ISAM: um *Middleware* para Aplicações Móveis Distribuídas

Iara Augustin[1]
Adenauer Correa Yamin[1]
Edson Nascimento Silva Junior[1]
Jorge Luis Victoria Barbosa[1]
Cláudio Fernando Resin Geyer[1]
Gerson Geraldo Homrich Cavalheiro[2]

Resumo: A computação móvel é um novo paradigma computacional advindo da combinação das tecnologias de redes sem fio e sistemas distribuídos. A produção de software para este ambiente é complexa. O desafio é projetar aplicações cujos níveis de serviço e disponibilidade de recursos são imprevisíveis, e cujo comportamento é variável no tempo e no espaço. Para abordar esta questão está em desenvolvimento o projeto ISAM. ISAM é um *middleware* concebido para tratar com o dinamismo do ambiente móvel, projetado com mobilidade, flexibilidade e adaptabilidade intrínsecas.

Palavras-chave: *middleware*, computação móvel, sistemas adaptativos

Abstract: The mobile computing is a new computational paradigm, it emerging from combination of wireless networks and distributed system technologies. Software production for this environment is complex. Challenges include the applications design which service levels and resources available are unpredictable, and which behavior is variable in the time and in the space. To approach this issue is ongoing the ISAM project. ISAM is a *middleware* conceived to deal with the dynamism of mobile environment, it was designed with mobility, flexibility and adaptability built-in.

Keywords: *middleware*, mobile computing, adaptive systems

1 Instituto de Informática, UFRGS, Porto Alegre, RS
{august, adenauer, barbosa, edsonj, geyer}@inf.ufrgs.br
2 Centro de Ciências Exatas e Tecnológicas, UNISINOS, São Leopoldo, RS
gersonc@exatas.unisinos.br

1 Introdução

A computação móvel é um novo paradigma computacional oriunda das tecnologias de rede sem fio e sistemas distribuídos. Nela, o usuário, portando dispositivos móveis como *palmtops* e *notebooks*, tem acesso a uma infra-estrutura compartilhada e independente da sua localização física. Isto disponibiliza uma comunicação flexível entre as pessoas e um acesso aos serviços de rede. Observa-se que a crescente introdução de facilidades de comunicação tem deslocado as aplicações da computação móvel de uma perspectiva de uso pessoal (atual) para outras mais avançadas e de uso corporativo, como as aplicações móveis distribuídas.

A produção de software no ambiente móvel é complexa. Seus componentes são variáveis no tempo e no espaço em termos de conectividade, portabilidade e mobilidade. O desafio que se apresenta é projetar aplicações móveis distribuídas cujos níveis de serviço e disponibilidade de recursos são imprevisíveis. Existem, portanto, requisitos emergindo para uma nova classe de aplicações projetadas especificamente para este ambiente dinâmico.

Esta nova classe de aplicações tem sido referenciada de muitas formas: *environment-aware, network-aware, resource-aware, context-aware applications*. Porém, todas têm um conceito embutido: adaptação. A diferença está no grau de adaptabilidade e nos recursos que são objetos da adaptação. Segundo Satyanarayanan [26], mobilidade exige adaptabilidade. Isto significa que os sistemas devem ter consciência da localização e da situação onde estão inseridos, e devem tirar vantagem desta informação para configurar-se dinamicamente de um modo distribuído. O foco da complexidade na implementação de aplicações móveis com comportamento adaptativo, está no fato que os componentes distribuídos das mesmas sofrem influência dos diversos ambientes onde estão inseridos.

Sistemas distribuídos tradicionais são construídos com suposições sobre a infra-estrutura física de execução, como conectividade permanente e disponibilidade dos recursos necessários. Porém, essas suposições não são válidas nos sistemas móveis. Isto impede o uso direto das soluções adotadas pelos sistemas distribuídos, as quais podem ser altamente ineficientes devido à variabilidade freqüente da conexão à rede e da disponibilidade de recursos e serviços.

Parece, portanto, ser necessário definir uma nova arquitetura para sistemas móveis, projetada com mobilidade, flexibilidade e adaptabilidade intrínsecas. Com esta motivação, está em desenvolvimento o projeto ISAM (Infra-estrutura de Suporte às Aplicações Móveis Distribuídas), que integra as universidades UFRGS, UFSM, UCPel e UNISINOS, e a empresa PROMON*IP S.A. Este artigo apresenta a arquitetura ISAM e está organizado como segue. A seção 2 introduz o escopo onde o projeto se situa. Na seqüência, seção 3, apresenta-se o *middleware* proposto para o ISAM, e discute-se os requisitos, as decisões da arquitetura e as principais questões em andamento. A seção 4 focaliza o aspecto do comportamento adaptador e adaptativo do escalonador na arquitetura. Os trabalhos relacionados são discutidos na seção 5, e as conclusões são apresentadas na seção 6.

2 O Escopo do ISAM

O termo computação móvel (*mobile computing*) não está ainda bem definido e é usado pelos autores em um espectro de ambientes, que envolvem alguma forma de mobilidade. De forma geral, pode-se dizer que "computação móvel é a computação distribuída que envolve elementos (software, dados, hardware, usuário) cuja localização se altera no curso da execução" [6]. Esta definição torna evidente a amplitude de abrangência desta nova área da computação.

2.1 Cenários Possíveis da Computação Móvel

Dependendo dos elementos que possuem a propriedade de mobilidade, pode-se definir diferentes cenários. Entre eles:

- computação nômade (*nomadic computing*) - a mobilidade está no hardware e não é transparente. É baseada usualmente em facilidades de comunicação via acesso discado. A cada movimentação, uma nova conexão é requerida;
- computação com redes sem fio (*wireless computing*) - usuário portando um equipamento pode se mover dentro de uma área de acesso, enquanto mantém a conexão a um conjunto fixo;
- *mobile computation* - os elementos da aplicação podem ser mover. Pode-se ter somente: a mobilidade de código; a mobilidade de dados; ou a mobilidade de todo o estado da execução da aplicação (agentes móveis);
- computação global *(pervasive computing)* - o usuário, de posse de um equipamento portátil, executando aplicações com estado, dados ou código móveis, se locomove para qualquer ponto enquanto mantém a conexão à rede.

