



**Universidade Estadual de Campinas**

Faculdade de Engenharia Elétrica e de Computação

DEPARTAMENTO DE TELEMÁTICA

---

**Propostas de mobilidade para os protocolos  
H.323 e SIP em Telefonia IP, utilizando  
linguagem de especificação formal SDL  
orientada à objetos**

**Autora: Stella Wing Kwan Chung**

Bacharel em Ciências da Computação - Universidade Católica de Goiás

**Orientador: Prof. Dr. Walter da Cunha Borelli**

**Banca Examinadora:**

Prof. Dr. Leonardo Guerra de Rezende Guedes      Universidade Federal de Goiás

Prof. Dr. Paulo Cardieri      FEEC/UNICAMP

Prof. Dr. Reginaldo Palazzo Junior      FEEC/UNICAMP

Tese apresentada na Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas, como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Elétrica.

Campinas - SP

Julho 2004

FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

C472p Chung, Stella Wing Kwan  
Propostas de mobilidade para os protocolos H.323 e SIP em telefonia IP, utilizando linguagem de especificação formal SDL orientada à objetos / Stella Wing Kwan Chung.-Campinas, SP: [s.n.], 2004.

Orientador: Walter da Cunha Borelli  
Dissertação (mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. SDL (Linguagem de programação de computador). 2. Simulação (Computadores digitais). 4. Redes de computação-Protocolos. I. Borelli, Walter da Cunha. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.



FEEC

## COMISSÃO JULGADORA - TESE DE MESTRADO

**Nome do Candidato** : Stella Wing Kwan Chung

**Data da Defesa** : 28.07.2004

**Título da Tese** : “Propostas de Mobilidade para os Protocolos H.323 e SIP em  
Telefonia IP, Utilizando Linguagem de Especificação Formal SDL  
Orientada a Objetos”

### AVALIAÇÃO FINAL

Será considerada aprovada a tese que obtiver aprovação da maioria dos membros da Comissão Julgadora.

1. (Aprovada ou Reprovada)

Aprovada

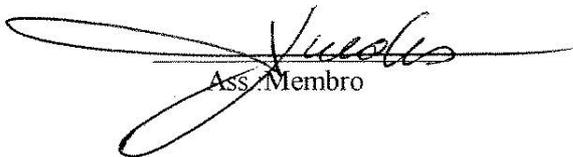


2. Modificações sugeridas pela Comissão Julgadora que deverão ser introduzidas na versão final da tese, especificar o prazo máximo para entrega da versão final da tese na CPG:

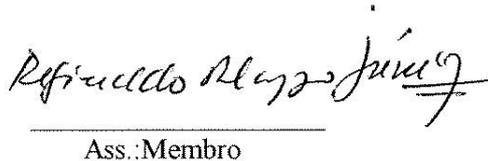
As sugestões feitas pelos membros da banca examinadora serão incorporadas no texto final sob o acompanhamento do orientador. Fica estipulado o prazo de 2 meses para entrega da versão final da tese na CPG/FECC.

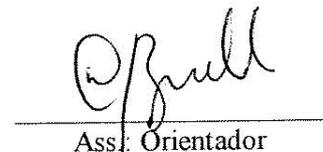
3. Comentários Finais:

A candidata fez uma apresentação segura e demonstrou maturidade no tema de pesquisa. O trabalho desenvolvido contempla contribuições à temática em consideração, tendo gerado uma publicação em congresso internacional altamente relevante.

  
Ass. Membro

  
Ass. Membro

  
Ass. Membro

  
Ass. Orientador



# Agradecimentos <sup>1</sup>

A Deus pelas ricas bençãos.

Em especial aos meus pais Chung e Dália e ao meu irmão Daniel pela força, carinho e amor e por fazerem parte da minha vida.

Ao meu orientador, Prof. Dr. Walter da Cunha Borelli, pela oportunidade do mestrado e dedicação.

Aos grandes amigos que conheci durante estes anos.

---

<sup>1</sup>Este trabalho foi financiado pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - CAPES

# Resumo

Este trabalho apresenta propostas de mobilidade para os protocolos H.323 e SIP. Para o protocolo H.323, são desenvolvidos sistemas com mobilidade intra-domínio, inter-domínios e inter-domínios em conjunto com os novos conceitos do protocolo H.510. Estas propostas se basearam na solução de mobilidade desenvolvida por Liao utilizando a expansão da conferência *ad hoc* multiponto para realizar *roaming* e *handoff* em um estabelecimento de chamada entre terminais móveis H.323. Para o SIP, é desenvolvido o sistema com mobilidade inter-domínios para o estabelecimento de sessão entre terminais móveis SIP. Esta proposta se baseia na solução de mobilidade desenvolvida por Chen, Moh e Berquin utilizando a conferência multiponto com a presença da MCU para realizar *roaming* e *handoff*. Todos estes sistemas foram especificados utilizando a linguagem de especificação formal SDL orientada a objeto a partir da ferramenta SDL TAU Suite. Esta ferramenta permite simular os sistemas a partir dos diagramas em MSC e validar os sistemas para detecção e correção de eventuais erros de lógica e de especificação.

**Palavras-chave:** Telefonia IP, H.323, H.510, SIP, Mobilidade, Roaming, Handoff, Conferência Ad Hoc Multiponto, Conferência Multiponto, SDL - Specification and Description Language, SDL TAU Suite, MSC - Message Sequence Chart, Simulação, Validação.

# Abstract

This work describes some proposals for mobility extension to H.323 and SIP protocols. The new proposals for H.323 involve intrazone and interzone mobility, and interzone mobility using resources with new entities, signalling and functions from H.510. Based on the mobility solution developed by Liao, these new systems have demonstrated how H.323 could be extended in order to offer mobility using ad hoc multipoint conference for roaming and handoff during a call establishment between H.323 mobile terminals. The other proposal is a system with interzone mobility using multipoint conference with MCU for roaming and handoff during a call establishment between SIP mobile terminals. This system is based on Chen, Moh and Berquin mobility solution. All these systems were specified using the SDL object-oriented formal language with SDL TAU Suite tool. This tool allows the simulation of the systems with MSC diagrams and the validation of the systems by detecting and correcting logical and specification errors.

**Keywords:** IP Telephony, H.323, H.510, SIP, Mobility, Roaming, Handoff, Ad Hoc Multipoint Conference, Multipoint Conference, SDL - Specification and Description Language, SDL TAU Suite, MSC - Message Sequence Chart, Simulation, Validation.

# Publicações

S.W.K Chung, W.C. Borelli. A New Proposal for Mobility Extension to H.323 using SDL. In: *11th International Conference on Telecommunications (ICT'2004)*, Fortaleza - CE, Agosto 2004.

# Sumário

|   |            |
|---|------------|
| <b>Agradecimentos</b>   | <b>i</b>   |
| <b>Resumo</b>   | <b>ii</b>  |
| <b>Abstract</b>   | <b>iii</b> |
| <b>Publicações</b>  | <b>iv</b>  |
| <b>Lista de Figuras</b>   | <b>vii</b> |
| <b>1 Introdução</b>   | <b>1</b>   |
| 1.1 Motivação e Objetivos . . . . .   | 1          |
| 1.2 Organização do Trabalho . . . . .   | 3          |
| <b>2 Arquiteturas e as Propostas de Mobilidade para os Protocolos H.323 e SIP</b> | <b>5</b>   |
| 2.1 Arquitetura do Protocolo H.323 . . . . .                                      | 6          |
| 2.1.1 <i>Terminal</i> H.323 - TL . . . . .  | 6          |
| 2.1.2 <i>Gateway</i> H.323 - GW . . . . .   | 6          |
| 2.1.3 <i>Multipoint Control Unit</i> - MCU . . . . .                              | 6          |
| 2.1.4 <i>Multipoint Controller</i> - MC . . . . .                                 | 7          |
| 2.1.5 <i>Gatekeeper</i> - GK . . . . .  | 7          |
| 2.1.6 <i>Zona</i> . . . . .   | 7          |
| 2.1.7 Procedimentos para comunicação entre dois terminais . . . . .               | 7          |
| 2.1.8 Conjunto de protocolos do H.323 . . . . .                                   | 8          |
| 2.1.9 Expansão da conferência <i>ad hoc</i> multiponto . . . . .                  | 10         |
| 2.1.10 Gerenciamento de mobilidade . . . . .                                      | 10         |
| 2.1.11 Protocolo H.510 . . . . .  | 11         |
| 2.2 Arquitetura do Protocolo SIP . . . . .  | 12         |
| 2.3 A linguagem SDL e a ferramenta TAU SDL Suite (SDT) . . . . .                  | 15         |
| 2.4 Propostas de mobilidade para H.323 e SIP em SDL . . . . .                     | 16         |

---

|          |   |           |
|----------|---|-----------|
| <b>3</b> | <b>Especificação formal em SDL dos sistemas H.323 e SIP com mobilidade</b>                                    | <b>18</b> |
| 3.1      | H.323 . . . . .   | 19        |
| 3.1.1    | Estabelecimento de chamada entre terminais fixos . . . . .  | 19        |
| 3.1.2    | Estabelecimento de chamada com mobilidade em um mesmo domínio<br>(intra-domínio) . . . . .                    | 24        |
| 3.1.3    | Estabelecimento de chamada com mobilidade em domínios diferentes<br>(inter-domínios) . . . . .                | 27        |
| 3.1.4    | Estabelecimento de chamada com mobilidade inter-domínios e os<br>novos conceitos do protocolo H.510 . . . . . | 33        |
| 3.2      | SIP . . . . .   | 37        |
| <b>4</b> | <b>Resultados da Validação e das Simulações</b>   | <b>40</b> |
| 4.1      | Validação . . . . .   | 40        |
| 4.1.1    | Detecção de erros de especificação . . . . .  | 44        |
| 4.2      | Simulações . . . . .  | 45        |
| <b>5</b> | <b>Conclusões</b>   | <b>55</b> |
|          | <b>Referências Bibliográficas</b>   | <b>58</b> |
| <b>A</b> | <b>A linguagem SDL e a ferramenta TAU SDL Suite</b>   | <b>62</b> |
| A.1      | Estrutura do sistema . . . . .  | 62        |
| A.2      | Troca de sinais . . . . .   | 63        |
| A.3      | Principais elementos para a descrição de processos . . . . .  | 63        |
| A.4      | Herança e Especialização de tipos . . . . .   | 63        |
| A.5      | <i>Package</i> . . . . .  | 65        |
| A.6      | <i>SDL TAU Suite</i> . . . . .  | 66        |

# Lista de Figuras

|      |   |    |
|------|---|----|
| 2.1  | Pilha de protocolos para o H.323[1] . . . . .   | 8  |
| 2.2  | <i>Roaming</i> em um mesmo domínio ( <i>intrazone</i> ) . . . . .                                     | 11 |
| 2.3  | <i>Roaming</i> entre domínios diferentes ( <i>interzone</i> ) . . . . .                               | 11 |
| 2.4  | Gerenciamento de mobilidade para o protocolo SIP com MCU . . . . .                                    | 14 |
| 3.1  | <i>Organizer</i> do estabelecimento de chamada entre dois terminais fixos H.323                       | 20 |
| 3.2  | Pacote <i>PackBasicCall</i> do sistema entre terminais fixos . . . . .                                | 20 |
| 3.3  | Definição do sistema entre terminais fixos <i>BasicCall</i> . . . . .                                 | 21 |
| 3.4  | Definição do sistema entre terminais fixos <i>BasicCall</i> . . . . .                                 | 22 |
| 3.5  | Bloco <i>BasicCall</i> do sistema entre terminais fixos . . . . .                                     | 22 |
| 3.6  | Comportamento funcional do <i>Terminal1</i> no sistema entre terminais fixos .                        | 23 |
| 3.7  | Organizer do sistema sem e com mobilidade intra-domínio . . . . .                                     | 25 |
| 3.8  | Package <i>PackIntraZoneMob</i> contendo o sistema com mobilidade intra-domínio                       | 26 |
| 3.9  | Sistema <i>IntraZoneMob</i> com mobilidade intra-domínio . . . . .                                    | 26 |
| 3.10 | Bloco <i>CallingTerminal</i> redefinido para sistema com mobilidade intra-domínio                     | 27 |
| 3.11 | Novos sinais do processo <i>TermMob</i> no sistema intra-domínio . . . . .                            | 27 |
| 3.12 | Comportamento funcional do processo <i>Terminal2</i> no sistema intra-domínio                         | 28 |
| 3.13 | Comportamento funcional redefinido do processo <i>Terminal2</i> no sistema<br>intra-domínio . . . . . | 29 |
| 3.14 | Organizer do sistema sem e com mobilidade inter-domínios . . . . .                                    | 30 |
| 3.15 | Pacote <i>PackHandInterZone</i> para sistema com mobilidade inter-domínios .                          | 31 |
| 3.16 | Bloco <i>virtual CallMobile</i> para sistema sem mobilidade inter-domínios . . .                      | 31 |
| 3.17 | Sistema <i>HandInterZone</i> com mobilidade inter-domínios . . . . .                                  | 32 |
| 3.18 | Bloco <i>redefined CallMobile</i> para sistema com mobilidade inter-domínios . .                      | 33 |
| 3.19 | Organizer do sistema sem e com mobilidade inter-domínios e H.510 . . . .                              | 34 |
| 3.20 | Pacote <i>PackInterZoneH.510M</i> do sistema com mobilidade inter-domínios<br>e H.510 . . . . .       | 35 |
| 3.21 | Sistema <i>InterZoneH.510M</i> com mobilidade inter-domínios e H.510 . . . . .                        | 35 |
| 3.22 | Bloco <i>redefined CallUserMob</i> no sistema com mobilidade inter-domínios e<br>H.510 . . . . .      | 36 |
| 3.23 | Organizer do sistema com mobilidade para SIP . . . . .  | 37 |

|   |    |
|---|----|
| 3.24 Sistema SIP com mobilidade . . . . .   | 38 |
| 3.25 Pacote <i>PackSipConf</i> SIP com mobilidade . . . . .   | 39 |
| 4.1 Resultados da validação do sistema com mobilidade entre domínios diferentes e H.510 (ver figura 3.19) . . . . .             | 41 |
| 4.2 Cobertura do sistema inter-domínios com mobilidade e H.510 entre dois terminais H.323 (ver figura 3.19) . . . . .           | 42 |
| 4.3 Resultados da validação do sistema SIP (ver figura 3.23) . . . . .  | 43 |
| 4.4 Erro de "loop" detectado na validação do sistema com mobilidade inter-domínios (ver figura 3.14) . . . . .                  | 44 |
| 4.5 Erro de "loop" corrigido no sistema com mobilidade inter-domínios . . . . .   | 44 |
| 4.6 MSC do estabelecimento de chamada com mobilidade inter-domínios com erro de RRJ (ver figura 3.14) . . . . .                 | 46 |
| 4.7 MSC do estabelecimento de chamada com mobilidade inter-domínios e conceitos de H.510 (ver figura 3.19) . . . . .            | 47 |
| 4.8 MSC do estabelecimento de chamada com mobilidade inter-domínios e H.510 com erro de atualização (ver figura 3.19) . . . . . | 48 |
| 4.9 MSC do estabelecimento de sessão com mobilidade inter-domínios entre terminais SIP (ver figura 3.23) . . . . .              | 49 |
| 4.10 MSC com o erro gerado na simulação do estabelecimento de sessão entre terminais SIP (ver figura 3.23) . . . . .            | 50 |
| 4.11 Processo MCUC com erro de especificação para o sistema SIP (ver figura 3.23) . . . . .                                     | 51 |
| 4.12 Especificação do processo MCUC corrigido para o sistema SIP (ver figura 4.11) . . . . .                                    | 52 |
| A.1 Descrição e representação de algumas estruturas SDL . . . . .   | 64 |
| A.2 Descrição e representação de algumas estruturas SDL . . . . .   | 65 |
| A.3 Relação entre linguagens que fazem interface com o SDL TAU Suite . . . . .  | 67 |
| A.4 Visão geral dos módulos que compõem o SDL TAU Suite . . . . .   | 67 |

# Capítulo 1

## Introdução

### 1.1 Motivação e Objetivos

Com o crescimento exponencial da Internet, a comunicação sobre IP tende a ser predominante para a transmissão de dados, voz e vídeo. Espera-se que a rede de telefonia convencional baseada em circuitos, PSTN (*Public Switched Telephony Network*) realize uma completa convergência para a rede baseada em pacotes, a **Telefonia IP**, também conhecida como **VoIP** (Voice over IP) em um futuro próximo. Esta tendência se deve a alguns fatores importantes que incluem a diminuição do custo como chamadas telefônicas convencionais, adição de novos serviços, unificação das infra-estruturas para dados, voz e mensagens.

Os dois principais protocolos que competem entre si na Telefonia IP são o SIP (*Session Initiation Protocol*), desenvolvido pelo IETF (*Internet Engineering Task Force*), e o H.323, desenvolvido pelo ITU-T (antigo CCITT) (*International Telecommunication Union - Telecommunication Standardization Sector*).

O H.323 [13] é uma especificação guarda-chuva que descreve componentes, protocolos e procedimentos para sistemas multimídia de tempo real, como áudio, vídeo e dados sobre uma rede de pacotes sem qualidade de serviço (QoS), incluindo redes baseadas em protocolo IP.

O SIP [24] é um protocolo textual baseado no modelo cliente-servidor que permite o estabelecimento de sessões fim-a-fim entre duas ou mais entidades padronizado pelo IETF no RFC 3261. O protocolo SIP reutiliza algumas características, como endereços, cabeçalhos, de outros protocolos utilizados na Internet, como o HTTP e o SMTP. Não é um protocolo de transporte e também não oferece QoS. Portanto, o protocolo necessita do RTP (*Real Time Transport Protocol*) e do SDP (*Session Description Protocol*) para negociação e identificação. O SIP não é específico apenas para a telefonia IP, mas também estabelece recomendações para serviços adicionais como mensagens instantâneas (*instant messaging*), conferência, transferência e redirecionamento de chamadas, autenticação e

identificação de chamadas, entre outros.

A Telefonia IP foi desenvolvida para permitir o serviço de voz em tempo real entre um terminal IP fixo e outro terminal IP baseado na rede IP ou telefone baseado na rede de circuitos através de um *gateway*. Com o crescimento da popularidade de PDA's, laptops, wireless LAN's e o desejo da conectividade global, a mobilidade na Internet tem se tornado uma característica relevante. É desejável que se integre as características de mobilidade na telefonia IP. Por exemplo, quando um usuário origina uma chamada PSTN para um telefone IP na Internet, o terminal IP pode ser fixo ou móvel. O terminal pode se mover entre sub-redes e mudar o endereço IP. Assim como na telefonia celular, a mobilidade na telefonia IP requer *handoff* durante uma chamada em andamento e *roaming* toda vez que estiver mudando de sub-redes. O processo de *handoff* permite que uma chamada ativa seja mantida durante as mudanças dos terminais móveis entre sub-redes e de modo transparente para o terminal(usuário).

A versão atual do protocolo SIP contempla a mobilidade para usuários através dos servidores *proxy* e de redirecionamento, que encaminham as requisições para a localidade atual do usuário. No entanto, o protocolo H.323 não oferece nenhuma solução de mobilidade na recomendação atual. Apesar da existência de mobilidade a nível de usuários no protocolo SIP, ambos os protocolos não contemplam os procedimentos de *roaming* e *handoff* entre terminais móveis durante o estabelecimento de chamada.

Alguns trabalhos já foram desenvolvidos para prover mobilidade para os protocolos H.323 e SIP. Liao [1] propôs a expansão da conferência *ad hoc* multiponto para realizar o processo de *roaming* e *handoff* durante o estabelecimento de chamada entre terminais H.323 móveis. Esta solução permite que o terminal móvel se mova entre domínios iguais ou diferentes mantendo uma chamada em andamento ativa. A expansão da conferência *ad hoc* multiponto é um recurso existente no protocolo H.323, portanto a solução não introduz entidades adicionais e requer mínimas modificações no protocolo. Baseado nesta solução de Liao, Chen, Berquin e Moh [2] propuseram uma solução a partir de recursos existentes no protocolo SIP utilizando a conferência com presença de MCU (*Multipoint Control Unit*) para realizar o processo de *roaming* e *handoff* durante a sessão entre dois terminais SIP móveis.

Este trabalho desenvolve propostas de mobilidade em SDL para os protocolos H.323 e SIP. Para o H.323, as propostas dos sistemas com mobilidade tiveram como ponto de partida o estabelecimento de chamada entre terminais fixos. A partir do conceito de herança foram desenvolvidos os sistemas com mobilidade intra-domínio, inter-domínios e inter-domínios em conjunto com as novas entidades, sinalizações e funcionalidades do protocolo H.510 [20] para o estabelecimento de chamada entre terminais H.323 móveis. O processo de *roaming* e *handoff* são obtidos a partir da expansão da conferência *ad hoc* multiponto onde dois terminais H.323 móveis mantêm uma conexão ativa ao se moverem entre domínios diferentes, mudando consequentemente de endereço IP. O protocolo H.510

foi padronizado em 2002 pelo ITU-T e oferece mobilidade para sistemas H.323 a nível de usuário e terminal, mas apesar destas novas facilidades, a recomendação não oferece solução para *roaming* e *handoff*, mobilidade para manter a conexão ativa durante mudanças de sub-redes. Para o SIP, este trabalho propôs o sistema com mobilidade inter-domínios entre terminais móveis a partir da conferência com presença de MCU para realizar *roaming* e *handoff*.

As propostas são especificadas utilizando a linguagem orientada a objeto SDL (*Specification Description Language*) [22] através da ferramenta TAU SDL Suite <sup>1</sup>. Os modelos SDL propostos conferem uma formalização dos sistemas de acordo com os protocolos H.323, H.510 e SIP, permitindo a validação de suas principais funcionalidades, bem como a simulação de casos críticos envolvidos nos sistemas, auxiliando a identificação e correção de quaisquer erros existentes antes de uma eventual implementação.

## 1.2 Organização do Trabalho

Este trabalho está dividido em cinco capítulos e um apêndice.

O **Capítulo 2** descreve uma introdução às arquiteturas dos protocolos H.323, H.510 e SIP, considerando os pontos mais relevantes para o trabalho desenvolvido. Ao final do capítulo são apresentadas as propostas dos sistemas para os Capítulos 3 e 4.

O **Capítulo 3** apresenta as propostas de especificação formal orientada a objeto para os sistemas com mobilidade para os protocolos H.323, SIP, e H.323 em conjunto com o H.510. No capítulo são apresentados diagramas em SDL dos sistemas desenvolvidos a partir dos conceitos das arquiteturas apresentadas no Capítulo 2. É mostrada também a possibilidade de reuso das especificações através do conceito de herança. Desta forma, é possível simplificar a construção dos novos blocos e processos SDL a partir das especificações já existentes.

O **Capítulo 4** apresenta os resultados da validação dos sistemas desenvolvidos e exemplos de simulações dos comportamentos funcionais e casos críticos através de diagramas MSC (*Message Sequence Chart*). O capítulo apresenta também técnicas para identificar e corrigir erros existentes de lógica e especificação destes sistemas através da validação e simulação, garantindo a corretude para futura expansão e implementação.

No **Capítulo 5** são apresentadas as conclusões finais e sugestões para eventuais trabalhos futuros.

