

5

Uma Arquitetura para Implementação do Modelo de Transações Proposto

Este capítulo tem por objetivo apresentar uma arquitetura para implementação do modelo de transações proposto, integrando SGBDs a um ambiente de computação móvel, usando as tecnologias de framework e agentes de software. Inicialmente, será apresentada uma descrição do ambiente em que a arquitetura se insere, bem como seu esquema genérico. Em seguida, a arquitetura será descrita e representada, em seus aspectos estáticos e dinâmicos, usando a notação MAS-ML.

5.1

O Ambiente para Integração e Implementação do Modelo de Transações Proposto

O ambiente para computação móvel considerado neste trabalho é uma rede de comunicação de dados que possui equipamentos fixos e móveis (*hosts fixos e móveis*). Os dados podem estar distribuídos por vários Sistemas de Gerência de Bancos de Dados (*servidores de banco de dados*), localizados na rede fixa e nos equipamentos móveis (*servidores de banco de dados móveis*).

Os equipamentos móveis podem ser de diferentes tipos, variando desde PDAs a computadores pessoais móveis do tipo notebook. Os equipamentos móveis agem, na maioria das vezes, como *clientes*, e deles partem as consultas e atualizações para os bancos de dados, operações essas que compõem as *transações de bancos de dados*. A figura 5.1 apresenta uma caracterização do ambiente móvel considerado, destacando seus componentes fixos e móveis.

Os equipamentos móveis se comunicam com os servidores de banco de dados através das Estações de Suporte à Mobilidade (ESM) e podem mudar a sua localização, inclusive migrando para a área de controle de outra ESM. Essa migração pode ocorrer no momento em que as transações, solicitadas pelas aplicações residentes nos equipamentos móveis (*clientes móveis*), estão sendo processadas, nos servidores de banco de dados localizados na rede

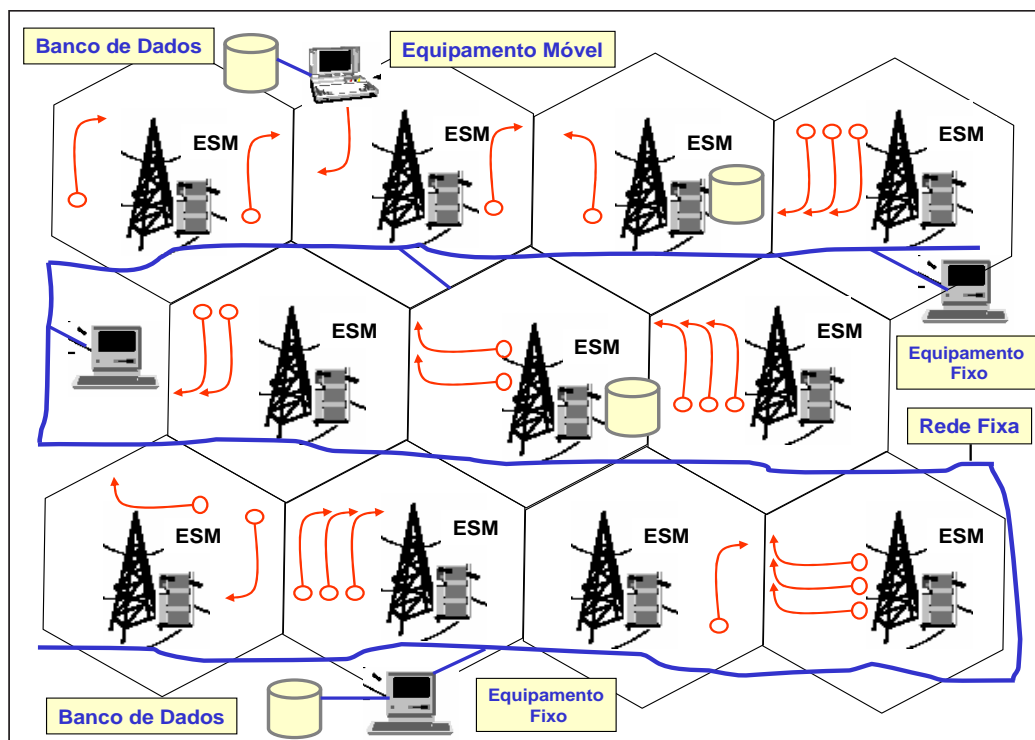


Figura 5.1: Caracterização do ambiente de computação móvel considerado

fixa. Tanto os equipamentos móveis quanto as ESM devem ter a capacidade de executar os módulos de software do SGBD, em prol da execução das transações de banco de dados.

5.2 A Arquitetura Proposta

A arquitetura proposta é uma extensão da arquitetura cliente/servidor apresentada em [20], [44] e [40] e revista no capítulo 2. Nessa arquitetura, é introduzida uma terceira camada, que executará um conjunto de funções particulares do ambiente da computação móvel.

Usuários da computação móvel, inúmeras vezes, necessitam acessar dados, em bancos de dados localizados em equipamentos fora da área de controle da Estação de Suporte à Mobilidade (ESM), à qual eles estão conectados no momento da solicitação do acesso aos dados, o que gera um *overhead* no ambiente. A arquitetura proposta busca minimizar esse *overhead*. Suas *principais contribuições* podem ser resumidas em:

- Proporcionar maior agilidade e desempenho na busca, recuperação e atualização dos dados solicitados pelas operações geradas pelas transações de banco de dados, a partir das aplicações móveis, utili-

zando a característica de migração dos agentes (*agentes móveis*), para diminuir o volume de mensagens trocadas entre o ambiente móvel e o de rede fixa, reduzindo o tempo de conexão necessário para a evolução da transação [33];

- Criar um mecanismo que permita que toda a gerência dos dados processados no ambiente da computação móvel seja efetuada pelo próprio SGBD, assegurando consistência e segurança dos dados;
- Resolver alguns problemas típicos do ambiente da computação móvel, tais como desconexões freqüentes e mobilidade dos clientes móveis, que podem afetar diretamente o processamento das transações de banco de dados e causar um grande desperdício e *overhead* nos recursos do ambiente;
- Construir uma solução capaz de ser adaptada e ajustada para diferentes tipos de ambientes móveis, garantindo a qualidade e a exatidão dos dados armazenados nos servidores de banco de dados, que podem ser de diferentes tipos de SGBD, dados esses manipulados pelas transações geradas nos equipamentos móveis a partir de diferentes linguagens de programação.

A figura 5.2 apresenta o *esquema geral* da arquitetura proposta, visando principalmente destacar a integração entre os ambientes de um sistema de computação móvel e o uso da migração dos agentes móveis.

A arquitetura será definida com o uso da tecnologia de framework e de agentes de software, explorando, principalmente, a sua capacidade de autonomia, percepção e adaptação ao ambiente, reação e mobilidade. Os agentes de software serão utilizados no framework para disponibilizar diversas funcionalidades, necessárias ao ambiente móvel, tais como controle da perda de conexão, tratamento das transações que necessitam de várias interações entre os clientes móveis e o SGBD da rede fixa, controle sobre o término de transações longas e tratamento do deslocamento dos clientes móveis das transações em execução no SGBD da rede fixa.

O esquema geral da arquitetura proposta, apresentado na figura 5.2, está dividido em três camadas:

1. *Camada 1* - Aplicações móveis;
2. *Camada 2* - Estações de Suporte à Mobilidade (ESM) ou Estações de Base (EB);
3. *Camada 3* - Sistemas de Gerência de Bancos de Dados (SGBDs) da rede fixa.

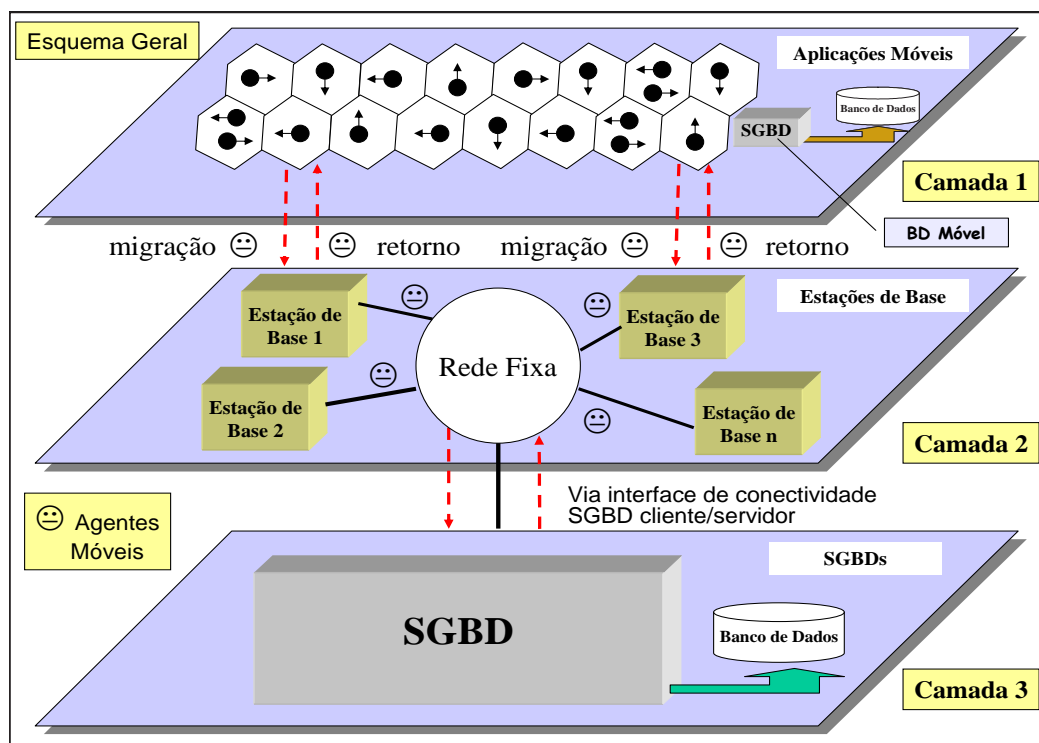


Figura 5.2: Esquema geral da arquitetura proposta

Cada camada possui seus componentes e funções específicos dentro da arquitetura proposta. A seguir são apresentados os detalhes de cada camada.

Camada 1 - Aplicações Móveis

Esta camada, composta pelos equipamentos móveis, contém as aplicações processadas nos equipamentos móveis (*aplicações móveis*), que podem ser desenvolvidas em diferentes tecnologias e diferentes linguagens de programação, e os bancos de dados locais (*BD móvel*). Nela, existirá um sistema de agentes de software, composto por agentes fixos e móveis, que será o responsável por:

1. Criar as *Transações Móveis (TMs)* de banco de dados;
2. Processar as TMs nos equipamentos móveis;
3. Migrar as TMs para a rede fixa e receber os seus resultados da rede fixa;
4. Confirmar ou cancelar as TMs invocadas pelas aplicações móveis.

Vale ressaltar que a utilização dos agentes de software nessa camada traz algumas facilidades, principalmente devido à sua mobilidade, que

propicia maior eficiência na migração das Transações Móveis (TMs) para a rede fixa e na gerência da localização dos clientes móveis, que podem ser identificados pelos agentes das aplicações, de modo que esses agentes possam se comunicar com os agentes da segunda camada e se reconfigurar dinamicamente segundo a necessidade do ambiente em que estejam. Além disso, poderão também desempenhar papéis importantes na segurança e autenticação dos processos executados no equipamento móvel.

Camada 2 - Estações de Suporte à Mobilidade (ESM)

Esta segunda camada da arquitetura, composta pelas ESM, também contém seus componentes desenvolvidos com agentes de software fixos e móveis, visando, principalmente, facilitar a comunicação entre as aplicações móveis e os SGBDs.

Inúmeras vezes, os dados necessários para a aplicação móvel não residem na mesma rede local da ESM, no qual o equipamento móvel está conectado, o que dificulta o acesso à fonte de dados e pode não garantir que as requisições de dados das aplicações sejam atendidas, principalmente em função de inúmeras falhas que podem ocorrer. Na arquitetura proposta, o sistema de agentes residente nas ESM será o responsável por:

1. Receber, controlar e garantir a execução das Transações Móveis (TMs) de banco de dados, nos SGBDs localizados na rede fixa;
2. Entregar corretamente os resultados do processamento das Transações Móveis (TMs) aos clientes móveis que as geraram, independentemente de qualquer tipo de falha que possa ocorrer durante o seu processamento;
3. Manter a comunicação com os sistemas de agentes residentes nos equipamentos móveis (*cliente móvel*);
4. Perceber a mudança de localização do cliente móvel para a área de outra ESM e manter atualizada a sua localização para otimizar a entrega dos resultados de suas transações móveis;
5. Detectar uma possível ruptura na comunicação com o equipamento móvel e não cancelar a Transação Móvel (TM);
6. Trocar mensagens com outros agentes na ESM e entre ESM;
7. Coordenar a execução das Transações Móveis (TMs) na rede fixa;

8. Detectar e controlar uma possível ruptura na comunicação com o ambiente do SGBD;
9. Manter a comunicação com os sistemas de agentes residentes nos SGBDs localizados na rede fixa, quando o SGBD for desenvolvido com a tecnologia de agentes de software.

Camada 3 - Sistemas de Gerência de Bancos de Dados (SGBDs) da Rede Fixa

Esta terceira camada da arquitetura é composta pelos Sistemas de Gerência de Banco de Dados (SGBD) da rede fixa. Na arquitetura proposta os SGBDs podem ser de diferentes tipos, inclusive o próprio SGBD pode ser desenvolvido com o uso de agentes de software, visando facilitar, principalmente, a gerência do SGBD e o armazenamento e a recuperação dos dados por ele mantido. O SGBD é responsável por:

1. Armazenar e garantir a persistência dos dados dos bancos de dados;
2. Garantir a exatidão e a consistência dos dados armazenados;
3. Processar as transações de banco de dados;
4. Garantir o processamento concorrente das transações de banco de dados;
5. Manter a comunicação com os sistemas de agentes residentes nas ESM, através de interfaces específicas;
6. Implementar a segurança para acesso aos bancos de dados.

