authentication Approach Based on Behavioral Definitions
 - **Evento:** IKE 2010 – The 2010 International Conference on Information and Knowledge Engineering
 - **Local:** Las Vegas, Nevada, EUA
 - **Data:** Julho de 2010
 - **Autores:** Cristiano C. Rocha, João Carlos D. Lima, Matheus A. Viera, Miriam Capretz, Michael A. Bauer, Iara Augustin e Mário A. R. Dantas

- **Estrato Qualis/CAPES:** B4

- **Título:** Toward Developing Knowledge Representation in Emergency Medical Assistance through a Ontology-based Semantic Cache Model
- **Evento:** SEKE 2009 – 21st International Conference on Software Engineering and Knowledge Engineering
- **Local:** Boston, Massachusetts, EUA
- **Data:** Julho de 2009
- **Autores:** Heloíse Manica, Cristiano C. Rocha, José Leomar Tosdesco e Mário A. R. Dantas
- **Estrato Qualis/CAPES:** B2

- **Título:** Uma Arquitetura de Reserva Antecipada de Recursos Centrada no Usuário para Ambientes de Grades Móveis
- **Evento:** I2TS 2009 – 8th International Information and Telecommunication Technologies Symposium
- **Local:** Florianópolis, SC, Brasil
- **Data:** Dezembro de 2009
- **Autores:** Cristiano C. Rocha, Matheus A. Viera, Luis Fernando Friedrich, Iara Augustin e Mário A. R. Dantas

- **Título:** Um Modelo para Desenvolvimento de Terminologia no Domínio de Emergência Médica com Utilização de Dispositivos Portáteis
- **Evento:** I2TS 2009 – 8th International Information and Telecommunication Technologies Symposium
- **Local:** Florianópolis, SC, Brasil

- **Data:** Dezembro de 2009
 - **Autores:** Heloíse Manica, Cristiano C. Rocha, José Leomar Todesco e Mário A. R. Dantas
-
- **Título:** Grid Computing Middleware Survey
 - **Evento:** I2TS 2008 – 7th International Information and Telecommunication Technologies Symposium
 - **Local:** Foz do Iguaçu, PR, Brasil
 - **Data:** Novembro de 2008
 - **Autores:** Cristiano C. Rocha, Denise J. Ferreira, Matheus A. Viera, Miguel Cartagena, Rodrigo G. Silva e Mário A. R. Dantas

Apêndice B

Configuração do Ambiente de Desenvolvimento

Esse apêndice tem como objetivo apresentar algumas das aplicações utilizadas para configuração do ambiente de desenvolvimento do protótipo.

B.1 Ambiente de Software

B.1.1 Java 2 Micro Edition (J2ME)

A plataforma *Java 2 Platform, Micro Edition* (J2ME) é destinada para o desenvolvimento de aplicativos para dispositivos com recursos restritos, tais como, memória, tela e poder de processamento limitados. Tal plataforma inclui a máquina virtual Java e um conjunto de APIs padrão do Java, definidos através da *Java Community*.

A arquitetura J2ME define configurações, perfis e APIs opcionais, como ilustrado na Figura B.1. Esta classificação indica as informações específicas sobre diferentes dispositivos. Uma configuração J2ME é composta de uma máquina virtual e um conjunto reduzido de bibliotecas. As bibliotecas do J2ME fornecem funcionalidades básicas para diversos modelos de dispositivos que compartilham características similares.

B.1.2 Framework SuMMIT

Devido ao advento das grades computacionais, fornecendo vários serviços, recursos e a capacidade de resolver problemas complexos, está se tornando cada vez mais comum a execução de aplicações com várias tarefas para a resolução de um único problema. Em geral, este agregado de tarefas tem interações e dependências, necessitando do uso de algumas ferramentas computacionais (por exemplo, software e banco de dados) que são compartilhadas pelas organizações virtuais da grade computacional.