O **Apêndice A** apresenta uma introdução à linguagem SDL com as características principais e relevantes para o trabalho desenvolvido. O apêndice também apresenta a ferramenta SDL TAU Suite que possui os módulos *Simulator* e *Validator* utilizados para

---

<sup>1</sup>Pacote Telelogic Tau SDL Suite 4.2 : adquirido pelo DT/FEEC/UNICAMP através do Projeto Temático - FAPESP (Proc. 91/3660-0)

gerar os resultados das simulações dos casos críticos e validação dos sistemas respectivamente, para o desenvolvimento das especificações dos sistemas em SDL.

## Capítulo 2

# Arquiteturas e as Propostas de Mobilidade para os Protocolos H.323 e SIP

A comunicação de voz em redes IP, chamado de **VoIP**, consiste no uso da rede de dados que utiliza o conjunto de protocolos da rede IP (IP/TCP/UDP) para transmissão de sinais de voz em tempo real no formato de pacotes de dados. A **Telefonia IP** é uma evolução natural do conceito de VoIP que consiste no fornecimento de serviços de telefonia utilizando a rede IP para o estabelecimento de chamadas e comunicação de voz.

A telefonia IP requer sinalizações para que duas entidades possam se comunicar, para estabelecer e gerenciar as chamadas e trocar informações entre as entidades.

Atualmente existem dois protocolos padronizados que competem entre si para prover sinalização e controle para a telefonia IP, o **H.323** e o **SIP**. O protocolo H.323 [13], desenvolvido pelo ITU-T, define padrões para comunicação multimídia através de redes que não oferecem qualidade de serviço (QoS) garantida, por exemplo, redes do tipo LAN, IP e Internet. O conjunto de padrões utilizados no H.323 e suas aplicações para os sistemas de telefonia IP englobam os protocolos H.225 [14] para conexão, H.245 [17] para controle, H.332 [16] para conferências, H.235 [15] para segurança, H.246 [18] para interoperabilidade com RTPC [26] e a série H.450x [19] para serviços complementares. Todos estes padrões compõem a série H de recomendações. O SIP [24], desenvolvido pelo IETF, é um protocolo de geração mais recente. Como os outros protocolos da família Internet (HTTP [27], FTP [28], SMTP [29]), é baseado em código texto. O SIP é um protocolo de controle e sinalização para criar, modificar e terminar sessões entre um ou mais terminais. Os protocolos H.323 e SIP têm funções equivalentes, ricos em funcionalidades e serviços, mas nenhum padrão suporta mobilidade na telefonia IP.

A mobilidade é um assunto que tem se desenvolvido bastante nos últimos tempos. O uso difundido de equipamentos que provêm mobilidade aos usuários, por exemplo, *laptops*,

PDA's, celulares, faz com que pesquisas e desenvolvimento de soluções em mobilidade continue crescendo. A mobilidade é uma característica que está sendo incorporada em várias tecnologias. A tendência é que a rede de telefonia convencional (PSTN) realize a completa convergência para a telefonia IP e que esta tenha a mobilidade incorporada em sua tecnologia. A proposta deste trabalho é prover mobilidade para os protocolos H.323 e SIP onde os terminais móveis envolvidos durante o estabelecimento de chamada devem ser capazes de realizar o processo de *handoff*, manter a conexão ativa durante mudanças de sub-redes.

Para a compreensão de todos os sistemas propostos neste trabalho, este capítulo apresenta os principais conceitos e arquiteturas associadas aos protocolos H.323 e SIP, bem como uma introdução à linguagem SDL utilizada para especificar os sistemas.

A seção 2.1 descreve as arquiteturas dos protocolos H.323 com as respectivas entidades, sinalizações e funcionalidades. A seção 2.2 descreve os conceitos e a arquitetura SIP. A seção 2.3 descreve brevemente as características da linguagem SDL e por fim, a seção 2.4 relata a proposta de mobilidade IP para os protocolos H.323 e SIP.

## **2.1 Arquitetura do Protocolo H.323**

### **2.1.1 Terminal H.323 - TL**

O terminal H.323 é um dispositivo em que está implementado o serviço de telefonia IP, realizando as funções de terminal de voz, vídeo e dados ou multimídia. Os terminais possuem as funcionalidades e protocolos necessários instalados para suportar comunicação bidirecional de voz em tempo real e a sinalização das chamadas.

### **2.1.2 Gateway H.323 - GW**

Elemento situado entre uma rede IP e outra de padrão diferente, tais como: sistemas telefônicos convencionais (PSTN), rede integrada de serviços digitais (ISDN) de forma a permitir uma interoperabilidade e funções de sinalização entre as redes. O GC (*Gateway Controller*) é um equipamento responsável pelo controle das chamadas em andamento realizadas pelos GW. O GC utiliza e gera as informações de sinalização e comanda os GW para iniciar, acompanhar e terminar uma chamada entre dois terminais distintos.

### **2.1.3 Multipoint Control Unit - MCU**

Elemento responsável pelos serviços de conferência multiponto entre três ou mais terminais. É composto por um Controlador Multiponto (MC - *Multipoint Controller*), responsável pela sinalização das chamadas, e por um Processador Multiponto (MP - *Multipoint Processor*), responsável pelo processamento dos pacotes de dados dos sinais de

voz dos terminais envolvidos na conferência.

#### **2.1.4 *Multipoint Controller* - MC**

Elemento responsável pelo controle das funções para suporte a conferências entre três ou mais terminais em uma conferência multiponto. O MC possui as capacidades de cada terminal em uma conferência multiponto e envia o conjunto de informações para os terminais na conferência indicando o modo em que os terminais devem transmitir os seus dados. O terminal se conectará ao MC presente dentro do GK (*gatekeeper*) através de uma conexão implícita no canal de controle H.245 estabelecido inicialmente. O MC localizado no GK pode ser utilizado para suportar as conferências *ad hoc* multiponto quando o GK recebe o canal H.245 do terminal. Desta forma, o GK pode traçar a rota do canal de controle H.245 para o MC no início da chamada ou quando a conferência migra para multiponto.

#### **2.1.5 *Gatekeeper* - GK**

Elemento responsável pelo gerenciamento de um conjunto de equipamentos dedicados a telefonia IP, TL, GW e MCU. Suas principais funções são executar a tradução de endereçamento dos diversos equipamentos, controlar o acesso dos equipamentos à rede dentro de sua zona e gerenciar a banda utilizada.

Outras funcionalidades opcionais podem ser adicionadas, como autorização de chamadas, localização de GW e serviços de gerenciamento de chamadas.

#### **2.1.6 Zona**

Zona é um conjunto de terminais, GW's e MCU's, gerenciados por um único GK. Uma zona deve ter pelo menos um terminal e pode ou não conter GW's ou MCU's. Entretanto, uma zona tem apenas um GK gerenciando. Fisicamente, a zona pode ser composta por um ou mais segmentos de rede interligados através de roteadores ou outros equipamentos semelhantes.

#### **2.1.7 Procedimentos para comunicação entre dois terminais**

A comunicação entre dois terminais na telefonia IP ocorre através de dois processos simultâneos: a sinalização e controle de chamadas e o processamento de voz.

O primeiro processo (sinalização e controle de chamadas) é constituído por três fases: *call setup*, *call handling* e *call termination*. O estabelecimento de chamada (*call setup*) ocorre entre dois ou mais terminais e envolve um ou mais GK's para obtenção da informação dos terminais de uma mesma zona ou de zonas distintas. Pode envolver também GW's, caso incluam terminais de padrões diferentes, ou MCU's caso seja estabelecida

uma conferência multiponto. Estabelecida a chamada, são criados canais virtuais de controle entre todos os equipamentos envolvidos. O acompanhamento da chamada (*call handling*) é feito através dos canais de controle no decorrer da chamada para identificar perda de conexão e outros eventos relevantes e dependentes dos serviços adicionais permitidos pelos terminais, como atendimento simultâneo, chamada em espera etc. A finalização da chamada (*call termination*) libera os terminais e outros equipamentos envolvidos, libera os canais de controle e atualiza o status dos terminais junto aos equipamentos da rede.

O segundo processo (processamento de voz) é constituído por duas fases: *transport control* e *media stream transport*. No controle do transporte de voz (*transport control*), ao estabelecer a chamada, os terminais iniciam um processo de definição do mecanismo de transporte de voz onde é eleito um mestre, identifica-se o tipo de mídia a ser transportada (voz) e são criados os canais virtuais de controle e de mídia. O transporte de mídia (*media stream transport*) inicia o transporte bidirecional em tempo real de mídia (voz) entre os terminais envolvidos através dos canais virtuais criados na fase anterior.

### 2.1.8 Conjunto de protocolos do H.323

O H.323 trabalha em conjunto com os protocolos da série H para estabelecer a comunicação entre os terminais na telefonia IP. A seguir são descritos os principais protocolos.

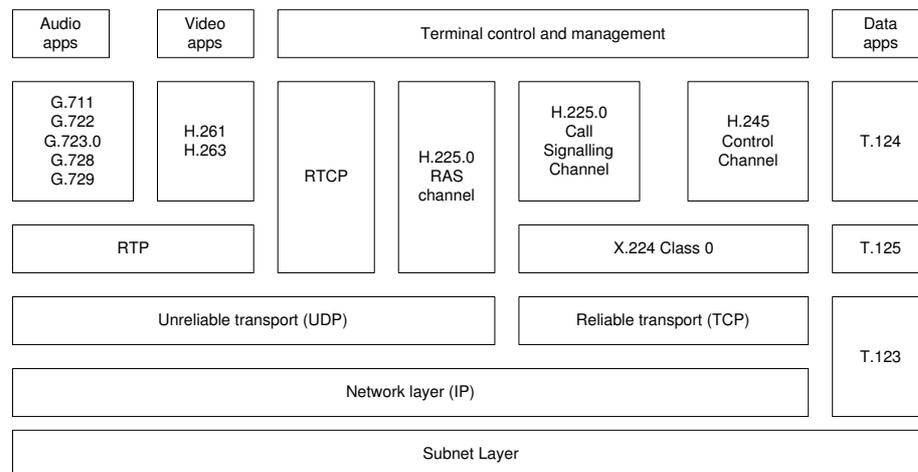


Figura 2.1: Pilha de protocolos para o H.323[1]

A figura 2.1 apresenta a pilha de protocolos relacionados ao H.323.

#### 1. *H.225 Call Signalling Protocols and Media Stream Packetization for Packet-based Multimedia Communication Systems*

Esta recomendação estabelece padrões para sinalização e empacotamento de mídia (voz) para chamadas em sistemas baseados em redes de pacotes. Suas principais aplicações são:

- Sinalização de chamadas: define um conjunto de mensagens que usa o formato da recomendação Q.931[21] sobre pacotes TCP da rede IP, com a finalidade de estabelecer e finalizar as chamadas. Estas mensagens são trocadas entre os equipamentos envolvidos na chamada, GC (*Gateway Controller*) e MCU's.
- Controle de equipamentos na rede (**zona**): define um conjunto de mensagens para a funcionalidade RAS (*Registration, Admission and Status*), responsável pelo registro, admissão e status dos equipamentos na rede. As mensagens são trocadas entre o GK e os terminais, GW, GC e MCU's para o controle de uma determinada zona. Estas mensagens usam como suporte os pacotes UDP da rede IP.
- Comunicação entre *Gatekeepers* (**GK**): define um conjunto de mensagens para a sinalização *gatekeeper-gatekeeper*, que estabelece o processo de sinalização e controle para chamadas entre zonas distintas.
- Transporte de mídia (voz): esta recomendação baseia-se no uso dos protocolos RTP[26] (*Real-Time Transport Protocol*) e RTCP (*Real-Time Transport Control Protocol*) como padrão para o transporte de mídia.

#### **2. H.245 Control Protocol for Multimedia Communication**

Esta recomendação estabelece padrões para a comunicação entre terminais para o processo de controle de transporte de voz (*transport control*). Estas mensagens usam como suporte os pacotes TCP da rede IP, e são trocadas entre os terminais, GW e MCU's envolvidos em chamadas do tipo ponto-a-ponto e ponto-multiponto.

#### **3. H.235 Security and Encryption for H-series (H.323 and other H.245-based) Multimedia Terminals**

Esta recomendação estabelece padrões adicionais de autenticação e segurança (criptografia) para terminais que usam o protocolo H.245[17] para comunicação ponto-a-ponto e multiponto.

#### **4. H.450.X Generic Functional Protocol for the Support of Supplementary Services**

Conjunto de recomendações que estabelece padrões de sinalização para serviços adicionais para terminais, tais como transferência e redirecionamento de chamadas, atendimento simultâneo, chamada em espera, identificação de chamadas, entre outros.

Estas mensagens usam como suporte os pacotes TCP da rede IP, e são trocadas entre os terminais, GW e MCU's envolvidos em chamadas do tipo ponto-a-ponto e ponto-multiponto que possuam as funcionalidades dos serviços adicionais.

### 2.1.9 Expansão da conferência *ad hoc* multiponto

Este recurso, presente na recomendação atual do H.323, é utilizado na solução de mobilidade proposta neste trabalho. A conferência *ad hoc* multiponto pode ser expandida a partir da conferência ponto-a-ponto com a presença do MC (*Multipoint Controller*). Inicialmente, a conferência ponto-a-ponto é criada entre dois terminais, por exemplo terminal 1 e terminal 2. Um terminal ou um GK, no mínimo, deve conter o MC. Uma vez que a conferência ponto-a-ponto é criada, a conferência pode ser expandida para uma conferência multiponto através de duas formas: o terminal na conferência convida (*invite*) outro terminal (terminal 3) para participar da conferência chamando este terminal 3 através do MC presente no mesmo ou o terminal 3 ingressa (*join*) na conferência existente chamando um terminal (terminal 1 ou terminal 2) participante da conferência.

### 2.1.10 Gerenciamento de mobilidade

O gerenciamento de mobilidade (*mobility management*) é essencial para prover mobilidade para a telefonia IP. As operações básicas devem incluir o registro, o estabelecimento de chamada, *roaming* e *handoff* entre os terminais. O *roaming* é o processo que garante a conectividade global para o usuário móvel e acontece toda vez que é necessário um novo registro na rede, por exemplo mudanças de endereços IP. Quando este usuário móvel muda de sub-rede, é necessário o processo de *handoff* para manter a conexão atual ativa. O usuário móvel se associa a um novo canal na nova sub-rede, de modo transparente para os usuários. Caso contrário, a conexão ou chamada é interrompida. Este processo de *handoff* ocorre durante mudanças de sub-redes sem novos registros e modificações de endereços IP.

No protocolo H.323, o terminal se registra no GK para admissão no sistema através de um canal RAS não confiável (UDP), estabelece e termina chamadas através de um canal confiável (TCP). Mas a versão atual do protocolo H.323 não possui mecanismos para prover *roaming* e *handoff*. Por exemplo, quando um terminal H.323 está em *handoff* entre sub-redes diferentes, mudando o endereço IP, é necessário que o mesmo terminal suporte *roaming* para que a conexão estabelecida não termine. O *handoff* deve acontecer de maneira transparente para os terminais envolvidos. O processo de *roaming* pode ser em um mesmo domínio (*intrazone roaming*) gerenciado por um mesmo GK ou entre domínios diferentes (*interzone roaming*) gerenciados por GK's diferentes.

As figuras 2.2 e 2.3 descrevem estes dois cenários. Assuma que o terminal H.323 TL1 esteja na sub-rede 1 e o terminal H.323 TL2 corresponde ao TL1 em *roaming* da sub-rede 1 para a sub-rede 2 durante uma chamada. Em ambos os cenários, é criado apenas uma conexão entre TL1 e TL onde se comunicam durante o processo de *roaming*. Quando TL1 se move da sub-rede 1 para a sub-rede 2, é realizado o *handoff* entre o TL e o terminal H.323 TL1 (TL1 se torna TL2 na sub-rede 2). Momentaneamente, irá coexistir duas

conexões ao mesmo tempo até que se finalize o processo de *handoff*. Uma vez finalizada, a conexão entre TL1 e TL é cancelada restando a conexão entre TL e TL2. Os recursos alocados na conexão antiga na sub-rede 1 são liberados e ficam disponíveis para novas requisições.

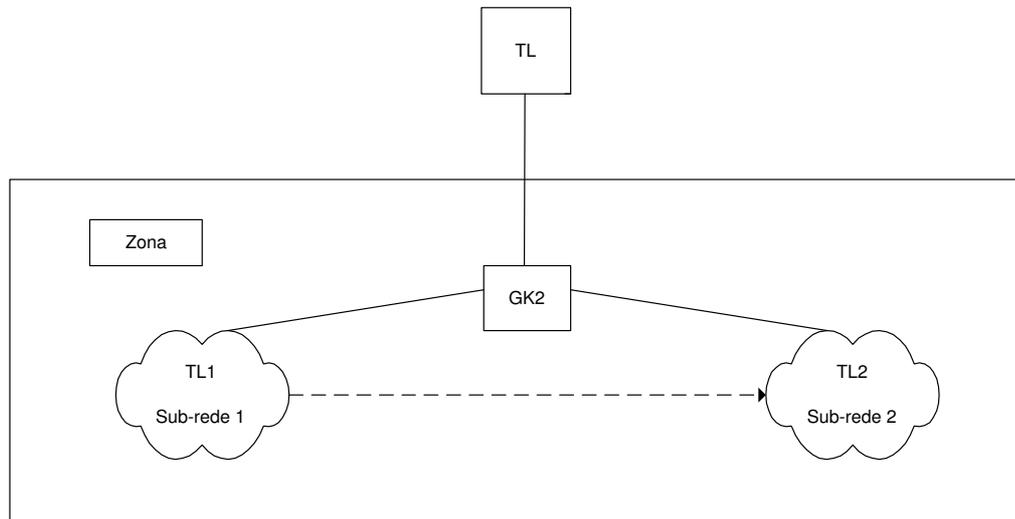


Figura 2.2: *Roaming* em um mesmo domínio (*intrazone*)

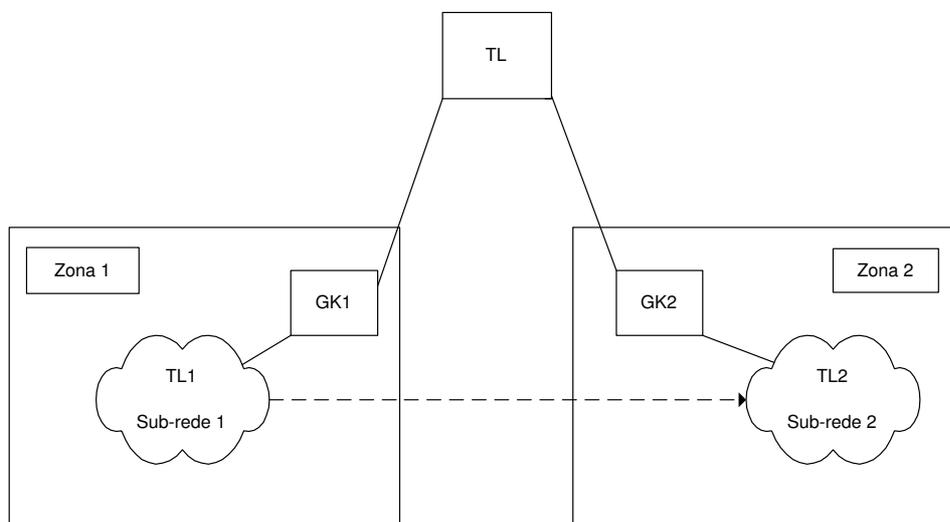


Figura 2.3: *Roaming* entre domínios diferentes (*interzone*)

### 2.1.11 Protocolo H.510

O protocolo H.510[20], desenvolvido pelo ITU-T em 2002, é uma recomendação que provê mobilidade através de novas funções para sistemas multimídia H.323. A recomendação atual do protocolo H.323 não suporta mobilidade e o H.510 foi desenvolvido recentemente para suprir as necessidades de mobilidade para terminais e usuários. Apesar de prover

mobilidade para os terminais e usuários, esta recomendação não contempla também o processo de *handoff*, ou seja, os terminais não mantêm as conexões ativas durante mudanças de sub-redes.

Novas entidades foram adicionadas ao protocolo H.510, VLF (*Visitor Location Function*), HLF (*Home Location Function*) e AuF (*Authentication Function*) que atuam em conjunto com as entidades do protocolo H.323, *gatekeepers*, *gateways* e terminais, além das novas sinalizações, *DescriptorUpdate* e *DescriptorUpdateAck*, que são utilizados nos novos procedimentos descritos a seguir.

Para prover mobilidade para os sistemas H.323 são necessários três procedimentos desenvolvidos na nova recomendação:

- Registro dos terminais no HLF: esse procedimento deve ser realizado antes que um usuário associado ao HLF possa ser contactado. Esse procedimento é realizado entre HLF's e GK's ou BE's (elemento de borda) para indentificação dos usuários durante a localização dos mesmos pelo HLF.
- Atualização dos endereços (*location update*): procedimentos necessários quando o terminal móvel muda de sub-rede, mudando a zona ou o endereço de rede. Esses procedimentos incluem o processo de descoberta de GK, registro e cancelamento dos registros dos terminais.
- Localização do terminal móvel (*callee location tracking*): troca de informações para localização dos terminais H.323 móveis.

## 2.2 Arquitetura do Protocolo SIP

O protocolo SIP, desenvolvido pelo IETF no RFC 3261, é um protocolo de controle e sinalização para criar, modificar e terminar sessões entre um ou mais participantes. É de documentação mais compacta, de fácil implementação e realiza algumas das características de outros protocolos, como por exemplo, os cabeçalhos, erros e regras de codificação do protocolo HTTP[27]. As mensagens SIP são codificadas usando a sintaxe de mensagem HTTP/1.1 e os endereços são URL's[30] (*Uniform Resource Locators*). Essas sessões podem incluir conferências de multimídia, chamadas de telefonia IP e distribuição de arquivos com multimídia.

O SIP convida usuários ou processos para participar da sessão. O protocolo suporta cinco categorias para estabelecimento e término de sessões: pela localização, pela característica e disponibilidade, início de sessões e modificação das características das sessões. Enfoca principalmente os aspectos de sinalização, criando, alterando e terminando as sessões. Os terminais são identificados pelo endereço SIP e não por endereços IP ou MAC. As mensagens utilizam os pacotes UDP da rede IP. A transação SIP consiste numa

solicitação SIP e uma resposta apropriada. Para tal finalidade, existem vários campos com valores idênticos na solicitação e na resposta, como o identificador de chamadas (*call ID*), o número da sequência do comando (*CSeq*), o campo (*to*), o campo (*from*), o campo (*via*), campo (*Content-Type*) e campo (*Content-length*).

Os terminais SIP podem usar sinalização UDP ou TCP, a sintaxe das mensagens é independente do protocolo de transporte usado. O protocolo SIP não é um protocolo de transporte, portanto é necessário o protocolo RTP (*Real Time Transport Protocol*) e depende do SDP [25] (*Session Description Protocol*) para cuidar da negociação e identificação nas sessões.

O protocolo SDP descreve sessões multimídia para telefonia e aplicações de difusão, como rádio ou TV via Internet, inclui informações sobre fluxo de mídia, endereços, portas, tipo de conteúdo, horário do início e término e origem da sessão. O SDP leva esta informação em formato textual, assim como o protocolo SIP. Quando uma chamada SIP é estabelecida, a mensagem *invite* contém informações SDP descrevendo os parâmetros aceitáveis para o equipamento chamador, incluindo uma descrição dos recursos e capacidades. A resposta ao chamado contém uma versão modificada dessa descrição, incorporando suas capacidades.