Essas camadas estão detalhadas na figura 5.3, em que cada camada deu lugar a um módulo.

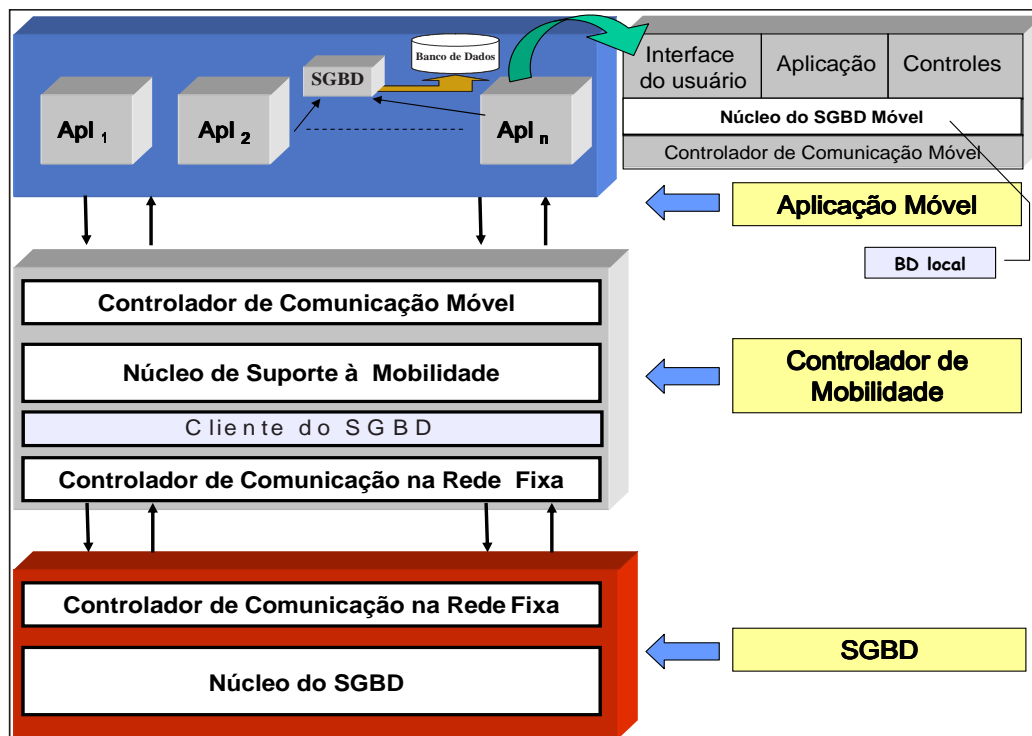


Figura 5.3: Detalhamento do esquema geral da arquitetura proposta

Aplicação Móvel

O módulo *Aplicação Móvel* abrange todas as funcionalidades necessárias para que cada aplicação móvel possa utilizar os serviços de um banco de dados. Várias aplicações, desenvolvidas em diferentes tecnologias, compõem esse modelo. De uma forma geral, abrange os módulos *Interface do Usuário*, *Aplicação*, *Controles ou Outros Softwares*, *Núcleo do SGBD Móvel* e *Controlador de Comunicação Móvel*, além do sistema operacional utilizado no equipamento. O módulo *Núcleo do SGBD Móvel* interage com os demais componentes para realizar as funções de banco de dados.

Controlador de Mobilidade

O módulo *Controlador de Mobilidade* é responsável, principalmente, por manter a comunicação entre as aplicações móveis e o SGBD, localizado na rede fixa. Acompanha a localização de cada cliente móvel e as transações iniciadas nos mesmos, quando esses estão em sua área de controle. Uma vez iniciada uma transação, assume a responsabilidade por sua conclusão, tendo um representante de cada transação móvel, consideradas transações locais. Assume os controles e a responsabilidade de administrar a conclusão das transações, bem como a entrega dos resultados, ao cliente móvel, ou

a comunicação de eventuais falhas em sua execução, mesmo que o cliente móvel tenha alterado a sua localização, se desconectado ou migrado para a área de controle de uma outra ESM. Dessa forma, diferentes protocolos de comunicação podem ser utilizados para prover a comunicação entre os equipamentos móveis e os SGBDs, sempre passando pelas ESM.

SGBD

O módulo *SGBD* é responsável pelo armazenamento e pela distribuição dos dados, execução de todas as transações locais, bem como por todas as atividades inerentes a um SGBD, tais como controle de concorrência, manutenção da consistência e integridade dos dados, persistência, entre outras. Quando desenvolvido como um sistema baseado em agentes, terá a facilidade de integrar todos os seus componentes ao sistema de agentes proposto neste trabalho.

5.3

O Framework Proposto

O framework é um sistema ou subsistema de software parcialmente completo, criado com o objetivo de ser instanciado. Para o projeto do framework, foram considerados as tecnologias de bancos de dados convencionais e distribuídos, as características particulares das aplicações móveis, bem como os controles necessários para implementar o processamento das *transações móveis* de banco de dados.

O framework proposto possui as seguintes funcionalidades:

- De âmbito geral do ambiente móvel:
 1. Fornece um ambiente para execução de aplicações que necessitam utilizar bancos de dados em suas aplicações móveis;
 2. Possibilita a solicitação de consultas e atualizações a bancos de dados, localizados nos equipamentos móveis e em uma rede fixa, a partir de aplicações localizadas em equipamentos móveis;
 3. Integra os componentes localizados em redes fixas e móveis;
 4. Acompanha o processo de deslocamento dos clientes móveis;
 5. Acompanha e reconhece os processos de conexão e desconexão dos clientes móveis;

6. Garante uma perfeita utilização dos dados armazenados nos bancos de dados, mantendo as propriedades de consistência e persistência;
 7. Proporciona a implantação de novos procedimentos de segurança nos dados.
- De âmbito específico do banco de dados:
1. Gerencia todo o processamento das transações móveis;
 2. Proporciona o processamento de transações móveis locais e globais de banco de dados;
 3. Garante a execução das transações móveis, independentemente de eventuais falhas no sistema de comunicação;
 4. Proporciona a distribuição e o compartilhamento de dados;
 5. Garante a consistência e a persistência dos dados armazenados nos SGBDs locais e globais/distribuídos;
 6. Mantém o sincronismo entre os bancos de dados locais e globais no menor intervalo de tempo possível;
 7. Cria mecanismos de segurança para controle e acesso aos dados.

Para documentar o framework proposto, foi usada *MAS-ML* (*Multi-Agent System Modeling Language*) [17], uma extensão da *UML* para a representação de sistemas de agentes. *MAS-ML* é uma linguagem de modelagem que representa os aspectos estruturais e dinâmicos de um sistema baseado em agentes, explorando as características peculiares do agente de software. Os aspectos estruturais (ou estáticos) abrangem a definição de entidades, suas propriedades e seus relacionamentos. Os aspectos dinâmicos estão relacionados ao comportamento das entidades, abrangendo a execução interna das entidades (intra-ação) e as interações entre entidades. *MAS-ML* também define uma gramática para formalizar a sintaxe para descrição dos aspectos estruturais. *MAS-ML* pode ser usada nas fases de análise e projeto de um sistema de agentes.

Na documentação do framework, inicialmente serão apresentados os *casos de uso* de uma instanciação qualquer do framework proposto, representados pela notação do diagrama de casos de usos da *UML* [64, 36]. Para a representação do aspecto estático do framework proposto, serão utilizados o *Diagrama de Organização* e a *gramática da MAS-ML*, enquanto para a representação do aspecto dinâmico será utilizado o *diagrama de seqüência*, com as extensões da *MAS-ML*. A principal vantagem em utilizar a *MAS-ML*

para representar o framework proposto consiste em sua expressividade para representar agentes de software e, especialmente, a mobilidade dos agentes.

5.3.1 Os Casos de Uso do Framework

O Diagrama de Casos de Uso apresenta uma visão externa do software e suas interações com o mundo exterior.

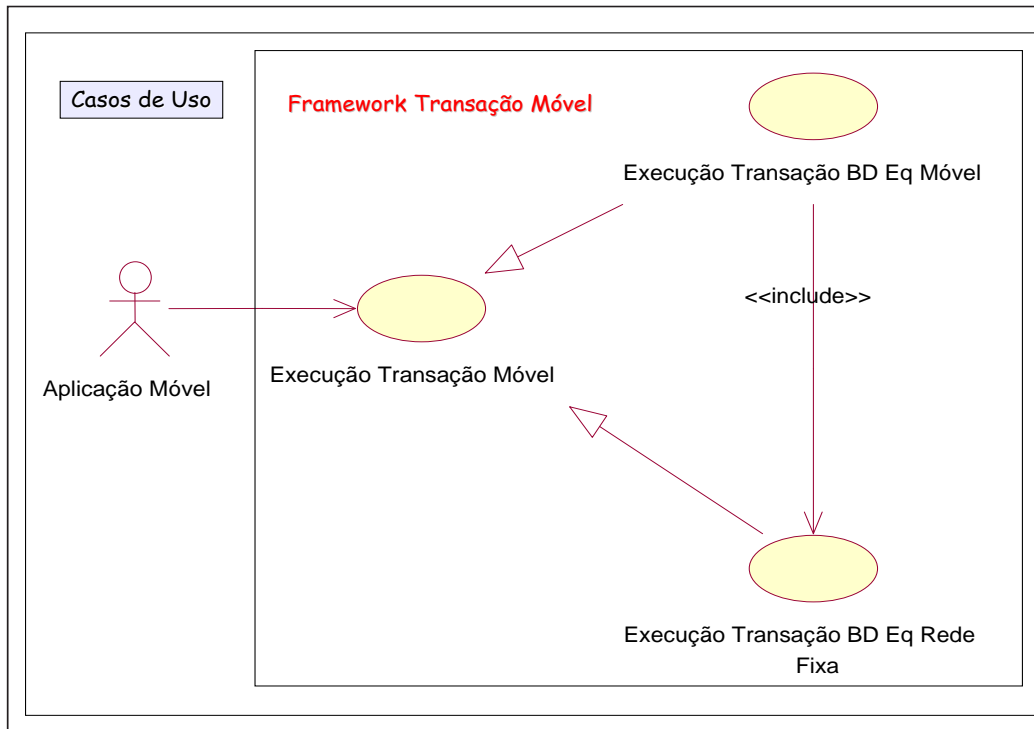


Figura 5.4: Diagrama de Casos de Uso do framework proposto

Conforme apresentado na figura 5.4, têm-se os seguintes casos de uso:

- **Execução Transação Móvel** - este caso de uso começa quando a *Aplicação* localizada no equipamento móvel (*Aplicação Móvel*) solicita a execução de uma transação, considerada *transação móvel*, no contexto do framework. O sistema, então, registra o início da *transação móvel* no histórico de execução de transações localizado no equipamento móvel (*Histórico-M*) e verifica se existe um banco de dados no equipamento móvel (*BD local*). Caso não exista, trata-se de uma transação a ser executada apenas no SGBD da rede fixa, o que corresponde ao caso de uso *Execução Transação BD Eq Rede Fixa*. Caso exista um BD no equipamento móvel, trata-se de uma transação a ser executada primeiramente no equipamento móvel, caso

de uso, *Execução Transação BD Eq Móvel*. Caso sua execução termine com êxito, essa transação deverá ser enviada para ser executada no equipamento da rede fixa, caso de uso, *Execução Transação BD Eq Rede Fixa*.

- **Execução Transação BD Eq Móvel** - este caso de uso refere-se à execução da transação quando existe um banco de dados no equipamento móvel. O sistema encaminha a transação para ser executada pelo SGBD localizado no equipamento móvel, registra os eventos significativos da transação e os resultados da sua execução no histórico de execução de transações (*Histórico-M*), e, se a transação foi terminada com sucesso (*commit*), solicita a execução da transação no SGBD localizado na rede fixa, caso de uso, *Execução Transação BD Eq Rede Fixa*.
- **Execução Transação BD Eq Rede Fixa** - este caso de uso refere-se à execução da transação no equipamento da rede fixa. Uma transação é encaminhada para a rede fixa quando não existe um banco de dados local no equipamento móvel ou quando a transação é executada no equipamento móvel, que tem um banco de dados local, e é terminada com sucesso (*commit*). O sistema encaminha a transação para ser executada pelo SGBD, registra os eventos significativos da transação e os resultados de sua execução no histórico de execução de transações (*Histórico-F*), bem como acompanha a localização do cliente móvel, para efetuar a entrega dos resultados do processamento da transação móvel.

Para melhor distinguir os ambientes e facilitar a compreensão do framework proposto, a figura 5.5 apresenta o *esquema genérico de funcionamento de uma instância do framework*.

5.3.2

Modelos Estáticos do Framework

Na computação móvel existem dois *ambientes de computação* bastante distintos: o *ambiente móvel*, que utiliza uma rede de comunicação sem fio, e o *ambiente de rede fixa*, que utiliza uma rede de comunicação com fio ou cabeada. Conforme proposto em [17], o modelo estático é representado por meio de três diagramas: diagrama de organização, diagrama de classes e diagrama de papéis. Inicialmente apresentaremos o diagrama de organização para cada ambiente abrangido pelo framework proposto. A seguir, serão apresentados os diagramas de classes e de papéis.