As várias tarefas de uma aplicação representam um fluxo de trabalho no qual dados são enviados e recebidos entre as tarefas, obedecendo a certas regras. Neste contexto, torna-se necessário o emprego de um

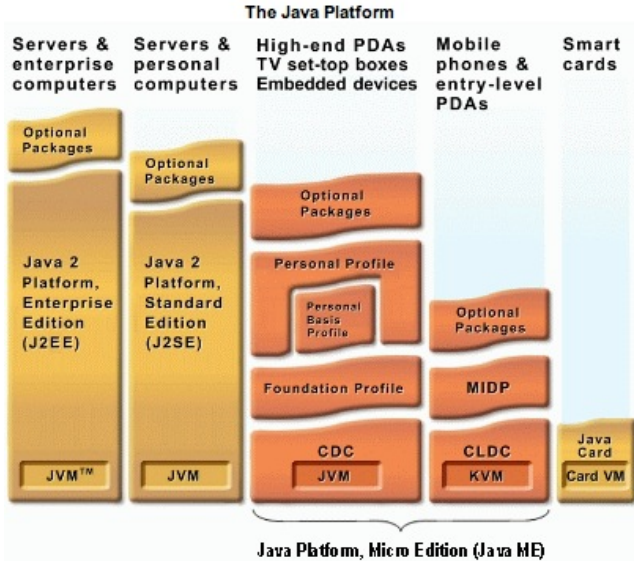


Figura B.1: Visão geral da plataforma Java

mecanismo para controlar, organizar e automatizar este fluxo de tarefas. Para este propósito, o conceito de *workflow* foi empregado na arquitetura proposta em [Rossetto et al., 2007]. A proposta apresenta uma solução genérica e de granularidade fina para definição de recursos da grade computacional usados em cada estágio da execução de aplicação em um modo coordenado e automatizado.

A arquitetura proposta em [Rossetto et al., 2007] é apresentada na Figura B.2. Tal arquitetura foi concebida com quatro componentes principais: *Mobile GUI*, *Workflow Manager*, *Agent* e *Resource Selector*.

O componente *Mobile GUI* foi projetado e desenvolvido, inicialmente, para PDAs, provendo algumas funcionalidades que permitem a interface de acesso à grade computacional. Tal componente foi implementado utilizando o J2ME Wireless Toolkit. Exemplos de aplicações providas por este componente são: submissão de *workflows* para a grade computacional e visualização de resultados parciais finais do *workflow* submetido à grade.

Por outro lado, o componente *Workflow Manager*, que se encontra no servidor, tem como atribuições processar e gerenciar os pedidos

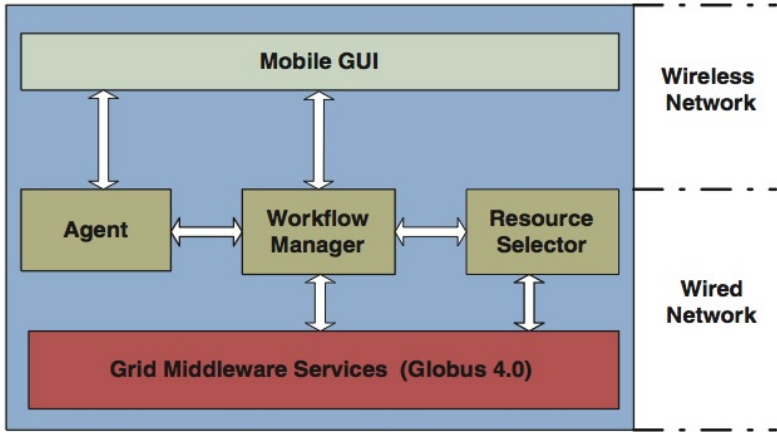


Figura B.2: Arquitetura do framework SuMMIT [Rossetto et al., 2007]

vindos dos dispositivos móveis para executar no ambiente de grade, além de coletar informações relacionadas à execução das tarefas. Ademais, tal componente fornece automatização para o usuário móvel, remetendo todas as tarefas para o escalonador de tarefas da grade computacional sem a necessidade de interação do usuário. Desta forma, ele permite mais agilidade na execução de tarefas que trabalham juntas para resolver o mesmo problema.

