

## ADAPTAÇÃO DE TRANSAÇÕES PARA SISTEMAS DE COMPUTAÇÃO MÓVEL

### TRANSACTIONS ADAPTATION FOR MOBILE COMPUTING SYSTEMS

**Daniela Eloise Flor<sup>1,2,\*</sup>, Maurício Gonçalves Vieira Ferreira<sup>1</sup>, Nilson Sant'Anna<sup>1</sup>, Ricardo Paulino<sup>2</sup>, Yandre Maldonado e Gomes da Costa<sup>3</sup>, Munif Gebara Junior<sup>1</sup>**

**Resumo:** Características inerentes ao contexto da computação móvel impactam em transações de banco de dados. Fatores como a instabilidade do meio sem fio, a heterogeneidade de recursos de computação dos dispositivos móveis e a inflexibilidade dos modelos transacionais propostos na literatura, culminaram na motivação para esta pesquisa. Este trabalho propõe uma acomodação harmoniosa entre aplicações que processam transações e o dinamismo do ambiente de computação móvel através do modelo de transações móveis adaptadas ao contexto e o MiD-Mobile. Tal união viabiliza a adaptação colaborativa provendo sensibilidade ao contexto, êxito nas transações e uso consciente de recursos. Um estudo de caso legitima as propostas apresentadas.

**Palavras-chave:** transações móveis. adaptação de transações.

**Abstract:** The mobile computation context has special characteristics and these characteristics lead to challenger researches, including database transactions. Factors as the dynamism of the wireless environment, the lack of the mobile devices computations resources and the inflexibility of the proposed transactional model in literature, carry out the motivation for this research. This research presents an accordant arrangement among applications that process transactions and dynamic specific of mobile computation environment through the context adaptive mobile transaction model and MiD-Mobile. A case study legitimates the proposals proposed in this research.

**Key words:** mobile transactions. transactions adaptation.

## 1. Introdução

A utilização de dispositivos móveis no acesso as informações é uma realidade. A qualquer momento, de qualquer lugar, usuários assistem a vídeos, recebem e-mails, consultam dados remotos, entre outras tarefas. O cenário descrito é o da computação móvel, resultado da adição da portabilidade e comunicação sem fio à dispositivos com capacidade de processamento e armazenamento.

As possibilidades para Sistemas de Computação Móveis (SCM) são muitas, mas os desafios também. A portabilidade veio com a miniaturização dos computadores, mas resultou em dispositivos com restrição de recursos. A comunicação sem fio apesar de oferecer liberdade, não garante a mesma qualidade de serviço que as redes com fio. A mobilidade dos usuários acrescenta novas questões como gerência de localização.

Devido às várias possibilidades de negócios que podem usufruir das características da computação móvel, é grande o interesse em sistemas de informação que processem transações nesse contexto, mas a complexidade no desenvolvimento de aplicações fortemente sujeitas a requisitos não funcionais maximiza potencialmente as falhas. A manipulação de dados em equipamentos temporariamente desconectados da rede fixa exige uma revisão nas técnicas tradicionais de gerenciamento de dados.

Em um levantamento bibliográfico inicial, vários modelos transacionais foram vistos, mas a singularidade de propósito identificada inviabiliza sua aplicação a uma diversidade maior

<sup>1</sup> Instituto Nacional de Pesquisas Espaciais (INPE).

<sup>2</sup> Universidade Paranaense (UNIPAR) - Campus Paranavaí.

<sup>3</sup> Departamento de Informática (DIN) - Universidade Estadual de Maringá (UEM).

\* Autora correspondente: Av. Huberto Brüning, 360 - Jardim Santos Dumont - Zona Leste - CEP 87706-490 - Paranavaí - Paraná - Brasil.

de domínios de negócio. Diante da inflexibilidade e incompletude dos modelos a busca por um formato de processamento mais maleável e de maior abrangência encontrou apoio na adaptação colaborativa.

Esta pesquisa trata os aspectos que influenciam o gerenciamento de transações em SCM, sumariza os modelos transacionais encontrados na literatura, detalha a proposta de um modelo para transação móvel e por fim a arquitetura de um middleware de suporte à adaptação das transações adequadas ao modelo proposto.

A seção introdutória é seguida por outra que oferece um panorama sobre os modelos transacionais encontrados na literatura. Da seção 3 constam os modelos que aplicam a estratégia de adaptação em transações móveis. Os detalhes da arquitetura do MiD-Mobile e do modelo das transações móveis adaptadas ao contexto estão nas seções 4 e 5. A seção 6 apresenta o estudo de caso e a sete exhibe as conclusões.

## 2. Transações em Ambiente de Computação Móvel

Em (Ferreira e Finger, 2000), os modelos de transações para ambiente de computação móvel são identificados como abstrações de alto nível de operações de leitura/escrita em dados compartilhados e distribuídos entre estações fixas e móveis. Genericamente, é desejável que transações móveis sejam capazes de suportar conexões intermitentes, atualizar seu deslocamento em função do movimento, manter transações de longa duração devido à invariabilidade do meio, fornecer autonomia em desconexões e reintegrar atualizações locais para manter a consistência dos dados.

Diante da pluralidade de exigências impostas pelo meio, vários modelos foram propostos. Os que mais se destacam são o *Isolation-Only Transactions* (IOT) tratado em (Lu e Satyanarayanan, 1984), o *Two-tier Replication* sugerido por (Gray et al., 1996), *Clustering* proposto por (Pitoura e Bhargava, 1995), *Pro-motion* explicado por (Walborn e Chrysanthis, 1998), *Moflex* tratado em (Ku e Kim, 2000) e *Kangaroo* por (Dunham et al., 1997). Uma revisão completa é apresentada em (Serrano-Alvarado et al., 2004).

Por razões de espaço, os modelos são apenas agrupados em suas características determinantes. (Serrano-Alvarado et al., 2004) consideram o local de execução das transações como critério de classificação, a tabela 1 sumariza tal proposta.

Local de processamento	Modelo
A transação parte da estação móvel mas é executada na rede fixa	<i>Kangaroo</i>
	<i>Pré-serialization</i>
	<i>MDSTPM</i>
A transação é executada na estação móvel e o resultado é reintegrado na rede fixa	<i>HiCoMo</i>
	<i>Pro-motion</i>
	<i>Clustering</i>
	<i>Two-tier replication</i>
	<i>IOT e Prewrite</i>
O processamento é distribuído entre estação móvel e rede fixa	<i>Semantics-based</i>
	<i>Reporting</i>
	<i>Clustering</i>
	<i>Two-tier replication</i>

**Tabela 1.** Modelos agrupados pelo local de processamento da transação.

Outra abordagem para agrupar os modelos foi usada por (Rocha e Toledo, 2005), sendo considerados os modelos que suportam desconexões, variação de largura de banda, limitações dos suprimentos de energia, escassez de espaço em disco e mobilidade. A tabela 2 apresenta os modelos e as técnicas usadas em cada situação.