Este último cenário categoriza a mobilidade de todos os elementos, o qual permite ao usuário deslocar-se junto com seu ambiente computacional. Este é o cenário foco do ISAM.

2.2 O Ambiente de Rede

Para dar suporte de comunicação aos sistemas móveis, distinguem-se dois tipos de redes [21]: as redes infra-estruturadas e as redes *ad hoc*. As **Redes Infra-Estruturadas** compõem-se de *host móveis* (dispositivos portáteis que se comunicam por meio sem fio) com acesso de comunicação a *estações-bases* ou pontos-de-acesso (servidores de rede com interface para redes sem fio). As estações-bases estão ligadas entre si por uma rede fixa, permitindo o acesso indireto dos *host* móveis a toda estrutura de rede. O *host* móvel pode se deslocar fisicamente dentro de uma *célula* (área de abrangência da comunicação com a estação-base), a qual pode variar de dimensões: pico, micro, macro, e entre as células.

As **Redes *Ad-Hoc*** são compostas exclusivamente de *host* móveis, formando um cenário dinâmico, sem o suporte de uma rede fixa. A topologia da rede é altamente variável, constituída a partir das intersecções das áreas de abrangência (células) dos *host* móveis. A tecnologia para este tipo está começando a ser disponibilizada, com o protocolo *BlueTooth* (http://www.bluetooth.com).

Para o desenvolvimento de aplicações móveis distribuídas no ISAM, considera-se que os *hosts* móveis devam usufruir da infra-estrutura de rede fixa existente, beneficiando-se de ambientes como o oferecido pela Internet. Este modelo é refletido nos elementos básicos do ambiente de execução do sistema ISAM[3] (figura 1):

- HoloBase - é o ponto de contato do *host* móvel com os serviços ISAM residentes na parte fixa da rede. Possui as funções de identificação, autenticação e de ativação das ações básicas do sistema;
- HoloCélula - denota a área de atuação de uma HoloBase, e é composta pela mesma e por HoloSítios;
- HoloSítio - são os nodos do sistema responsáveis pela execução da aplicação móvel distribuída propriamente dita. Nestes também processam serviços de gerenciamento ISAM;
- HoloSítioMóvel - são os nodos móveis do sistema. Análogos aos HoloSítios, atuam na execução das aplicações e em algumas funções de monitoramento de recursos, conforme seu poder computacional. Os de menor porte tratam somente da interação com o usuário;
- HoloHome - é um ponto de referência único por usuário móvel no âmbito de toda rede. Está associado a um HoloSítio registrado para tal na arquitetura.

2.3 Características do Ambiente

A computação móvel é caracterizada por três propriedades: **portabilidade**, **mobilidade**, e **conectividade** [2], que introduzem restrições no ambiente. Para ser portátil, um computador deve ser pequeno, leve e requer fontes pequenas de energia. Isto significa que um computador portátil tem restrições no tamanho de memória, na capacidade de armazenamento, no consumo de energia e na interface do usuário. Além disso, a portabilidade potencializa o risco de perda, queda ou roubo. Quando em movimento, o dispositivo móvel pode alterar sua localização e, possivelmente, seu ponto de contato com a rede fixa. Essa natureza dinâmica do deslocamento introduz questões relativas ao endereçamento dos nós, localização do usuário e informações dependentes da localização. Além da mobilidade física, a aplicação com seu código, dados e estado também pode se deslocar entre os nós da rede. A conexão à rede através do meio sem fio levanta outros obstáculos: comunicação intermitente (desconexões freqüentes, bloqueio no caminho do sinal, ruído), restrita (e altamente variável) largura da banda, alta latência e alta taxa de erros.

[3] O prefixo *Holo* é uma referência ao Holoparadigma [7] adotado para a modelagem do ambiente.

Outro aspecto importante a considerar no projeto de aplicações móveis é o grau de conectividade à rede. Estudos, a partir dos sistemas de arquivos, têm demonstrado que existem basicamente três **modos de operação** de um sistema móvel [2]: fortemente conectado – conexão sobre uma rede fixa, rápida e confiável; fracamente conectado – conexão sobre um canal sem fio com largura de banda restrita; e desconectado – sem conexão à rede. O **modo desconectado é eletivo**[4] e utilizado, principalmente, para economia de recursos do *host* móvel, como o consumo de energia. Observa-se que o *host* móvel estará a maior parte do tempo desconectado da rede, sendo a desconexão um modo normal no ambiente móvel, enquanto que no ambiente distribuído é uma exceção. Desta forma, o sistema deve prover uma "ilusão" de conexão para o usuário móvel.

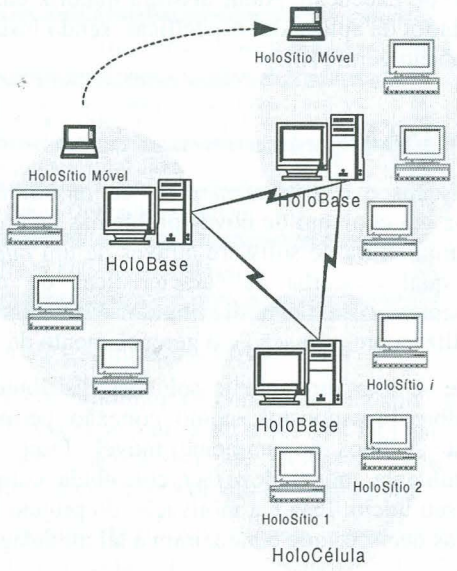

Figura 1 – Componentes do Ambiente ISAM

Três outros aspectos existentes na distribuição assumem uma maior relevância quando a mobilidade está presente. São eles: **escalabilidade,** refere-se ao tamanho do conjunto potencial de usuários; **heterogeneidade**, introduzida por diferentes equipamentos e redes móveis; e **dinamismo** introduzido pela alta variabilidade da disponibilidade de

[4] Termo introduzido por Dan Duchamp [14], significando que o *host* pode informar antecipadamente ao sistema que ocorrerá a desconexão, e este pode executar um "protocolo de desconexão".

recursos e pela mobilidade do usuário. Esses aspectos não são isolados, e devem ser abordados na arquitetura do sistema móvel.