- gibilidade. Tais elementos foram, então, especificados em texto, nas próximas subseções, usando a *gramática* MAS-ML;
- O agente *Gerente mobilidade transação* e o agente *Executor transação local* aparecem tanto no Modelo do Ambiente Móvel quanto no modelo do Ambiente de Rede Fixa. O agente *Executor transação local* desempenha exatamente o mesmo papel (*agent role*) em ambos os ambientes. Logo, o agente *Gerente mobilidade transação* e o agente *Executor transação local*, assim como seu papel (*agent role*), estão especificados uma única vez, após a especificação dos elementos específicos de cada ambiente;
 - Uma organização pode ter suas próprias metas (*goals*), crenças (*beliefs*), ações (*actions*) e planos (*plans*), em acréscimo às características dos agentes que desempenham papéis nessa organização. No entanto, no caso das organizações *Equipamento Móvel* e *Equipamento Rede Fixa*, as características de seus agentes são suficientes para representá-las, não havendo necessidade de outras especificações [17].

Diagrama de Organização do Ambiente Móvel

No Ambiente Móvel, o agente *Gerente interação aplicação móvel* desempenha o papel *Interação aplicação móvel*, o agente *Gerente transação eq móvel*, o papel *Gerência transação eq móvel*, o agente *Gerente mobilidade transação* desempenha o papel *Gerência mobilidade transação eq móvel*. O agente *Executor transação local* desempenha o papel *Execução transação local* em ambos os ambientes.

A figura 5.6 apresenta o *diagrama de organização do ambiente móvel*.

Cada um dos agentes e seus respectivos papéis estão especificados a seguir, na gramática MAS-ML. Um agente (*agent*) tem objetivos (*goal*), e, para alcançá-los, executa planos (*plan*). Um plano é composto de ações (*action*), que estão relacionadas a um conjunto de objetivos que o agente atinge quando executa o plano.

O papel de um agente (*agent role*) guia e restringe o comportamento de um agente em um ambiente, descrevendo os objetivos (*goal*) do agente quando está desempenhando o papel, suas crenças (*belief*) e definindo as ações (*duty*) que o agente tem que realizar nesse ambiente. Os protocolos (*protocol*) definem as interações entre papéis através da especificação de mensagens (*message*). Um protocolo estabelece o conjunto de mensagens que um agente pode enviar para outros agentes em uma interação e as mensagens que pode receber deles. A mensagem é caracterizada por

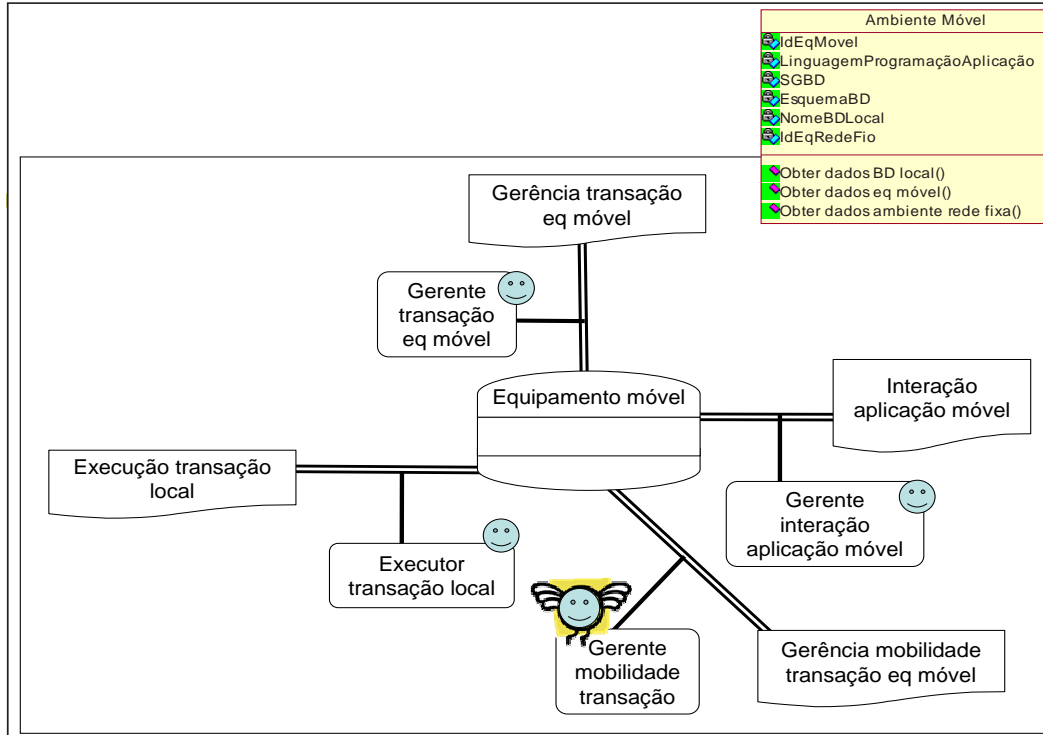


Figura 5.6: Diagrama de Organização do ambiente móvel do framework proposto

um nome (*label*) especificando o tipo de mensagem, o tipo de conteúdo (*content*), o emissor (*sender*) e o receptor (*receive*) da mensagem.

AGENT

(Gerente interação aplicação móvel

```

    GOAL ("Gerenciamento interação aplicação móvel"
        RELATED TO PLAN "Solicitando transação móvel"
        RELATED TO PLAN "Apresentando resultado transação móvel"
    )
    
```

```

    PLAN (Solicitando transação móvel
        COMPOSED OF ACTION "Obter transação móvel"
        COMPOSED OF ACTION "Executar transação móvel"
        RELATED TO GOAL "Gerenciamento interação aplicação móvel"
    )
    
```

```

    PLAN (Apresentando resultado transação móvel
        COMPOSED OF ACTION "Retornar resultado"
        RELATED TO GOAL "Gerenciamento interação aplicação móvel"
    )
    
```

```
    ACTION Obter transação móvel
    ACTION Executar transação móvel
    ACTION Retornar resultado
)

AGENT ROLE
( Interação aplicação móvel

    ROLEGOAL ("Gerenciamento interação aplicação móvel"
    )

    DUTY "Executar transação móvel"
    DUTY "Retornar resultado"

    PROTOCOL SolicitaçãoTransaçãoMóvel
        MESSAGE
        (LABEL "Solicitação"
            CONTENT "Transação móvel a executar"
            SENDER "Interação aplicação móvel"
            RECEIVER "Gerência transação eq móvel"
        )
        MESSAGE
        (LABEL "Resposta"
            CONTENT "Resultado a apresentar"
            SENDER "Gerência transação eq móvel"
            RECEIVER "Interação aplicação móvel"
        )
        MESSAGE
        (LABEL "Aviso"
            CONTENT "Resultado a apresentar"
            SENDER "Gerência transação eq móvel"
            RECEIVER "Interação aplicação móvel"
        )
    )
)

AGENT
( Gerente transação eq móvel
```

```
GOAL ("Gerenciamento transação execução eq móvel"
      RELATED TO PLAN "Gerenciando início transação móvel"
      RELATED TO PLAN "Executando transação eq móvel"
      RELATED TO PLAN "Encerrando transação eq móvel"
    )
GOAL ("Gerenciamento transação execução eq rede fixa"
      RELATED TO PLAN "Gerenciando início transação móvel"
      RELATED TO PLAN "Enviando transação eq rede fixa"
      RELATED TO PLAN "Encerrando transação eq móvel"
    )

PLAN (Gerenciando início transação móvel
      COMPOSED OF ACTION "Registrar início transação eq móvel"
      COMPOSED OF ACTION "Obter dados BD local"
      RELATED TO GOAL "Gerenciamento transação execução eq móvel"
      RELATED TO GOAL "Gerenciamento transação execução eq rede fixa"
    )
PLAN (Enviando transação eq rede fixa
      COMPOSED OF ACTION "Registrar envio transação eq rede fixa"
      COMPOSED OF ACTION "Enviar transação eq rede fixa"
      RELATED TO GOAL "Gerenciamento transação execução eq rede fixa"
    )
PLAN (Encerrando transação eq móvel
      COMPOSED OF ACTION "Encerrar transação eq móvel"
      COMPOSED OF ACTION "Retornar resultado"
      COMPOSED OF ACTION "Avisar resultado"
      COMPOSED OF ACTION "Executar transação replicada eq rede fixa"
      COMPOSED OF ACTION "Executar transação compensatória"
      RELATED TO GOAL "Gerenciamento transação execução eq móvel"
      RELATED TO GOAL "Gerenciamento transação execução eq rede fixa"
    )
PLAN (Executando transação eq móvel
      COMPOSED OF ACTION "Executar aplicação"
      COMPOSED OF ACTION "Encerrar transação eq móvel"
      RELATED TO GOAL "Gerenciamento transação execução eq móvel"
    )

ACTION Registrar início transação eq móvel
ACTION Obter dados BD local
```



```
ACTION Registrar envio transação eq rede fixa
ACTION Enviar transação eq rede fixa
ACTION Encerrar transação eq móvel
ACTION Retornar resultado
ACTION Avisar resultado
ACTION Executar transação replicada eq rede fixa
ACTION Executar transação compensatória
ACTION Executar aplicação
```

)

AGENT ROLE

(Gerência transação eq móvel

```
ROLEGOAL ("Gerenciamento transação execução eq móvel"
)
```

```
ROLEGOAL ("Gerenciamento transação execução eq rede fixa"
)
```

DUTY "Enviar transação eq rede fixa"

PROTOCOL EnvioTransaçãoRedeFixa

MESSAGE

(LABEL "Solicitação"

CONTENT "Transação a enviar"

SENDER "Gerência transação eq móvel"

RECEIVER "Gerência mobilidade transação eq móvel"

)

MESSAGE

(LABEL "Solicitação"

CONTENT "Transação a encerrar"

SENDER "Gerência mobilidade transação eq móvel"

RECEIVER "Gerência transação eq móvel"

)

PROTOCOL ExecuçãoAplicação

MESSAGE

(LABEL "Solicitação"

CONTENT "Aplicação a executar"

SENDER "Gerência transação eq móvel"

RECEIVER "Execução transação local"

```
)
MESSAGE
(LABEL "Resposta"
  CONTENT "Resultado execução transação"
  SENDER "Execução transação local"
  RECEIVER "Gerência transação eq móvel"
)
)

AGENT ROLE
( Gerência mobilidade transação eq móvel

  ROLEGOAL ("Gerenciamento deslocamento ambiente rede móvel"
  )

  DUTY "Procurar ambiente rede fixa"

  PROTOCOL MudançaDeAmbiente
    MESSAGE
    (LABEL "Role_deactivate"
      SENDER "Gerência mobilidade transação eq móvel"
      RECEIVER "Gerência mobilidade transação eq móvel"
    )
    MESSAGE
    (LABEL "Role_change"
      SENDER "Gerência mobilidade transação eq móvel"
      RECEIVER "Gerência mobilidade transação eq rede fixa"
    )
  )
)
```

Diagrama de Organização do Ambiente Rede Fixa

A figura 5.7 apresenta o *diagrama de organização do ambiente rede fixa*.

O agente *Executor transação local*, responsável por tratar da execução da transação no SGBD existente no local, também aparece no Ambiente Móvel, visto que pode existir um banco de dados no equipamento móvel.

No Ambiente Rede Fixa, o agente *Gerente mobilidade transação* desempenha o papel Gerência mobilidade transação eq rede fixa; o agente

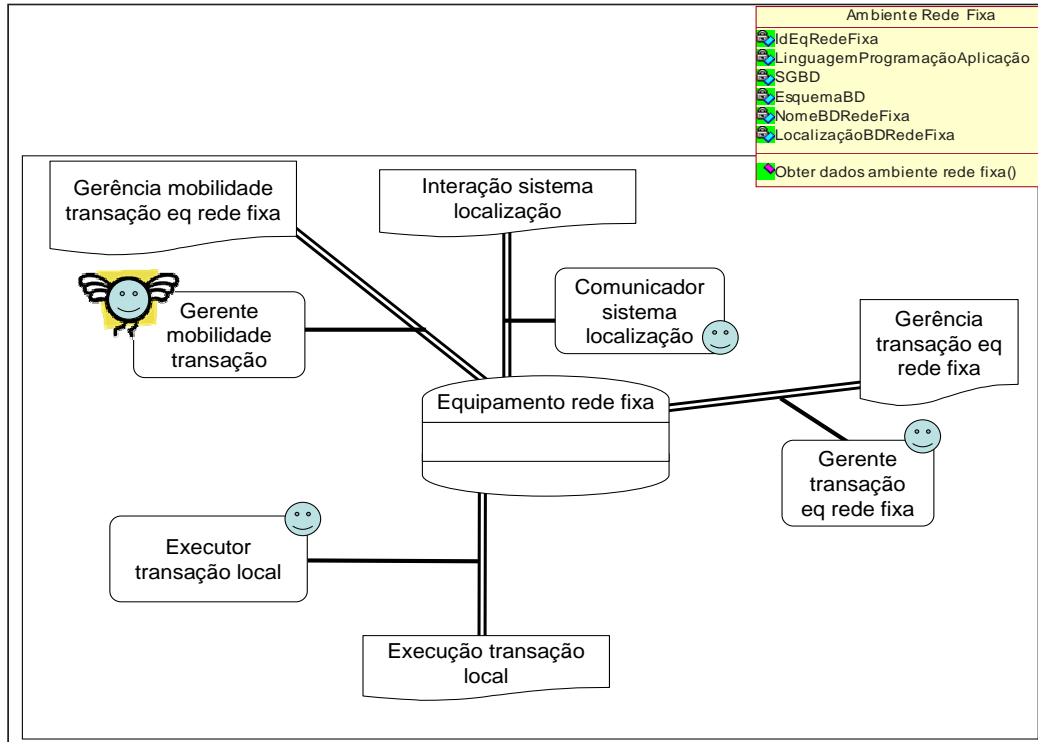


Figura 5.7: Diagrama de Organização do ambiente de rede fixa do framework proposto

Comunicador sistema localização, o papel *Interação sistema localização*; o agente *Gerente transação eq rede fixa*, o papel *Gerência transação eq rede fixa*. Finalmente, o agente *Executor transação local* desempenha o papel *Execução transação local* em ambos os ambiente.