A arquitetura do SIP é composta por clientes e servidores. O cliente pode ser qualquer elemento de rede que envia requisições SIP e recebe respostas SIP. O cliente pode ou não interagir com o usuário humano. O cliente pode ser um *agente cliente* ou um *proxy*. O servidor é qualquer elemento de rede que recebe as requisições com o objetivo de atendê-los e envia as respostas referente às requisições ao originador da chamada. O servidor pode ser um *proxy*, um *agente servidor*, um *servidor de redirecionamento* ou um *registrador*.

O *agente cliente* é uma entidade que cria novas requisições. O *agente servidor* é uma entidade que gera uma resposta para as requisições SIP. Esta resposta aceita, rejeita ou redireciona a requisição.

O *servidor proxy* é uma entidade intermediária que atua como servidor e cliente com o propósito de fazer outras requisições para outros clientes. Inicialmente o *servidor proxy* exerce a função de roteamento, ou seja, certifica-se de que a requisição está sendo encaminhada para um destino próximo ao destino final. O *proxy* interpreta as requisições e se necessário, reescreve partes específicas da requisição antes de encaminhá-la.

O *servidor de redirecionamento* informa aos seus clientes o endereço do servidor requisitado e permite que o cliente entre em contato direto com este servidor.

O *registrador* é um servidor que registra e armazena os respectivos registros.

O SIP define seis tipos diferentes de métodos e trinta e sete cabeçalhos. A combinação destes métodos e cabeçalhos permite um controle completo de uma sessão multimídia. O método *Invite* é utilizado para convidar um usuário para participar de uma chamada (sessão) e estabelecer uma nova conexão. O método *ACK* é utilizado para facilitar trocas de mensagens entre usuários (convites) e confirmar que a entidade chamada recebeu uma

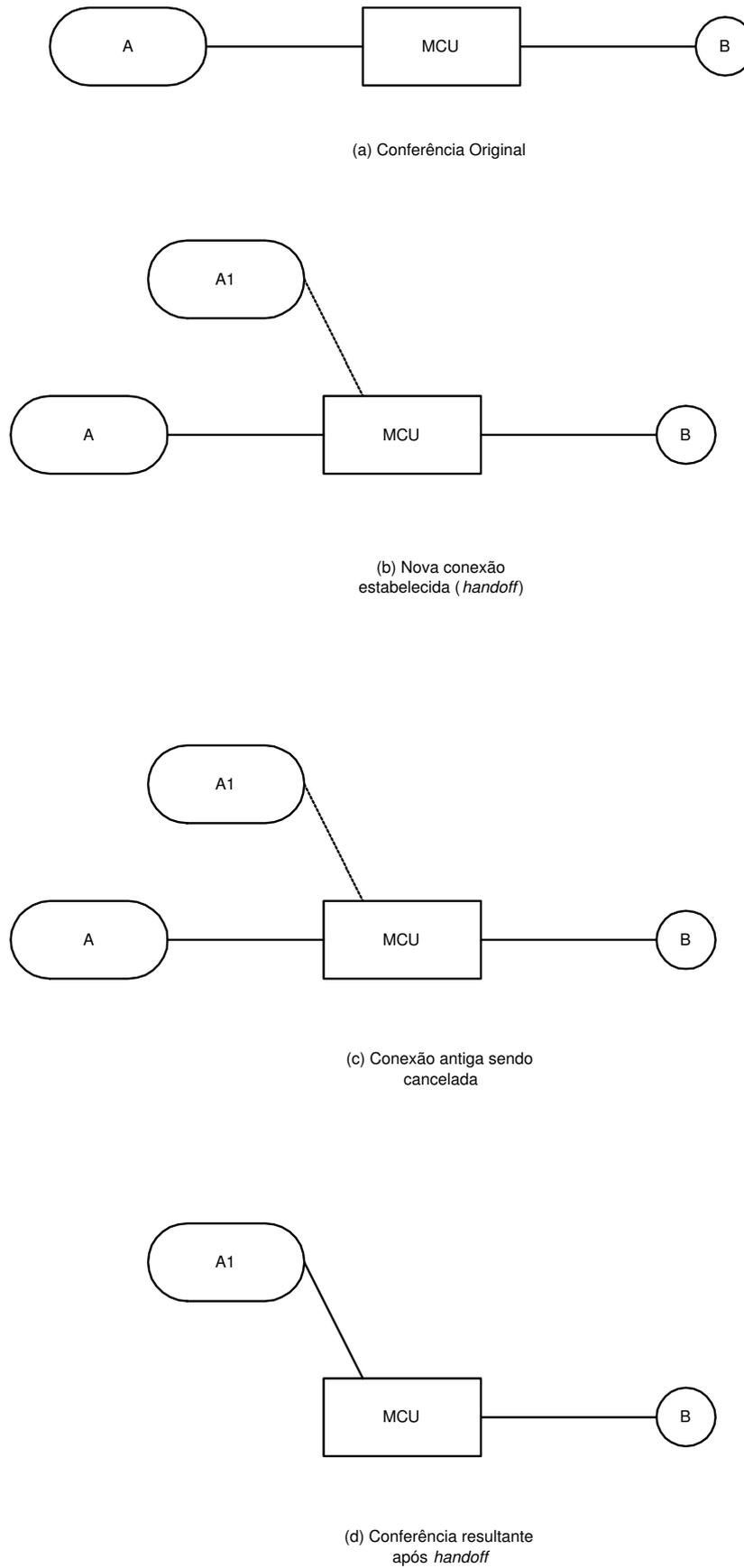


Figura 2.4: Gerenciamento de mobilidade para o protocolo SIP com MCU

resposta final do servidor. O método *Bye* termina conexões entre dois usuários numa chamada ou rejeita um convite. O método *Cancel* termina uma requisição que foi enviado anteriormente enquanto o servidor ainda não tiver enviado uma resposta final. O método *Register* transporta informações da localização atual do usuário para um servidor SIP. O método *Options* é utilizado para solicitar informações sobre as capacidades dos servidores SIP.

O SIP possui uma variedade de cenários para conferência multiponto. O SIP [24] possui definido na recomendação atual o MCU (*Multipoint Control Unit*) para oferecer conferência multiponto. O MCU pode ser expandido como solução para prover *handoff* e consequentemente mobilidade. Inicialmente, os usuários A e B estabelecem uma conexão ponto-a-ponto, A envia o endereço da MCU para B e este convida a MCU, formando uma conferência entre três participantes. Em algum momento, o usuário A se move para uma nova sub-rede e com um novo endereço se torna o usuário C (o elemento A com um novo endereço) para participar da conferência. O usuário C contacta a MCU indicada e requisita a sua entrada na conferência, mantendo o mesmo conteúdo do campo *Call-ID* iniciado na conexão entre A e B. A conexão entre A e C é estabelecida e a conexão entre A e B é encerrada, mantendo a conexão inicial ativa. A figura 2.4 descreve este gerenciamento da mobilidade a partir da conferência no protocolo SIP.

## **2.3 A linguagem SDL e a ferramenta TAU SDL Suite (SDT)**

A linguagem de especificação formal SDL [22] teve seu desenvolvimento iniciado pela CCITT *Comité Consultatif International Téléphonique*, hoje ITU-T, em 1972 visando a especificação e descrição de sistemas telefônicos, sendo também depois utilizada para a especificação de protocolos em redes de telecomunicações. Atualmente sua aplicação é muito abrangente, sendo utilizada na especificação de sistemas de informação, sistemas distribuídos, sistemas de tempo real, dentre outros. Trata-se de uma linguagem que incorpora conceitos de orientação a objetos, como tipos, reuso e herança.

A linguagem SDL é estruturada em diversos níveis de abstração, desde o nível mais alto até o nível mais detalhado do sistema especificado. Um sistema é dividido em blocos, que se comunicam uns com os outros e com o ambiente. Cada bloco é composto por um conjunto de processos que também se comunicam entre si e com outros blocos. O processo, por sua vez, trata-se do nível de abstração onde é especificado o comportamento do sistema através da definição de estados e transições. Sempre que necessário, podem ser especificados procedimentos (dentro dos processos) que realizem alguma tarefa específica.

Os processos se comunicam entre duas formas diferentes: através de troca de sinais ou através da chamada a procedimentos. Através destes sinais e dos procedimentos,

os processos podem trocar dados e trabalhar cooperativamente, dando a funcionalidade desejada ao sistema especificado.

Uma das principais características do SDL é que, diferentemente de grande parte das linguagens de especificação formal como Estelle e LOTOS, ele apresenta uma representação gráfica, chamada SDL-GR, que facilita bastante a compreensão do usuário e o desenvolvimento das especificações. Esta representação gráfica possui uma representação textual equivalente, chamada SDL-PR.

Neste trabalho, todo desenvolvimento em SDL foi realizado fazendo-se o uso da ferramenta TAU SDL Suite, que permite o desenvolvimento das especificações em SDL através de sua representação gráfica (SDL-GR), facilitando assim a compreensão do sistema como um todo. O TAU SDL Suite disponibiliza também módulos capazes de realizar a validação da especificação desenvolvida e simulações de suas funcionalidades e casos críticos. Estas facilidades foram utilizadas para a detecção e correção de erros, inclusive as eventuais falhas de especificação, o que garante a correção das especificações geradas.

O Apêndice A apresenta mais detalhes das principais construções SDL e da estrutura da linguagem. É também apresentada a ferramenta TAU SDL Suite e as principais facilidades na versão 4.2, utilizada neste trabalho.

## **2.4 Propostas de mobilidade para H.323 e SIP em SDL**

Este trabalho propõe sistemas com mobilidade especificados em SDL para os protocolos H.323 e SIP.

Ambos os protocolos não possuem mobilidade para manter uma conexão ativa durante mudanças de localidades por terminais móveis nas recomendações atuais. O estabelecimento de chamada ocorre entre terminais fixos. Para se ter a mobilidade durante o estabelecimento de chamada entre terminais móveis, é necessário que estes terminais realizem o processo de *roaming* e *handoff* para que a conexão inicialmente estabelecida não seja interrompida. Neste sentido, Liao [1] desenvolveu uma solução para prover mobilidade ao protocolo H.323 e Chen, Moh e Berquin [2], para o protocolo SIP, ambos utilizando recursos disponíveis na recomendação atual dos protocolos. Para o H.323, o processo de *handoff* é realizado através da expansão da conferência *ad hoc* multiponto com a entrada e saída dos terminais móveis nesta conferência. Já para o protocolo SIP, a MCU cria o ambiente de conferência e o *handoff* é realizado com a entrada do terminal móvel na conferência.

O capítulo 3 apresenta as propostas em SDL dos sistemas com mobilidade intra-domínio, inter-domínios e inter-domínios em conjunto com os novos conceitos de H.510 utilizando a expansão da conferência *ad hoc* multiponto para realizar *handoff* entre termi-

nais móveis H.323. Para o SIP, é apresentado o sistema com mobilidade inter-domínio a partir da conferência com MCU para o estabelecimento de sessão entre terminais móveis.

Os sistemas propostos apresentam as suas estruturas em SDL com orientação à objetos através da ferramenta SDL TAU Suite, simplificando as especificações, além de validar e simular alguns casos críticos a fim de garantir a corretude destes sistemas, possibilitando uma possível modificação, expansão ou implementação dos sistemas.

## Capítulo 3

# Especificação formal em SDL dos sistemas H.323 e SIP com mobilidade

As recomendações atuais dos protocolos H.323 [13] e SIP [24] não oferecem mobilidade durante o estabelecimento de chamada entre terminais. Quando um terminal móvel muda de sub-rede, a conexão não se mantém ativa por não haver mecanismos de suporte ao processo de *handoff*, portanto o estabelecimento de chamada ocorre entre dois terminais fixos. O objetivo deste trabalho é desenvolver sistemas com soluções de mobilidade para os protocolos H.323 e SIP propondo suas especificações em SDL (*Specification Description Language*).

Os sistemas H.323 com mobilidade propostos se basearam inicialmente no trabalho de Liao [1] que utilizou a expansão da conferência *ad hoc* multiponto para realizar *handoff*. O terminal móvel pode efetuar mudanças de sub-rede em um mesmo domínio (*intrazone roaming*) gerenciado pelo mesmo GK (*gatekeeper*) ou em domínios diferentes (*interzone roaming*) gerenciados por diferentes GK's em cada zona respectivamente.

Os sistemas H.323 com mobilidade se baseiam no sistema que realiza o estabelecimento de chamada entre dois terminais fixos. A partir destas entidades e sinalizações especificadas, os sistemas com estabelecimento de chamada com mobilidade intra-domínio e inter-domínios são gerados. No sistema intra-domínio, o terminal móvel se move entre sub-redes em um mesmo domínio ou zona gerenciado por um GK. No sistema inter-domínios, o terminal móvel se move entre sub-redes localizadas em diferentes domínios ou zonas, gerenciadas por diferentes GK's.

A partir do sistema H.323 com mobilidade inter-domínios especificado, é apresentado neste trabalho uma outra proposta de mobilidade para o protocolo H.323. Novos conceitos que incluem entidades, sinalizações e funcionalidades do H.510 [20] são adicionadas ao sistema que realiza o estabelecimento de chamada entre terminais H.323 móveis com

mobilidade inter-domínios.

Para o sistema SIP, a solução para mobilidade se baseou no trabalho de Chen, Moh e Berquin [24] que utilizaram a conferência com presença da MCU para realizar *handoff*. A partir do estabelecimento de sessão entre os dois terminais móveis, a MCU é convidada para a sessão, abrindo uma conferência entre os três participantes permitindo realizar o processo de *handoff*.

A linguagem de especificação formal SDL orientada a objetos permite que estes sistemas propostos com mobilidade utilizem a herança para fazer o reuso dos tipos especificados na super-classe, simplificando o desenvolvimento das especificações dos sistemas.

A seção 3.1.1 apresenta o sistema do estabelecimento de chamada entre dois terminais H.323 fixos em SDL. As entidades e as sinalizações são especificadas utilizando a estrutura hierárquica orientada à objetos da linguagem SDL. Nas seções 3.1.2 e 3.1.3 são descritos os sistemas com mobilidade intra-domínio e inter-domínios, respectivamente. Os sistemas herdam e redefinem as entidades e funcionalidades do sistema desenvolvido na seção 3.1.1 para oferecer mobilidade entre terminais móveis através do processo de *handoff* obtido a partir da expansão da conferência *ad hoc* multiponto.

A proposta de mobilidade para o protocolo H.323 apresentada na seção 3.1.4, herda e redefine o sistema com mobilidade inter-domínios entre dois terminais H.323 móveis, adicionando os novos conceitos do protocolo H.510.

A seção 3.2 apresenta a proposta do sistema em SDL do estabelecimento de sessão entre dois terminais SIP móveis com mobilidade através da conferência com presença de MCU para realizar *handoff*.

## 3.1 H.323

### 3.1.1 Estabelecimento de chamada entre terminais fixos

A proposta da especificação deste sistema é detalhar as entidades e sinalizações do protocolo H.323 necessárias para o estabelecimento de chamada entre dois terminais H.323 fixos. Este sistema é a base para o desenvolvimento das propostas de mobilidade para os sistemas com terminais H.323 móveis.

A figura 3.1 apresenta a organização hierárquica em SDL do sistema especificado. A hierarquia é composta de **Package**, **System**, **Block** e **Process**. O pacote **PackagePackBasicCall** é o diagrama raiz que contém o sistema **System TypeBasicCall**. Este sistema contém o bloco **virtual Block TypeBasicCall** que contém os processos **virtual Process Type Terminal1**, **Terminal2** e **GK1**.

O sistema foi especificado dentro de uma estrutura de **Package** (figura3.2) o qual permite que todo o sistema seja herdado por outra especificação. A declaração de um sistema, bloco ou processo como **Type**, **System Type**, **Block Type** e **Process Type**,



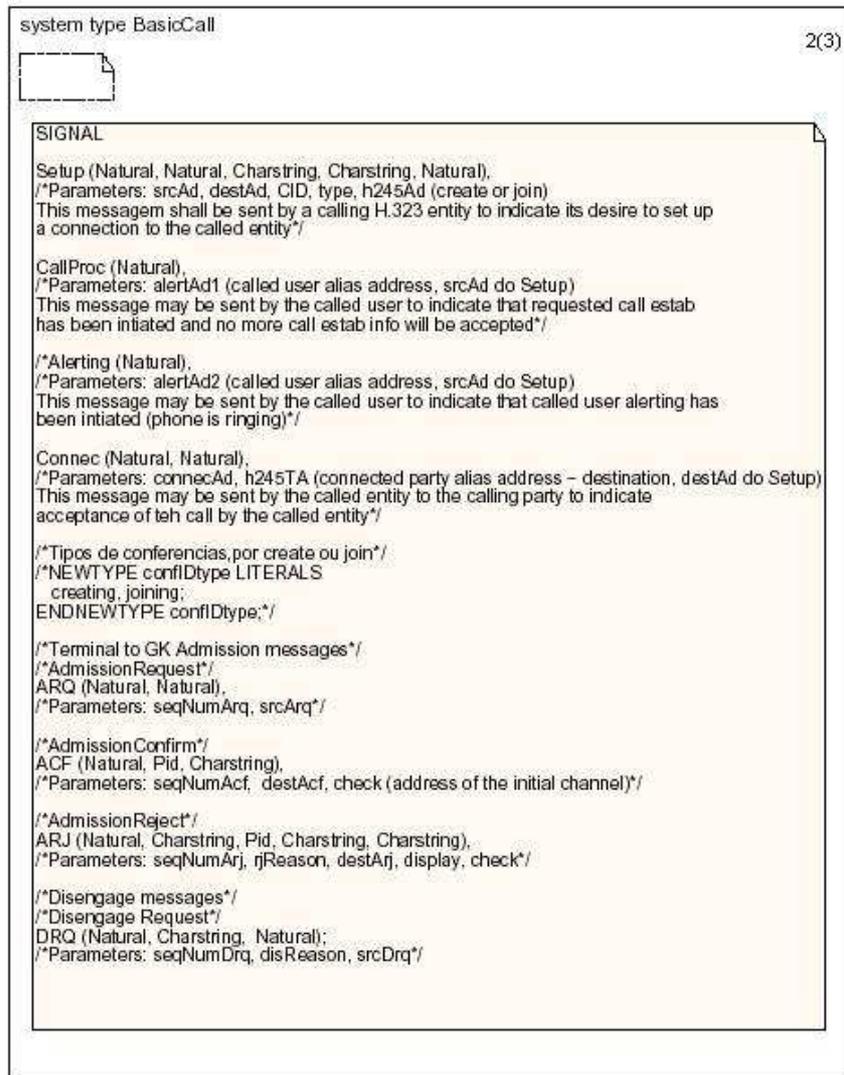


Figura 3.3: Definição do sistema entre terminais fixos *BasicCall*

As entidades, as funcionalidades e as sinalizações (figuras 3.3 e 3.4) para o estabelecimento de chamada estão especificadas no bloco *virtual block type BasicCall* (figura 3.5). Nas definições do sistema, cada sinal possui vários campos para armazenar mensagens. Os principais campos são os endereços de fonte e destino dos terminais, o campo CID que representa o identificador único de conferência. O objetivo da conferência é indicar se o sinal *Setup* deseja criar (*create*) uma conferência, convidar um terminal para uma conferência existente (*invite*) ou entrar em uma conferência existente (*join*). O campo com o endereço do canal de controle H.245 também é essencial para a negociação de capacidades entre os terminais durante a chamada em uso, na lista de definição estão representados pelos campos *h245TA* ou *h245Ad*.

Os processos (figura 3.5) *Terminal1* e *Terminal2* representam os terminais H.323 fixos e o processo *GK1* representa o *gatekeeper*. Os terminais estabelecem chamadas entre si para troca de informações e são dispositivos ou computadores em que estão implemen-

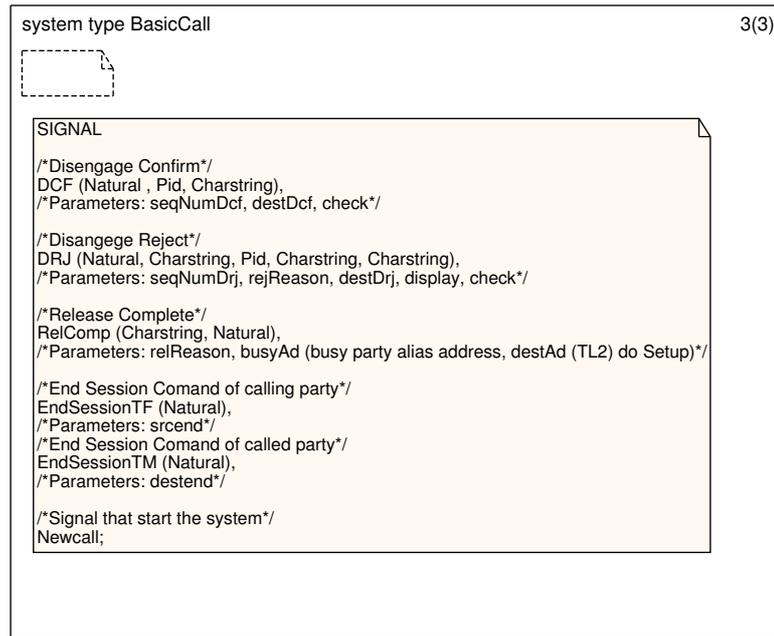


Figura 3.4: Definição do sistema entre terminais fixos *BasicCall*

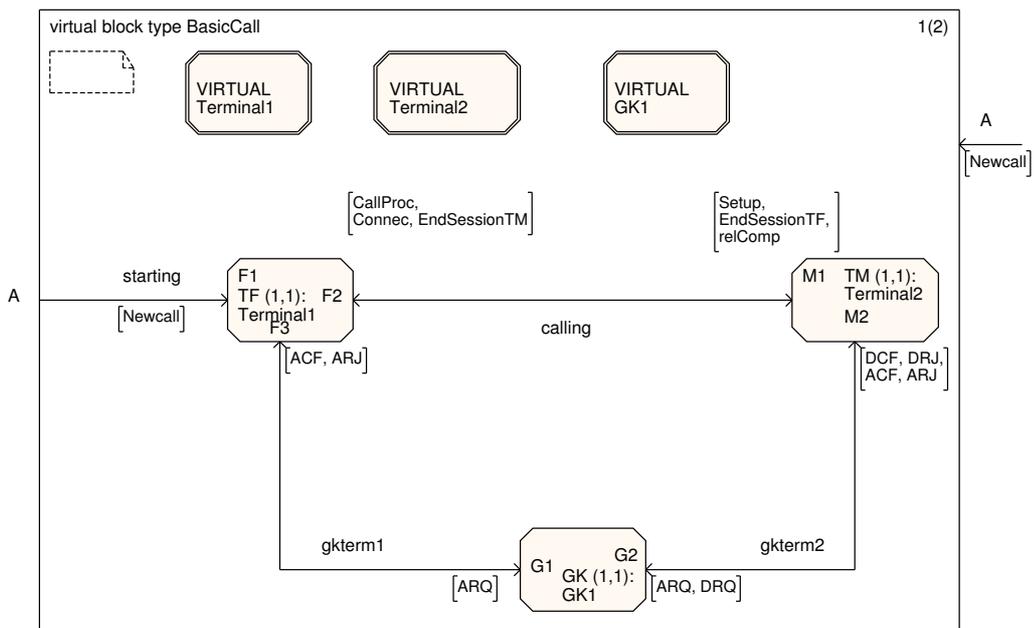


Figura 3.5: Bloco *BasicCall* do sistema entre terminais fixos

tados serviços de telefonia IP, realizam as funções de terminal de voz, vídeo e dados ou multimídia. O GK1 gerencia esta zona e é um servidor de tradução, localização, admissão, atualização e gerenciador de largura de banda. Cada processo possui uma instância (TF, TM e GK) representadas pela notação (1,1). O bloco *BasicCall* recebe o sinal *Newcall* através do gate A do ambiente externo e o *Terminal1* recebe este sinal através da rota *starting* para iniciar o estabelecimento de chamada.