Como visto, as restrições são da natureza da mobilidade e colocam novas demandas no projeto de *middlewares* [12] [15] [17] As aplicações móveis devem ser mais flexíveis que as atuais aplicações distribuídas, quando um recurso está indisponível/inacessível ou tem seu nível de disponibilidade/acessibilidade reduzido. Para serem efetivos e apresentarem um desempenho compatível com a expectativa do usuário, esses sistemas exigem a capacidade de **adaptação** às freqüentes e rápidas alterações no ambiente de execução durante o curso de evolução da aplicação. As soluções *middlewares* para ambientes distribuídos tradicionais (redes fixas) não são apropriadas, pois não permitem uma adaptação dinâmica em face às alterações do contexto de execução. Além disso, a maioria das soluções *middleware* atuais satisfazem as necessidades de aplicações específicas, sendo insuficientes para serem reusadas em aplicações de propósito geral [10].

3 O *Middleware* ISAM

Pelas características e restrições naturais da mobilidade, vê-se que a mobilidade lógica e física introduz um conjunto de novos problemas na área de sistemas distribuídos. A organização de uma arquitetura de software através de um *middleware* parece adequada ao ambiente móvel, o qual exacerba as características de dinamismo, escalabilidade e heterogeneidade, presentes no ambiente distribuído. Além disso, a organização em camadas do *middleware* flexibiliza a programação e o gerenciamento da aplicação.

O problema de se usar diretamente soluções distribuídas é que esses *middlewares* foram construídos sobre pressupostos, como conexão permanente e disponibilidade de recursos, que não são válidos no ambiente móvel. Desta forma, argumenta-se que a arquitetura para o ambiente móvel deve ser concebida com premissas de mobilidade e adaptabilidade desde seu início. Esta é a motivação do projeto ISAM. A seguir, apresenta-se a arquitetura ISAM e as decisões que conduziram a tal modelagem.

3.1 A Arquitetura ISAM

O principal desafio no tratamento da adaptação é a sua complexidade. Desta forma, buscou-se modelar a arquitetura ISAM em módulos independentes, abordando cada um os vários aspectos envolvidos no comportamento adaptativo da aplicação. Neste momento do projeto, está-se tratando da questão do como executar a adaptação através do comportamento colaborativo entre aplicação e sistema.

A arquitetura proposta é organizada em camadas com níveis diferenciados de abstração e está direcionada para a busca da manutenibilidade da qualidade de serviços oferecida ao usuário móvel através do conceito de adaptação. No ISAM, o sistema se adapta para fornecer qualidade, enquanto que a aplicação se adapta para manter a qualidade dentro da expectativa do usuário móvel. Uma visão organizacional desta arquitetura é apresentada

na figura 2. Salientam-se dois pontos. Primeiro, a adaptação permeia todo o sistema, por isto está colocada em destaque na representação do ISAM. Segundo, o escalonador é o "core" da arquitetura. As decisões de adaptação, tanto da aplicação quanto do escalonador, são baseadas no perfil do comportamento de três entidades: o usuário móvel, a aplicação e o sistema em si, os quais compõem o contexto de execução.

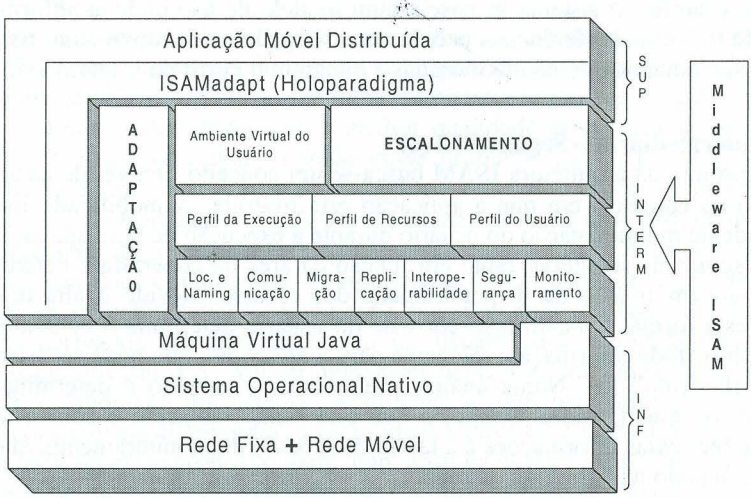

Figura 2 – Arquitetura ISAM

A camada superior (SUP) da arquitetura é composta pela aplicação móvel distribuída. A construção desta aplicação baseia-se nas abstrações do Holoparadigma [7], as quais permitem expressar mobilidade, acrescidas de novas abstrações para expressar adaptabilidade (ISAMadapt). Esta questão será objeto de futura publicação.

Por sua vez, a camada inferior (INF) é composta pelas tecnologias dos sistemas distribuídos existentes, tais como sistemas operacionais nativos e a Máquina Virtual Java. Além disso, a arquitetura ISAM foi concebida de forma a não excluir os mecanismos de adaptação nativos das tecnologias que compõem esta camada e, assumindo-se a existência de uma rede infra-estruturada (rede fixa + rede móvel) que forneça o suporte necessário para o acesso aos recursos da rede em escala global.

Camada Intermediária – Primeiro Nível

A camada intermediária (INTERM) é o núcleo funcional da arquitetura ISAM, sendo fornecida em três níveis de abstração. O primeiro nível é composto por dois módulos de serviço à aplicação: Escalonamento e Ambiente Virtual do Usuário. O escalonamento, por sua vez, é o componente-chave da adaptação na arquitetura ISAM. Sua funcionalidade é detalhada na seção 4. O Ambiente Virtual do Usuário (AVU) compõe-se dos elementos que integram a interface de interação do usuário móvel com o sistema. O modelo foi projetado para suportar a exploração de aplicações contextualizadas (adaptadas aos recursos, serviços e

localização corrente) e individualizadas (adaptadas aos interesses e preferências do usuário móvel). O desafio da adaptabilidade é suportar os usuários em diferentes localizações com diferentes sistemas de interação que demandam diferentes sistemas de apresentação dentro dos limites da mobilidade. Este módulo deve caracterizar, selecionar e apresentar as informações de acordo com as necessidades e o contexto em que o usuário se encontra. Para realizar estas tarefas, o sistema se baseia num modelo de uso onde as informações sobre o ambiente de trabalho, preferências, padrões de uso, padrões de movimento físico e hardware do usuário são dinamicamente monitoradas e integram o Perfil do Usuário e da Aplicação.