Cada um dos agentes e seus respectivos papéis estão especificados a seguir, na gramática MAS-ML.

AGENT ROLE

(*Gerência mobilidade transação eq rede fixa*

```
ROLEGOAL ("Gerenciamento deslocamento ambiente rede fixa"
)
```

```
DUTY "Executar transação eq rede fixa"
```

```
DUTY "Localizar cliente móvel"
```

```
PROTOCOL ExecuçãoTransaçãoRedeFixa
```

```
MESSAGE
```

```
(LABEL "Solicitação"
```

```
        CONTENT "Transação a executar"
        SENDER "Gerência mobilidade transação eq rede fixa"
        RECEIVER "Gerência transação eq rede fixa"
    )
MESSAGE
(LABEL "Resposta"
    CONTENT "Resultado execução transação"
    SENDER "Gerência transação eq rede fixa"
    RECEIVER "Gerência mobilidade transação eq rede fixa"
)
MESSAGE
(LABEL "Solicitação"
    CONTENT "Cliente móvel a localizar"
    SENDER "Gerência mobilidade transação eq rede fixa"
    RECEIVER "Interação sistema localização"
)
MESSAGE
(LABEL "Resposta"
    CONTENT "Localização cliente móvel"
    SENDER "Interação sistema localização"
    RECEIVER "Gerência mobilidade transação eq rede fixa"
)
MESSAGE
(LABEL "Aviso"
    CONTENT "Retorno transação"
    SENDER "Gerência mobilidade transação eq rede fixa"
    RECEIVER "Gerência transação eq rede fixa"
)

PROTOCOL MudançaDeAmbiente
MESSAGE
(LABEL "Role_deactivate"
    SENDER "Gerência mobilidade transação eq rede fixa"
    RECEIVER "Gerência mobilidade transação eq rede fixa"
)
MESSAGE
(LABEL "Role_change"
    SENDER "Gerência mobilidade transação eq rede fixa"
    RECEIVER "Gerência mobilidade transação eq móvel"
```

)
)

AGENT

(Gerente transação eq rede fixa

GOAL ("Gerenciamento transação eq rede fixa"
RELATED TO PLAN "Executando transação eq rede fixa"
)

PLAN (Executando transação eq rede fixa
COMPOSED OF ACTION "Registrar início transação eq rede fixa"
COMPOSED OF ACTION "Executar aplicação"
COMPOSED OF ACTION "Registrar fim transação eq rede fixa"
COMPOSED OF ACTION "Registrar envio resultado transação eq móvel"
RELATED TO GOAL "Gerenciamento transação eq rede fixa"
)

ACTION Registrar início transação eq rede fixa
ACTION Executar aplicação
ACTION Registrar fim transação eq rede fixa
ACTION Registrar envio resultado transação eq móvel

)

AGENT ROLE

(Gerência transação eq rede fixa

ROLEGOAL ("Gerenciamento transação eq rede fixa"
)

DUTY "Executar aplicação"

PROTOCOL ExecuçãoAplicação

MESSAGE

(LABEL "Solicitação"

CONTENT "Aplicação a executar"

SENDER "Gerência transação eq rede fixa"

RECEIVER "Execução transação local"

```
)
  MESSAGE
    (LABEL "Resposta"
      CONTENT "Resultado execução transação"
      SENDER "Execução transação local"
      RECEIVER "Gerência transação eq rede fixa"
    )
)

AGENT
( Comunicador sistema localização

  GOAL ("Gerenciamento interação sistema localização cliente móvel"
    RELATED TO PLAN "Localizando cliente móvel"
  )

  PLAN (Localizando cliente móvel
    COMPOSED OF ACTION "Localizar cliente móvel"
    RELATED TO GOAL "Gerenciamento interação sistema localização
      cliente móvel"
  )

  ACTION Localizar cliente móvel
)

AGENT ROLE
( Interação sistema localização

  ROLEGOAL ("Gerenciamento interação sistema localização cliente móvel"
  )

  DUTY "Localizar cliente móvel"
)
```

Entidades Comuns aos Ambientes Móvel e de Rede Fixa

A seguir, estão apresentadas, na *gramática* MAS-ML, as especificações das entidades comuns aos dois ambientes.

AGENT

(Gerente mobilidade transação

```
GOAL ("Gerenciamento deslocamento ambiente rede móvel"
      RELATED TO PLAN "Deixando ambiente móvel"
      RELATED TO PLAN "Encerrando transação eq móvel"
    )
GOAL ("Gerenciamento deslocamento ambiente rede fixa"
      RELATED TO PLAN "Recebendo transação eq rede fixa"
      RELATED TO PLAN "Deixando ambiente rede fixa"
    )

BELIEF ("IdEqMóvel")
BELIEF ("IdTransação")
BELIEF ("Transação")
BELIEF ("ResultadoTransação")
BELIEF ("IdEqRedeFixa")

PLAN (Deixando ambiente móvel
      COMPOSED OF ACTION "Obter dados eq móvel"
      COMPOSED OF ACTION "Procurar ambiente rede fixa"
      RELATED TO GOAL "Gerenciamento deslocamento ambiente rede móvel"
    )
PLAN (Encerrando transação eq móvel
      COMPOSED OF ACTION "Encerrar transação eq móvel"
      RELATED TO GOAL "Gerenciamento deslocamento ambiente rede móvel"
    )
PLAN (Recebendo transação eq rede fixa
      COMPOSED OF ACTION "Obter dados ambiente rede fixa"
      COMPOSED OF ACTION "Executar transação eq rede fixa"
      RELATED TO GOAL "Gerenciamento deslocamento ambiente rede fixa"
    )
PLAN (Deixando ambiente rede fixa
      COMPOSED OF ACTION "Localizar cliente móvel"
      COMPOSED OF ACTION "Encerrar transação eq rede fixa"
      RELATED TO GOAL "Gerenciamento deslocamento ambiente rede fixa"
    )
```

```
    ACTION Obter dados eq móvel
    ACTION Procurar ambiente rede fixa
    ACTION Encerrar transação eq móvel
    ACTION Obter dados ambiente rede fixa
    ACTION Executar transação eq rede fixa
    ACTION Localizar cliente móvel
    ACTION Encerrar transação eq rede fixa
)

AGENT
( Executor transação local

    GOAL ("Execução transação local"
        RELATED TO PLAN "Executando aplicação"
    )

    PLAN (Executando aplicação
        COMPOSED OF ACTION "Executar aplicação"
        RELATED TO GOAL "Execução transação local"
    )

    ACTION Executar aplicação
)

AGENT ROLE
( Execução transação local

    ROLEGOAL ("Execução transação local"
    )

    DUTY "Executar aplicação"
)
```

Diagrama de Classes do Framework Proposto

O *Diagrama de Classes* modela agentes, organizações e ambientes. Representa os relacionamentos entre classes e ambientes, classes e agentes e classes e organizações. Também representa os relacionamentos entre agentes, entre ambientes e entre organizações. Esse diagrama permite modelar

as classes que habitam os ambientes, as classes associadas a agentes e organizações, a associação entre classes de ambientes e a especialização entre agentes, organizações e ambientes.

A figura 5.8 apresenta o *diagrama de classes do framework proposto*.

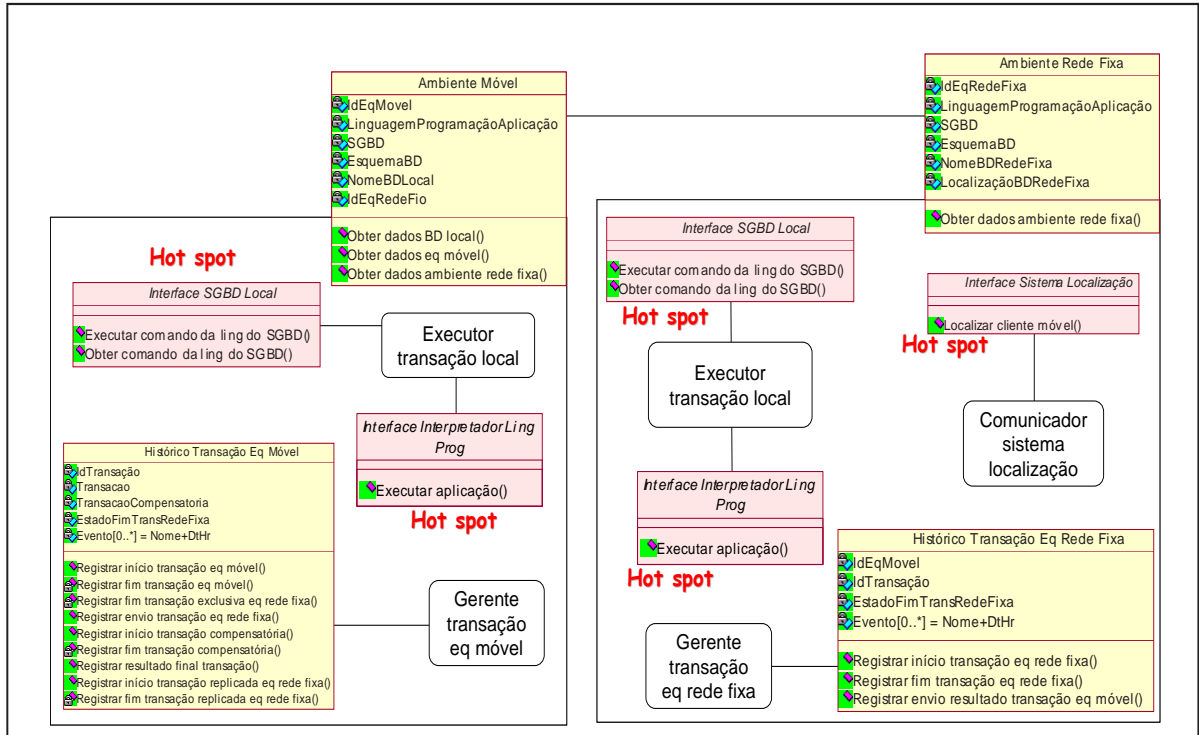


Figura 5.8: Diagrama de Classes do framework proposto

No **Ambiente Móvel**, as classes *Interface SGBD Local* e *Interface Interpretador Ling Prog* correspondem aos **hot spots** do framework, ou seja, devem ser configuradas com as funcionalidades específicas desejadas para uma determinada instanciação do framework. O framework proposto permite que os dados acessados por aplicações móveis sejam gerenciados por qualquer SGBD. Para que um determinado SGBD seja usado na arquitetura proposta, basta que se implemente para ele uma subclasse da classe *Interface SGBD Local*, redefinindo seus serviços para anexar o SGBD específico.

O framework proposto permite que as aplicações móveis sejam implementadas em qualquer linguagem de programação. Para que uma determinada linguagem de programação possa ser usada na arquitetura proposta, basta que se implante para ela uma subclasse da classe *Interface Interpretador Ling Prog*, redefinindo seus serviços para interagir com a linguagem específica.

Os demais elementos representados no diagrama correspondem aos **frozen spots** do framework, ou seja, conservam-se inalteráveis em qualquer

de suas instanciações.

Gerente Interação aplicação móvel, *Gerente transação eq móvel* e *Executor transação local* correspondem aos **agentes fixos**, enquanto *Gerente mobilidade transação* é o **agente móvel** que migrará para o ambiente de rede fixa quando a transação tiver que ser executada nesse ambiente. O agente fixo *Executor transação local* também aparece no Ambiente Rede Fixa. Cabe a ele tratar a execução da transação no SGBD existente no local.

Para a execução de suas ações, o agente *Gerente transação eq móvel* usa a classe *Histórico Transação Eq Móvel*, enquanto o agente *Executor transação local* usa as classes *Interface Interpretador Ling Prog* e *Interface SGBD Local*.

No **Ambiente Rede Fixa**, as classes *Interface SGBD Local*, *Interface Interpretador Ling Prog* e *Interface Sistema Localização* correspondem aos **hot spots** do framework, ou seja, devem ser configurados com as funcionalidades específicas desejadas para uma determinada instanciação do framework. Os hot spots *Interface SGBD Local* e *Interface Interpretador Ling Prog* são os mesmos do Ambiente Móvel, tendo sido discutidos na seção anterior.

O framework proposto permite que o gerenciamento da localização dos clientes móveis seja efetuado por qualquer tipo de sistema de localização. Para que um determinado sistema de localização seja usado na arquitetura proposta, basta que se implemente para ele uma subclasse da classe *Interface Sistema Localização*, redefinindo seus serviços para interagir com sistemas específicos. Os demais elementos representados no diagrama correspondem aos **frozen spots** do framework, ou seja, conservam-se inalteráveis em qualquer de suas instanciações.

Gerente Transação eq Rede Fixa, *Executor Transação local* e *Comunicador sistema localização* correspondem aos **agentes fixos**, enquanto *Gerente mobilidade transação* é o **agente móvel**, que migrará do ambiente móvel para que a transação móvel seja executada no ambiente de rede fixa.