<b>Característica</b>	<b>Técnica</b>	<b>Modelo</b>
Desconexão	Caching de dados	<i>HiCoMo, Pro-motion, Clustering, IOT, Prewrite, Reporting, Clustering e Two-tier replication</i>
Largura de banda	Transferência total de dados Transferência parcial de dados	<i>IOT e Pro-motion Semantics-based</i>
Consumo de energia	Divisão da computação Modo de operação desconectado Estação móvel inativa em períodos ociosos	<i>Reporting Semantics-based Kangaroo</i>
Restrição das estações móveis	Transferência de pequenos fragmentos Computação distribuída	<i>Semantics-based Pro-motion Reporting</i>
Mobilidade	Subdivisão da transação	<i>Kangaroo e MDSTPM</i>

**Tabela 2.** Modelos agrupados pelas variáveis do contexto do ambiente móvel

A visão proposta por (Côrtes, 2004) usa como critério de agrupamento a unicidade de execução do processamento, o isolamento diante de processamento concorrente, a consistência e a durabilidade das transações, características conhecidas por propriedades ACID. Os modelos acabam por optar pela flexibilização de uma ou mais propriedades como forma de viabilizar a transação móvel, conforme a tabela 3.

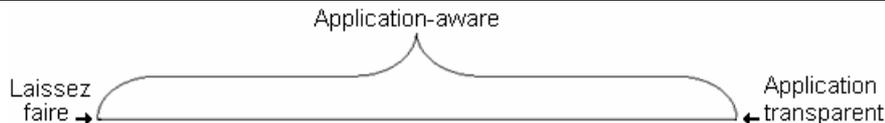
<b>Propriedade ACID</b>	<b>Estratégia de flexibilização</b>	<b>Modelo</b>
Atomicidade	Transações aninhadas	<i>Pro-motion e Reporting Kangaroo e HiCoMo</i>
Consistência	Formas de acesso aos dados	<i>Pro-motion e Semantics-based</i>
	Regras de consistência	<i>Moflex e HiCoMo Two-tier replication</i>
Isolamento	Liberação de resultados parciais	<i>IOT, Reporting e HiCoMo Pré-serialization Semantics-based</i>
	Níveis de compartilhamento	<i>Pro-Motion</i>
Durabilidade	Mecanismos de efetivação	<i>HiCoMo</i>

**Tabela 3.** Modelos agrupados pelo tratamento das propriedades ACID.

Os modelos de transações citados possuem características singulares e são inflexíveis para domínios de aplicações diferentes daqueles para os quais foram criados. Por essa razão é interessante que existam estratégias diferenciadas de atuação de uma transação, visando o aumento das chances de êxito das mesmas. A atuação diversificada é possível através da adaptação, assunto tratado a seguir.

### 3. Adaptação de Transações Móveis

As estratégias de adaptação, segundo (Satyanarayanan, 1996), podem ocorrer sob os dois eixos identificados na figura 1. Em um deles, na abordagem denominada *Laissez-faire*, a adaptação é responsabilidade das aplicações, dispensando a necessidade de um sistema de suporte. Neste caso, cabe a cada aplicação tratar convenientemente os recursos disponíveis no ambiente.



**Figura 1.** Eixos de adaptação (extraído de Satyanarayanan, 1996)

No outro extremo está a *Application-transparent*, onde a responsabilidade pela adaptação é do sistema. Entre estes dois extremos existe a possibilidade de uma adaptação colaborativa chamada de *Application-aware*. Neste caso, as aplicações definem a forma de adaptação que mais lhe interessam no momento, cabendo ao sistema monitorar e alocar recursos dinâmicos.

Segundo (Coulouris et al., 2007), adaptar significa modelar soluções computacionais em função de alguns fatores. Sendo assim, antes de tratar a adaptação das transações é necessário identificar quais são os fatores que influenciam nos resultados. A tabela 4 apresenta o resultado desse levantamento.

Fatores	Influência
Dispositivos portáteis	Restrição do tamanho dos dados disponíveis localmente Heterogeneidade de recursos de computação Limitação do trabalho autônomo devido a restrição de energia
Comunicação sem fio	Variabilidade da largura de banda e Indisponibilidade de conexão Heterogeneidade de tecnologias de comunicação sem fio Intensidade insatisfatória do sinal de comunicação
Mobilidade	Movimento do usuário Indisponibilidade de comunicação momentânea (área de sombra)

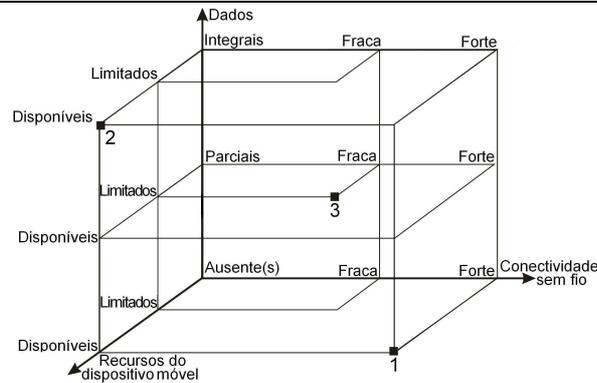
**Tabela 4.** Fatores que influenciam a execução de transações móveis.

Após a verificação das possíveis interferências a um sistema de computação móvel se faz oportuna a proposta de um middleware, que encapsule funções de monitoramento e suporte à adaptação das transações, visando a diminuição do cancelamento das mesmas. Estratégias de adaptação são exploradas em diferentes áreas da ciência da computação, inclusive em banco de dados.

Nesse aspecto dois trabalhos correlatos se destacam o STM e a PGTA. O STM (Serviço de Transações Móveis) foi proposto por (Serrano-Alvarado, 2004) para adaptar transações a partir da avaliação da configuração do dispositivo móvel e da rede sem fio. A adaptação colaborativa entre as transações e a Plataforma de Gerenciamento de Transações Adaptáveis (PGTA) é proposta por (Rocha e Toledo, 2007). A plataforma monitora o nível dos recursos e às transações cabe reagir às mudanças do ambiente.

Nesta pesquisa a adaptação também é colaborativa e é apoiada pelo modelo de transações móveis adaptadas ao contexto e pelo middleware denominado MiD-Mobile. As maiores contribuições dizem respeito à manutenção das características tradicionais das propriedades ACID e o acréscimo da aprendizagem na etapa de adaptação.

Em sua atuação, o middleware considera três dimensões: a conectividade sem fio, os dados do dispositivo móvel e os recursos de computação do mesmo. A figura 2 apresenta as dimensões avaliadas para a indicação da adaptação e exemplifica algumas ocorrências onde é preciso decidir como executar uma transação.



**Figura 2.** Dimensões consideradas para a adaptação das transações

O ponto 1 representa uma situação em que o dispositivo móvel apesar de apto a exercer processamento e de usufruir de boa conectividade, não executa a transação por não ter dados locais, passando a execução para a rede fixa. O segundo ponto exemplifica o processamento local da transação, quando há recursos e dados disponíveis na estação móvel, mas a conectividade deixa a desejar.

O ponto 3 ilustra o processamento da transação distribuído entre dispositivo móvel e rede fixa, esse exemplo ocorre quando os dados em cache são incompletos ou quando a conectividade é insuficiente para efetuar uma transferência de dados.

O MiD-Mobile atende diferentes domínios de aplicação através da harmonização entre requisitos funcionais e a disponibilidade de recursos computacionais, sem sacrificar as funcionalidades de cada software. Já o modelo altera a estrutura interna da transação para viabilizar o processamento isolado, remoto ou compartilhado.