Para o início do estabelecimento de chamada, o *Terminal1* requisita primeiramente a admissão para o *GK1* através dos comandos *ARQ/ACF* que contêm o endereço do

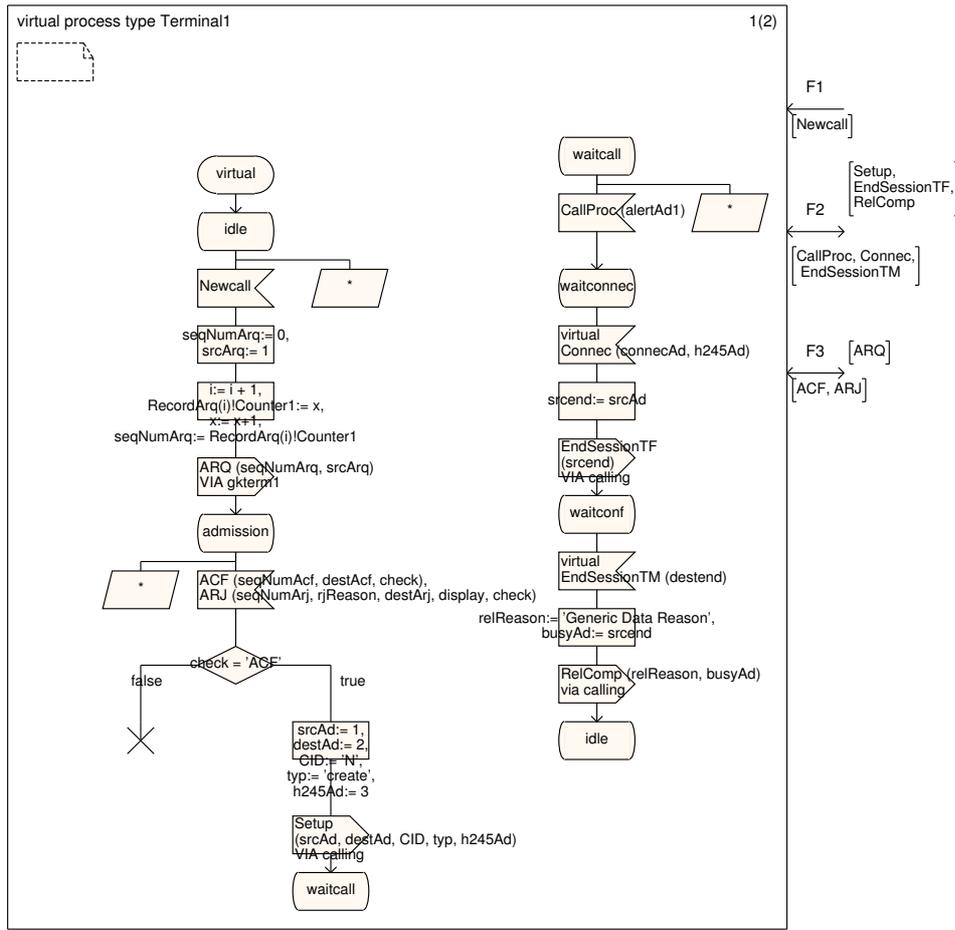


Figura 3.6: Comportamento funcional do *Terminal1* no sistema entre terminais fixos

destino, o identificador único da chamada (*CallID*) e a largura de banda para trafegar áudio e vídeo na conexão. O *Terminal1* envia a mensagem *Setup* para o *Terminal2* e este também requisita admissão ao *GK1* para prosseguir com a aceitação da conexão. A figura 3.6 apresenta o processo que descreve o comportamento funcional do *Terminal1*. A mensagem *Setup* possui como parâmetros os campos com o endereço do *Terminal1* (*srcAd*), o endereço do *Terminal2* (*destAd*), o *CID* indicando a criação de uma conferência através do comando *create* e o endereço do canal lógico H.245 (*h245Ad*) para troca de capacidades entre os terminais. O *Terminal2* aceita a conexão através da mensagem *Connect* em resposta a mensagem *Setup*. Através do endereço H.245, é aberto o canal lógico para tráfego de voz, dados ou vídeo e se mantém ativo até o encerramento da chamada. O término da chamada é realizada entre os terminais através das mensagens *Release Complete*, *EndSessionComand*, *DRQ* e *DCF*.

### 3.1.2 Estabelecimento de chamada com mobilidade em um mesmo domínio (intra-domínio)

Liao [1] propôs mobilidade ao protocolo H.323 através da expansão da conferência *ad hoc* multiponto para realizar *handoff* e adicionou esta solução ao sistema descrito na seção 3.1.1. Portanto, o estabelecimento de chamada passa a ser realizado entre dois terminais H.323 móveis, com mobilidade em um mesmo domínio (**intra-domínio**) ou em domínios diferentes (**inter-domínios**).

Para mobilidade intra-domínio (*intrazone roaming*), o terminal móvel se move entre sub-redes localizadas no mesmo domínio ou zona gerenciada por um GK. Esta sub-seção descreve em detalhes as principais características para o estabelecimento de chamada entre dois terminais móveis com mobilidade intra-domínio utilizando a linguagem SDL. O sistema com mobilidade inter-domínios será apresentado na próxima subseção.

A figura 3.7 apresenta a estrutura hierárquica orientada a objeto do sistema intra-domínio. A cláusula *virtual* no sistema ancestral permite que blocos e processos possam ser redefinidos no novo sistema através do comando *redefined* para as modificações necessárias. O sistema com mobilidade intra-domínio definido no *Package PackIntraZoneMob* herda através da cláusula *USE*, a especificação do estabelecimento de chamada entre dois terminais fixos H.323 definido no *Package PackIntraZoneFix* (figura 3.8). A cláusula *inherits* (figura 3.9) permite a herança do sistema sem mobilidade *IntraZoneFix* contendo todas as suas características.

O bloco redefinido *Calling Terminal* (figura 3.10) descreve as entidades e sinalizações para o estabelecimento da chamada com mobilidade intra-domínio representadas pelos processos *TF (Terminal1)*, *TM Terminal2 móvel*, *GK* e *TM2 (Terminal3 móvel)*. Os processos pontilhados (TF, TM e GK) são herdados do sistema ancestral e o processo com linha cheia (TM2) representa uma nova entidade móvel com novas sinalizações (figura 3.11).

A figura 3.12 apresenta o comportamento funcional do *Terminal2* pertencente ao sistema ancestral. Este processo é redefinido no novo sistema para prover mobilidade intra-domínio entre os terminais móveis durante uma conexão. A figura 3.13 apresenta as características redefinidas. Os gates pontilhados M1 e M2 transportam os sinais de entrada e saída herdados do sistema ancestral que serão reutilizados pelo processo e o gate M3 traz nova sinalização. Comparando as figuras 3.12 e 3.13, foi alterado apenas o trecho entre os estados *waitendsession* e *waitdisengage*. O restante da especificação herdada permanece inalterada e é reutilizada no novo sistema. Desta forma, o sistema se torna mais simples e fácil de se especificar e acrescentar modificações futuras.

Portanto, para se obter um sistema com mobilidade intra-domínio, inicialmente é estabelecida a conexão inicial com conferência entre Terminal1 (TF) fixo e Terminal2 móvel (TM). Em algum momento, o Terminal2 móvel se move para uma nova sub-rede

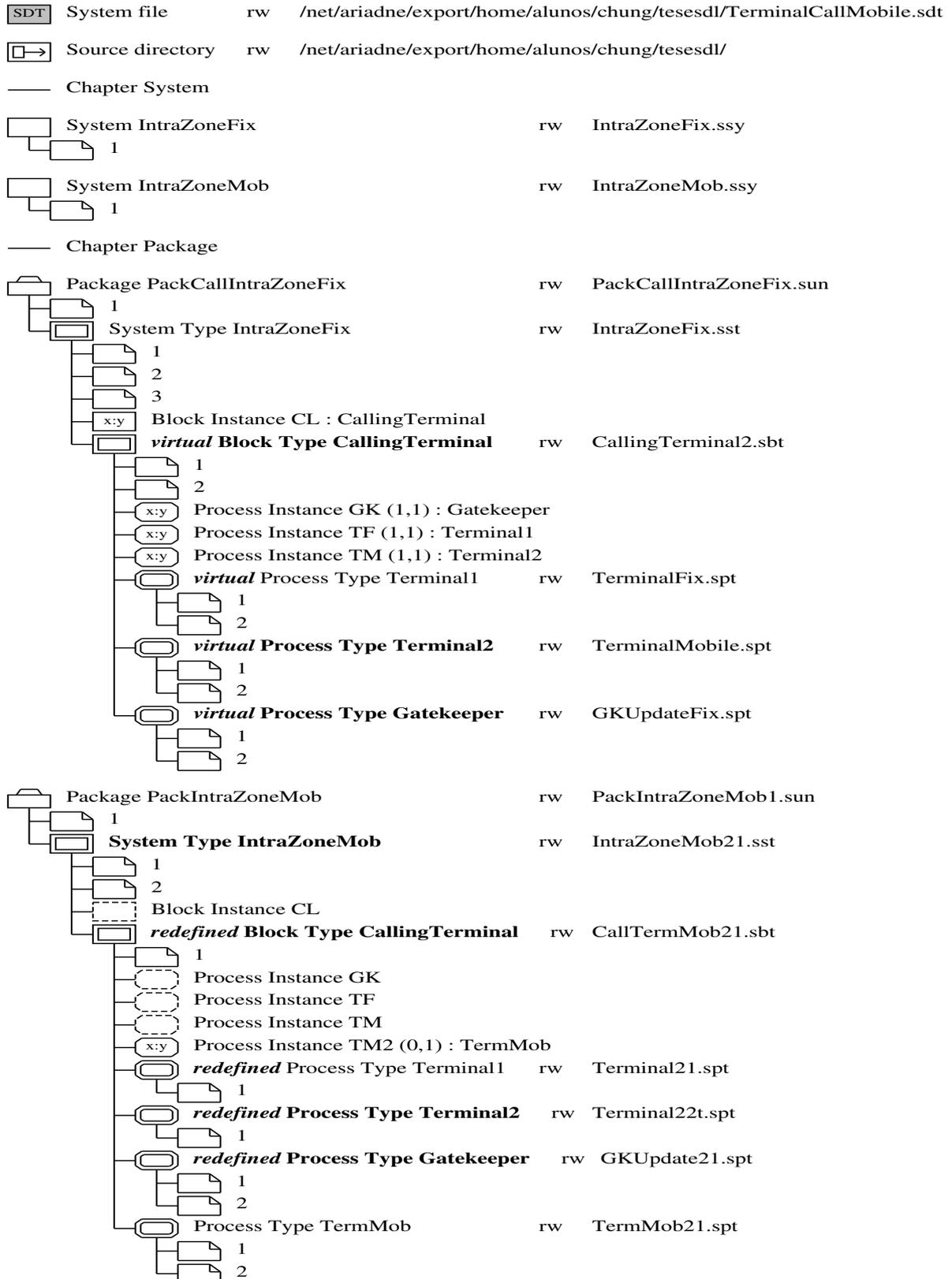


Figura 3.7: Organizer do sistema sem e com mobilidade intra-domínio

localizada no mesmo domínio (zona) e gerenciado pelo GK1. O endereço IP da nova

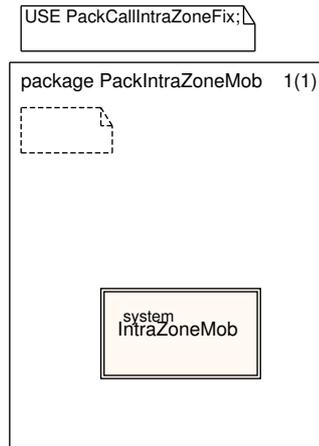


Figura 3.8: Package *PackIntraZoneMob* contendo o sistema com mobilidade intra-domínio

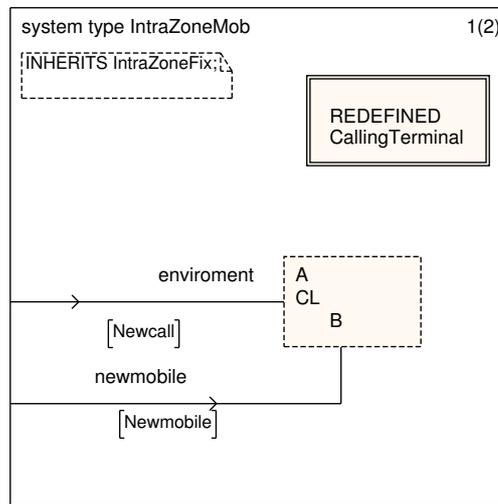


Figura 3.9: Sistema *IntraZoneMob* com mobilidade intra-domínio

sub-rede continua o mesmo referente ao endereço do GK1 (*home GK*), pois o Terminal2 móvel permanece na mesma zona. Para que a conexão não seja interrompida, é necessário o processo de *handoff*. O Terminal2 móvel (TM) ao mover para a nova sub-rede se torna o Terminal3 móvel que entra na conferência existente através do comando (*join*) enviado no sinal *SetupMob* a partir da expansão da conferência *ad hoc* multiponto, estabelecendo uma nova conexão com o Terminal1 fixo. Por alguns instantes, duas conexões com o Terminal1 (TF) irão coexistir. A conexão com o Terminal2 móvel é então encerrada, permanecendo apenas a conexão entre terminal1 (TF) e Terminal3 (TM2), finalizando o processo de *handoff* e mantendo a conexão inicial ativa sem interrupções.

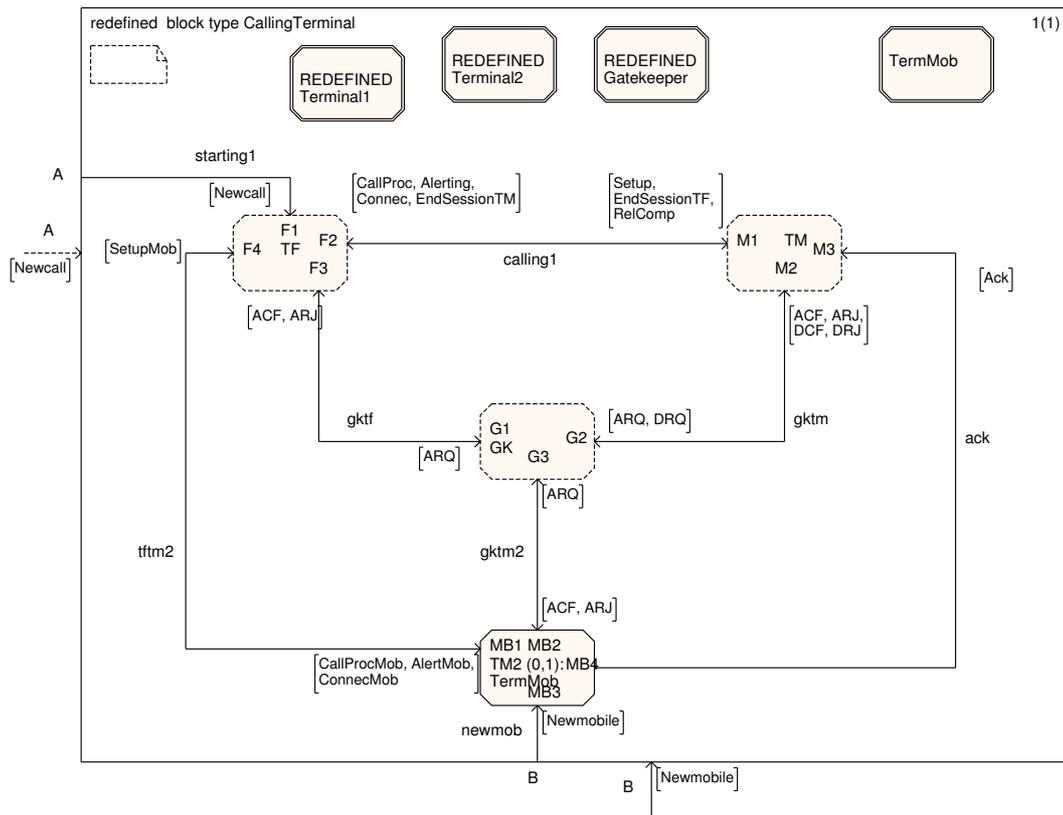


Figura 3.10: Bloco *CallingTerminal* redefinido para sistema com mobilidade intra-domínio

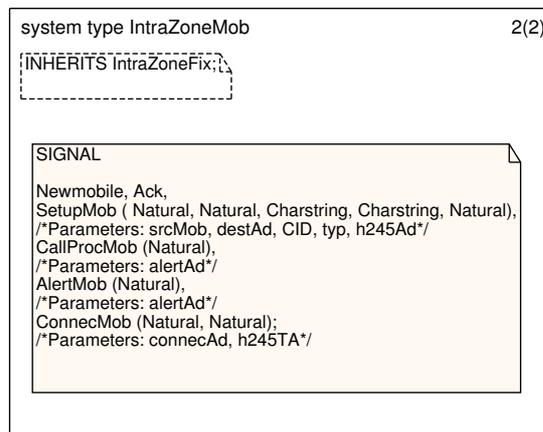


Figura 3.11: Novos sinais do processo *TermMob* no sistema intra-domínio

### 3.1.3 Estabelecimento de chamada com mobilidade em domínios diferentes (inter-domínios)

Esta sub-seção descreve em detalhes as principais características para o estabelecimento de chamada entre dois terminais móveis com mobilidade em domínios diferentes (inter-domínios) utilizando SDL.

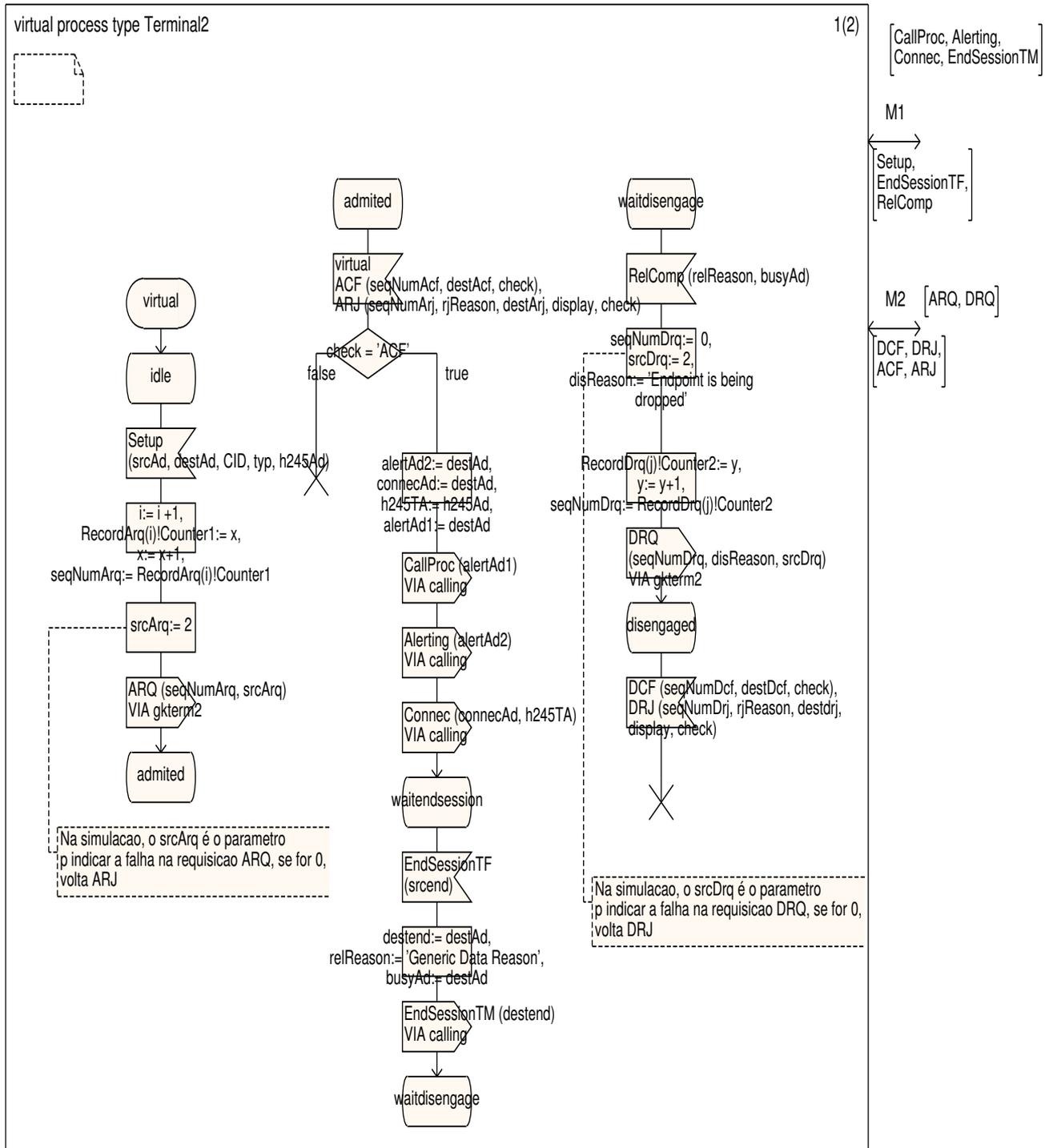


Figura 3.12: Comportamento funcional do processo *Terminal2* no sistema intra-domínio

Para a mobilidade inter-domínios (*interzone roaming*), o terminal móvel se move entre sub-redes localizadas em zonas diferentes gerenciadas por GK's distintos. O terminal móvel deve se registrar no GK visitante (*foreign GK*) toda vez que se mover para uma nova zona e atualizar os novos endereços no GK permanente (*home*) a qual pertence para que possa ser localizado.