Para a execução de suas ações, o agente *Gerente transação eq rede fixa* usa a classe *Histórico Transação Eq Rede Fixa*, enquanto o agente *Comunicador sistema localização* usa a classe *Interface Sistema Localização*. O agente *Executor transação local* usa as classes *Interface Interpretador Ling Prog* e *Interface SGBD Local*, conforme mencionado na seção anterior.

Diagrama de Papéis do Framework Proposto

O *Diagrama de Papéis* modela os papéis de objetos, papéis de agentes e classes. O digrama de papéis representa os relacionamentos entre os papéis de agentes e papéis de objetos identificados nos diagramas da organização. Esse diagrama também identifica as classes acessadas pelos papéis de agentes e papéis de objetos.

A figura 5.9 apresenta o *diagrama de papéis dos agentes do framework proposto*.

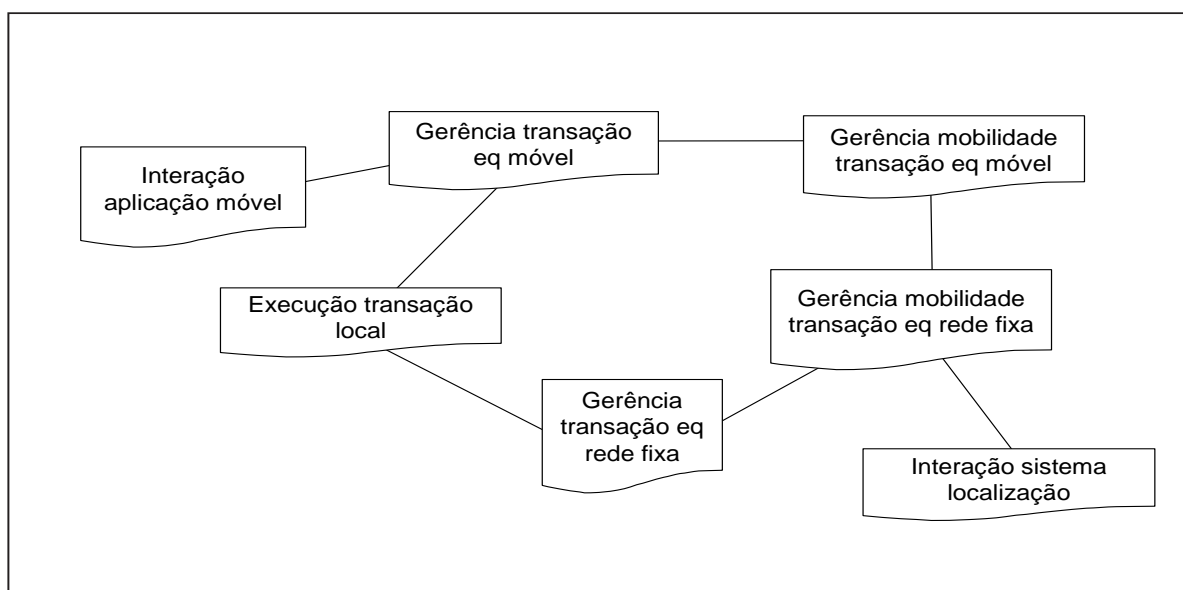


Figura 5.9: Diagrama de Papéis do framework proposto

5.3.3 Modelos Dinâmicos do Framework

Esta seção tem como objetivo apresentar o comportamento do framework através das interações entre as entidades de cada ambiente, mostrando como essas entidades colaboram entre si para a execução das transações móveis nos bancos de dados localizados no equipamento móvel e na rede fixa.

Conforme proposto em [17], o comportamento do framework foi representado através de diagramas de seqüência estendidos pela MAS-ML. Vale ressaltar que, em MAS-ML, a mudança de um agente de um ambiente para outro foi representada, nos modelos dinâmicos (diagramas de seqüência), através de dois estereótipos: <<role_deactivate>> e <<role_change>>. Para mover-se para um outro ambiente, o agente precisa parar de de-

sempenhar papéis no ambiente que ele quer abandonar. O estereótipo <<role_deactivate>> representa a desativação dos papéis desempenhados no ambiente corrente. O agente, então, identifica o ambiente para o qual ele quer ir, negocia sua entrada em uma organização do novo ambiente e se associa a um papel do ambiente de destino. O movimento do ambiente de origem para o ambiente de destino é representado pelo estereótipo <<role_change>>. A negociação entre o agente e a organização do novo ambiente está embutida nesse estereótipo.

O nome das instâncias apresentadas nos diagramas de seqüência segue a lei de formação estabelecida em [17]: <nome do agente> <nome do papel> <nome da organização> <nome do ambiente>.

As figuras de 5.10 a 5.17 apresentam os diagramas de seqüência com esse comportamento.

Caso de Uso Execução Transação Móvel

O diagrama de seqüência definido na figura 5.10 detalha o comportamento do caso de uso Execução Transação Móvel. O início da transação móvel é registrado no histórico de transações, do equipamento móvel, e é verificada a existência de banco de dados, no equipamento móvel, para decidir que tratamento deve ser dado à transação móvel.

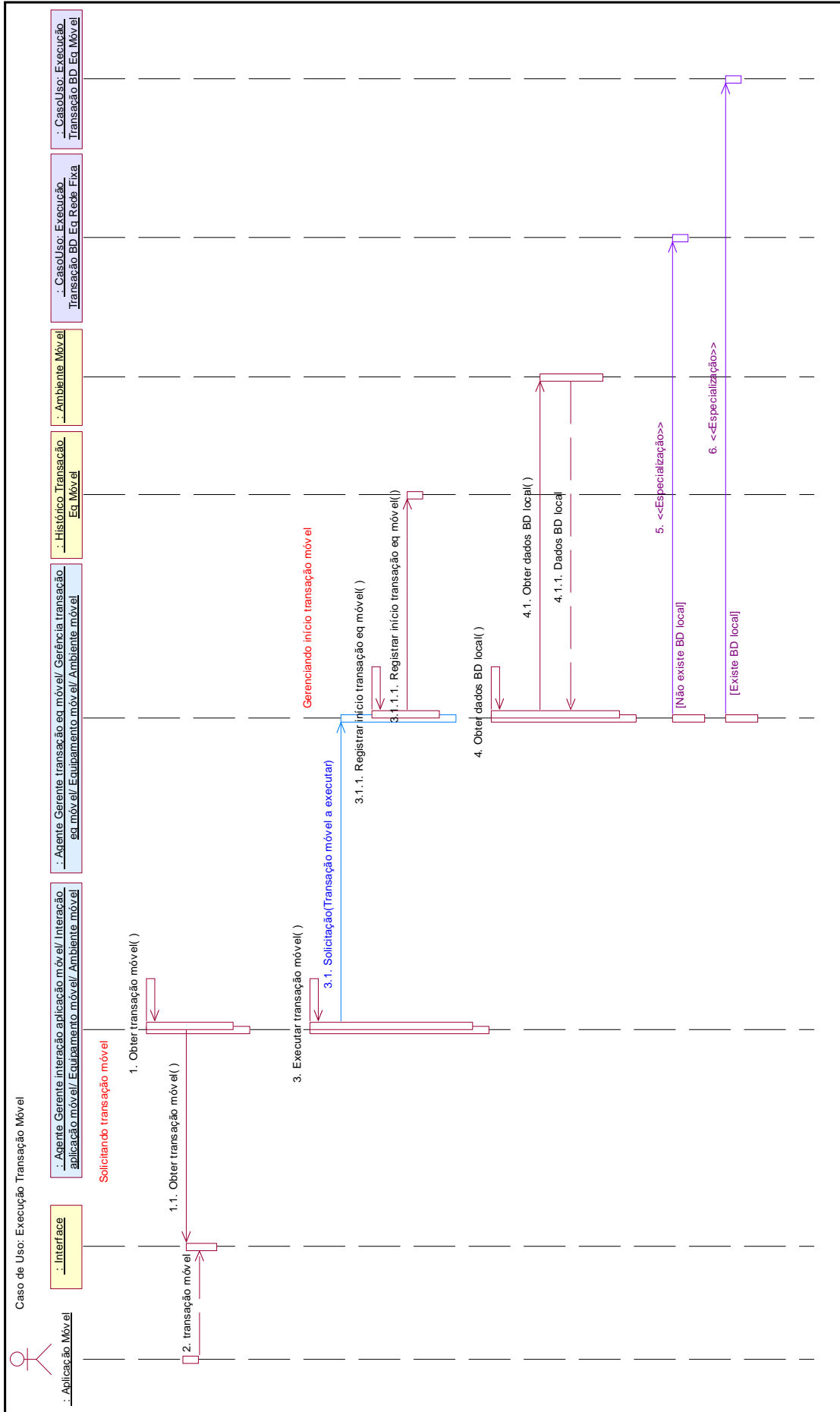


Figura 5.10: Diagrama de Seqüência Execução Transacção Móvel

Caso de Uso Execução Transação BD Eq Rede Fixa

O caso de uso execução transação BD Rede Fixa foi detalhado em três diagramas de seqüência: figuras 5.11, 5.12 e 5.13. Esse caso de uso refere-se à situação em que a transação móvel é encaminhada para ser executada no equipamento da rede fixa ou porque não existe banco de dados no equipamento móvel ou porque a transação foi executada e terminou com sucesso no equipamento móvel.

O diagrama de seqüência da figura 5.11 representa o processamento realizado no equipamento móvel para o envio da transação para ser executada no equipamento da rede fixa. O *agente móvel* se desliga do ambiente móvel e se desloca para o ambiente rede fixa, carregando a transação a ser executada lá.

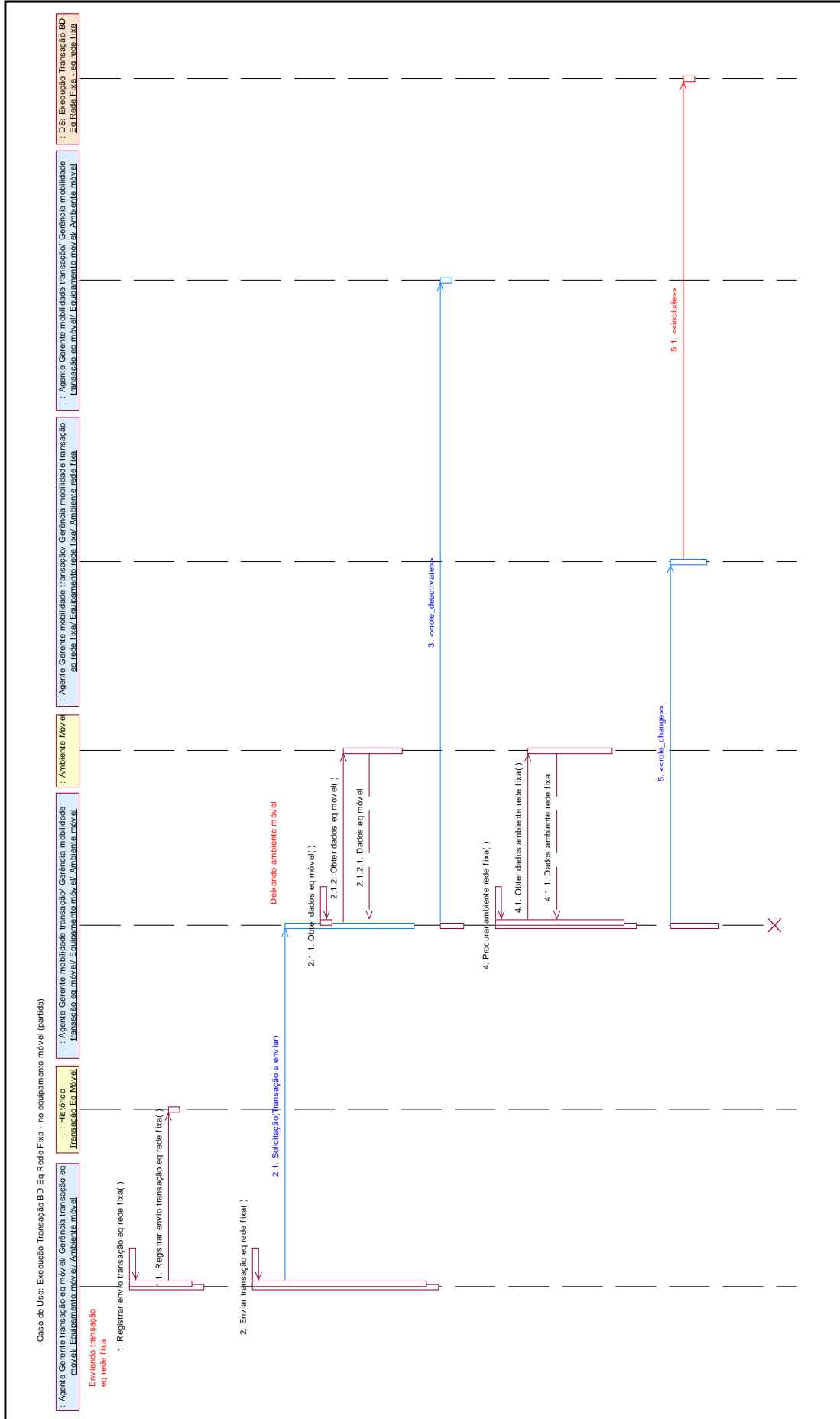


Figura 5.11: Diagrama de Seqüência Execução Transação BD Eq. Rede Fixa - no equipamento móvel (partida)

O diagrama de seqüência da figura 5.12 representa a *chegada do agente móvel* no equipamento da rede fixa, trazendo a transação a ser executada, o encaminhamento da transação para execução no SGBD local e o *retorno do agente móvel* para o ambiente móvel, levando o resultado da transação executada.