Camada Intermediária – Segundo Nível

No projeto da arquitetura ISAM busca-se um conceito flexível de adaptação que está relacionado ao **contexto** em que a aplicação está inserida. A mobilidade física introduz a possibilidade de movimentação do usuário durante a execução de uma aplicação. Podendo os recursos disponíveis se alterar, tanto em função da área de cobertura e heterogeneidade das redes, quanto em função da disponibilidade dos recursos devido à alta dinamicidade do sistema. Desta forma, a localização corrente do usuário determina o contexto de execução, definido como "toda informação, relevante para a aplicação, que pode ser usada para definir seu comportamento" [3]. Numa análise preliminar, o contexto é determinado através de informações de quem, onde, quando, o que está sendo realizado e com o que está sendo realizado. Obter essas informações é a tarefa do módulo de monitoramento, que atua tanto na parte móvel quanto na parte fixa da rede.

As informações que dirigem as decisões do escalonador e dão suporte à aplicação para sua decisão de adaptação (ISAMadapt) são advindas de três fontes: perfil da execução, perfil dos recursos e perfil do usuário e da aplicação.

O módulo monitoramento do ISAM obtém informações do acompanhamento das aplicações executadas pelo usuário, em um dado tempo e em um dado local, com determinados parâmetros. O que permite determinar a evolução histórica e quantitativa das entidades monitoradas. A interpretação destas informações estabelece o perfil do usuário e das aplicações. Desta forma, as aplicações móveis ISAM poderão se adaptar à dimensão pessoal, além das dimensões temporal e espacial presentes nos demais sistemas móveis [3].

Camada Intermediária – Terceiro Nível

No terceiro nível da camada intermediária estão os serviços básicos do ambiente de execução ISAM que provêm a funcionalidade necessária para o segundo nível e cobrem vários aspectos, tais como migração [28] – mecanismos para deslocar um ente de uma localização física para outra; replicação otimista [2] – mecanismo para aumentar a disponibilidade e o desempenho do acesso aos dados; localização e *naming* – para dar suporte ao movimento dos dispositivos móveis entre diferentes HoloCélulas, mantendo a execução durante o deslocamento. Está-se analisando a adequação do uso das redes Bayesianas e das Cadeias de Markov para processar as informações advindas do

monitoramento do ambiente. Alguns trabalhos em monitoração do ambiente estão em andamento, considerando quatro aspectos: informações da rede [27], informações dinâmicas da aplicação [1], informações estáticas da aplicação [5] e informações do sistema [28].

3.2 Decisões da Arquitetura

A adaptação é um processo disparado em resposta a uma situação ou evento externo, resultando na troca de um recurso ou por outro ou pela alteração na qualidade do serviço prestado. A adaptação faz o sistema se acomodar às alterações nos recursos disponíveis para ser capaz de continuar trabalhando com a melhor qualidade possível, em cada momento. Pesquisas recentes estão começando a tratar desse problema [6] [12] [20] [23], porém o fazem aplicados a um tipo de aplicação em específico, tais como processamento de imagens ou multimídia. A inovação deste *middleware* está em projetar uma arquitetura com **um tratamento uniforme da adaptação** e não comprometido com um domínio específico de aplicação. Isto é um fator complicador na arquitetura, porém justifica-se por se considerar que o potencial de aplicabilidade da computação móvel é amplo. Novos tipos de aplicações estão surgindo derivados do comportamento do usuário móvel (ainda não totalmente entendido/conhecido). Com isto, projetar aplicações móveis é uma tarefa difícil e esta deve ter uma ampla colaboração do sistema. Argumenta-se que para simplificar a tarefa de projetar aplicações móveis distribuídas, o programador deve ter à disposição uma linguagem para expressar aplicações de propósito geral, com abstrações para expressar a mobilidade e a adaptabilidade, e um ambiente de execução que lhe forneça mecanismos para especializar o comportamento móvel adaptativo ao domínio específico da aplicação.

Dimensões e Níveis de Adaptação

Pelo exposto, a adaptação ocorre como reação às variações no contexto de execução da aplicação. Desta forma, vê-se que as aplicações no ISAM podem se adaptar ao longo de três dimensões: **temporal** (quando ela ocorre), **espacial** (escolha dos recursos e serviços) e **pessoal** (preferências e comportamento individual do usuário). Por sua vez, em cada dimensão, pode-se pensar na adaptação em três níveis de abrangência: **recursos** - relativo às reações às mudanças em recursos físicos, como banda e latência da rede, alterando à qualidade da informação; **conteúdo** - relativo às alterações na semântica da aplicação, desde que esta pode alterar não somente o formato da informação, mas também seu conteúdo, como as *location-aware applications* [30]; **situação** - relativo às mudanças do comportamento do usuário final em face a um novo contexto. Por exemplo, a funcionalidade da aplicação em execução se altera quando o usuário está em determinado local: escritório, casa ou carro [18]. Nos sistemas móveis atuais somente o nível básico, de recursos, é normalmente presente nas arquiteturas de software analisadas [3].

Transparência x Consciência da Mobilidade

Uma grande diferença entre sistemas distribuídos e sistemas móveis é que os primeiros procuram fornecer transparência da distribuição ao usuário, enquanto que nos segundos um certo grau de consciência da mobilidade (localização, contexto) é importante para o comportamento adaptativo. Assim, é necessário estabelecer um equilíbrio entre consciência e transparência (em geral, conflitantes) da mobilidade: o sistema deve fornecer informações sobre o ambiente requeridas pela aplicação para que esta possa reagir (adaptar-se) conforme suas necessidades. Os *middlewares* correntemente disponíveis não fornecem facilidades para controlar o grau de transparência requerido pela aplicação móvel. Também neste aspecto, a arquitetura ISAM se diferencia dos demais.