A figura 3.14 apresenta a estrutura hierárquica em SDL do sistema sem mobilidade

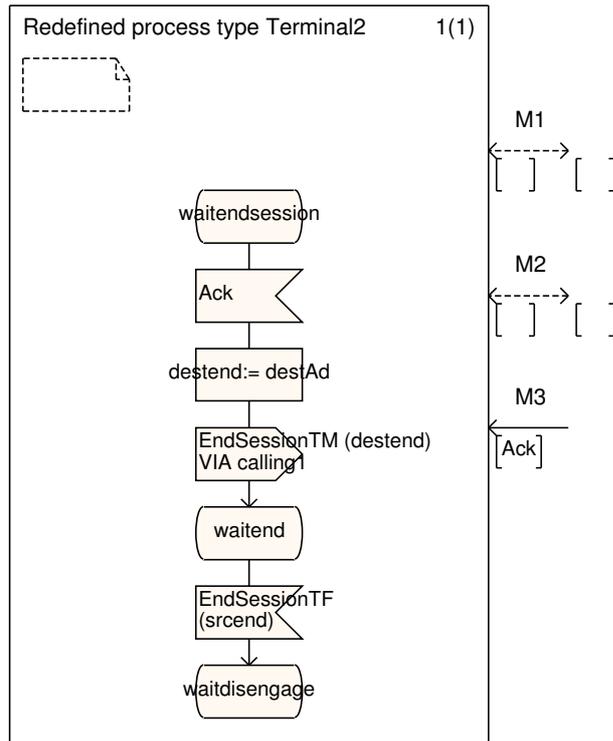


Figura 3.13: Comportamento funcional redefinido do processo *Terminal2* no sistema intra-domínio

*System Type InterZone* e do sistema com mobilidade inter-domínios *System Type HandInterZone*. O sistema com mobilidade foi definido em um pacote *PackHandInterZone* (figura 3.15). Este pacote herda todas as características do pacote *PackCallInterZone* que contém o sistema *InterZone* com o estabelecimento de chamada entre dois terminais fixos. A partir da adição da solução de Liao [1] ao sistema sem mobilidade, é possível obter o sistema com o estabelecimento de chamada entre dois terminais móveis com mobilidade inter-domínios a partir da expansão da conferência *ad hoc* multiponto para realizar *handoff* e *roaming*. Além da estrutura de *package*, os sistemas utilizam outras características de orientação a objeto. O sistema *InterZone* utiliza a cláusula *virtual* para o bloco e processos para que possam ser redefinidos no sistema *HandInterZone* através da cláusula *redefined*. A redefinição inclui modificação do comportamento funcional do bloco ou processo, adição de novas entidades e sinalizações ou reutilização de características especificadas no bloco e processos herdados.

O bloco do sistema herdado *InterZone* (figura 3.16) foi abordado com uma visão diferente comparado ao bloco do sistema especificado na sub-seção 3.1.2. O sistema *InterZone* é uma outra opção para se estabelecer uma chamada entre dois terminais H.323 fixos e é importante salientar a diferença entre estes sistemas para a compreensão da especificação do sistema com mobilidade inter-domínios. A figura 3.16 apresenta o bloco *virtual CallMobile* com as entidades (processos) e sinalizações. Os terminais fixos TL e TL1 estão registrados no GK1 e GK2, respectivamente. Cada GK gerencia uma zona distinta. Para

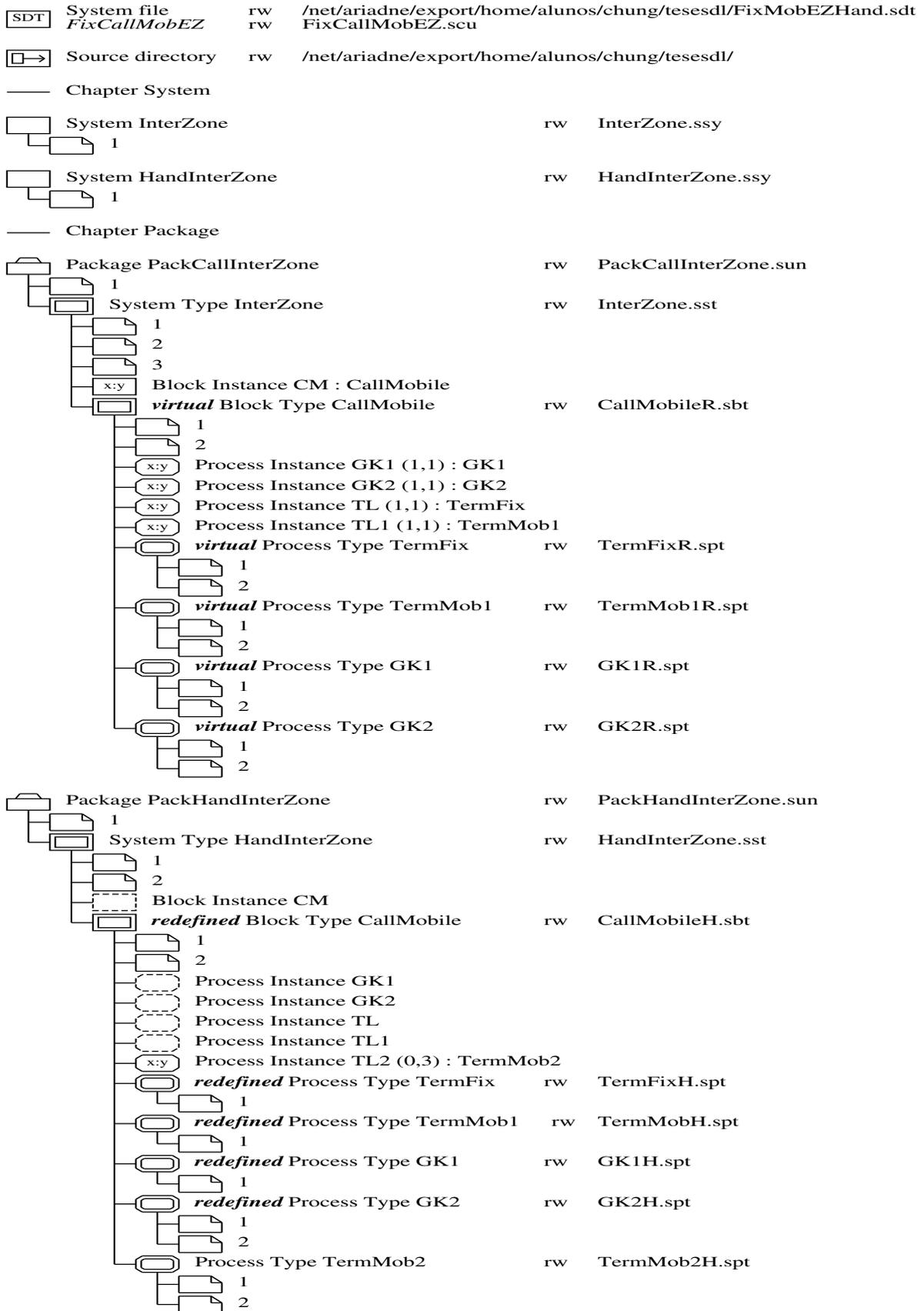


Figura 3.14: Organizer do sistema sem e com mobilidade inter-domínios

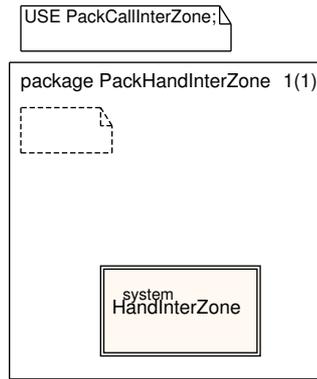


Figura 3.15: Pacote *PackHandInterZone* para sistema com mobilidade inter-domínios

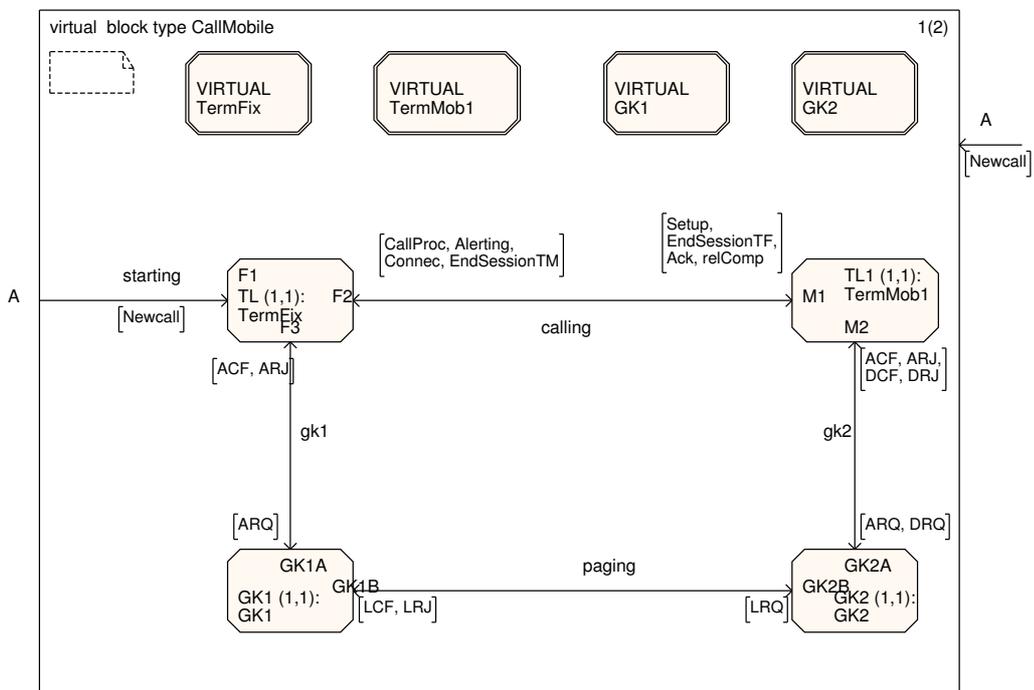


Figura 3.16: Bloco *virtual CallMobile* para sistema sem mobilidade inter-domínios

estabelecer a chamada entre os terminais TL e TL1, inicialmente o terminal TL requisita o endereço do terminal TL1. Devido ao fato de TL1 estar registrado em um GK que pertence a outra zona, o GK1 envia uma mensagem de localização (LRQ) em busca do GK2 responsável pelo TL1. Após obter a resposta de localização confirmada (LCF), inicia-se o processo comum para o estabelecimento da chamada e criação da conferência através do parâmetro *create* presente na mensagem *Setup*.

Outra característica importante na figura 3.16 é a declaração do número de instâncias dos processos. O termo *tipo (type)* utilizado nos blocos e processos indica que os mesmos podem ser instanciados. A instância desses blocos ou processos é apenas uma cópia do funcionamento desse *tipo*, possuindo características diferentes. Uma instância de um bloco é uma entidade diferente, é igual apenas no seu conteúdo, todas as instâncias podem ser

de um determinado *tipo*. A instância no processo funciona da mesma forma, apenas cada processo terá uma identidade (Pid) diferente, ou seja, um endereço diferente. Para uma melhor visualização, várias instâncias de um mesmo processo podem ser utilizadas, por exemplo para realizar um cenário de simulação. Esses processos possuem as mesmas características, a especificação é exatamente igual, mas são independentes, ou seja, cada processo pode estar em um estado diferente em um mesmo instante. Neste trabalho, foi necessário usar apenas uma instância de cada processo por não haver necessidade de utilizar mais instâncias nas simulações. Mas caso necessite em alguma implementação futura, o uso das instâncias pode ser útil para criar vários tipos de cenários de simulação. Na declaração dos processos deste bloco, os números que estão entre parênteses em cada processo indicam o número de instâncias que cada processo pode ter. Por exemplo, o conjunto (1,1) permite no máximo uma instância do processo.

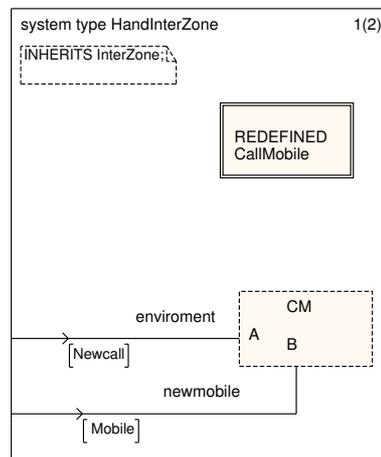


Figura 3.17: Sistema *HandInterZone* com mobilidade inter-domínios

A figura 3.17 apresenta o sistema com mobilidade *HandInterZone* que contém o bloco *redefined CallMobile* (figura 3.18). Este bloco contém os processos TL, TL1, GK1 e GK2 herdados do sistema *InterZone*. O bloco foi redefinido com a adição de um novo processo, o terminal móvel TL2. As entidades representadas pelos processos e as sinalizações presentes no bloco apresentam a especificação para obter o estabelecimento de chamada com mobilidade inter-domínios. Portanto, inicialmente ocorre o estabelecimento de chamada entre os terminais TL e TL1, conforme descrito acima. Em algum momento, o terminal móvel TL1 se move para uma nova sub-rede em um domínio diferente. Para que a conexão não seja interrompida entre TL e TL1, é necessário o processo de *roaming* e *hand-off*. Durante o *handoff*, TL1 ao se mover para a nova sub-rede em um domínio diferente se torna TL2. Na nova zona, é necessário que TL2 procure pelo GK gerenciador (mensagens GRQ e GCF) e se registre no GK (mensagens RRQ e RCF). Em seguida, TL2 entra na conferência existente através do comando (*join*) enviado no sinal *SetupMob* a partir da expansão da conferência *ad hoc* multiponto, estabelecendo uma nova conexão com TL.

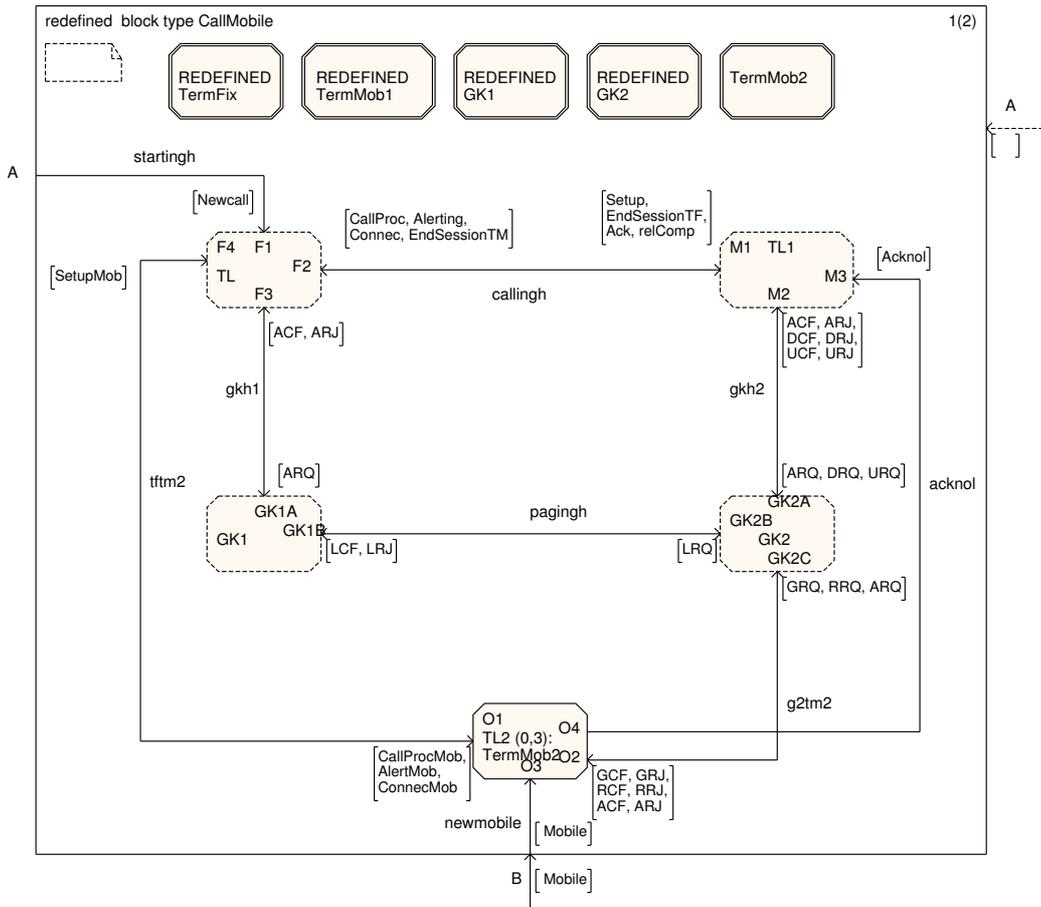


Figura 3.18: Bloco *redefined CallMobile* para sistema com mobilidade inter-domínios

Após o *handoff*, a conexão entre TL e TL1 é cancelada, mantendo a conexão ativa entre TL e TL2.

### 3.1.4 Estabelecimento de chamada com mobilidade inter-domínios e os novos conceitos do protocolo H.510

A partir da solução de mobilidade inter-domínios, desenvolveu-se neste trabalho outra proposta de mobilidade para o protocolo H.323. O sistema propõe mobilidade inter-domínios em conjunto com as novas entidades e funcionalidades do protocolo H.510, utilizando a expansão da conferência *ad hoc* multiponto para realizar *handoff* durante o estabelecimento de chamada entre terminais H.323 móveis.

Os novos conceitos do H.510 englobam novas entidades, sinalizações e funcionalidades. O HLF (*Home Location Function*) é a base de dados permanente (*home*) onde o terminal móvel armazena os seus dados e a sua atualização atual (apontando para o VLF). Esta entidade funcional está associada ao domínio permanente. O VLF (*Visitor Location Function*) é a base de dados temporária (*foreign*) de um terminal móvel visitante, indicando

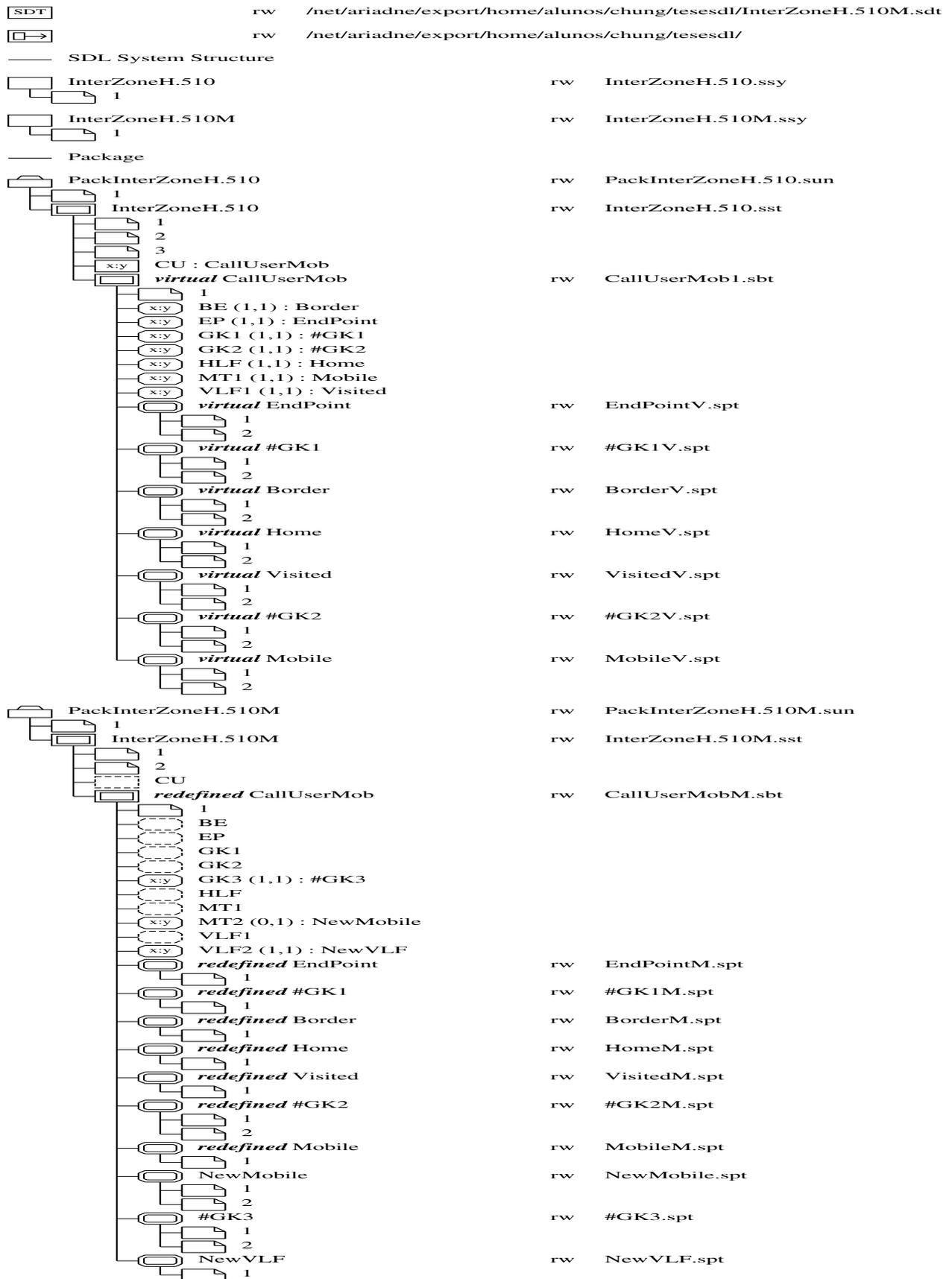


Figura 3.19: Organizador do sistema sem e com mobilidade inter-domínios e H.510

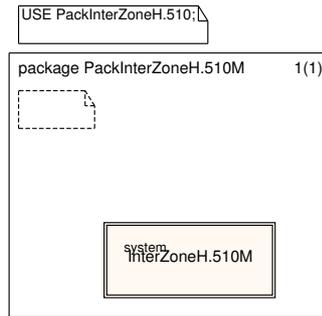


Figura 3.20: Pacote *PackInterZoneH.510M* do sistema com mobilidade inter-domínios e H.510

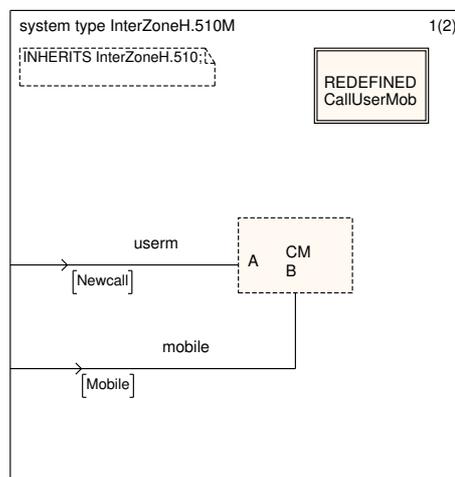


Figura 3.21: Sistema *InterZoneH.510M* com mobilidade inter-domínios e H.510

o GK onde o móvel está registrado e o HLF correspondente. Esta entidade funcional está associada ao domínio permanente e visitante. Estas entidades são responsáveis pelos novos serviços de localização (**Location Tracking**) e atualização de endereços (**Location Update**) do terminal móvel. Para a atualização das entidades após o processo de *handoff*, utiliza-se as novas sinalizações **DescriptorUpdate** e **DescriptorUpdateAck**.

A figura 3.19 descreve a estrutura da linguagem SDL orientada a objeto dos novos sistemas sem e com mobilidade inter-domínios. O pacote *PackInterZoneH.510M* (figura 3.20) define o sistema com mobilidade inter-domínios e H.510 *InterZoneH.510M*. A proposta herda as características do sistema com o estabelecimento de chamada entre terminais H.323 móveis com características de um ambiente inter-domínios (sem mobilidade) e as novas entidades pertencentes ao protocolo H.510, através da cláusula *inherits* (figura 3.21). O sistema contém o bloco *redefined CallUserMob* (figura 3.22) que por sua vez possui os processos herdados e redefinidos correspondentes às entidades e as sinalizações para o estabelecimento de chamada e troca de capacidades. Desta forma, as características inter-domínios herdadas podem ser expandidas para gerar o sistema com mobilidade inter-



uma conexão com o terminal EP através da sua entrada na conferência com o parâmetro *join* no sinal *SetupMob* enviado ao terminal EP. Após o *handoff*, a conexão entre MT1 e EP (*endpoint*) é liberada, permanecendo apenas a conexão entre MT2 e EP.