O diagrama de seqüência da figura 5.13 representa a *chegada do agente móvel* no ambiente móvel, disparando o encerramento da transação móvel.

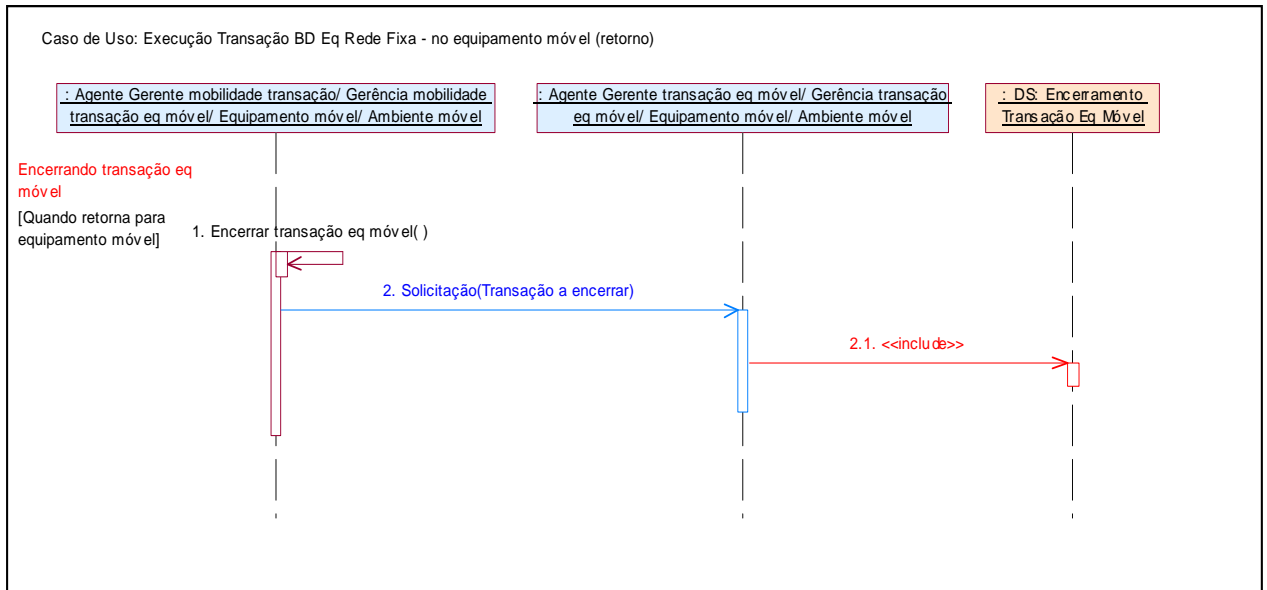


Figura 5.13: Diagrama de Seqüência Execução Transação BD Eq Rede Fixa - no equipamento móvel (retorno)

Caso de Uso Execução Transação BD Eq Móvel

O diagrama de seqüência da figura 5.14 detalha o comportamento do caso de uso Execução Transação BD Eq Móvel. Quando existe banco de dados no equipamento móvel, a transação móvel é encaminhada para ser executada no SGBD local e depois é encerrada.

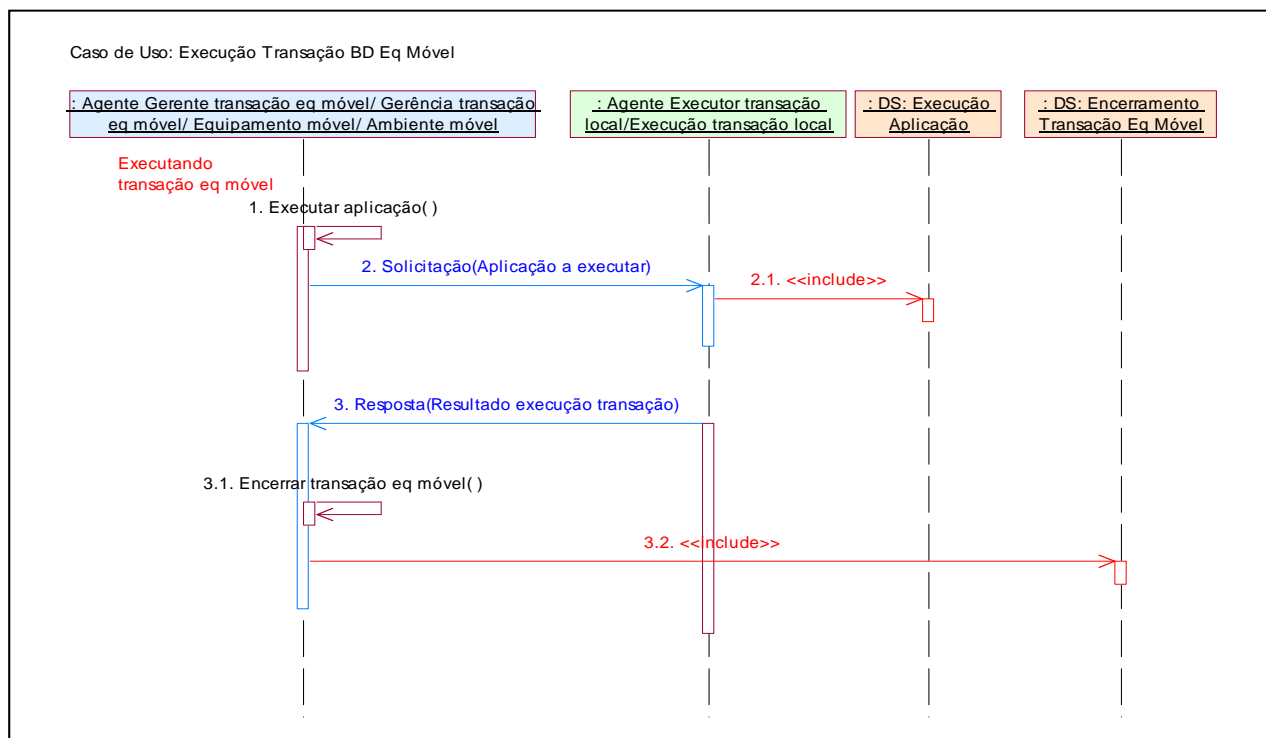


Figura 5.14: Diagrama de Seqüência Execução Transação BD Eq Móvel

Diagramas de Seqüência Complementares

Devido à complexidade dos diagramas de seqüência que detalham os casos de usos, bem como à existência de funcionalidades comuns a eles, foram criados diagramas de seqüência complementares, objetivando melhorar a legibilidade dos modelos e o compartilhamento de funcionalidades comuns.

O diagrama de seqüência da figura 5.15 representa o encaminhamento da transação para ser executada pelo SGBD local.

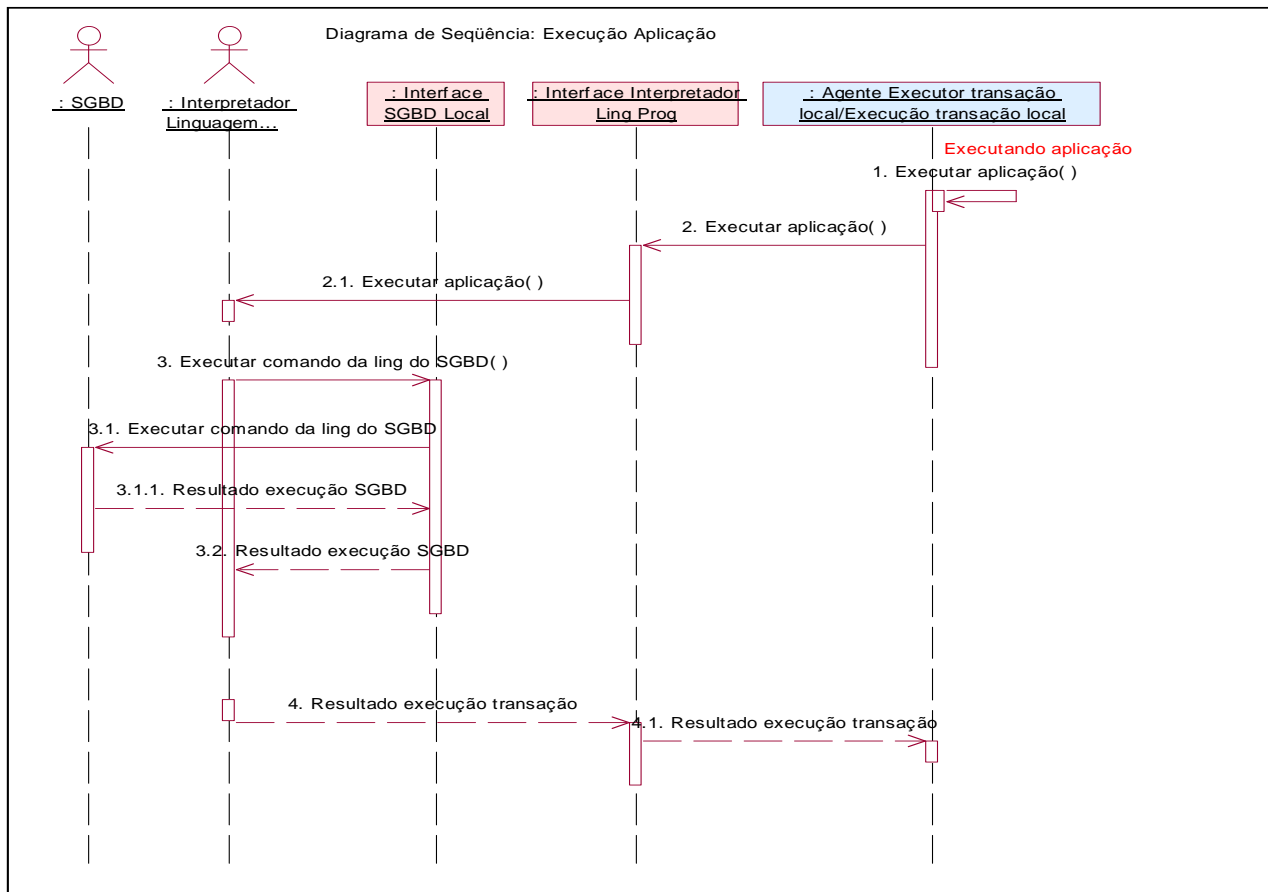


Figura 5.15: Diagrama de Seqüência Execução Aplicação

O diagrama de seqüência da figura 5.16 representa a interação com o sistema de localização para obter a localização do cliente móvel, de forma que o *agente móvel* possa retornar ao equipamento móvel, com o resultado da transação executada no equipamento da rede fixa.

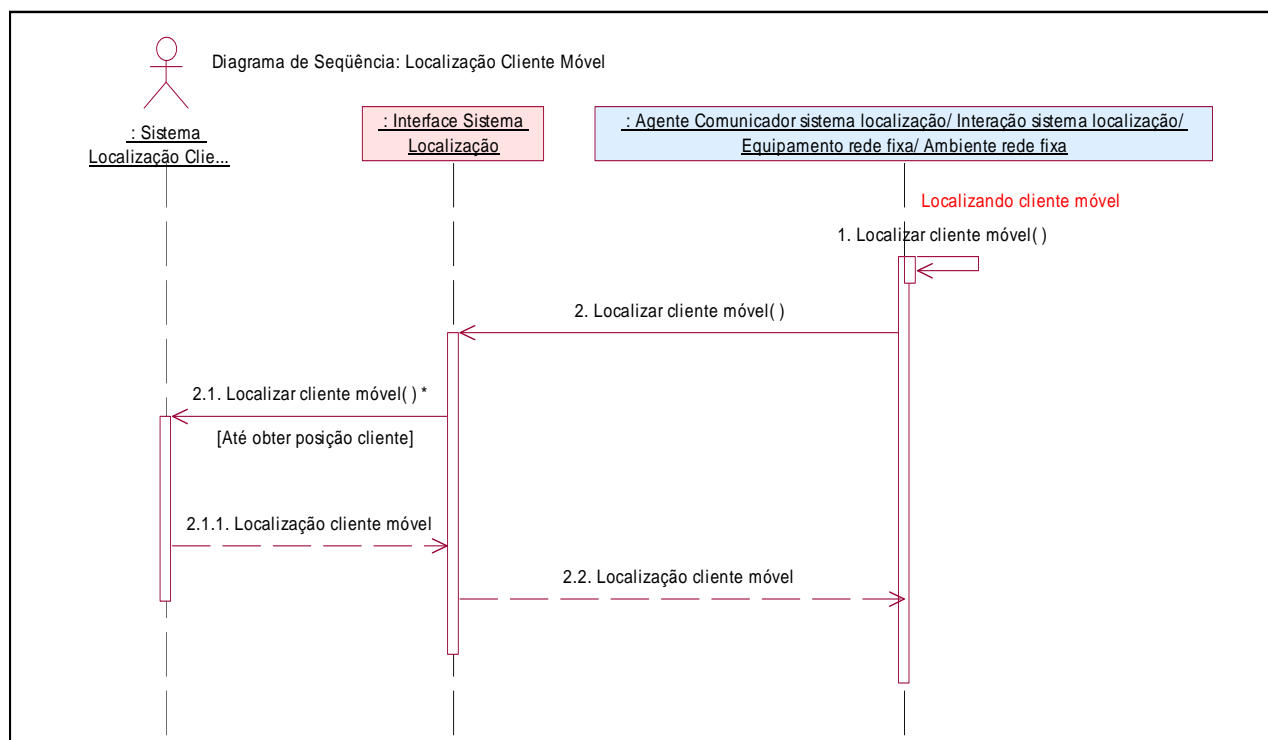


Figura 5.16: Diagrama de Seqüência Localização Cliente Móvel

O diagrama de seqüência da figura 5.17 representa o encerramento da transação móvel no equipamento móvel.