4 O Escalonamento no *Middleware* ISAM

No ISAM, as aplicações solicitam direta ou indiretamente recursos do escalonador. Algumas podem especificar uma determinada necessidade de qualidade de serviço (QoS), outras podem aceitar o "melhor-possível" nos níveis de serviço [22]. Assim, o módulo de escalonamento do *middleware*, tem a estratégia de trabalhar com diferentes políticas de gerenciamento para diferentes aplicações, usuários e/ou domínios de execução. Neste caso, as estratégias de adaptação exigem do mecanismo de escalonamento o tratamento de problemas de otimização utilizando critérios múltiplos [31].

Como forma de reduzir a complexidade do escalonamento, e prover comportamento adaptativo nas diferentes dimensões e níveis de adaptação (vide item 3.2), o ISAM adota a estratégia denominada **Adaptação Colaborativa Multinível**, onde:

- o sistema (escalonador) é responsável por determinadas adaptações, normalmente relativas ao desempenho e gerência de recursos do ambiente;
- a aplicação é responsável por decisões de adaptações específicas de seu domínio e situações de uso;
- ambos, são responsáveis por uma negociação de decisões de adaptação.

Neste sentido, um problema de pesquisa em andamento, é estabelecer o nível de cooperação requerido entre projetistas de sistemas e projetistas de aplicações para criar protocolos aceitáveis para ambos os grupos [4].

Por outro lado, o ISAM gerencia diversas aplicações que concorrem na utilização dos recursos. A otimização da execução de um subconjunto do total de aplicações não pode comprometer o nível mínimo de QoS necessário para o restante. Desta forma, o escalonador precisa trabalhar com uma visão global das execuções em andamento [29]. Nas próximas seções, serão descritos os princípios utilizados na sua modelagem.

4.1 Organização Física do Escalonamento

A forma como é organizada a distribuição dos equipamentos afeta diretamente todos os serviços do *middleware*, e naturalmente o escalonamento. A organização adotada no ISAM é a **celular hierárquica** (vide figura 1). Nesta, os equipamentos pertencentes a uma mesma célula comunicam-se diretamente (utilizando uma organização plana). Nas comunicações com o exterior um equipamento específico atua como fronteira. A característica hierárquica faculta que uma célula possa recursivamente conter outras.

Esta organização atende a necessidade de confinamento de contexto inerente ao modelo de programação adotado para o ISAM, o Holoparadigma [7]. Este modelo é baseado em eventos associados a escopos. Outrossim, a organização celular hierárquica é orgânica com o Holoparadigma, o qual trabalha com o conceito de entes (entidade de modelagem). O conceito de entes também contempla a possibilidade de agrupamento hierárquico. Tal associação se mostra oportuna ao mapeamento e/ou à alocação dinâmica de tarefas.

Em função das características do software e do hardware, o escalonamento no *middleware* ISAM utiliza uma organização fisicamente distribuída e cooperativa representada na figura 3. Como principais premissas passíveis de serem atingidas por esta organização tem-se: tolerância-a-falhas, escalabilidade, autonomia dos domínios administrativos (HoloCélulas) e suporte a múltiplas políticas de escalonamento.

A proposta está baseada em dois escalonadores: (*i*) **EscWAN** e (*ii*) **EscEnte**:

EscWAN: fica localizado no nodo HoloBase com atuação entre as HoloCélulas. O mesmo tem atribuições no gerenciamento global da arquitetura, tais como:

- localizar recursos (hardware e software) mais próximos, para reduzir custos de comunicação;
- decidir quando e onde replicar serviços e/ou componentes de software (entes);
- decidir quando e para onde migrar os componentes de software;
- instanciar o Ambiente Virtual do Usuário nos HoloSítios. Esta instanciação é feita sob duas óticas: (*i*) balanceamento de carga - neste caso é escolhido o nodo menos carregado, (*ii*) aspectos de afinidade da aplicação - exigência de memória, bases de dados, etc.;
- disponibilizar antecipadamente, por usuário, a demanda de componentes das aplicações e dos dados;
- repassar ao escalonador EscEnt a carga de trabalho (componentes de software) proveniente de outras HoloBases.

Pelas suas atribuições, além da consideração de custos de comunicação e balanceamento de carga, o escalonador EscWAN atua de forma intensiva sobre aspectos de replicação e migração.

EscEnte: também existente em todas as HoloBases, tem atribuições no gerenciamento interno da HoloCélula, tais como:

- efetuar o mapeamento dos componentes da aplicação nos HoloSítios da HoloCélula. Os critérios utilizados também são balanceamento de carga e de afinidade funcionais;
- dar suporte aos procedimentos de adaptação colaborativa multinível com a aplicação.

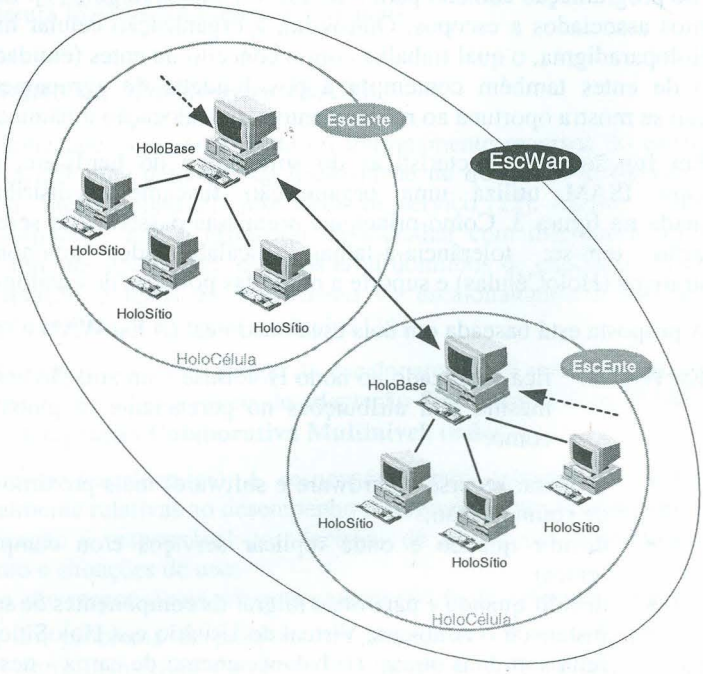

Figura 3 - Organização do Escalonamento

Uma estratégia do escalonador EscEnte é associar o contexto (a HoloCélula, as aplicações e os usuários) a grupos de escalonamento [11], onde cada grupo pode definir políticas especificas de balanceamento de carga. Cabe ao mecanismo de escalonamento gerir a evolução da execução das aplicações dos diferentes grupos segundo as políticas por eles selecionadas.