As próximas seções detalham a arquitetura do middleware e o modelo proposto.

#### 4. A arquitetura do MiD-Mobile

O principal objetivo do MiD-Mobile é garantir que as transações móveis iniciadas por um SCM obtenham êxito independente dos recursos disponíveis. Para isso, indica um plano de execução baseado nos itens ilustrados na figura 2. Um plano de execução é a estratégia de processamento mais adequada à transação no momento em que ela foi requisitada. Uma transação adequada ao plano de execução é denominada Transação Móvel Adaptada ao Contexto ( $T_{MAC}$ ).

As dimensões apresentadas na figura 2 foram mineradas e minúcias foram identificadas. Da tabela 5 constam as dimensões gerais, os detalhes e a discretização dos conteúdos tratados pelo middleware.

Dimensões	Detalhes	Conteúdos
Rede	Estado da conexão	Conectado (0) Desconectado (1)
	Taxa de transferência	Suficiente (0) Intermediária (1) Insuficiente (2)
	Intensidade do sinal	Suficiente (0) Intermediária (1) Insuficiente (2)
Dados	Situação dos dados	Possui dados (0) Não possui dados (1)
	Completeness dos dados	Completo (0) Incompleto (1) Indefinido (2)
	Validade dos dados	Dados Válidos (0) Inválidos (1) Indefinido (2)
Dispositivo	Capacidade de Bateria	Suficiente (0) Intermediária (1) Insuficiente (2)
	Capacidade de hardware	Suficiente (0) Intermediária (1) Insuficiente (2)

**Tabela 5.** Aspectos avaliados durante a adaptação.

Para que MiD-Mobile execute suas funções, vários serviços foram desenvolvidos para dar o suporte necessário, a figura 3 oferece uma visão geral da arquitetura do middleware.

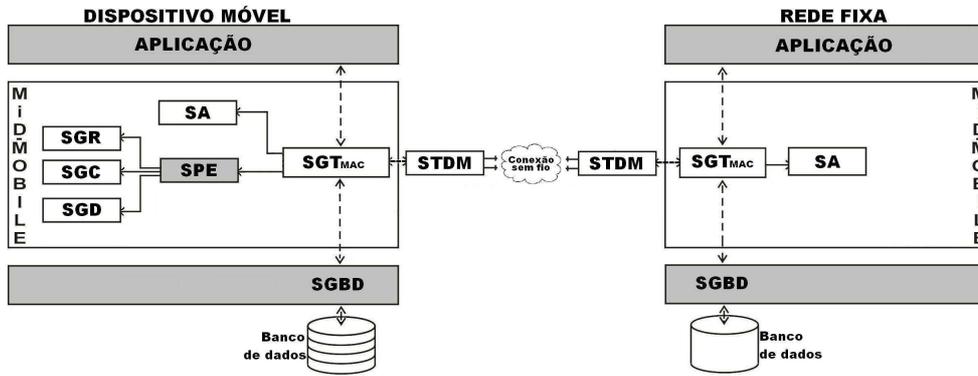


Figura 3. Arquitetura do MiD-Mobile

Em sua função principal o middleware, através do serviço gerenciador de transações móveis adaptadas ao contexto (SGT<sub>MAC</sub>), aciona o serviço de plano de execução (SPE) para indicar o plano de processamento mais adequado à transação. Essa decisão depende da caracterização do contexto móvel elaborado pelo serviço de gerenciamento de recursos do dispositivo móvel (SGR), serviço de gerenciamento da comunicação (SGC) e o serviço de gerenciamento dos dados (SGD).

Cada serviço atua junto a uma das dimensões citadas na figura 2 e detalhadas na tabela 5.

Para que o plano de execução possa oferecer chances reais de êxito, ele é proposto logo que a transação é requerida pela aplicação. Ao término da aplicação do plano, o SGT<sub>MAC</sub> convoca o Serviço de Aprendizagem (SA) para registrar a ocorrência e os resultados obtidos, dados necessários para refinar o processo de aprendizagem. A comunicação entre as camadas do SGT<sub>MAC</sub> é auxiliada pelo serviço de troca de dados e mensagens (STDM). É responsabilidade do STDM encaminhar os dados e as mensagens geradas durante as desconexões assim que o dispositivo móvel se reconectar à rede fixa.

A decisão sobre o plano de execução mais adequado à transação é vista como um problema de classificação. Classificar significa qualificar um exemplo em uma classe, o presente trabalho fez uso de Árvores de Decisão (AD) nessa tarefa. Existem vários métodos de classificação, a escolha por árvores de decisão deve-se a características como representatividade discreta dos valores dos atributos, inclusive o de classificação, robustez ao verificar ruídos (classificações incorretas) no conjunto de treinamento ou ausência de valores de atributos nas instâncias, entre outros.

A criação de uma árvore de decisão consiste no particionamento recursivo de um conjunto de treinamento, até que cada subconjunto obtido contenha casos de uma única classe. Neste estudo, o conceito de classe diz respeito ao plano de execução da transação. A figura 4 ilustra um modelo de AD e seu respectivo conjunto de treinamento.

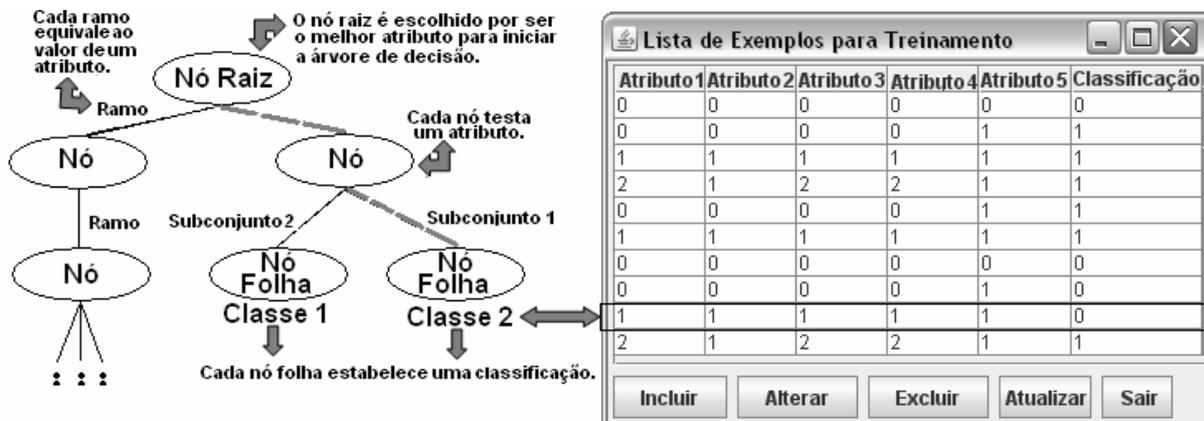


Figura 4. Modelo de árvore de decisão e de conjunto de treinamento

A responsabilidade pela indução da árvore é do Serviço de Aprendizagem (SA) da rede fixa. O algoritmo utilizado foi o ID3. A escolha justifica-se pela completude no espaço de busca da árvore, pela seleção estatística (entropia) das instâncias do problema, pelo ganho de desempenho devido à ausência de *backtracking*, ou seja, ausência de reconsideração de atributos anteriores, entre outros.