Caso a transação móvel tenha sido executada apenas no equipamento da rede fixa ou apenas no equipamento móvel, o resultado de seu processamento será retornado para a aplicação móvel.

Caso a transação móvel tenha sido executada no equipamento móvel e terminada com sucesso (*commit*), a transação será encaminhada para execução no equipamento da rede fixa. No diagrama de seqüência, essa nova transação é identificada como *Transação replicada eq. rede fixa*, que corresponde à *Transação Móvel Global (TMG)* do modelo de transações proposto no capítulo 4.

Caso a transação replicada para a rede fixa, ou seja, a transação que terminou com sucesso no equipamento móvel e que foi encaminhada para execução no equipamento da rede fixa, não tenha sido terminada com sucesso no ambiente de rede fixa, avalia-se o motivo do insucesso. Se for por incompatibilidade de esquemas dos bancos de dados, a transação móvel terá que ter seus efeitos desfeitos semanticamente através da execução de uma transação compensatória, no equipamento móvel. A transação compensatória não pode falhar, e é ressubmetida até que a conclua com sucesso.

Caso a transação replicada tenha encerrado com insucesso por qualquer outra razão que não incompatibilidade de esquema, o administrador do banco de dados, DBA, é avisado e ela é reenviada para o equipamento rede fixa até que encerre com sucesso nesse ambiente.

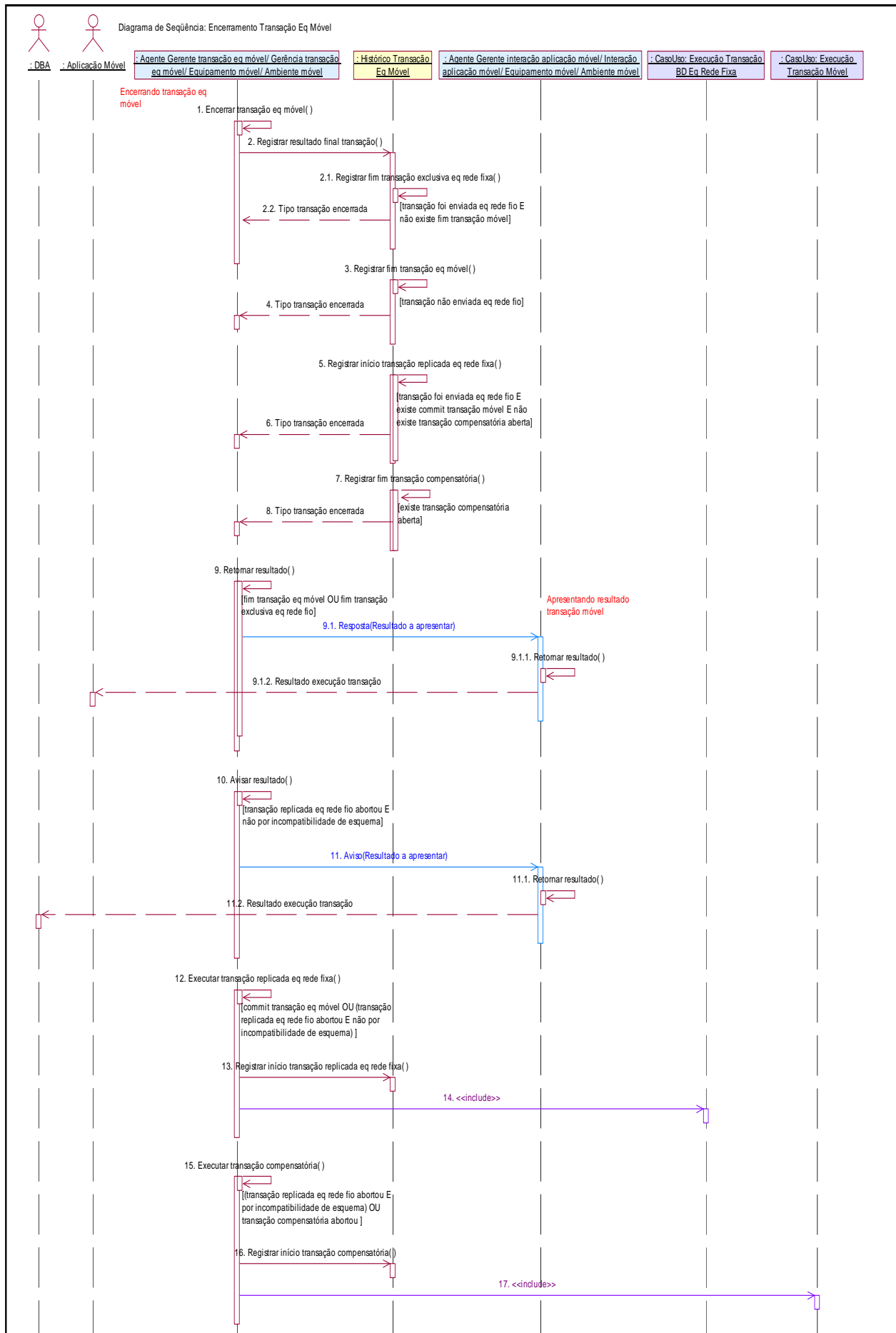


Figura 5.17: Diagrama de Seqüência Encerramento Transação Eq Móvel

5.3.4 Instanciação do Framework

Para que se possa utilizar o framework proposto para a execução de transações móveis, torna-se necessário preencher-se seus *hot spots*, de forma a se obter um sistema completo e executável.

As figuras 5.18 e 5.19 apresentam a instanciação do framework para atender às seguintes características:

- **Interface SGBD Local** - foi criada uma especialização dessa classe (classe *Interface Tino*) para que o SGBD Tino possa ser utilizado nos ambientes móvel e de rede fixa.
- **Interface Interpretador Linguagem de Programação** - para que as aplicações implementadas na linguagem de programação Java possam submeter transações móveis, uma subclasse (classe *Interface Java*) da classe *Interface Interpretador Ling Prog* foi definida.
- **Interface Sistema Localização** - o sistema de comunicação do ambiente móvel é o GSM (Global System for Mobile), largamente usado na Europa e agora no Brasil. Para recuperar a localização do cliente móvel, criou-se a classe *Interface Sistema GSM* como uma especialização da classe *Interface Sistema Localização*.

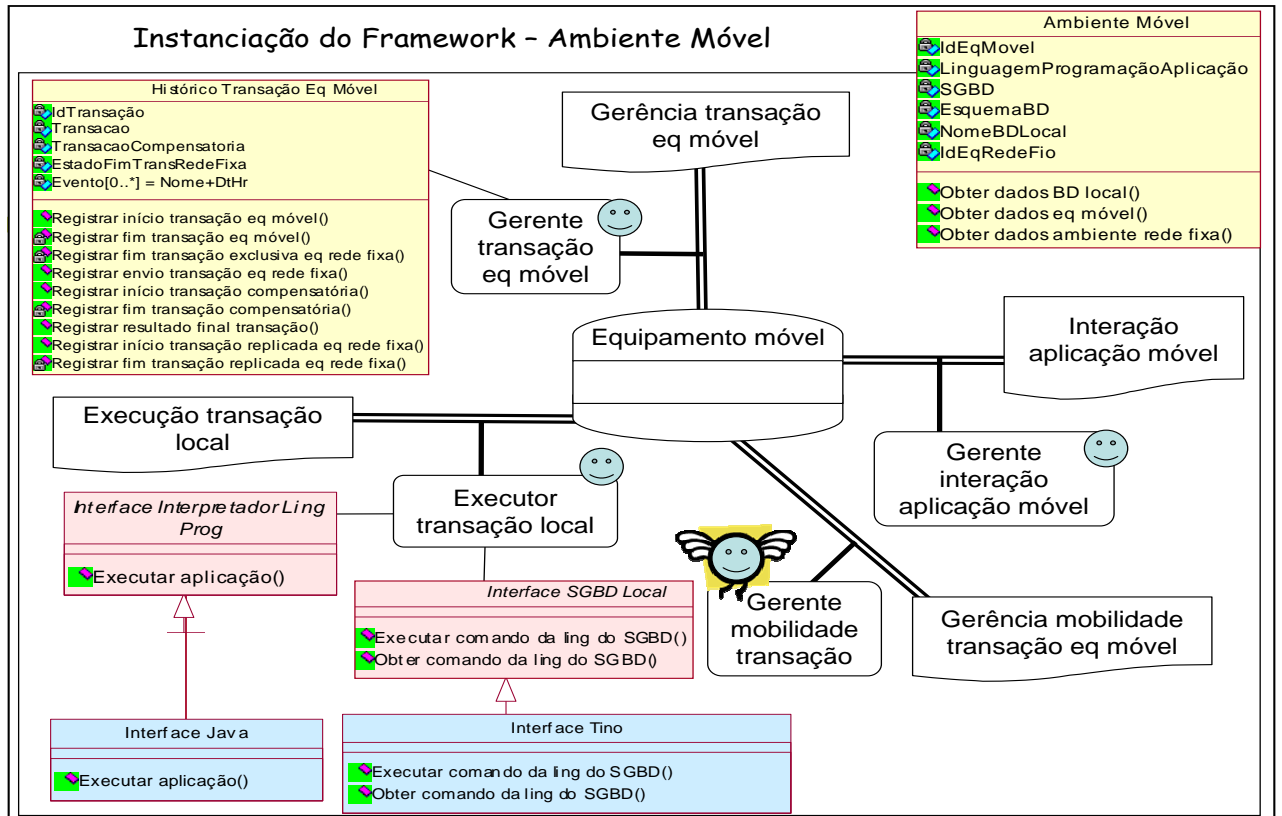


Figura 5.18: Diagrama Estrutural do ambiente móvel do framework proposto instanciado

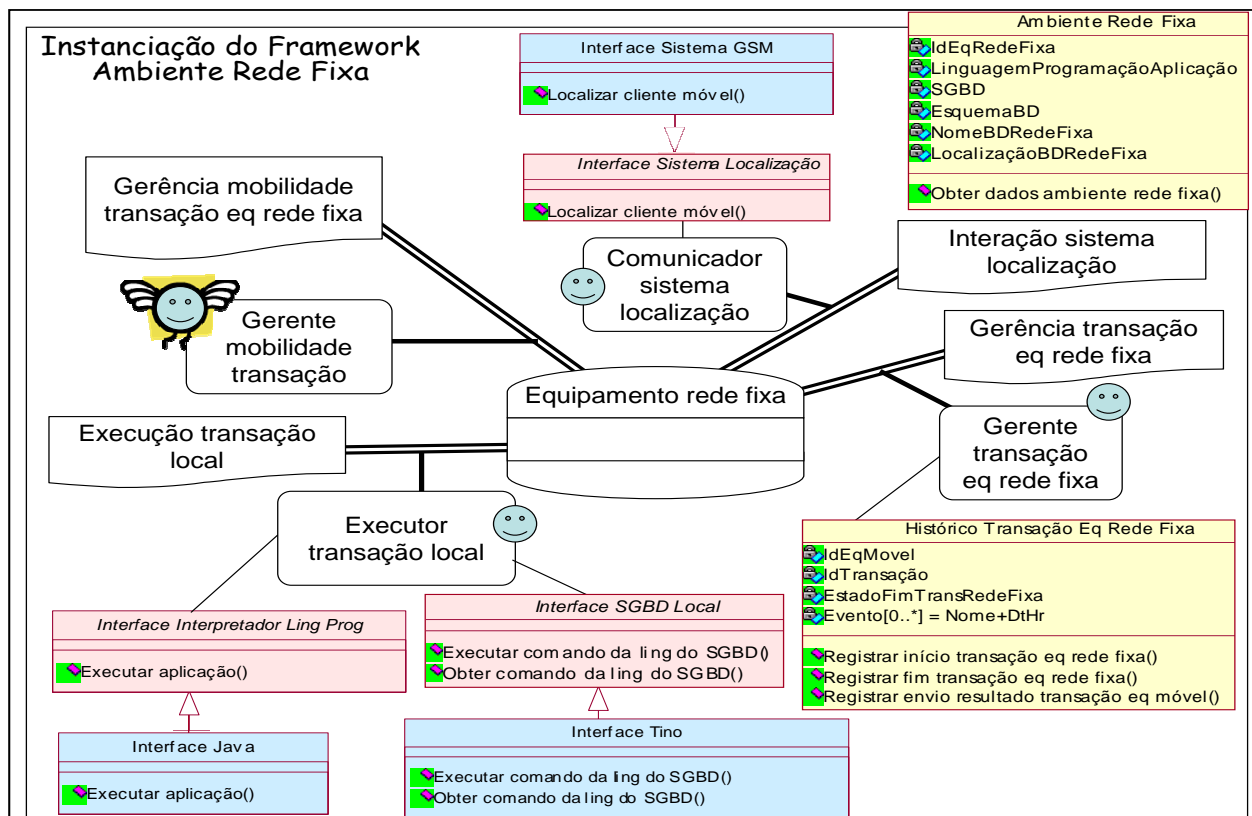


Figura 5.19: Diagrama Estrutural do ambiente de rede fixa do framework proposto instanciado

5.4

Garantia das Propriedades do Modelo Proposto pelo Framework

No capítulo 4 foi apresentado o modelo de transações proposto, e neste capítulo apresenta-se como esse modelo será implementado em um framework utilizando a tecnologia de agentes de software. A seguir será apresentado como as propriedades do modelo de transações proposto serão garantidas pelo framework.