## 3.2 SIP

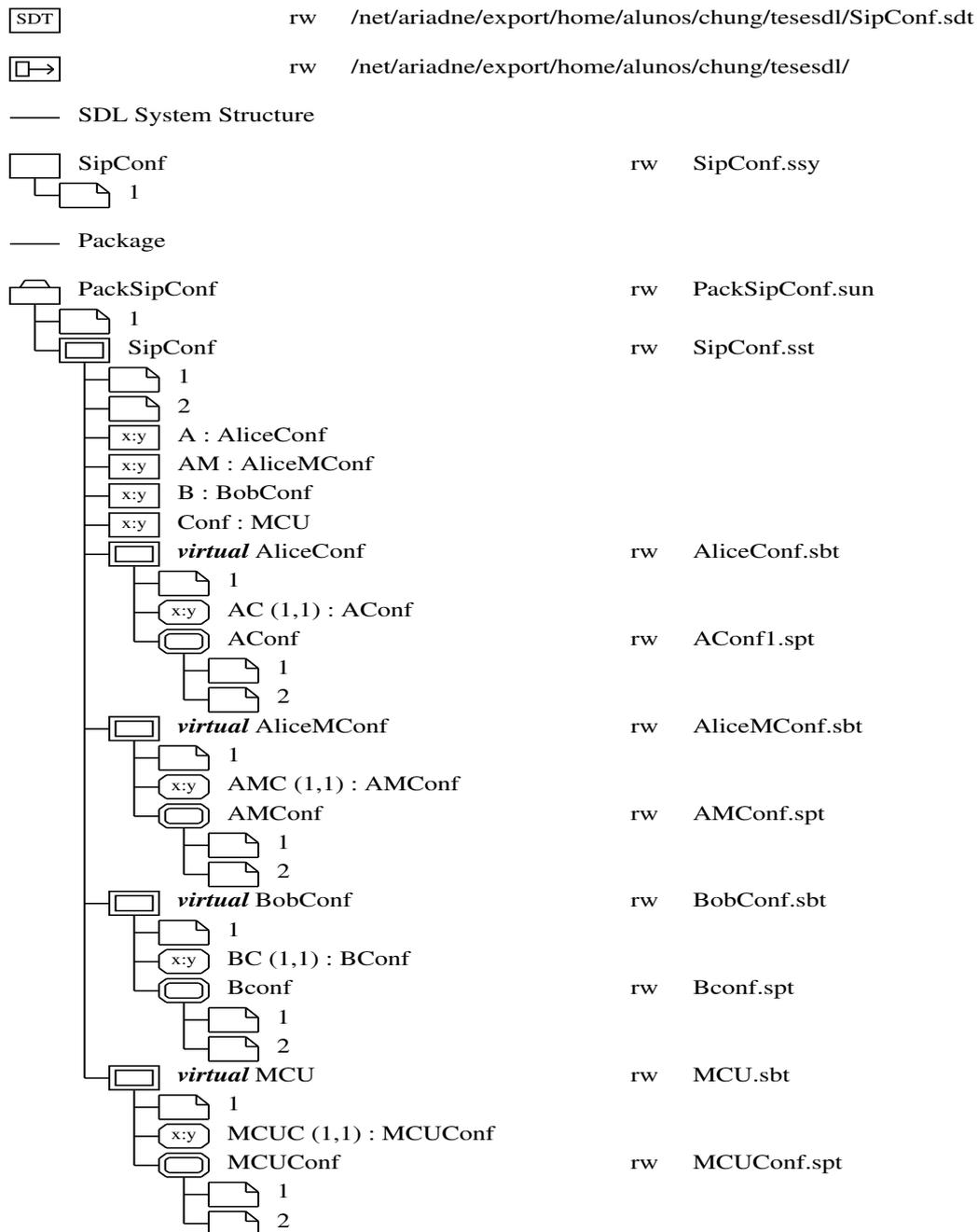


Figura 3.23: Organizador do sistema com mobilidade para SIP

Uma segunda linha de pesquisa, que compete com o protocolo de telefonia IP H.323, é o protocolo de geração mais recente denominado SIP, desenvolvido pelo IETF [24]. A

recomendação atual do protocolo SIP contempla a mobilidade para usuários através dos servidores *proxy* e de redirecionamento que encaminham as requisições para a localidade atual do usuário. Porém, não existe nenhum mecanismo de *handoff* para manter a conexão ativa durante o estabelecimento de sessão ou chamada entre dois terminais móveis. O trabalho de Chen, Berquin e Moh [2] se baseou na mesmo conceito de Liao [1] para prover mobilidade entre dois terminais SIP, realizando *handoff* a partir da conferência com presença de MCU (*Multipoint Control Unit*).

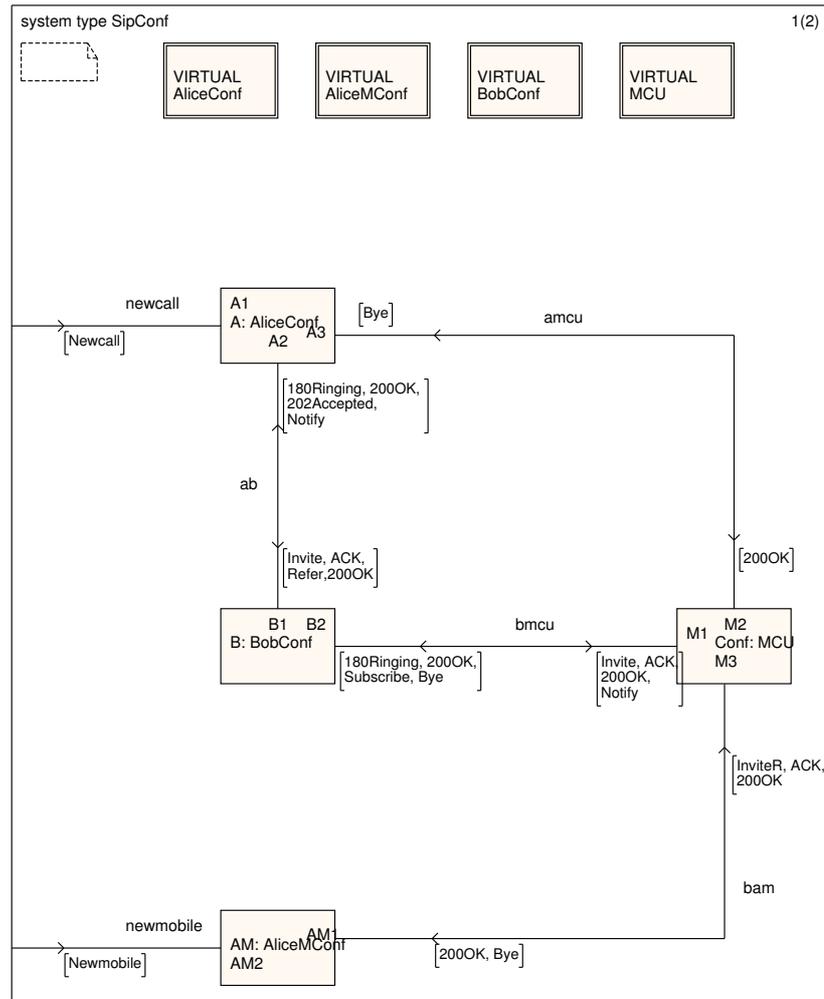


Figura 3.24: Sistema SIP com mobilidade

O organizador do sistema SIP com mobilidade descreve a estrutura hierárquica desenvolvida em SDL (figura 3.23).

O sistema *SipConf* (figura 3.24) do estabelecimento de chamada com mobilidade entre dois terminais SIP foi definido na estrutura de pacote *PackSipConf* (figura 3.25) e não utilizou herança de outro sistema. O sistema com mobilidade é o ancestral ou a super-classe. Os blocos e processos estão declarados como *virtual* para possível expansão e redefinição do sistema. Cada bloco *AliceConf*, *BobConf*, *MCU* e *AliceMConf* possui apenas um processo (o próprio bloco) e cada processo possui uma instância. Os processos

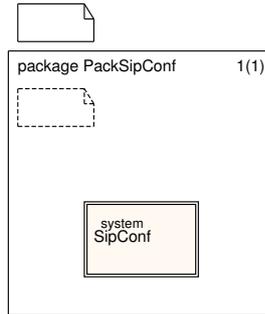


Figura 3.25: Pacote *PackSipConf* SIP com mobilidade

*AliceConf* e *BobConf* são clientes, terminais móveis, que requisitam tarefas e recebem respostas através do estabelecimento de chamada entre os mesmos. A MCU é um servidor responsável pelos serviços de conferência entre dois ou mais terminais.

Para estabelecer a chamada, *AliceConf* envia um *invite* para *BobConf* para estabelecer uma chamada. O sinal *invite* envia o endereço da MCU. Com este endereço, *BobConf* convida a MCU com o *invite* e estabelecem uma conferência entre as três entidades. Em um dado momento, o terminal móvel *AliceConf* se move para uma nova sub-rede em um outro domínio. Para não perder a conexão ativa, é necessário realizar o processo de *roaming* e *handoff*. Ao mover para a nova sub-rede, o terminal móvel *AliceConf* adquire um novo endereço temporário (*care-of-address*), se tornando *AliceMConf*. O novo terminal móvel convida a MCU com um *invite* contendo o *care-of-address* e o seu endereço antigo e estabelecem uma nova conexão. Após o *handoff*, a conexão entre *AliceConf* e MCU é cancelada, mantendo a conexão entre *AliceMconf* e MCU.

# Capítulo 4

## Resultados da Validação e das Simulações

A validação e as simulações dos sistemas especificados neste trabalho foram realizadas através do *Validator* e *Simulator* respectivamente, ferramentas integrantes do TAU SDL Suite [37].

A validação identifica possíveis erros ocorridos durante a especificação do sistema e verifica todos os estados possíveis do sistema analisando o comportamento de cada sinal. Os erros encontrados são corrigidos na especificação e os sistemas são validados para a implementação.

A simulação verifica a dinâmica do sistema analisando em detalhes o comportamento de casos críticos.

Na seção 4.1 são apresentados os resultados das validações dos sistemas especificados no capítulo 3 e é analisado como o processo de validação pode ser utilizado para identificar erros de especificação. Na seção 4.2 são apresentadas algumas simulações para analisar casos críticos e o comportamento funcional dos sistemas, além de identificar erros não detectados na validação, mostrados através de gráficos MSC (*Message Sequence Chart*).

### 4.1 Validação

O módulo *Validator* do TAU SDL Suite apresenta três formas diferentes para realizar o processo de validação do sistema: *Exhaustive Exploration*, *Random Walk* e *Bit State Exploration*.

O *Exhaustive Exploration* é o algoritmo utilizado para percorrer todos os caminhos possíveis dos estados do sistema, sendo o mais adequado para sistemas pequenos. À medida que o número de estados cresce significativamente, a utilização do algoritmo se torna impraticável.

O *Random Walk* é o mais indicado para sistemas de maior porte, onde o número de

estados é bastante considerável, mas também é eficiente para sistemas menores. Este algoritmo percorre de forma aleatória os ramos da árvore com os estados possíveis do sistema, simplificando o gerenciamento de estados percorridos e não percorridos.

O *Bit State Exploration* possui a característica de ser mais complexo e utilizado para analisar sistemas razoavelmente grandes, como é o caso dos sistemas especificados neste trabalho. O resultado deste algoritmo fornece dados estatísticos percorridos no sistema especificado e utiliza de uma estrutura denominada *Hashtable* para gerenciar os estados percorridos.

O *Bit State Exploration* é o algoritmo que fornece resultados e análise mais completa e detalhada dos sistemas especificados neste trabalho. A seguir são apresentados os resultados obtidos para os sistemas durante o processo de validação e discutidos os resultados de maior relevância.

|   |   |
|---|---|
| <p>Sistema Interzone e H.510</p> <p><b>** Bit State Exploration **</b><br/>           Number of reports: 0<br/>           Generated states: 40<br/>           Truncated paths: 0<br/>           Unique System States: 39<br/>           Colision risk: 0%<br/>           Max depth: 39<br/>           Current depth: -1<br/>           Min State size: 620<br/>           Max state size: 916<br/>           Symbol Coverage: 82,64 %</p> | <p>Sistema Interzone com Mobilidade e H.510</p> <p><b>** Bit State Exploration **</b><br/>           Number of reports: 0<br/>           Generated states: 85<br/>           Truncated paths: 0<br/>           Unique System States: 83<br/>           Colision risk: 0%<br/>           Max depth: 166<br/>           Current depth: -1<br/>           Min State size: 988<br/>           Max state size: 1380<br/>           Symbol Coverage: 78,86%</p> |
|---|---|

Figura 4.1: Resultados da validação do sistema com mobilidade entre domínios diferentes e H.510 (ver figura 3.19)

A figura 4.1 apresenta os resultados da validação dos sistemas sem mobilidade e com mobilidade inter-domínios (figura 3.19) e conceitos de H.510. Interpretando os resultados da validação do sistema sem mobilidade, o número de mensagens (*Number of Reports*) igual a 0 (zero) significa que nenhum erro ou aviso foi encontrado; nenhum caminho foi truncado (*Truncated Path*); o número de estados únicos do sistema é de 39; o número real de estados do sistema é de 40 e significa que a validação percorreu todos os estados existentes no sistema. O risco de colisão (*Colision Risk*) do sistema foi de 0%.

Os resultados da validação do sistema com mobilidade apresenta o número de mensagens (*Number of Reports*) igual a 0 (zero) e significa que nenhum erro ou aviso foi encontrado; nenhum caminho foi truncado (*Truncated Path*); o número de estados únicos do sistema é de 83; o número real de estados do sistema é de 85 e significa que a validação

percorreu todos os estados existentes no sistema. O risco de colisão (*Collision Risk*) do sistema também foi de 0%.

O número de estados únicos se refere aos números de estados declarados nas especificações. A existência de um número maior de estados gerados ocorre quando existe fila de sinais em um estado, pois um estado pode consumir mais de um sinal.

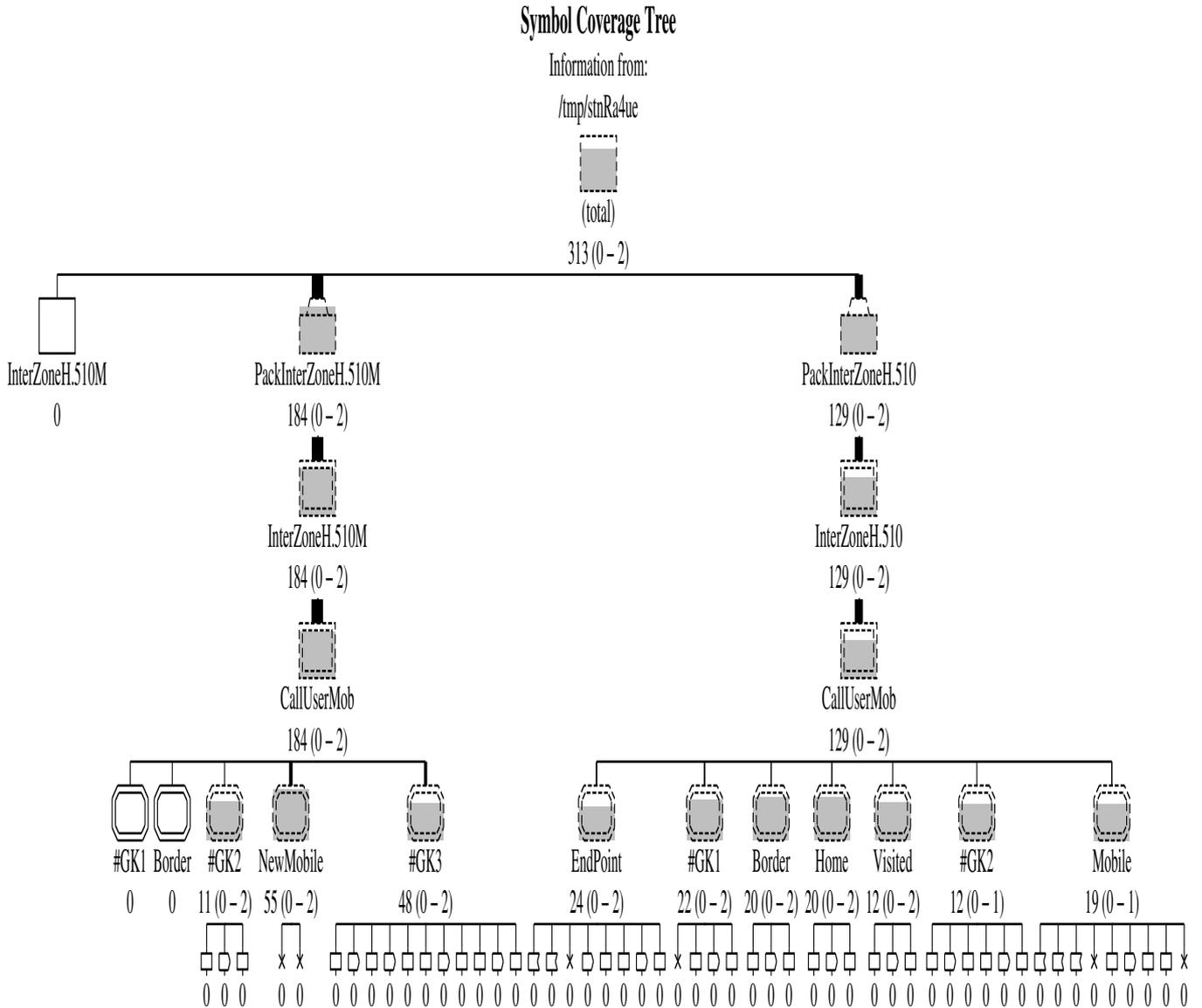


Figura 4.2: Cobertura do sistema inter-domínios com mobilidade e H.510 entre dois terminais H.323 (ver figura 3.19)

Conforme mostra a figura 4.1, o sistema sem mobilidade apresenta 82,64% dos símbolos percorridos (*Symbol Coverage*) enquanto que o sistema com mobilidade, 78,86%. A porcentagem de símbolos percorridos é menor no sistema com mobilidade devido à herança das características do sistema ancestral. Os blocos e processos herdados podem ser totalmente ou parcialmente redefinidos, modificando apenas as características necessárias

para atender o novo sistema. Na validação deste sistema, a varredura dos símbolos é feita em ambos os sistemas, as partes que não foram redefinidas são percorridas no sistema ancestral (herdado) e as partes redefinidas são percorridas no novo sistema. Portanto, a porcentagem de símbolos percorridos no sistema com mobilidade diminuem. Outro motivo para a diminuição dessa porcentagem de cobertura dos símbolos é a presença de ações condicionais na especificações, por exemplo, a condição de *if*. Apenas uma das condições, verdadeiro ou falso, será percorrida, diminuindo a quantidade de símbolos percorridos.

Os sistemas também podem ser validados a partir da ferramenta gráfica *Coverage Viewer*. Os resultados apresentam a quantidade de símbolos percorridos e as transições executadas nos sistemas. A figura 4.2 descreve a árvore composta pelos sistemas sem mobilidade herdado (*PackInterZoneH.510*) e o novo sistema com mobilidade e herança (*PackInterZoneH.510M*). No sistema com mobilidade, os dois processos *GK1* e o *Border* estão sem preenchimento. Isto se deve ao fato dos processos herdarem todas as características funcionais do ancestral, sem haver a necessidade de redefini-los, diminuindo a porcentagem dos símbolos percorridos neste sistema. Os processos com preenchimento incompleto indicam a presença de ações condicionais, por exemplo, diminuindo também a área de varredura. Os números abaixo de cada processo indicam o número de vezes que cada processo foi percorrido durante a varredura dos sistemas.

Sistema Sip com  
Mobilidade

**\*\* Bit State Exploration \*\***  
 Number of reports: 0  
 Generated states: 40  
 Truncated paths: 0  
 Unique System States: 37  
 Colision risk: 0%  
 Max depth: 74  
 Current depth: -1  
 Min State size: 232  
 Max state size: 988  
 Symbol Coverage: 100 %

Figura 4.3: Resultados da validação do sistema SIP (ver figura 3.23)

A figura 4.3 apresenta os resultados da validação do sistema com mobilidade para SIP. A porcentagem de símbolos percorridos atingiu 100% no sistema. Este fato se deve à ausência de características de herança no sistema, além de não ter atingido nenhuma ação condicional na validação.

### 4.1.1 Detecção de erros de especificação

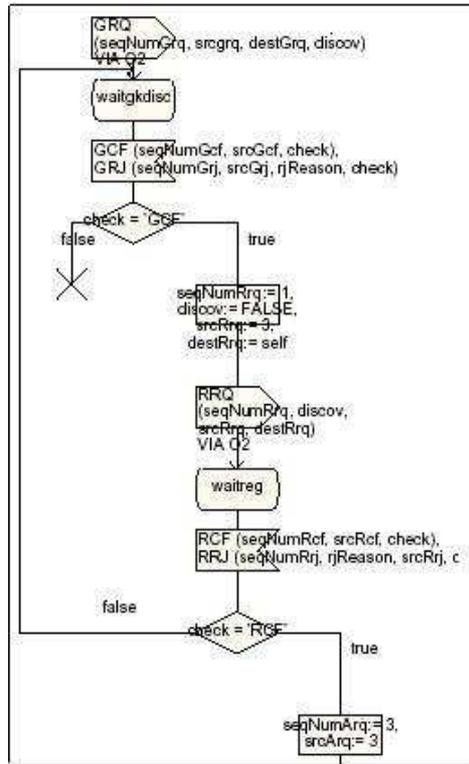


Figura 4.4: Erro de "loop" detectado na validação do sistema com mobilidade inter-domínios (ver figura 3.14)

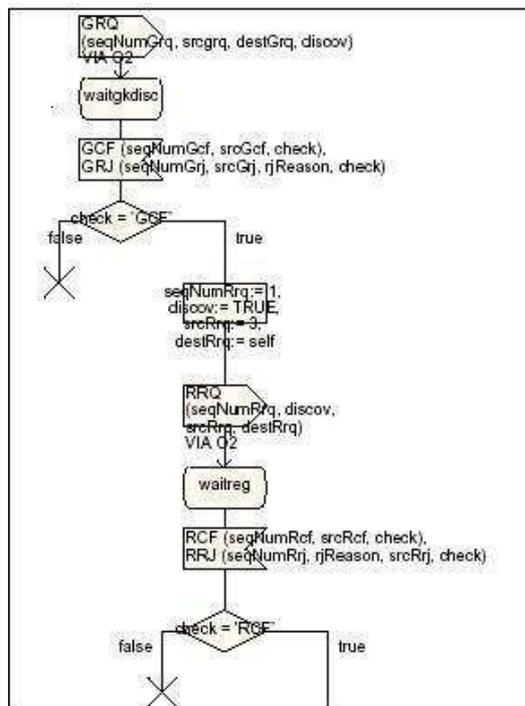


Figura 4.5: Erro de "loop" corrigido no sistema com mobilidade inter-domínios

A validação é uma ferramenta utilizada para facilitar a detecção de erros ocorridos na especificação. Através do *Coverage Viewer*, por exemplo, pode-se identificar símbolos que não foram alcançados na validação e detectar possíveis erros.