Segundo (Russel e Norvig, 2003), o ID3 é um dos algoritmos pioneiros na indução de árvores de decisão, sua essência consiste na seleção do atributo mais hábil para classificar o exemplo, essa escolha repete-se a cada nó descendente. Uma medida quantitativa, chamada ganho de informação, é usada para verificar quão bem um atributo classifica as instâncias de treinamento.

Mas antes do cálculo do ganho de informação é preciso verificar o ruído (classificação incorreta) inserido em um conjunto de treinamento. Essa avaliação é chamada de entropia. Em um conjunto de treinamento  $S$  o cálculo da entropia é dado por:

$$Entropia(S) = \sum_{j=1}^n - p_j \log_2(p_j)$$

Por razões de espaço, é apresentado um caso de cálculo de entropia para uma classificação binária. O exemplo contém 14 ocorrências no conjunto de treinamento, destas 9 são classificadas como A e 5 como B, a entropia de  $S$  é dada por:

$$Entropia([9_A, 5_B]) = -(9/14) \log_2(9/14) + -(5/14) \log_2(5/14)$$

$$Entropia([9_A, 5_B]) = 0.940$$

Após o cálculo da entropia de  $S$ , a fórmula do ganho de informação é aplicada para determinar o melhor atributo a ser usado como raiz.

$$Ganho(S, A) = Entropia(S) - \sum_{v \in \text{valores}(A)} \left[ \frac{|S_v|}{|S|} * Entropia(S_v) \right]$$

Para exemplificar dois atributos são considerados: estado e taxa. O atributo estado possui as opções "conectado" e "desconectado" abreviadas por "con" e "des". A taxa pode ser "satisfatória" ou "insatisfatória" identificada por "sat" ou "ins". A ocorrência do conteúdo "conectado" para o atributo estado é de 3 vezes, para "desconectado" 11. Taxa "satisfatória" ocorre 5 vezes e "insatisfatória" 9. Com esses dados é possível calcular o ganho para cada atributo. Para "estado" os cálculos são:

$$Ganho(S, estado) = 0.940 - (|S_{con}|/14) * (Entropia(S_{con})) - (|S_{des}|/14) * (Entropia(S_{des}))$$

$$Ganho(S, estado) : 0.940 - 0.476 - 0.274$$

$$Ganho(S, estado) : 0.19$$

Para o atributo "taxa" os cálculos são:

$$Ganho(S, taxa) = 0.940 - (|S_{sat}|/14) * (Entropia(S_{sat})) - (|S_{ins}|/14) * (Entropia(S_{ins}))$$

$$Ganho(S, taxa) : 0.940 - 0.530 - 0.442$$

$$Ganho(S, taxa) : 0.032$$

Comparando o ganho do atributo estado com o ganho do atributo taxa, percebe-se que o resultado de estado é maior, ou seja, ele é o atributo mais indicado para iniciar a classificação como nó raiz. A aplicação recursiva do algoritmo leva a escolha dos próximos atributos, criando assim a árvore de decisão. Uma árvore de decisão é útil se o seu aprendizado permite classificar corretamente exemplos não-treinados, essa é a principal função do SPE.

O ID3 emprega recursividade na escolha do melhor atributo, essa tática faz crescer progressivamente o uso da memória e do processamento, sendo assim, inicialmente, o algoritmo não é uma proposta coerente face à restrição de recursos dos dispositivos móveis. A

solução para manter a simplicidade, velocidade e economia de recursos computacionais do equipamento portátil foi inspirada em Autômatos Finitos Determinísticos (AFD).

Um AFD é uma quintupla  $\langle Q, \Sigma, \delta, q_0, F \rangle$ , onde  $Q$  é um conjunto finito de estados internos,  $\Sigma$  o alfabeto de entrada,  $\delta$  é um conjunto de funções de transição,  $q_0$  o estado inicial e  $F$  o conjunto de estados finais. Resumidamente, um AFD lê uma string de entrada que contém símbolos do alfabeto. A função de transição promove a mudança de estado de acordo com o símbolo lido, ao término da leitura diz-se que o AFD aceita a cadeia caso ela tenha partido do estado inicial e atingido um estado final.

Um AFD pode ser representado por uma tabela de transição, essa característica foi estendida às árvores de decisão. A figura 5 mostra uma tabela de transição para um AFD  $M$ , um exemplo de tabela de transição da árvore de decisão gerada pelo MiD-Mobile e um exemplo da tabela de transição usada na generalização de novos casos.

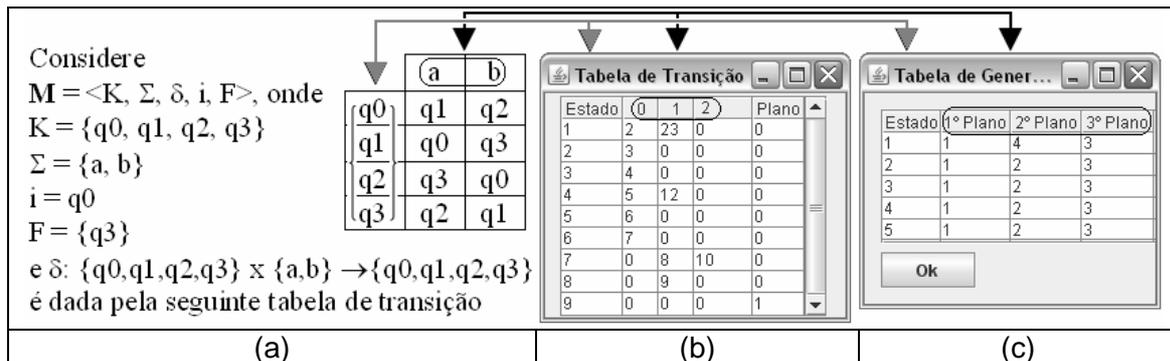


Figura 5. Tabela de transição para o AFD  $M$  (a), tabela de transição (b) e tabela de generalização (c).

Comparando a tabela de transição do AFD, figura 5(a), com a tabela de transição da árvore de decisão gerada pelo middleware, figura 5(b), é possível verificar que os nós da árvore equivalem aos estados do AFD e são descritos na coluna “Estado” da figura 5b. Os ramos 0, 1 e 2 tomaram o lugar dos símbolos do alfabeto da figura 5a. O acréscimo ficou por conta da coluna plano que só é preenchida quando precisar representar um nó folha, ou seja, uma classificação. Classificar casos de teste, ou novos casos, é uma forma de verificar a capacidade de generalização da árvore, ou seja, se ela adquiriu conhecimento.

Quando uma instância de teste for aplicada à tabela de transição, o algoritmo tradicional do AFD pode se perder ao encontrar em seu percurso um estado que não reconhece o símbolo lido. Duas operações complementares foram usadas nessa situação, uma adaptação no algoritmo tradicional e a criação de uma tabela de generalização.

O algoritmo do AFD foi adaptado para percorrer a tabela de transição da árvore de decisão até o subconjunto conhecido, em seguida passa para a tabela de generalização e quantifica a incidência de cada classificação nos subconjuntos descendentes, figura 5(c). Assim, a tabela de generalização difere da tabela de transição por quantificar as classificações subjacentes ao ponto de parada e não por indicar o próximo atributo.