5.4.1

Garantia das Propriedades da Transação Móvel (TM)

- *Dependência Forte de Confirmação* - O Agente "Gerente transação eq móvel", em seu plano "Encerrando transação eq móvel", possui a ação de "Encerrar transação eq móvel". Esta ação verifica se a TMG foi confirmada (*committed*) com sucesso e, em caso positivo, confirma a TM.

- *Dependência de Cancelamento* - O Agente "Gerente transação eq móvel", em seu plano "Encerrando transação eq móvel", possui a ação de "Encerrar transação eq móvel". Esta ação verifica se a TMG foi cancelada (*aborted*) e, em caso afirmativo, cancela a TM.
- *Dependência de Término* - O Agente "Gerente transação eq móvel", em seu plano "Encerrando transação eq móvel", possui a ação de "Encerrar transação eq móvel". Esta ação verifica se a TMG foi confirmada ou cancelada na rede fixa e, somente após essa verificação, permite o término da TM.
- *Dependência Serial* - O Agente "Gerente transação eq móvel", em seu plano "Gerenciando início transação móvel", possui a ação de "Registrar início da transação eq móvel". Esta ação registra no histórico móvel o início da transação T_i e libera essa transação para a execução de forma serial da transação T_{i-1} .

5.4.2

Garantia das Propriedades da Transação Móvel Local Compensável (TMLC)

- *Dependência de Cancelamento* - O Agente "Gerente transação eq móvel", em seu plano "Encerrando transação eq móvel", possui a ação de "Encerrar transação eq móvel". Esta ação verifica se a TMG foi confirmada ou cancelada. Caso tenha sido cancelada por divergência entre os esquemas dos bancos de dados móvel e da rede fixa, a TMLC será cancelada, e será invocada a ação de "Executar transação compensatória".
- *Dependência Serial* - O Agente "Gerente transação eq móvel", em seu plano "Gerenciando início transação móvel", possui a ação de "Registrar início da transação eq móvel". Esta ação registra no histórico móvel o início da transação TMLC_{*i*} e libera essa transação para a execução de forma serial da transação TMLC_{*i-1*}, invocando o agente "Executor transação local". Como cada TM_{*i*} gera uma correspondente TMLC_{*i*} e uma TMG_{*i*}, uma fila única de TMLCs e TMGs é gerada, e a cada término de uma TMLC_{*i*} sua correspondente TMG_{*i*} é liberada para execução na rede fixa, utilizando-se as ações do plano "Enviando transação eq rede fixa" desse mesmo agente.

5.4.3

Garantia das Propriedades da Transação Móvel Global (TMG)

- *Dependência de Confirmação* - O Agente "Gerente transação eq móvel", em seu plano "Enviando transação eq rede fixa", possui as ações de "Registrar envio da transação eq rede fixa" e "Enviar transação eq rede fixa". A primeira ação verifica se a correspondente $TMLC_i$ foi confirmada (*committed*). Caso a confirmação da $TMLC_i$ tenha ocorrido com sucesso, a segunda ação é executada.
- *Dependência Forte de Confirmação* - O Agente "Gerente transação eq móvel", em seu plano "Encerrando transação eq rede fixa", possui a ação de "Encerrar transação eq móvel". Esta ação verifica se a correspondente TCM_i foi confirmada na rede fixa. Caso a confirmação da TCM_i tenha ocorrido com sucesso, a sua correspondente TMG_i pode ser confirmada.
- *Dependência de Cancelamento* - O Agente "Gerente transação eq móvel", em seu plano "Encerrando transação eq rede fixa", possui a ação de "Encerrar transação eq móvel". Esta ação verifica se a correspondente TCM_i foi cancelada na rede fixa. Caso o cancelamento da TCM_i tenha ocorrido, a sua correspondente TMG_i deverá, também, ser cancelada. Caso o motivo do cancelamento não seja por divergência de esquemas entre o banco de dados móvel e o banco de dados da rede fixa, a ação "Executar transação replicada eq rede fixa" é então invocada para execução.
- *Dependência Término* - O Agente "Gerente transação eq móvel", em seu plano "Encerrando transação eq rede fixa", possui a ação de "Encerrar transação eq móvel". Esta ação verifica se a correspondente TCM_i foi concluída na rede fixa. Caso a TCM_i não tenha sido concluída, a sua correspondente TMG_i também não poderá ser concluída.
- *Dependência Fraca de Início na Confirmação* - O Agente "Gerente transação eq móvel", em seu plano "Enviando transação eq rede fixa", possui a ação de "Registrar envio da transação eq rede fixa". Esta ação verifica se a $TMLC_i$ correspondente à TMG_i foi confirmada. Caso a confirmação da $TMLC_i$ tenha ocorrido com sucesso, então TMG_i pode iniciar a sua execução na rede fixa.

5.4.4

Garantia das Propriedades da Transação Móvel Local Compensatória (TMLCP)

- *Dependência de Início na Confirmação* - O Agente "Gerente transação eq móvel", em seu plano "Encerrando transação eq móvel", possui a ação de "Executar transação compensatória". Esta ação verifica se a $TMLC_i$ correspondente foi confirmada e se a TMG_i não foi confirmada. Caso essas duas condições estejam satisfeitas, a $TMLCP_i$ pode ser executada.
- *Dependência de Início de Cancelamento* - O Agente "Gerente transação eq móvel", em seu plano "Encerrando transação eq móvel", possui a ação de "Executar transação compensatória". Esta ação verifica se a TMG_i correspondente foi cancelada. Caso o cancelamento tenha ocorrido por divergência entre os esquemas dos bancos de dados, a $TMLCP_i$ é então executada.

5.4.5

Garantia das Propriedades da Transação Coordenadora de Mobilidade (TCM)

- *Dependência Forte de Confirmação* - O Agente "Gerente transação eq rede fixa", em seu plano "Executando transação eq rede fixa", possui a ação de "Registrar fim transação eq rede fixa". Esta ação verifica se a $TCBD_i$ e a $TSCM_i$ foram confirmadas. Caso ambas as transações tenham sido confirmadas, então a TCM_i também será confirmada.
- *Dependência de Cancelamento* - O Agente "Gerente transação eq rede fixa", em seu plano "Executando transação eq rede fixa", possui a ação de "Registrar fim transação eq rede fixa". Esta ação verifica se a $TCBD_i$ ou a $TSCM_i$ foram canceladas. Caso uma das transações tenha sido cancelada, a TCM_i também será cancelada.
- *Dependência de Término* - O Agente "Gerente transação eq rede fixa", em seu plano "Executando transação eq rede fixa", possui a ação de "Registrar fim transação eq rede fixa". Esta ação não permite a confirmação da TCM_i até que as suas $TCBD_i$ ou $TSCM_i$ correspondentes tenham terminado as suas execuções.

5.4.6

Garantia das Propriedades da Transação Coordenadora de Banco de Dados (TCBD)

- *Dependência Forte de Confirmação* - O Agente "Gerente transação eq rede fixa", em seu plano "Executando transação eq rede fixa", possui a ação de "Registrar fim transação eq rede fixa". Esta ação verifica se todas as TL_{ij} , $j=1..n$, foram confirmadas. Caso isso aconteça, então a TCBD também é confirmada.
- *Dependência de Cancelamento* - O Agente "Gerente transação eq rede fixa", em seu plano "Executando transação eq rede fixa", possui a ação de "Registrar fim transação eq rede fixa". Esta ação verifica se pelo menos uma das TL_{ij} , $j=1..n$, foi cancelada. Caso isso aconteça, então a TCBD também é cancelada.

5.4.7

Garantia das Propriedades da Transação de Suporte à Mobilidade do Cliente (TSCM)

- *Dependência Serial* - O Agente "Comunicador sistema localização", em seu plano "Localizando cliente móvel", possui a ação de "Localizar Cliente Móvel". Esta ação verifica se a $TCBD_i$ correspondente foi concluída. Caso isso já tenha ocorrido, então a $TSCM_i$ pode iniciar a sua execução.

5.4.8

Garantia das Propriedades da Transação Local (TL)

- *Dependência de Cancelamento* - O Agente "Executor transação local", em seu plano "Executar aplicação", possui a ação de "Execução transação local". Esta ação será invocada pelo agente "Gerente transação eq rede fixa" sempre que a $TCBD_i$ tenha sido cancelada e cancelará todas as TL_{in} .
- *Dependência Serial* - O Agente "Executor transação local", em seu plano "Executar aplicação", possui a ação de "Execução transação local". Esta ação preserva a ordenação das transações TL_{in} .

5.4.9 Um Exemplo de Aplicação do Framework

Para melhor esclarecer a utilização de um software, instanciado a partir do framework proposto, será considerado o *Modelo Lógico Relacional do Banco de Dados* apresentado no Anexo C (figura C.1).

Um vendedor, atendendo à solicitação de um cliente, faz um pedido de cinco produtos. O processamento desse pedido de compra está detalhado a seguir. O Anexo C contém as interfaces dessa aplicação.

Pedido de Compra:

1. Vendedor se identifica na aplicação;
2. Vendedor identifica o Cliente na aplicação;
3. Vendedor informa a **Forma de Pagamento do Pedido** e a aplicação, utilizando algoritmos próprios, gera o **Número do Pedido** e a **Data da Venda do Pedido**. O **Valor Total do Pedido** será calculado ao final da seleção dos produtos;
4. Repete para cada produto solicitado pelo Cliente;
 - (a) Recupera o produto na tabela *Produto*;
 - (b) Informa a **Quantidade** do produto selecionado;
 - (c) Prepara uma linha (registro) para a tabela *Produto Encomendado*. A aplicação calcula o **Valor Total do Item do Produto**;
5. Calcula o **Valor Total do Pedido**;
6. Encerra o *Pedido de Compra*.
7. Grava uma linha na tabela *Pedido de Compra*;
8. Grava todas as linhas de produtos solicitados na tabela *Produto Encomendado*;
9. Atualiza, para cada produto solicitado, o valor armazenado no atributo **Quantidade em Estoque** da tabela *Produto*, subtraindo do seu valor, o valor armazenado no atributo **Valor Total do Item do Produto**.

Esse conjunto de instruções, devidamente preparado pela aplicação móvel, gera uma *Transação Móvel (TM)* de banco de dados que deverá ser executada no SGBD localizado no equipamento móvel (caso exista) e no SGBD localizado na rede fixa. Assim, em linhas gerais, o processamento da instanciação do framework proposto, para execução dessa transação, simulando uma interação entre seus agentes e supondo a existência de um banco de dados no equipamento móvel, seria:

1. O ator *Aplicação Móvel* informa uma **Transação Móvel (TM)** para ser executada;
2. O agente "Gerente interação aplicação móvel" gera a TM e solicita a sua execução ao agente "Gerente transação eq móvel";
3. O agente "Gerente transação eq móvel" registra seus controles sobre a TM;
4. O agente "Gerente transação eq móvel" verifica a existência de banco de dados local no equipamento móvel (BD Móvel);
5. O agente "Gerente transação eq móvel" solicita a execução da transação ao agente "Executor transação local";
6. O agente "Executor transação local" executa a TM no BD local e devolve o seu resultado para o agente "Gerente transação eq móvel";
7. Caso a execução da TM tenha sido cancelada (*abort*), o agente "Gerente de transação eq móvel" cancela a TM e avisa ao agente "Gerente interação aplicação móvel" para que a aplicação seja comunicada do fato. Nesse caso, todo o processamento é encerrado;
8. Caso a execução da TM tenha sido confirmada (*commit*), o agente "Gerente transação eq móvel" solicita ao "Gerente mobilidade transação", a migração da TM para o ambiente rede fixa para sua execução;
9. O agente "Gerente mobilidade transação" migra para o ambiente rede fixa para execução da TM;
10. Uma vez na ESM, o agente "Gerente mobilidade transação" solicita ao agente "Gerente transação eq rede fixa" a execução da transação;
11. O agente "Gerente transação eq rede fixa" faz seus controles e solicita a execução da transação ao agente "Executor transação local";

12. O agente "Executor transação local" executa a TM, no SGBD da rede fixa, e devolve o seu resultado para o agente "Gerente transação eq rede fixa";
13. O agente "Gerente transação eq rede fixa" registra seus controles e envia o resultado do processamento da TM ao agente "Gerente mobilidade transação";
14. O agente "Gerente mobilidade transação" solicita ao agente "Interação sistema localização" a localização atual do cliente móvel que gerou a TM;
15. O agente "Interação sistema localização" informa a localização atual do cliente móvel gerador da TM;
16. O agente "Gerente mobilidade transação", de posse da localização do cliente móvel, retorna ao seu local de origem e devolve o resultado do processamento da TM ao agente "Gerente transação eq móvel";
17. O agente "Gerente transação eq móvel" registra seus controles e envia os resultados do processamento da TM ao agente "Gerente interação aplicação móvel";
18. O agente "Gerente interação aplicação móvel" entrega os resultados da transação móvel solicitada para execução à "Aplicação Móvel".

Caso não exista um banco de dados local no equipamento móvel, os itens 5, 6 e 7 não serão executados. Diversas situações de exceção podem ocorrer na execução de uma Transação Móvel (TM) e deverão ser tratadas na construção do software. Entretanto, este item objetiva apenas demonstrar o funcionamento da instância do framework e a interação entre seus agentes.

5.5 Resumo do Capítulo

Este capítulo apresentou uma arquitetura para operacionalização do modelo de transações proposto no capítulo anterior, utilizando as tecnologias de framework e agentes de software. O framework teve seus aspectos estáticos e dinâmicos representados com a linguagem de modelagem de sistemas de agentes MAS-ML.

O próximo capítulo apresenta uma comparação entre os modelos de execução de transações, avaliados no capítulo 3, e os modelos descritos nos capítulos 4 e 5.