4.2 Características Operacionais Gerais

O suporte de escalonamento para a ISAM tem como meta ser flexível e extensível. Suas características mais significativas são:

- sua operação ocorre sobre o sistema operacional (escalonamento aplicativo), e sem exigir alteração do mesmo. Isto potencializa a portabilidade;
- suporte a operações concorrentes e distribuídas sobre os componentes das aplicações móveis;
- não está comprometido com uma heurística de escalonamento em particular. Ao contrário, disponibiliza facilidades para que novas heurísticas sejam implementadas;
- a heurística a ser utilizada é selecionada e/ou contextualizada por usuário e aplicação;
- os componentes que tomam decisão são replicados, e são capazes de atividades autônomas e assíncronas;
- as metas de escalonamento são perseguidas em escopos. Cada componente que toma decisão escalona serviços no seu domínio;
- uso intensivo de registro histórico como auxiliar na tomada de decisão.

Algumas dessas características são típicas de propostas de balanceamento de carga difusas [13], voltadas para sistemas com aplicações de elevada dinamicidade de execução como no caso do ISAM.

4.3 Principais Estratégias Utilizadas

No ISAM, a adaptação permeia todas as decisões da arquitetura. O escalonador é tanto adaptador quanto adaptativo, ou seja, o escalonador é responsável pela execução do comportamento adaptativo da arquitetura, e ele próprio se adapta – altera-se conforme o ambiente corrente. Além disso, a adaptação ISAM é colaborativa, ou seja, a aplicação e o escalonador interagem quando da decisão de adaptação. Esta colaboração ocorre em níveis diferenciados, dependendo do domínio da aplicação. As principais estratégias utilizadas pelo *middleware* ISAM para o escalonamento são [31]:

Aprendizado por reforço: à medida que o usuário interage com o sistema, seu comportamento é monitorado e seu perfil é construído. O escalonador emprega uma abordagem estocástica com aprendizado por reforço, na qual são construídas correlações estatísticas entre o usuário, o comportamento das suas aplicações e o ambiente de execução.

Instanciação otimizada das aplicações: a premissa é buscar um modelo WYNIWYG (*What You Need Is What You Get*) [19]. O escalonador carrega nos HoloSítios (móveis ou fixos) um conjunto mínimo de componentes de software que garantam a execução da aplicação (valendo-se do perfil do usuário). Os outros componentes, se necessários, serão solicitados sob demanda, caracterizando uma **estratégia *pull*** de operação [2].

Instanciação antecipada das aplicações: o processo de instanciação começa no momento em que o usuário efetiva sua autenticação na HoloBase, antes de solicitar a execução de aplicações. Neste caso, adota-se caso uma **estratégia *push*** [2] de disseminação de componentes de software e informação. Esta instanciação também pode ocorrer com uma antecipação ainda maior, tendo por referência uma expectativa de roteiro de mobilidade do usuário já consolidada. Antecipar o tráfego na parte estruturada da rede (com conexão física) é uma opção da arquitetura proposta para aumentar o desempenho global da aplicação móvel, e conseqüentemente reduzir o tempo de espera/conexão do usuário do segmento de rede com suporte à mobilidade (conexão sem fio).

Monitoração em dois níveis: o escalonamento é alimentado por métricas oriundas de dois níveis distintos:

no nível de sistema têm-se dois tipos principais de métricas: (*i*) uma para caracterização do *workload* dos *hosts* e (*ii*) outra para construção de perfis de comunicação entre objetos. Os mecanismos para capturar os perfis de comunicação entre os objetos são integrados com a API RMI de Java, preservando a compatibilidade com a semântica nativa da linguagem [28]. No ISAM, as abstrações do Holoparadigma serão mapeadas em Java [8];

no nível de aplicação é feita uma monitoração utilizando a JVMPI (*Java Virtual Machine Profiler Interface*), a qual oferece a possibilidade de uma seleção dinâmica dos eventos de interesse da aplicação (por exemplo, ativação, interrupção e tempo de espera de métodos) que devem ser monitorados. No ISAM esta monitoração não exige cuidados de programação, e os eventos a serem monitorados podem ser individualmente ativados e/ou desativados para redução de *overheads* desnecessários [1].

Tratamento dos dados monitorados: a tomada das decisões de escalonamento e conseqüentemente de adaptação, está associada à propriedade de agilidade do sistema [23]. Esta é uma noção complexa no ISAM, pois as aplicações em execução no momento podem ter diferentes sensibilidades a diferentes recursos. Por exemplo, uma aplicação pode ser sensível a alterações na banda, enquanto que outra não.

Está sendo avaliado o uso simultâneo de duas técnicas para tratar esta diversidade de requisitos na monitoração. A primeira atua no nível mais baixo do processo de decisão, utilizando um modelo Bayesiano. Este modelo probabilístico atua como um filtro sobre os dados monitorados, com o objetivo de permitir uma diferenciação entre flutuações, e reais transições de estado da arquitetura. Seu objetivo é aumentar a estabilidade do sistema, reduzindo o *overhead* total com operações de escalonamento desnecessárias.

A segunda utiliza um modelo de previsão baseado nos estados de uma cadeia de Markov. Este modelo opera no nível mais alto do processo de decisão, sendo projetado para acompanhar o desempenho global da aplicação através da monitoração de métricas do desempenho. Utilizando dados de transição de estado, este modelo é capaz de prever o estado futuro do sistema e reagir a potenciais degradações no desempenho do mesmo.