Percorrer a tabela de transição e de generalização através de um algoritmo de mudança de estado é mais simples e mais econômico em termos de memória e processamento do que o ID3, conferindo ao SPE leveza e simplicidade para decidir onde executar a transação solicitada. A criação da árvore e dos mecanismos de aprendizagem e generalização é responsabilidade do serviço de aprendizagem disponível na rede fixa.

Ao iniciar o acesso primário ao sistema de computação móvel que emprega o MiD-Mobile, tanto a aplicação quanto as tabelas de transição e generalização são enviados para o dispositivo móvel. Os desenvolvedores de SCM em cada domínio de aplicação decidem o período mais apropriado para as instâncias de testes e os resultados armazenados pelo SA do dispositivo móvel serem enviados ao SA da rede fixa, visando novo treinamento. O aproveitamento das experiências colhidas em cada dispositivo permite à árvore melhorar seu desempenho em novos casos. Em seguida, os elementos são reenviados ao dispositivo móvel para continuar a avaliar casos inéditos.

## 5. Modelo para Transações Móveis Adaptadas ao Contexto

Transações planas são caracterizadas por apresentar uma única camada de controle e pelo seu breve tempo de vida. Possuem propriedades bastantes restritivas conhecidas por ACID, que garantem a integridade e a recuperabilidade dos dados. Mas não é em todo cenário que uma transação dura uma fração de segundos, a conexão intermitente, a largura de banda insuficiente ou o trânsito em áreas não abrangidas pelo sinal de comunicação fazem de uma transação móvel uma transação de maior duração.

Com o intuito de oferecer maior flexibilidade com o mínimo de transgressão às propriedades ACID, o presente trabalho propõe que uma transação móvel obedeça ao modelo de transações móveis adaptadas ao contexto. No modelo uma transação móvel é chamada de  $T_{MAC}$ , acrônimo do nome do modelo.

A estrutura de uma  $T_{MAC}$  depende do seu plano de execução. Cada plano pode optar por uma dentre as três estratégias possíveis, execução isolada, remota ou compartilhada. No plano de execução isolada a transação ocorre no dispositivo móvel, a opção remota prevê o envio do processamento para a rede fixa e a execução compartilhada distribui o processamento entre os extremos.

Uma  $T_{MAC}$  no plano de execução isolada ou no plano de execução remota possui estrutura plana, para a opção compartilhada a estrutura da transação passa a ser multicomponente. Dentre os trabalhos pesquisados, nenhum prevê a mudança de estrutura da transação como mecanismo de adaptabilidade. A figura 6 está dividida em duas partes para apresentar uma  $T_{MAC}$  com estrutura plana (a) e com estrutura multicomponentizada (b).

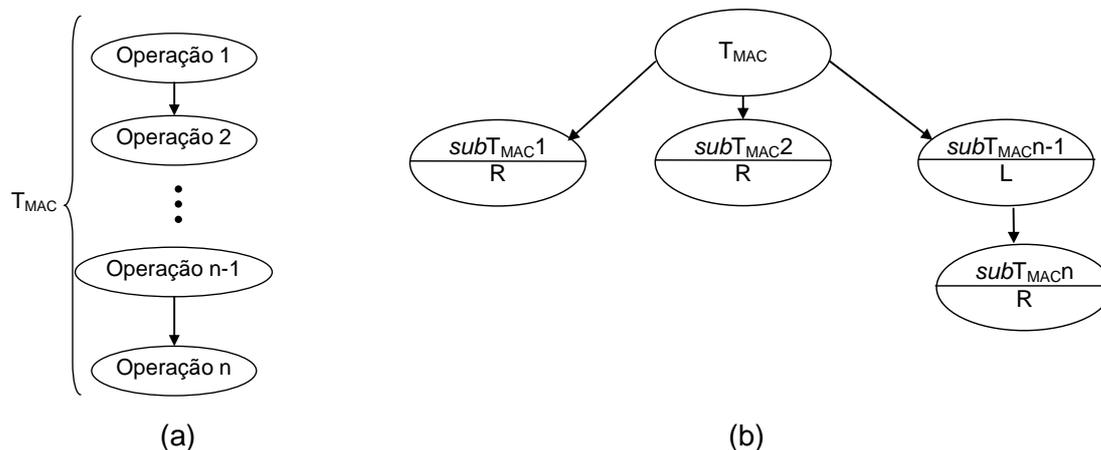


Figura 6. Estruturas de uma  $T_{MAC}$ .

A estrutura plana, figura 6.1 (a), é orientada pelo modelo clássico. A opção dividida em componentes, figura 6.1 (b), é baseada no paralelismo intra-transação explorado no modelo de transação de múltiplos níveis proposto por (Weikum, 1991).

O paralelismo é admitido internamente (intra-transaction parallelism) e externamente, ou seja, entre várias transações concorrentes (inter-transaction parallelism). Transações multicomponentes possuem componentes de computação e de acesso chamados de subtransações ou  $subT_{MACs}$ . Os componentes são concorrentes e possuem relação de dependência de execução entre eles.

A transação impressa na figura 7 é uma transação do tipo plana utilizada quando o plano de execução isolada ou remota é indicado, a figura 8 apresenta um exemplo de  $T_{MAC}$  multicomponente, estrutura indicada para o processamento compartilhado.

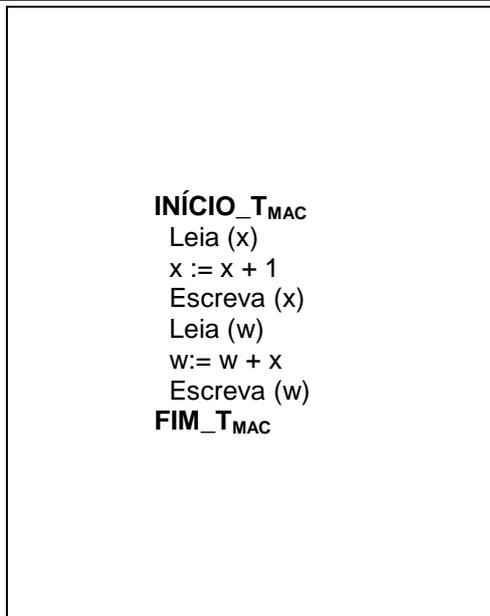


Figura 7. Exemplo de uma  $T_{MAC}$  plana.