A figura 4.4 exemplifica o erro de "loop" detectado durante a validação antes da correção. Após o terminal móvel ter requisitado o registro ao GK através do sinal RRQ (*Registration Request*) na nova sub-rede, a resposta à requisição foi negativa retornando RRJ (*Registration Reject*), o tempo de vida da requisição tinha expirado. Se a resposta for RRJ, o processo deve ser encerrado. Antes da correção, após a condição *check = 'RCF'*, o terminal móvel retornava o pedido de uma nova requisição para descobrir um novo GK (GRQ), entrando em um loop. A figura 4.5 apresenta o erro de loop corrigido, o processo deve ser encerrado quando a requisição RRQ receber o sinal RRJ.

Outros tipos de erros podem ser detectados durante a validação dos sistemas. Por exemplo, o erro de **consumação implícita de sinal** (*Implicit Signal Consumption*). O erro significa que um sinal foi enviado a um processo e não vai ser tratado pelo processo no estado em que ele se encontra, indicando que a especificação deve ser revista. Outro tipo de erro é o de **output**. Um processo envia um sinal, mas não existe um receptor para esse sinal. Isto indica uma reavaliação da especificação.

Após a correção dos erros detectados na validação, os sistemas podem ser submetidos a testes mais detalhados para verificar erros de lógica que comprometam o funcionamento dos processos especificados.

O recurso de simulação poderá ser utilizado para realizar uma análise mais detalhada e criteriosa dos sistemas especificados. Através de gráficos MSC [23], pode-se analisar o comportamento funcional dos sistemas e verificar passo-a-passo se as especificações estão de acordo com as recomendações.

## 4.2 Simulações

As simulações realizadas podem detectar erros de lógica nas especificações dos sistemas não identificados no processo de validação e também estuda casos críticos baseado nos sistemas especificados. Através de gráficos MSC gerados pela simulação, é possível visualizar de forma gráfica o comportamento funcional dos sistemas em detalhes e verificar se os mesmos estão de acordo com o funcionamento previsto nos protocolos propostos.

No capítulo 3 foram propostos sistemas especificados em SDL que envolvem o estabelecimento de chamada entre dois terminais fixos H.323 (figura 3.1) e sistemas com mobilidade intra-domínio (figura 3.7) e inter-domínio (figura 3.14) realizando *handoff* a partir da expansão da conferência *ad hoc* multiponto para continuidade da conexão em uma eventual mudança de sub-rede pelo terminal móvel. Baseado nos sistemas com mobilidade, a proposta (figura 3.19) foi especificada envolvendo o estabelecimento de chamada com mobilidade inter-domínio com suporte a *handoff* a partir da expansão da conferência

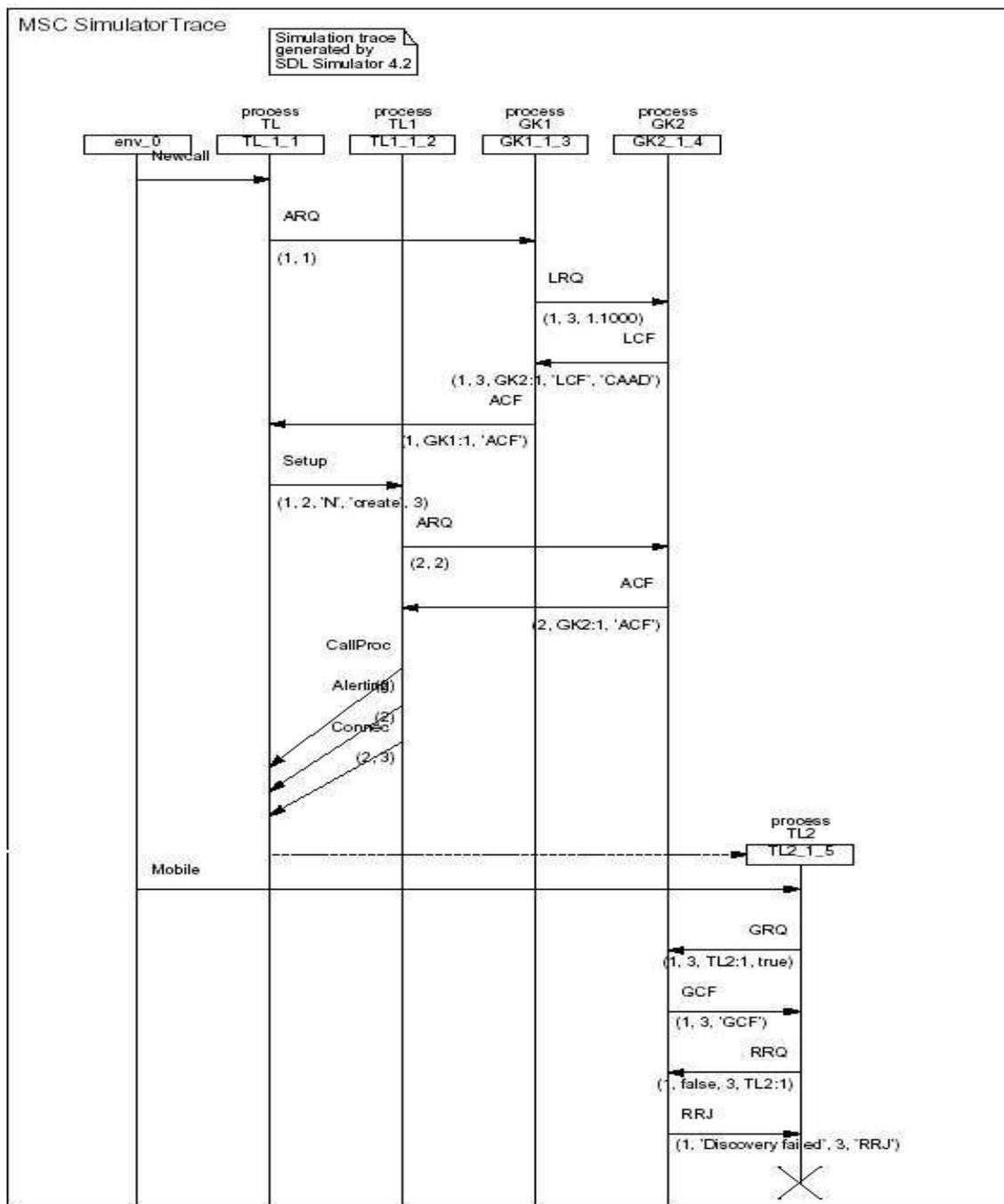


Figura 4.6: MSC do estabelecimento de chamada com mobilidade inter-domínios com erro de RRJ (ver figura 3.14)

*ad hoc* multiponto em conjunto com as novas entidades, funcionalidades e sinalizações do protocolo H.510. Para o protocolo SIP, foi especificado um sistema (figura 3.23) com estabelecimento de sessão entre terminais SIP móveis com mobilidade a partir da conferência com presença de MCU para realizar o processo de *handoff*.

Baseado nos sistemas propostos, são apresentadas algumas simulações de casos críticos considerados de maior importância e erros detectados durante o processo de simulação e posteriormente corrigidos, a partir de gráficos MSC.

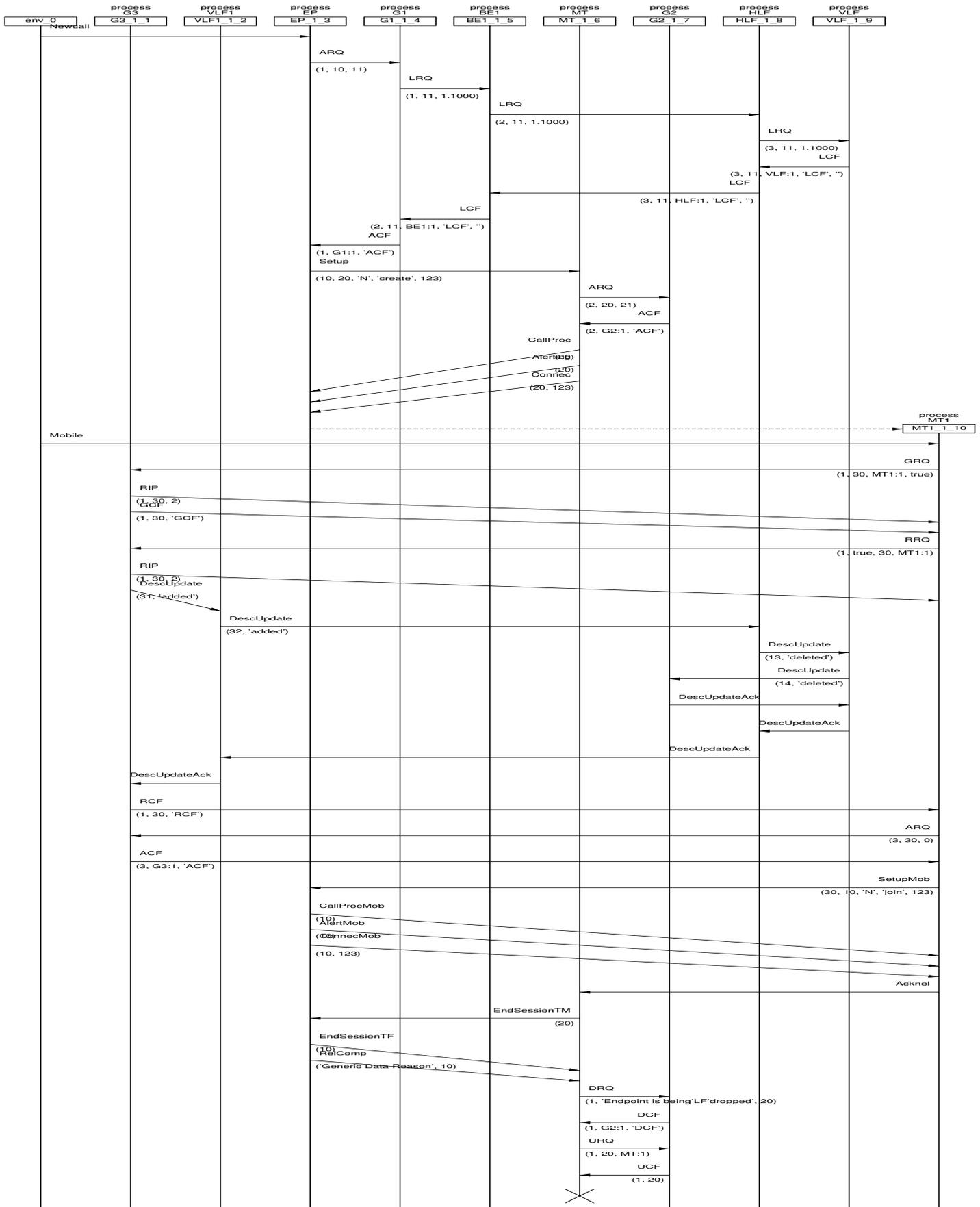


Figura 4.7: MSC do estabelecimento de chamada com mobilidade inter-domínios e conceitos de H.510 (ver figura 3.19)

A figura 4.6 apresenta o MSC de um erro identificado durante a simulação do estabelecimento de chamada entre terminais móveis com mobilidade inter-domínios (figura 3.14). O MSC descreve os processos ou entidades: *environment*, *TL*, *TL1*, *TL2*, *GK1* e *GK2* e as sinalizações para estabelecer a chamada. Durante a simulação foi identificado

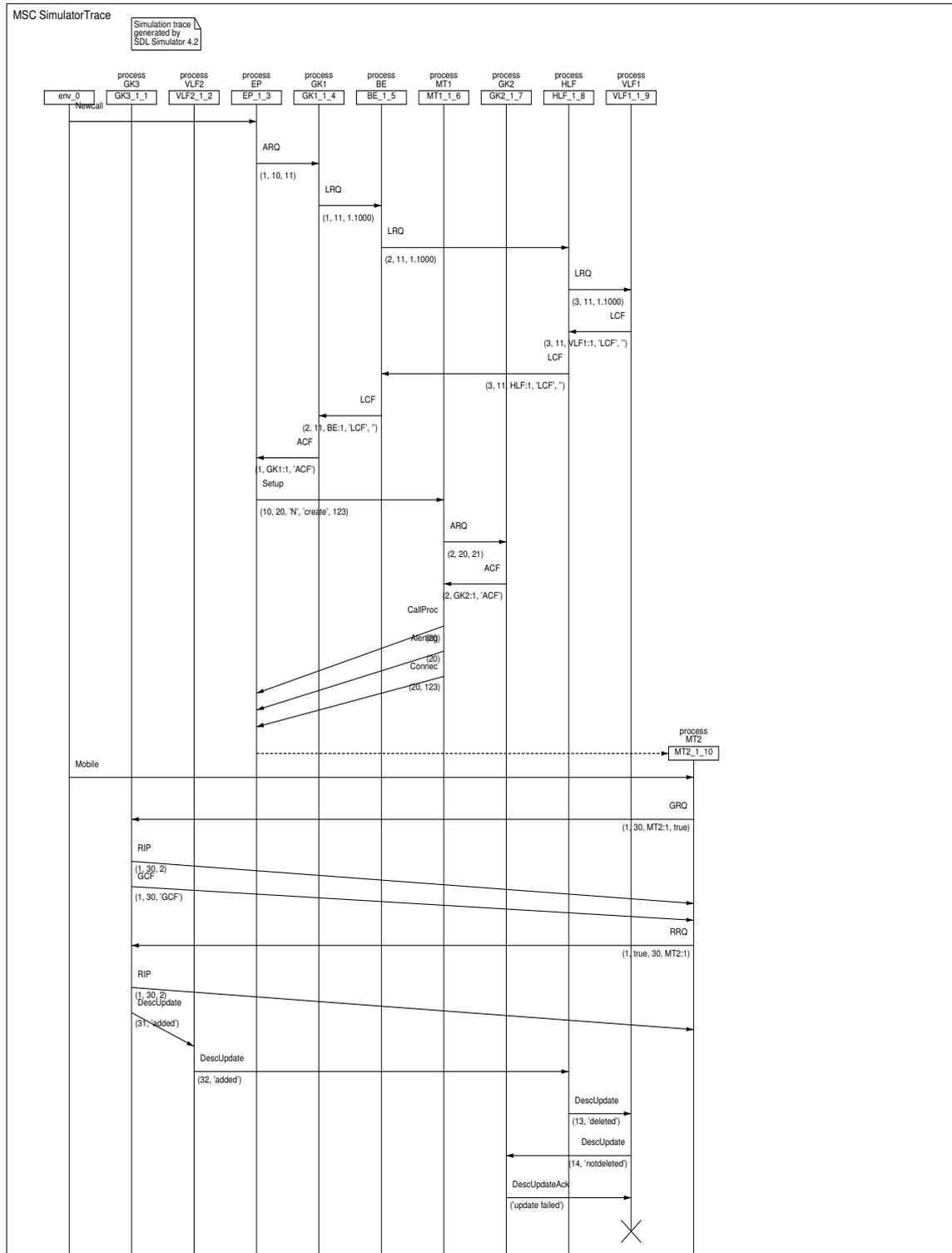


Figura 4.8: MSC do estabelecimento de chamada com mobilidade inter-domínios e H.510 com erro de atualização (ver figura 3.19)

o erro RRJ (*Registration Reject*) ocorrido durante o processo de *roaming* e *handoff*. Para que o terminal TL mantenha a conexão com o terminal móvel TL1, TL1 se move para uma nova sub-rede gerenciada por um novo GK e se torna o terminal móvel TL2. Antes de realizar o processo de *handoff*, é necessário descobrir em qual GK o novo terminal móvel TL2 visitante irá se registrar. TL2 envia uma mensagem de *broadcast* para todos os GK's e descobre o GK responsável pela nova zona. Após encontrado o GK, o terminal móvel TL2 deve fazer o registro no GK2. O erro durante a simulação ocorreu nesta fase,

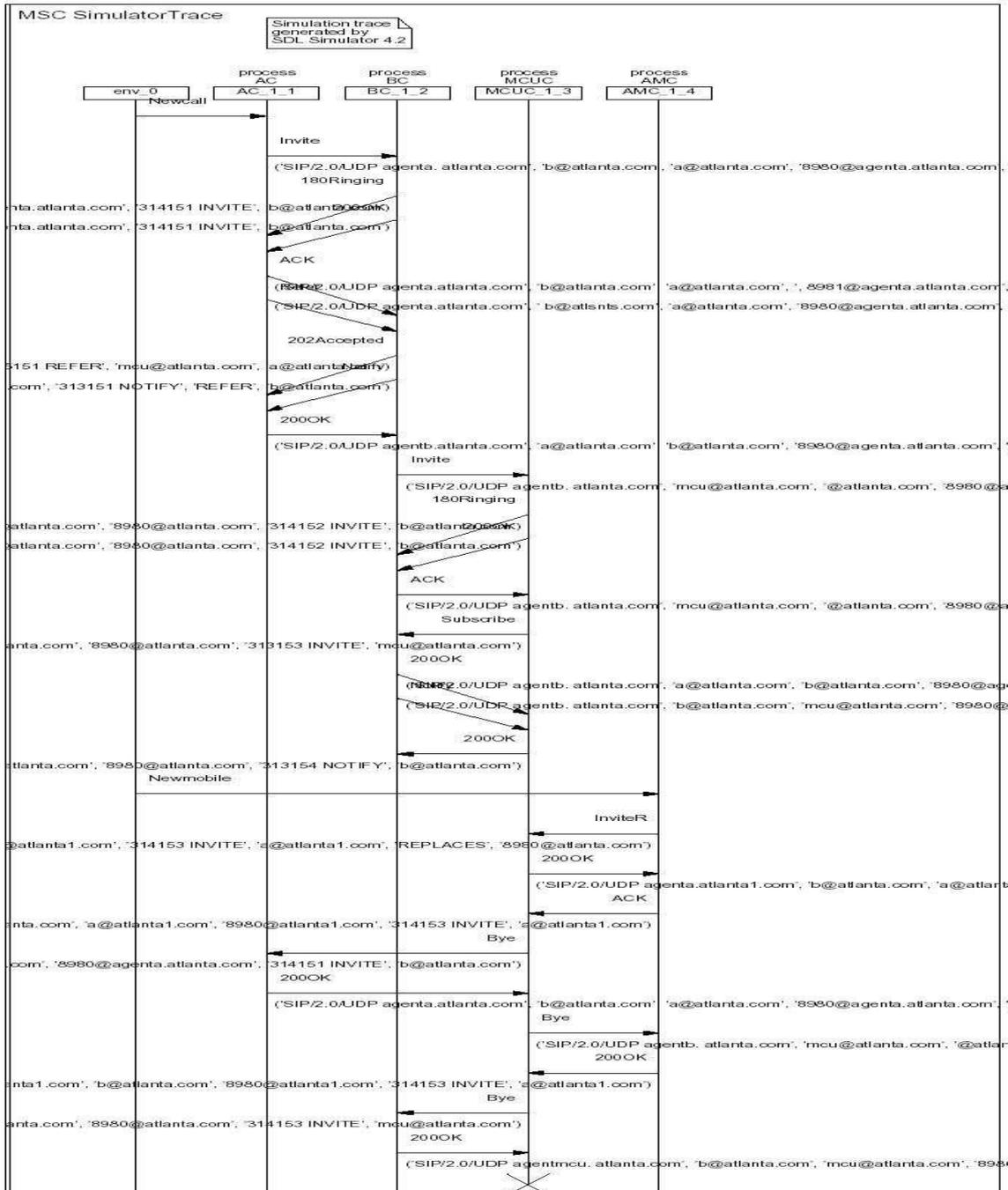


Figura 4.9: MSC do estabelecimento de sessão com mobilidade inter-domínios entre terminais SIP (ver figura 3.23)

TL2 ao enviar o sinal RRQ (*Registration Request*) para se registrar no GK correspondente, o GK2, recebeu o erro RRJ (*Registration Reject*) indicando que o tempo de vida (*time to live*) do terminal móvel TL2 expirou e por isso deve ser enviado o sinal de RRQ novamente para o GK2. O *time to live* é definido em segundos.

A figura 4.7 apresenta o MSC gerado durante a simulação do estabelecimento de chamada entre terminais móveis H.323 com mobilidade inter-domínios e conceitos de H.510 (figura 3.19). A nova proposta apresenta entidades e funcionalidades de mobili-

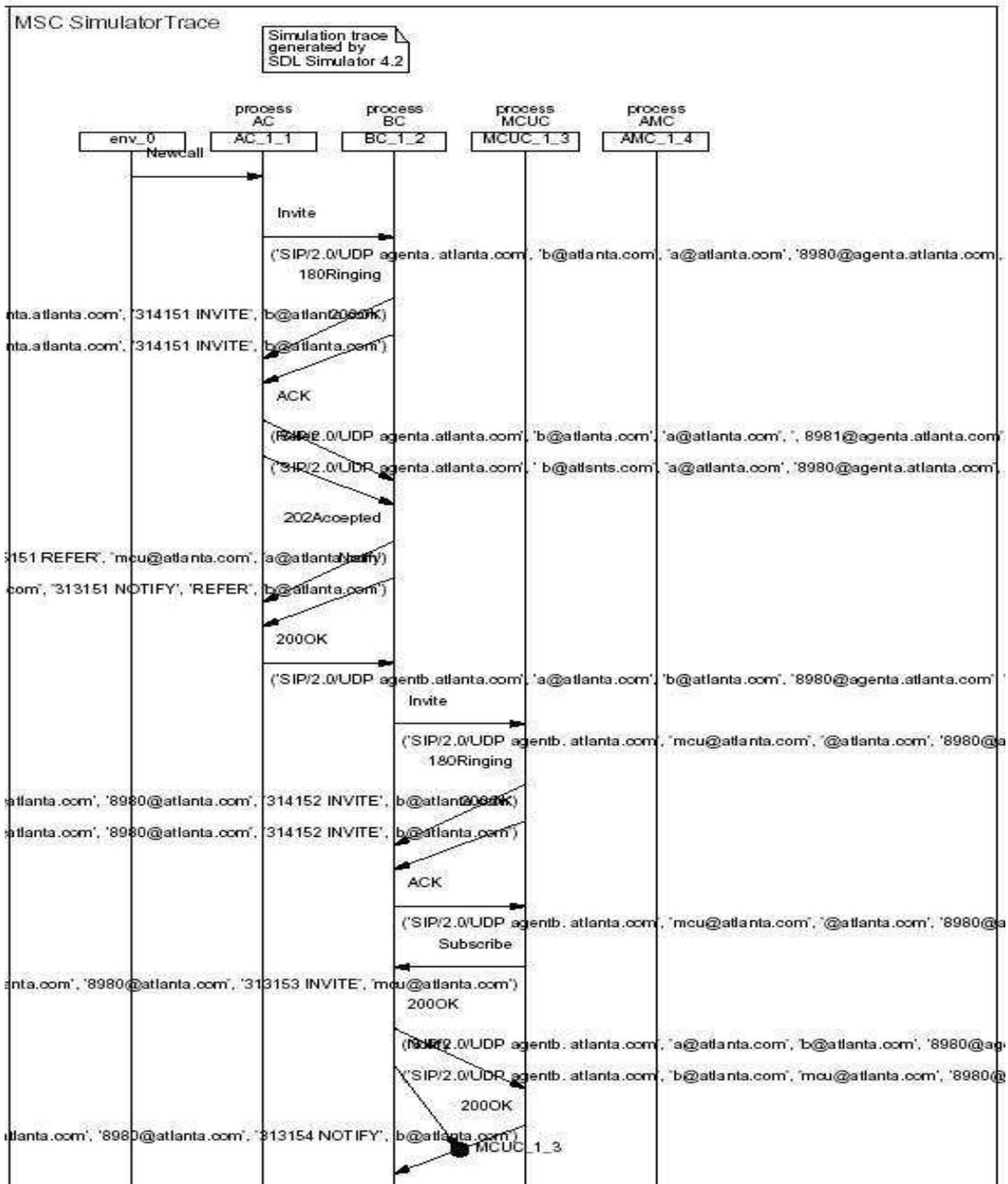


Figura 4.10: MSC com o erro gerado na simulação do estabelecimento de sessão entre terminais SIP (ver figura 3.23)

dade inexistentes na recomendação atual do protocolo H.323. As entidades HLF (*Home Location Function*) e VLF (*Visitor Location Function*) e as sinalizações *DescriptorUpdate* e *DescriptorUpdateAck* são responsáveis pela atualização dos endereços e localização dos terminais móveis. O gráfico MSC apresenta os processos *environment*, *EP*, *MT*, *MT1*, *BE*, *GK1*, *GK2*, *GK3*, *HLF*, *VLF1*, *VLF2* para o estabelecimento de chamada entre os terminais móveis H.323. Para iniciar a simulação, o EP recebe o sinal do ambiente externo (*environment*). EP pede admissão para o GK1 o qual está registrado e o endereço do ter-