5 Trabalhos Relacionados

Os sistemas móveis, em geral, fornecem adaptação de forma específica a um domínio de aplicações. Em muitos desses sistemas, a adaptação diz respeito ao uso de técnicas de redução, transformação, ou filtragem dos dados para trafegarem na rede [3] ativadas pela alteração na largura da banda, ou pelo dispositivo display em uso. A localização dos componentes é fixa (sistemas estáticos). A adaptação é alcançada pela troca do modo como a rede é usada. Já Sumatra [25] propõe o uso de execução remota para reduzir a comunicação entre máquinas móveis e a rede estática. Nesta, toda comunicação e migração acontece sob controle da aplicação. Isto facilita explorar diferentes políticas para monitoramento de recursos e adaptação à variação desses recursos; porém, essa exploração é responsabilidade do programador, o que restringe a utilidade da proposta. Diferente do ISAM, estas soluções cobrem somente uma pequena faixa das possibilidades de adaptação e tornam-se restritas para expressar a flexibilidade exigida pelos novos tipos de aplicações que são sensíveis ao contexto em que estão inseridas.

O ambiente de execução necessário ao ISAM é caracterizado por uma grande abrangência física (*wide-area*), podendo atingir uma escalabilidade no nível de Internet. Esta configuração de *network computing* tem forte semelhança com um ambiente tipo *Grid* [16]. Nos últimos anos, a pesquisa em *Grid Computing* vem se intensificando, porém diferentemente do ISAM, os trabalhos desenvolvidos não consideram a mobilidade física dos equipamentos.

No que diz respeito ao software, em *Grid Computing* destacam-se atualmente duas naturezas: *Grids Computacionais* - seu principal objetivo é reduzir o tempo necessário para execução das aplicações [16]; e *Grids de Dados* - constituído pelos sistemas que compõem informações a partir de repositórios de dados distribuídos [24]. As atividades de pesquisa dedicadas a *Grid Computing* focalizam uma das naturezas citadas, e dentro desta uma aplicação em particular, por exemplo; *Grid*Gene (Pesquisa Genômica no *Grid* Computacional), GriPhyN (*Grid Physics Network*) e PPDG (*Particle Physics Data Grid*) dentre outros [9]. O desenvolvimento do *middleware* nestes trabalhos, apesar de contemplar alguns esforços na mesma direção do ISAM, tem vários aspectos diferentes em função da especificidade das suas aplicações-alvo. ISAM se direciona a uma nova perspectiva: *Grid Colaborativo*.

6 Conclusões

As restrições naturais do ambiente móvel colocam novos desafios para os projetistas de aplicações, e exigem novas tecnologias para que as aplicações sejam úteis em sistemas com recursos limitados. Sente-se a necessidade de sistemas mais flexíveis que dividam a responsabilidade entre o projetista da aplicação e o sistema de suporte (*middleware*) para fornecer o comportamento dinâmico e adaptativo que a aplicação requer. O projeto ISAM, diferente dos demais, tem a premissa de abordar a adaptação de maneira uniforme,

fornecendo-a nas dimensões temporal, espacial e pessoal. Outro ponto-chave da arquitetura é o comportamento colaborativo da adaptação. Tanto o sistema quanto a aplicação colaboram na decisão de adaptação. Esta abordagem permite uma flexibilização maior na modelagem das aplicações móveis. Argumenta-se que, para simplificar a implementação de aplicações móveis, o sistema de suporte deve ser modelado sob uma ótica de propósito geral, fornecendo mecanismos e abstrações para a implementação de visões específicas das aplicações. A arquitetura proposta é ampla, e apresenta muitos desafios. Este artigo abordou a questão da modelagem do *middleware* considerando mobilidade, flexibilidade e adaptação. Vários outros aspectos estão em andamento, os quais abordam a questão de linguagem, ambiente de execução, monitoramento e integração com tecnologias existentes.

Referências

[1] Araujo, E. B.; Augustin, I; Yamin, A.; Silva, L.; Geyer, C.F.R. Uma Proposta de Monitoração para Visualização de Aplicações Distribuídas Java. JORNADAS CHILENAS DE COMPUTACIÓN 2001. *V WORKSHOP EN SISTEMAS DISTRIBUÍDOS Y PARALELISMO*. Chile. 5-9 Nov. 2001.

[2] Augustin, I. Acesso aos Dados no Contexto da Computação Móvel. PPGC/UFRGS. Porto Alegre. Dez. 2000 (Exame de Qualificação).

[3] Augustin, I.; Yamin, A.; Barbosa, J.; Geyer, C.F.R. Requisitos para o Projeto de Aplicações Móveis Distribuídas. *VIII CACIC CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN*. Argentina. Oct, 2001.

[4] Augustin, I.; Yamin, A.; Barbosa, J.; Geyer, C.F.R. Towards a Taxonomy for Mobile Applications with Adaptive Behavior. *INTERNATIONAL SYMPOSIUM ON PARALLEL AND DISTRIBUTED COMPUTING AND NETWORKS*. PDCN 2002.

[5] Azevedo, S.; Vargas, P.; Barbosa, J. .; Yamin, A.; Geyer, C.F.R. DEPAnalyzer: um analisador estático de dependências para programas Java. In: *WORKSHOP EM SISTEMAS COMPUTACIONAIS DE ALTO DESEMPENHO*, II, WSCAD 2001. Anais . Pirenópolis, Go, 10-12, set., 2001.

[6] Baggio, A. Adaptable and Mobile-aware Distributed Objects. PHD THESIS. Université Pierre & Marie Curie, Paris VI. June. 1999.

[7] Barbosa, J.; Geyer, C.F.R. Integrating Logic Blackboards and Multiple Paradigm for Distributed Software Development. Proceedings of *INTERNATIONAL CONFERENCE ON PARALLEL AND DISTRIBUTED PROCESSING TECHNIQUES AND APPLICATIONS (PDPTA)*. Jun, 2001.