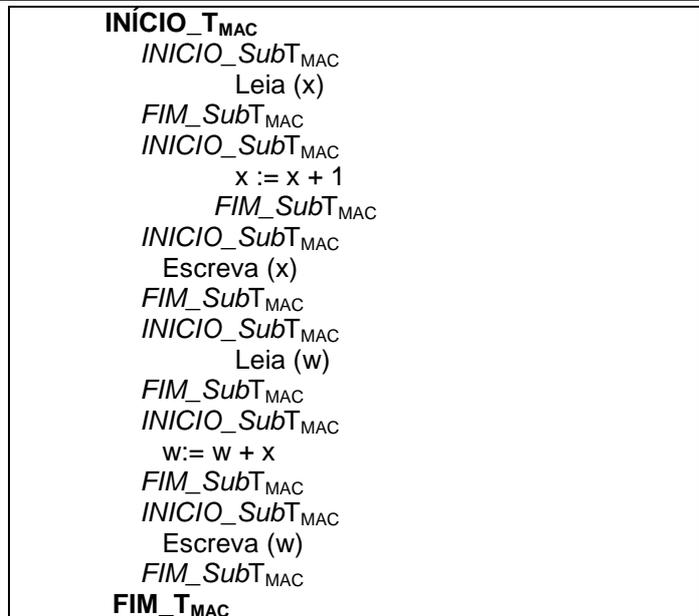


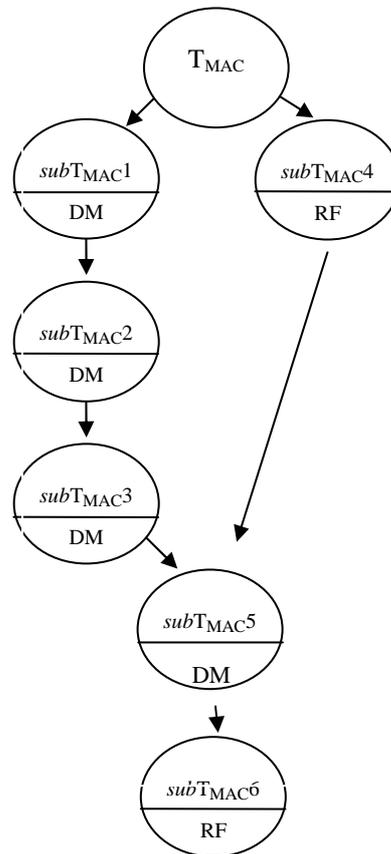
Figura 8. Exemplo de uma transação multicomponente.

A tabela 6 apresenta os detalhes dos planos de execução possíveis, suas respectivas plataformas de execução, estrutura, lista de componentes e dependência.

Plano de Execução	Estrutura da $T_{MAC}$	Plataforma de Execução	Componentes	Dependência
Isolada	Plana	Dispositivo Móvel	<i>não se aplica</i>	<i>não se aplica</i>
Remota	Plana	Rede Fixa	<i>não se aplica</i>	<i>não se aplica</i>
Compartilhada	Multicomponente	Dispositivo Móvel	$subT_{MAC1}$	0
		Dispositivo Móvel	$subT_{MAC2}$	$subT_{MAC1}$
		Dispositivo Móvel	$subT_{MAC3}$	$subT_{MAC2}$
		Rede Fixa	$subT_{MAC4}$	0
		Rede Fixa	$subT_{MAC5}$	$subT_{MAC3},$ $subT_{MAC4}$
		Rede Fixa	$subT_{MAC6}$	$subT_{MAC5}$

Tabela 6. Plano de Execução de uma  $T_{MAC}$ .

As colunas "Componentes" e "Dependência" mostram a divisão interna da  $T_{MAC}$  e a ordem de execução das subtransações para o plano de execução compartilhada. A figura 9 representa a dependência de subtransações como uma hierarquia disposta em formato de grafo acíclico dirigido.



**Figura 9.** Dependência entre componentes.

Manter a consistência em ambiente de execução concorrente onde transações ou subtransações acessam os mesmos dados é possível através de escalas de execução. A granularidade de uma  $subT_{MAC}$ , conforme representado na figura 9, é a menor possível, isso permite que cada subtransação seja composta por apenas uma operação de leitura, escrita ou computação. Dessa forma, é possível manter cada subtransação como uma unidade atômica passível de ser submetida a um escalonamento serializável.

### 5.1. Critério de correteza do modelo

Escalonamentos serializáveis são escalas de execução intercalada das operações de transações concorrentes equivalentes à execução serial das mesmas. Tais escalonamentos compõem a teoria da serializabilidade, critério de correteza mais amplamente usado pelos sistemas gerenciadores de banco de dados (SGBD).

Neste cenário são comuns as operações intercaladas que escrevem em um mesmo item de dados no mesmo site, chamadas operações conflitantes. Anomalias decorrentes de operações conflitantes são tratadas pelo SGBD através de mecanismos de controle de concorrência. Esse mecanismo garante a consistência e o isolamento das transações, os mais conhecidos são baseados em bloqueios, marcas de tempo (*timestamp*) ou validação. Neste trabalho, o controle da concorrência depende do plano de execução.

Uma  $T_{MAC}$  remota ou isolada é submetida ao controle de concorrência empregado pelo SGBD do dispositivo móvel ou da rede fixa. Para a execução distribuída, o middleware aplica o controle baseado em bloqueio adaptado para uso conjunto com um controlador de tempo, denominado protocolo de bloqueio bifásico rigoroso temporário ( $2PL\Delta t$ ).

Em seu formato primário, o protocolo de bloqueio em duas fases rigoroso (*two-phase locking* – 2PL) emite gradualmente as solicitações de bloqueio e ao término da transação permite o desbloqueio total. A figura 10 extraída de (Özsu e Valduriez, 2001) exhibe a execução do 2PL rigoroso.

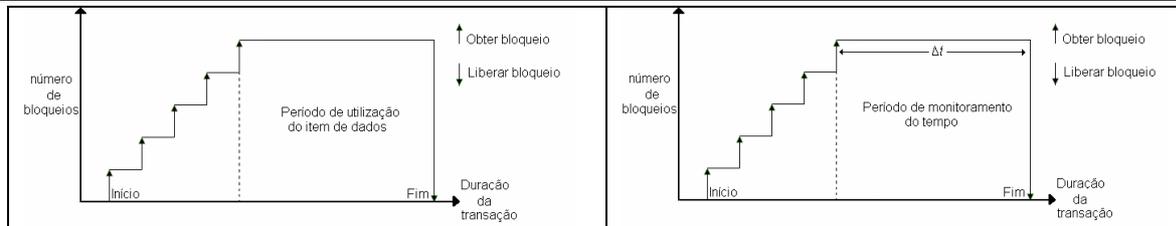


Figura 10. Fases de bloqueio 2PL.

Figura 11. Fases de bloqueio 2PL $\Delta t$ .

A adaptação feita ao 2PL a partir do controle de tempo é exemplificada na figura 11, o bloqueio permanece até o término da  $T_{MAC}$  ou até o tempo destinado para a transação terminar. No ambiente distribuído, o bloqueio é solicitado pelo gerenciador da transação ao gerenciador de bloqueio. No ambiente distribuído móvel, com a utilização do MiD-Mobile, o  $SGT_{MAC}$  ao receber a  $T_{MAC}$ , solicita os bloqueios ao gerenciador de bloqueio central que repassa o pedido aos gerenciadores locais. No momento adequado o  $SGT_{MAC}$  informa sobre as liberações.

Na utilização das técnicas de bloqueio os deadlocks ocorrem com maior frequência, controles adicionais também baseados em ajuste de tempo foram usados para minimizar o *overhead* ocasionado pela aplicação do esquema.

A consistência global dos dados é garantida pelo protocolo de consistência global (PCG) apresentado no diagrama de seqüência da figura 12. Uma comparação entre as etiquetas de tempo (*timestamp*) dos dados do dispositivo móvel e da rede fixa é realizada para avaliar a continuidade do processo de sincronização.