Com o propósito de garantir a consistência da aplicação em casos de desconexão, o componente *Agent* é responsável pela verificação do ambiente de execução e pela adaptação do fluxo de execução do *workflow* submetido a partir do dispositivo móvel. Portanto, tal componente é responsável pela detecção de desconexões ocorridas na rede e, então, adaptar o fluxo de execução a fim de evitar um estado de falha.

Por fim, o componente *Resource Selector* foi projetado e desenvolvido a fim de prover um ambiente mais flexível e extensível, considerando que as aplicações executadas em grades computacionais utilizam diversos recursos. Os módulos que integram este componente são apresentados na Figura B.3.

B.1.2.1 Extensões do Framework

A fim de possibilitar o processo de reserva de recursos da grade e, também, o cancelamento de tais reservas, fez-se necessária a modifica-

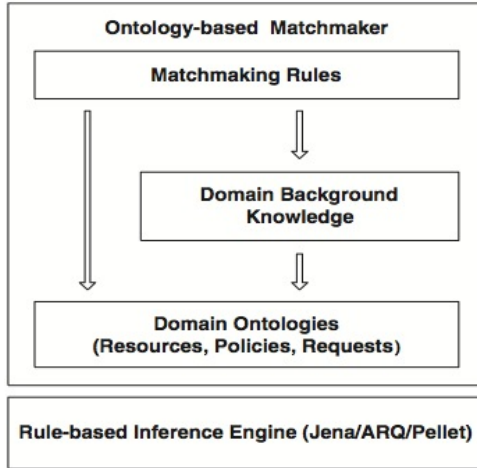


Figura B.3: Resource Selector - Interação entre os módulos [Rossetto et al., 2007]

ção do *framework* original. Conforme apresentado na Figura B.4, foram adicionados dois novos componentes à arquitetura do SuMMIT: *Resource Reservation* e *User-centric Authentication*. O componente *Resource Reservation* permite que os usuários da grade computacional móvel possam planejar uma utilização futura da grade. Então, tal módulo é responsável por garantir que as reservas de recursos realizadas sejam mantidas para uma futura submissão de *workflow* para a grade computacional. Além disso, tal componente possibilita o monitoramento dos recursos reservados e reservas canceladas. O módulo *User-centric Authentication* consiste na proposta apresentada nesta dissertação e é responsável por interceptar as requisições das aplicações que interagem com a grade computacional.

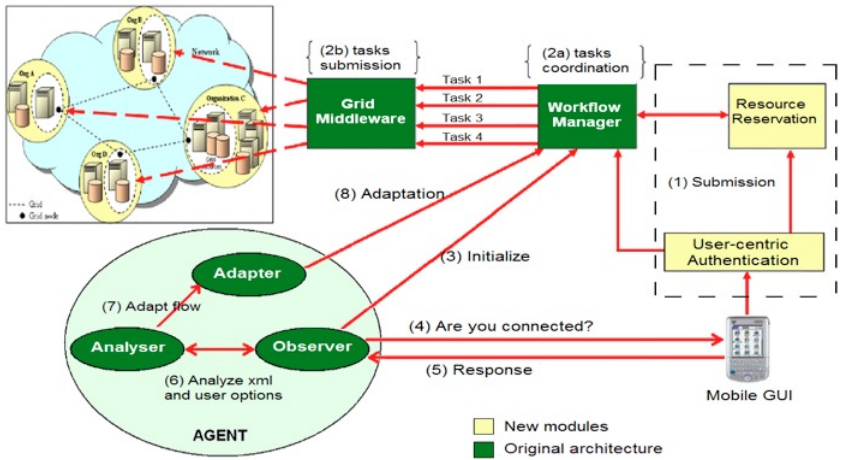


Figura B.4: Arquitetura estendida [Viera et al., 2010]