[8] Barbosa, J.; Yamin, A.; Vargas, P.; Ferrari, D.; Schaeffer, E.; Geyer, C.F.R. Using Mobility and Blackboards to Support a Multiparadigm Model Oriented to Distributed Processing. In: *SIMPÓSIO BRASILEIRO DE ARQUITETURAS DE COMPUTADORES E PPROCESSAMENTO DE ALTO DESEMPENHO*, SBAC PAD 2001. Proceedings ... Pirenópolis, Go, 10-12, Set., 2001.

[9] Buyya, R. et al. Architectural Models for Resource Management in the *Grid*. In *1ST. IEEE/ACM INTERNATIONAL WORKSHOP ON GRID COMPUTING. GRID* 2000. Dec. 2000. Disponível em http://citeseer.nj.nec.com/321320.html. Acesso em outubro de 2001.

[10] Capra, L.; Emmerich, W.; Mascolo, C. Reflective *Middleware* Solutions for Context-Aware Applications. UCL-CS Research. Note RN/01/12. University College London, Dept. of Computer Science. march, 2001. (submitted for publication)

[11] Cavalheiro, G. G. H.; Denneulin, Y.; Roch, J.-L. A General Modular Specification for Distributed Schedulers. In *PROCEEDINGS OF EUROPAR'98*. Southampton, Springer Verlag, LNCS 980. 1998.

[12] Corradi, A .; Bellavista, P.; Stefanelli, C. Mobile Agent *Middleware* for Mobile Computing. *IEEE COMPUTER*. March, 2001.

[13] Corradi, A; Letizia, Li; Zambonelli, Franco. Diffuse Load-Balancing Policies for Dynamic Applications. *IEEE CONCURRENCY*. New York, v7, n.1. 1999.

[14] Duchamp, D; Feiner, S; Maguire, G. Q. Software Technology for Wireless Mobile Computing, *IEEE NETWORK*, (5):6, pp. 12-18, November 1991.

[15] Emmerich, W. Software Engineering and *Middleware*: a Roadmap. In: The Future of Software Engineering. *22nd INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING* (ICSE2000). ACM Press. May, 2000.

[16] Foster, I.; Kesselman, C. Globus: A METACOMPUTING INFRASTRUCTURE TOOLKIT. IN *INTERNATIONAL JOURNAL OF SUPERCOMPUTING APPLICATIONS*, 11(2), 1997.

[17] Geihs, K. *Middleware* Challenges Ahead. *IEEE COMPUTER*. June. 2001.

[18] Harter, A. et al. The Anatomy of a Context-aware Application. Proceedings of 5th *INTERNATIONAL CONFERENCE ON MOBILE COMPUTING AND NETWORKING* (MOBICOM'99). Seattle, USA. Aug, 1999.

[19] Kon, F. et al. 2K: Distributed Operating System for Dynamic Heterogeneous Environments. Proceedings of the *NINTH IEEE INTERNATIONAL SYMPOSIUM ON HIGH PERFORMANCE DISTRIBUTED COMPUTING* - HPDC'00. Pennsylvania, USA. 2000.

[20] Kunz, T.; Black, J.P. An Architecture for Adaptive Mobile Applications. Proceedings 11th *INTERNATIONAL CONFERENCE ON WIRELESS COMMUNICATIONS*. Alberta, Canada. Jul. 1999.

[21] Loureiro, A.F.; Matheus, G.R. Introdução à Computação Móvel. Tutorial. *11ª ESCOLA DE COMPUTAÇÃO*, Rio de Janeiro, Brasil, 1998.

[22] Maheswaran, M. Quality of Service Driven Resource Management Algorithms for Network Computing. In *THE INTERNATIONAL CONFERENCE ON PARALLEL AND DISTRIBUTED PROCESSING TECHNIQUES AND APPLICATIONS*. PDPTA, June 1999.

[23] Noble, B. System Support for Mobile, Adaptive Applications. *IEEE PERSONAL COMPUTING SYSTEMS*. v.7,n.1,p. 44-9, Feb. 2000.

[24] Oldfield, R. Summary of Existing and Developing Data *Grids*. September, 2001. Disponível em www.cs.dartmouth.edu/~raoldfi/papers/data*Grids*.pdf.

[25] Ranganathan, M.; Acharya, A.; Saltz, J. Sumatra: a Language for Resource-aware Mobile Programs. In *MOBILE OBJECTS SYSTEMS: TOWARDS THE PROGRAMMABLE INTERNET*: Springer-Verlag Publisher, Serie Lecture Notes on Computer Science. v.1222. Apr. 1997.

[26] Satyanarayanan, M. Fundamental Challenges in Mobile Computing. Proceedings of *15th ACM SYMP. ON PRINCIPLES OF DIST. COMPUTING*. 1996.

[27] Silva Jr, E.N.; Augustin, I., Yamin, A.; Barbosa, J.; Geyer, C.F.R. Hierarquia de Gerenciamento de Redes com Componentes Móveis VIII CACIC *CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN*. Santa Cruz, Argentina. Oct, 2001.

[28] Silva, L.; Yamin, A.; Augustin, I.; Barbosa, J.; Geyer, C.F.R. Mecanismos de Suporte ao Escalonamento em Sistemas com Objetos Distribuídos Java. *VIII CACIC CONGRESO ARGENTINO DE CIENCIAS DE LA COMPUTACIÓN*. Santa Cruz, Argentina. Oct, 2001.

[29] Vahdat, A. Toward Wide-Area Resource Allocation. In The International Conference on Parallel and Distributed Processing Techniques and Applications. PDPTA 1999, June 1999. p.930-936.

[30] Want, R. and Schilit, B. Editors. *IEEE COMPUTER*. Special Issue on Location-aware Computing. V. 34, n.8. August, 2001.

[31] Yamin, A.; Augustin, I.; Barbosa, J.; Silva, L.; Geyer, C.F.R. Explorando o Escalonamento no Desempenho de Aplicações Móveis Distribuídas. In: *WORKSHOP EM SISTEMAS COMPUTACIONAIS DE ALTO DESEMPENHO*, II, WSCAD 2001. Anais . Pirenópolis, set., 2001.