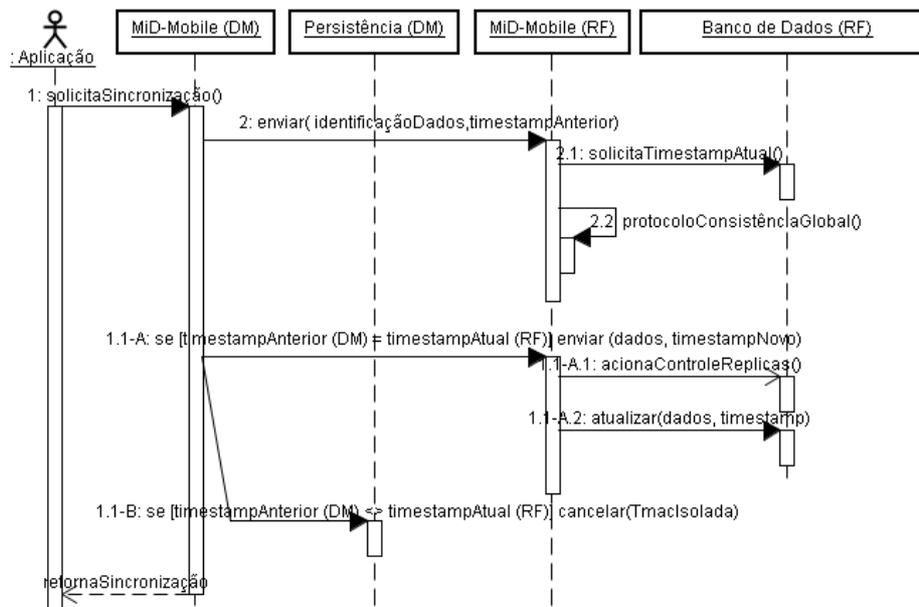


Figura 12. Diagrama de seqüência do PCG.

Caso a solicitação de sincronização seja iniciada na rede fixa, mecanismos de gerenciamento de cache devem ser aplicados para decidir a política de invalidação de cache mais adequada para o domínio da aplicação. Caso o banco de dados seja replicado, a consistência mútua entre todas as cópias dos itens de dados é responsabilidade do protocolo de controle de réplica empregado pelo SGBD.

Diante da apresentação do modelo de transação de uma  $T_{MAC}$  é relevante ressaltar que para tratar as especificações sugeridas foi desenvolvido um serviço específico dentro do MiD-Mobile, o  $SGT_{MAC}$ .

O  $SGT_{MAC}$  executa funções comuns aos coordenadores de transações distribuídas, acrescidos de necessidades particulares ao ambiente móvel, tais como: receber a solicitação da transação; acionar o SPE para decidir como a transação será executada; executar a transação conforme o modelo indicado pelo SPE; encaminhar as subtransações para a rede fixa; executar a efetivação ou o cancelamento das  $T_{MACs}$  e das  $subT_{MACs}$  em todos os sites;

encaminhar o envio de sincronização de dados. Todas essas funções possuem coordenação colaborativa entre as camadas do SGT<sub>MAC</sub>.

## 6. Estudo de caso

Para legitimar o modelo e o middleware proposto, foi desenvolvido um protótipo para execução de operações financeiras denominado Banco Online. A aplicação foi desenvolvida em Java e usou várias tecnologias como JSP, Servlets, JavaME, Web Services, JPA, Hibernate, Annotations e Generics. Os requisitos funcionais disponíveis no Banco Online tratam a visualização de extrato, efetivação de pagamento, simulação e realização de empréstimo. A figura 13 exibe a interface do Banco Online.

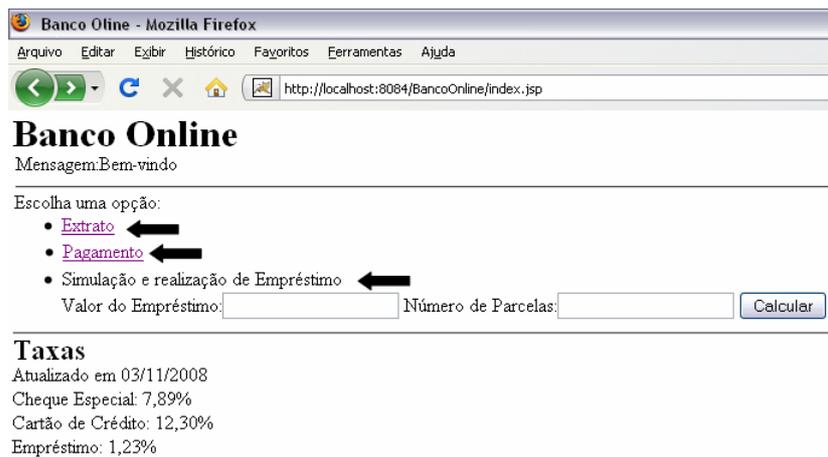


Figura 13. Interface principal do Banco Online.

A figura 14 apresenta o BancoMe, camada da aplicação disponível em celular para acesso aos dados do Banco Online.

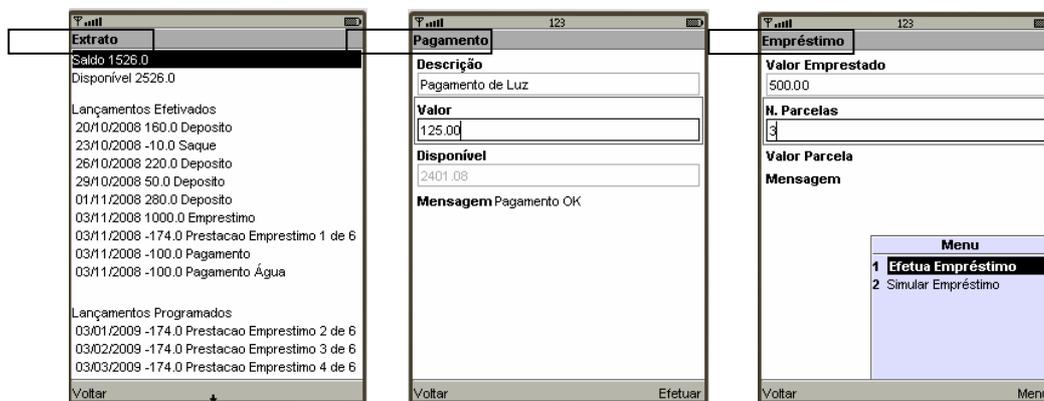


Figura 14. Execução das operações disponíveis do BancoMe.

Todas as transações requisitas pelo BancoMe são tratadas pelo MiD-Mobile que define o plano de execução de cada uma. O middleware disponível na rede fixa possui várias funcionalidades entre elas a indução da árvore de decisão e a verificação da aprendizagem, conforme ilustra a figura 15.

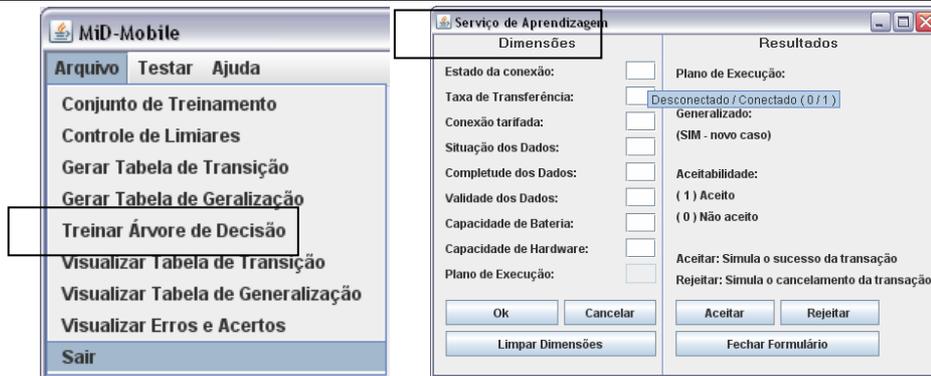


Figura 15. Interface de treinamento e de verificação da aprendizagem.

A figura 16 oferece uma visão geral da utilização do Banco Online, BancoMe e as camadas do MiD-Mobile reforçando os possíveis planos de execução das transações, as operações implementadas no protótipo e a troca dos dados que permitem a aprendizagem.

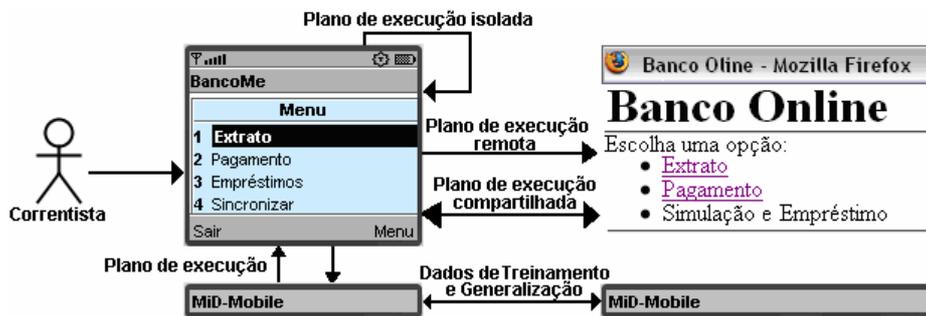


Figura 16. Visão geral do Banco Online, BancoMe e do MiD-Mobile.

## 7. Conclusões

Permitir a execução de requisitos funcionais em sistemas de computação móvel significa submeter transações às variações de recursos do dispositivo portátil e da conectividade sem fio. Um cenário sujeito às desconexões ou a falhas de site compromete as funções dos sistemas. Este trabalho apresentou um instrumento de decisão dinâmica para adaptar a transação ao processamento de maior chance de êxito.

Os três tipos de execução são isolada, remota ou compartilhada. O plano de processamento isolado prevê uma situação de falta de comunicação com a rede fixa e completa hospedagem dos dados necessários. A abordagem remota é indicada para dispositivos de recursos modestos, que usufruem de algum tipo de comunicação sem fio, configuração ideal para emitir requisições, mas não para processá-las. A execução compartilhada é ideal diante da instabilidade da rede de comunicação ou devido à parcialidade dos dados usados nas transações.

A adaptabilidade é exercida na organização interna das transações com base na situação da rede de comunicação, nas configurações do equipamento móvel e na disponibilidade dos dados. Todo o contexto móvel é identificado pelo MiD-Mobile.

O middleware foi proposto no formato de serviços acessíveis por interfaces. O mecanismo de aprendizagem refina as indicações do middleware e aumenta as chances de sucesso das  $T_{MACs}$ . Os retreinamentos posteriores propiciarão ao algoritmo da árvore de decisão generalizar corretamente um número cada vez maior de casos de testes.

## Referências

- Côrtes, S. C. *Um modelo de transações para integração de SGBD a um ambiente de computação móvel*. Rio de Janeiro, Departamento de Informática PUC, 2004. 182 p. Tese (Doutorado em Informática) Pontifícia Universidade Católica Rio de Janeiro, 2004.
- Couloris, G., Dollimore, J. & Kindberg, T. *Sistemas Distribuídos: conceitos e projetos*. Porto Alegre: Bookman, 2007.
- Dunham, M.H., Helal, A. & Balakrishnan, S. A mobile transaction model that captures both the data and movement behavior, 1997. *ACM/Baltzer Journal on Special Topics in Mobile Networks and Applications*, v.2, p. 149-162, 1997.
- Ferreira, J. E. & Finger, M. *Controle de concorrência e distribuição de dados: a teoria clássica, suas limitações e extensões modernas*. São Paulo: IME-USP, 2000.
- Gray, J., Helland, P., O'neil, P. & Shasha, D. The dangers of replication and a solution. In: *INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, 1996*, Montreal. Proceedings of the 1996 ACM SIGMOD. Montreal, Canada, 1996. P. 173-182.
- Ku, K. & Kim, Y. Moflex transaction model for mobile heterogeneous multidatabase systems. In: *IEEE WORKSHOP ON RESEARCH ISSUES IN DATA ENGINEERING*, San Diego, 2000. Proceedings of the 10th RIDE, San Diego, 2000. P. 39.
- Lu, Q. & Satyanarayanan, M. Isolation-only transactions for mobile computing. In: *ACM SIGOPS Operating Systems Review*, vol. 28, n. 2, p.81-87,1994.
- Özsu, M.T. & Valduriez, P. *Princípios de sistemas de bancos de dados distribuídos*. Rio de Janeiro, Campus, 2001.
- Pitoura, E. & Bhargava, B. *Maintaining Consistency of Data in Mobile Distributed Environments*. In: *INTERNATIONAL CONFERENCE ON DISTRIBUTED COMPUTING SYSTEMS, 1995*. Proceedings of 15th ICDCS, June, 1995.
- Rocha, T. & Toledo, M.B.F. *Estudo de Modelos de Transação para o ambiente de Computação Móvel*. Campinas: Instituto de Computação - Unicamp, Relatório Técnico IC-05-25, 2005.
- Rocha, T. & Toledo, M.B.F. Mecanismos de Adaptação para Transações em Ambientes de Computação Móvel, 2007. *Revista IEEE Latin America Transactions*, p.631-637, 2007.
- Russel, S. & Norvig, P. *Inteligência Artificial*. Rio de Janeiro: Campus, 2003.
- Satyanarayanan, M. Fundamental Challenges in Mobile Computing, 1996. In: *ANNUAL ACM SYMPOSIUM ON PRINCIPLES OF DISTRIBUTED COMPUTING, 1996*, Philadelphia. *Proceedings of the 15th ACM Principles of Distributed Computing*. Philadelphia, ACM, 1996. P. 1-7.
- Serrano-Alvarado, P. *Transactions Adaptables pour les Environments Mobiles*. Grenoble, Université Joseph Fourier, 2004. 184 p. Tese (Doutorado em Informática) - Université Joseph Fourier, 2004.
- Serrano-Alvarado, P., Roncancio, C. & Adiba, M. *A Survey of Mobile Transactions*. In: *Distributed and Parallel Databases*. Netherlands, Kluwer Publishers, 2004. p.193-230
- Walborn, G. & Chrysanthis, P. Transaction Processing in PRO-MOTION. In: *SYMPOSIUM ON APPLIED COMPUTING, 1998*, San Antonio. *Proceedings of the 1999 ACM Symposium on Applied Computing*. San Antonio, 1998. P.389-398.
- Weikum, G. & Schek, H.J. Concepts and Applications of Multilevel Transactions and Open Nested Transactions. In: *ELMAGARMID, A. (ed.). Database transaction models for advanced applications*. San Francisco: Morgan Kaufmann Publishers, 1991. P. 515-553.