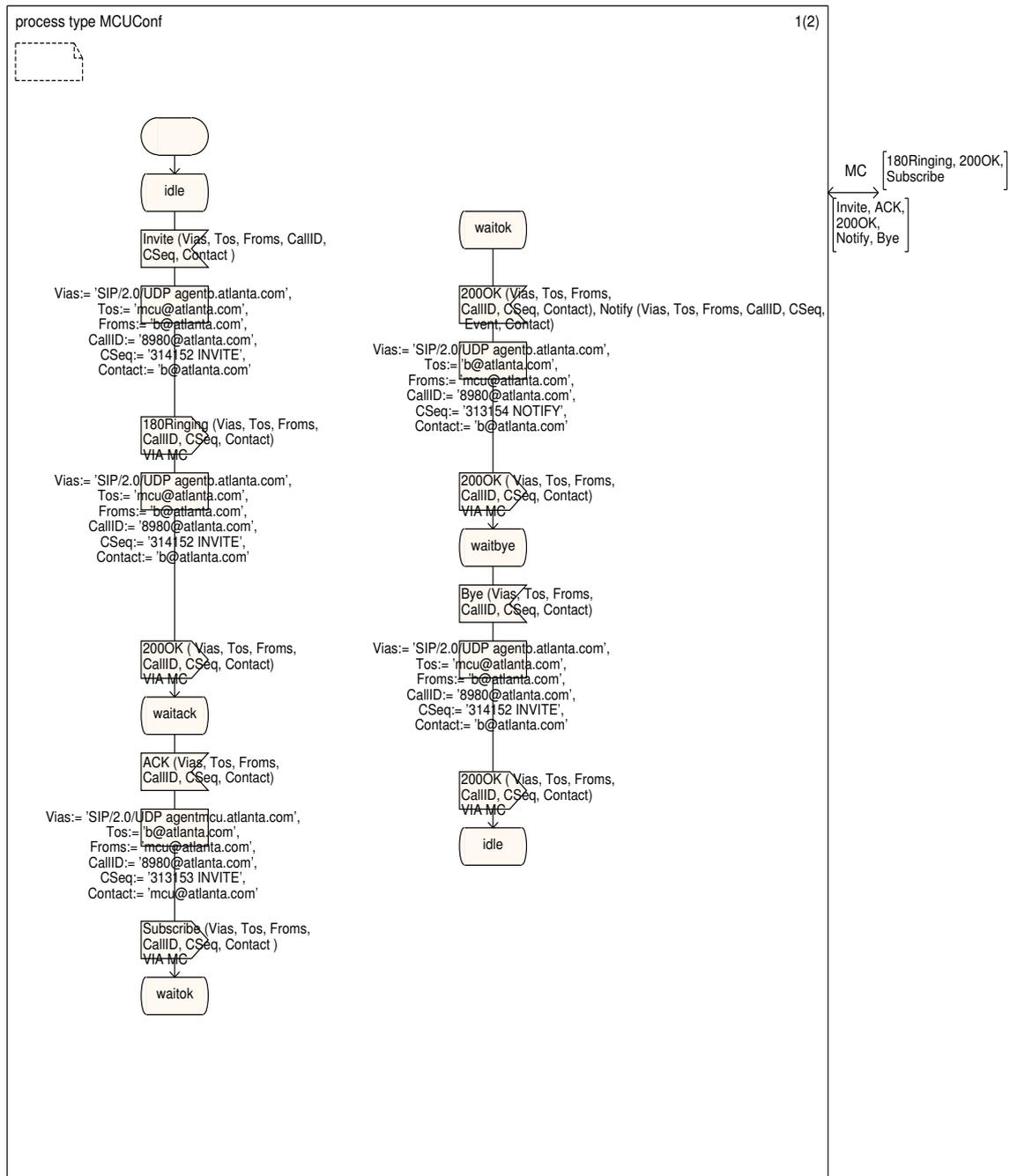


Figura 4.11: Processo MCUC com erro de especificação para o sistema SIP (ver figura 3.23)



terminal móvel MT se move para uma nova sub-rede localizada em um domínio diferente, tornando-se MT1 com um novo endereço IP *care-of-address*. Para manter a conexão inicial estabelecida sem interrupção, é necessário o processo de *roaming* e *handoff*. O terminal móvel MT1 envia o sinal de broadcast GRQ (*Gatekeeper Request*) e recebe GCF (*Gatekeeper Confirmation*) com o endereço do GK3 responsável pelo novo domínio. MT1 registra-se no GK3 e uma nova conexão com o terminal EP é estabelecida com a sua entrada na conferência *ad hoc* multiponto expandida. Momentaneamente coexistirá duas conexões com o terminal EP. Após realizado o *handoff*, a conexão antiga entre MT e EP é cancelada, mantendo apenas a conexão entre MT e MT1 sem perda de capacidades.

O registro no novo GK3 visitante requer a atualização do endereço de MT1 no seu *home* HLF e na nova base de dados temporária VLF2 visitante, além do GK2 e do VLF1 em que o terminal móvel MT1 pertencia. Essa atualização é necessária para não haver conflito no processo de localização do terminal móvel MT1. No HLF e no VLF2, o MT1 atualiza os dados com o seu novo endereço *care-of-address*. Para a base temporária de dados VLF1 e o GK2 onde estava registrado, é cancelado o registro. Ambos os processos de atualização e cancelamento do registro são realizados através dos sinais *DescriptorUpdate* para atualização e *DescriptorUpdateAck* para confirmação.

A figura 4.8 apresenta o MSC da simulação do estabelecimento de chamada com mobilidade inter-domínios e H.510 com erro de atualização da base temporária VLF1 e GK2 (figura 3.19). O terminal móvel MT1 ao mudar para a nova sub-rede precisa atualizar o seu status (registro) no novo GK3, na nova base temporária VLF2 e na base de dados permanente HLF em que o terminal pertence. GK2 e VLF1 devem ser atualizados com o cancelamento do endereço de MT1 em seus registros para não haver conflito de endereços durante o processo de localização do terminal móvel MT1. O erro de atualização no GK2 foi encontrado na simulação. A atualização no GK2 falhou e por isso não foi possível entrar na conferência para estabelecer a conexão com o terminal EP, o processo de *handoff* não foi finalizado e a ligação entre o terminal EP e o terminal móvel MT1 (MT na nova sub-rede) foi cancelada. A base de dados temporária VLF1 não possuía o endereço TSAP do GK2 para cancelar o registro de MT1. O erro foi corrigido com a informação deste endereço TSAP (endereço de transporte).

A figura 4.9 apresenta o MSC gerado pela simulação do estabelecimento de sessão entre terminais SIP móveis (figura 3.23). O MSC é composto pelos processos *environment*, *AC*, *BC*, *MCUC*, *AMC* que correspondem às entidades SIP. O terminal móvel AC envia um *invite* convidando BC para iniciar uma sessão. Nesta mensagem de *invite*, AC envia também para BC o endereço da MCU. BC convida (*invite*) a MCU formando uma conferência entre AC, BC e MCU. Em algum momento, AC move-se para uma nova sub-rede em um novo domínio e recebe um novo endereço temporário *care-of-address*. AC envia um *invite* para MCU, com o seu novo endereço *care-of-address* e o parâmetro *Replace* contendo o seu endereço antigo, e entra na conferência estabelecendo uma nova conexão

com a MCU e BC e realizando o *handoff*. Após o término do *handoff*, a conexão entre BC e AC é cancelada mantendo apenas a conexão entre BC e ACM e as capacidades inicialmente estabelecidas. A nova conexão entre BC e AMC possui o mesmo identificador de chamada (*Call-ID*) da conexão iniciada entre AC e BC, certificando o processo de *handoff*.

A figura 4.10 descreve o MSC com o erro de especificação durante a simulação do estabelecimento de sessão entre terminais móveis SIP, o sinal *Notify* não foi consumido. Na especificação do processo MCUC com erro (figura 4.11), os dois sinais *Notify* e *200OK* não podem ser declarados juntos no mesmo sinal de entrada. Este erro significa que o sinal *Notify* enviado pelo processo BC não pode ser recebido pelo processo MCUC, não existe nenhuma instância de processo escalada para a transição, após o consumo do sinal *200OK*. A figura 4.12 apresenta a especificação corrigida do processo MCUC, foi criado mais um estado *waitnotify* para consumir o sinal *Notify*.

# Capítulo 5

## Conclusões

Com o crescimento da Internet, o protocolo IP é uma tendência predominante para as redes de dados com o objetivo de fornecer inclusive serviços de telefonia para o estabelecimento de chamadas e comunicações de voz, substituindo a rede de telefonia convencional baseada em circuitos PSTN (*Public Switched Telephony Network*). Assim como a Internet, a característica de mobilidade tem sido bastante utilizada, visto o aumento da utilização de dispositivos móveis como laptops e PDA's (*Personal Digital Assistant*). Neste contexto, este trabalho visa integrar a telefonia IP com a mobilidade.

Os principais protocolos para sinalização e controle de telefonia IP que competem atualmente entre si são o H.323, padronizado pelo ITU-T, e o SIP, padronizado pelo IETF. Ambos os protocolos não possuem a característica de mobilidade, quanto a capacidade de manter uma conexão ativa durante mudanças de localidades, agregada às recomendações atuais.

A partir do estabelecimento de chamada entre os terminais fixos, desenvolveu-se neste trabalho propostas de mobilidade para os protocolos H.323 e SIP utilizando a linguagem de especificação formal SDL, baseadas nos trabalhos de Liao [1] para o protocolo H.323 e Chen, Moh e Berquin [2] para o protocolo SIP.

Para o protocolo H.323, foram desenvolvidas propostas em SDL de mobilidade intra-domínio e inter-domínio utilizando a expansão da conferência *ad hoc* multiponto para realizar o processo de *handoff*. Durante o estabelecimento de chamada entre terminais H.323 móveis, é necessário o suporte ao *roaming* e *handoff* para que o terminal móvel possa se mover entre sub-redes diferentes localizadas em um mesmo domínio (intra-domínio) ou em domínios distintos (inter-domínios) e manter a conexão inicialmente estabelecida ativa sem interrupção. A partir da proposta de mobilidade inter-domínios sugerida por Liao, foi desenvolvida neste trabalho uma nova proposta de mobilidade para o protocolo H.323 utilizando SDL [3]. A nova abordagem descreve um sistema com estabelecimento de chamada entre terminais móveis H.323 com mobilidade inter-domínios e a adição de novas entidades e funcionalidades do protocolo H.510, realizando *handoff* a partir da expansão da conferência *ad hoc* multiponto. As novas entidades HLF (*Home Location Function*)

e VLF (*Visitor Location Function*) são as bases de dados permanentes e temporárias responsáveis pelas novas funcionalidades para atualização de endereços e localização dos terminais móveis a cada mudança de sub-rede, através das sinalizações *DescriptorUpdate* e *DescriptorUpdateAck*. Estes conceitos de mobilidade do protocolo H.510 não são abordados no trabalho de Liao.

Para o protocolo SIP, foi desenvolvida uma proposta em SDL com mobilidade inter-domínios para terminais móveis baseado na conferência com presença de MCU para suporte a *roaming* e *handoff* e sinalizações SIP. A proposta deste sistema partiu da proposta de mobilidade inter-domínios no H.323 por utilizar a expansão da conferência *ad hoc* multiponto para realizar *handoff*.

As especificações de todos os sistemas propostos utilizaram a linguagem de especificação formal orientada a objeto SDL, padronizada pelo ITU-T, através da ferramenta SDL TAU Suite [37]. A partir destas especificações, os sistemas foram simulados e validados para garantir a corretude das propostas de acordo com as recomendações H.323 e SIP para futuras implementações. A validação permitiu a correção de erros de especificação e lógica e a simulação permitiu simular diferentes ambientes e casos críticos para verificar o comportamento funcional dos sistemas.

A partir das especificações e simulações com as entidades e sinalizações dos sistemas com mobilidade para os protocolos H.323 e SIP, é possível fazer uma análise de complexidade e comparação que envolve as duas recomendações para telefonia IP atualmente. Apesar da diferença de implementação das especificações, ambos os protocolos possuem funções equivalentes. O protocolo H.323 é uma recomendação mais complexa que envolve uma série de protocolos da série H, como o H.225, o H.245, o Q.931, necessários para as sinalizações e controle para estabelecer uma chamada. Mas apesar da complexidade, é um protocolo mais utilizado e difundido. O protocolo SIP é uma recomendação mais leve e única para sinalizações e controle para estabelecer uma sessão, mas ainda não é capaz de substituir o H.323. A tendência é que os protocolos H.323 e SIP se interoperem através de gateways capazes de reconhecer e traduzir respectivamente as sinalizações.

Baseado nas propostas de mobilidade desenvolvidas para os protocolos H.323 e SIP, pode-se propor a realização de alguns trabalhos futuros:

- Analisar a performance dos sistemas com mobilidade desenvolvidos para H.323 e SIP, inclusive a proposta de mobilidade para o H.323 com conceitos de H.510, através de simulações de um ambiente real para cada sistema proposto. Analisar o tempo gasto entre as sinalizações durante um estabelecimento de chamada entre terminais móveis, verificando se o processo de *roaming* e *handoff* é capaz de se realizar com um ambiente de conferência multiponto.
- Implementar um sistema com o estabelecimento de chamada entre um terminal H.323 e um terminal SIP com presença de gateways capazes de traduzir sinalizações

H.323 e SIP. Durante o estabelecimento de chamada, os terminais devem ser capazes de realizar *roaming* e *handoff* entre domínios diferentes a partir da conferência multiponto.

# Bibliografia

- [1] W. Liao, *Mobile Internet Telephony: Mobility Extension to H.323*, IEEE Transactions on Vehicular Technology, vol.50, n.6, 1403–1414, November 2001.
- [2] M. Moh, G. Berquin and Y. Chen, *Mobile IP Telephony: Mobility Support of SIP*, Eight International Conference on Computer Communication and Networks, 554–559, 1999.
- [3] S. Chung and W. Borelli *A New Proposal for Mobility Extension to H.323 using SDL*, 11th International Conference on Telecommunications, August 2004.
- [4] W. Liao, *VoIP Mobility in IP/Cellular Network Internetworking*, IEEE Communications Magazine, 70–75, April 2000.
- [5] G. A. Mills-Tettey and D. Kotz, *Mobile Voice over IP (MVOIP): An Application-level Protocol for Call Hand-off in Real Time Applications*, IEEE International Performance Computing and Communications Conference, 271–279, April 2002.
- [6] H. Liu and P. Mouchtaris, *Voice over IP Signalling: H.323 and Beyond*, IEEE Communications Magazine, 142–148, October 2000.
- [7] H. Schulzrinne and J. Rosenberg, *The Session Initiation Protocol: Internet-Centric Signaling*, IEEE Communications Magazine, 134–141, October 2000.
- [8] H. Schulzrinne and E. Wedlund, *Mobility Support using SIP*, Second ACM/IEEE International Conference on Wireless and Mobile Multimedia, August 1999.
- [9] H. Schulzrinne and E. Wedlund, *Application-Layer Mobility Using SIP*, Mobile Computing and Communications Review, vol.4, n.3, 29–36, July 2000.
- [10] H. Schulzrinne and J. Rosenberg, *Signaling for Internet Telephony*, Sixth International Conference on Network Protocols, 298–307, October 1998.
- [11] K. Chan and G. Bochmann, *Modeling IETF Session Initiation Protocol and its services in SDL*, SDL Forum 2003, 352–373, January 2003.

- 
- [12] H. Schulzrinne and J. Rosenberg, *A Comparison of SIP and H.323 for Internet Telephony*, Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV '98), July 1998, Cambridge, England.
- [13] *H.323 - Packet-based multimedia communications systems*, ITU-T Telecommunication Standardization Sector of ITU, November 2000.
- [14] *H.225.0 - Call signalling protocols and media stream packetization for packet-based multimedia communication systems*, ITU-T Telecommunication Standardization Sector of ITU, November 2000.
- [15] *H.235 - Security and encryption for H-series (H.323 and other H.245-based) multimedia terminals*, ITU-T Telecommunication Standardization Sector of ITU, August 2003.
- [16] *H.332 - H.323 extended for loosely coupled conferences*, ITU-T Telecommunication Standardization Sector of ITU, September 1998.
- [17] *H.245 - Control protocol for multimedia communication*, ITU-T Telecommunication Standardization Sector of ITU, July 2001.
- [18] *H.246 - Interworking of H-Series multimedia terminals with H-Series multimedia terminals and voice/voiceband terminals on GSTN and ISDN*, ITU-T Telecommunication Standardization Sector of ITU, February 1998.
- [19] *H.450.1 - Generic functional protocol for the support of supplementary services in H.323*, ITU-T Telecommunication Standardization Sector of ITU, February 1998.
- [20] *H.510 - Mobility for H.323 multimedia systems and services*, ITU-T Telecommunication Standardization Sector of ITU, March 2003.
- [21] *Q.931 - ISDN user-network interface layer 3 specification for basic call control*, ITU-T Telecommunication Standardization Sector of ITU, May 1998.
- [22] *Specification and description language (SDL)- Z.100*, ITU-T Telecommunication Standardization Sector of ITU, August 2002.
- [23] *Message Sequence Chart - Z.120*, ITU-T Telecommunication Standardization Sector of ITU, April 2004.
- [24] *RFC 3261 - SIP: Session Initiation Protocol*, J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley and E. Schooler, IETF - The Internet Engineering Task Force, June 2002.

- 
- [25] *RFC 2327 - SDP: Session Description Protocol*, M. Handley and V. Jacobson, IETF - The Internet Engineering Task Force, April 1998.
- [26] *RFC 1889 - RTP: A Transport Protocol for Real-Time Applications*, H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, IETF - The Internet Engineering Task Force, January 1996.
- [27] *RFC 2068 - HTTP 1.1: Hypertext Transfer Protocol*, R. Fielding, J. Gettys, J. Mogul, H. Frystyk, T. Berners-Lee, IETF - The Internet Engineering Task Force, January 1997.
- [28] *RFC 959 - FTP: File Transfer Protocol*, J. Postel, J. Reynolds, IETF - The Internet Engineering Task Force, October 1985.
- [29] *RFC 2821 - SMTP: Simple Mail Transfer Protocol*, J. Klensin, IETF - The Internet Engineering Task Force, April 2001.
- [30] *RFC 2396 - Uniform Resource Identifiers (URI): Generic Syntax*, JT. Berners-Lee, R. Fielding, U.C. Irvine, L. Masinter, IETF - The Internet Engineering Task Force, August 1998.
- [31] J. Rosenberg, H. Schulzrinne and P. Kyzivat, *Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)*, Dynamicsoft, Columbia University and Cisco Systems, June 2003.
- [32] J. Rosenberg, J. Peterson, H. Schulzrinne and G. Camarillo, *Best Current Practices for Third Party Call Control in the Session Initiation Protocol*, Dynamicsoft, Neustar, Columbia University and Ericsson Advanced Signalling Research Lab, June 2003.
- [33] J. Rosenberg, *A Framework for Conferencing with the Session Initiation Protocol*, Dynamicsoft, May 2003.
- [34] R. Sparks, *The SIP Refer Method*, Dynamicsoft, November 2002.
- [35] R. Mahy, R. Dean and B. Biggs, *The Session Initiation Protocol (SIP) "Replaces" Header*, Cisco Systems, Inc., June 2003.
- [36] A. Johnston and O. Levin, *Session Initiation Protocol Call Control - Conferencing for User Agents*, MCI and Radvision, June 2003.
- [37] Telelogic AB, Sweden. *Telelogic Tau 4.2 SDL Suite Getting Started*, September 2001.
- [38] Telelogic AB, Sweden. *Telelogic Tau 4.2 SDL Suite Methodology Guidelines*, September 2001.

- [39] J.M. Takahara *Especificação formal utilizando projeto orientado a objeto e SDL '92 como alternativa para a evolução de protocolos de interconexão de redes locais e ATM*, Tese de Mestrado/Departamento Telemática/UNICAMP, Campinas, Fevereiro - 1998.
- [40] N.A. Nassif *Especificação Formal em SDL e Simulação de Protocolos que combinam a Técnica de Label-Swapping com Roteamento de Camada 3 para Transporte do IP sobre o ATM*, Tese de Mestrado/Departamento Telemática/UNICAMP, Campinas, Março - 1999.
- [41] R.P. Guimarães *Uma metodologia para o desenvolvimento orientado a objetos de sistemas TINA utilizando a linguagem de especificação formal SDL com geração automática de código Java*, Tese de Mestrado/Departamento Telemática/UNICAMP, Campinas, Dezembro - 2001.
- [42] O. Hersent, D. Guide and J. Petit, *Telefonia IP - Comunicação Multimídia baseada em Pacotes*, Makron, Sao Paulo, Brazil, 2002.
- [43] C. Falbriad, *Protocolos e Aplicações para Redes de Computadores*, Ed. Erica, Brazil, 2002.
- [44] F. Belina, D. Hogrefe and A. Sarma, *SDL with Applications from Protocol Specification*, Prentice Hall, London, 1991.

# Apêndice A

## A linguagem SDL e a ferramenta TAU SDL Suite

A linguagem SDL (*Specification Description Language*)[22] foi publicada primeiramente como uma recomendação do CCITT (*Comité Consultatif International Téléphonique*), no começo dos anos 70. Atualmente o órgão CCITT é representado pelo ITU-T (*International Telecommunications Union - Telecommunication*).

O SDL é uma linguagem para a especificação e descrição de protocolos, sistemas de telecomunicações e sistemas de tempo real. O SDL se concentra basicamente na especificação dos aspectos comportamentais do sistema, envolvendo dados quando necessário. A linguagem é ao mesmo tempo textual e gráfica, ou seja, os sistemas especificados em SDL podem fazer uso de componentes gráficos da linguagem, os quais possuem equivalentes textuais.

O SDL possui algumas características que a diferencia de outras linguagens: é uma linguagem padrão, formal, orientada à objetos, portátil, escalável, um padrão aberto e permite reuso.

### A.1 Estrutura do sistema

Primeiramente, deve-se diferenciar os termos especificação, tipo e instância. Uma especificação define um sistema que é composto por diversas entidades que se comunicam entre si. Estas entidades possuem um comportamento definido que determina qual é o seu tipo. Cada um destes tipos pode assumir inúmeras instâncias, todas possuem o mesmo comportamento e as mesmas características, mas existem de forma independente umas das outras, ou seja, o tipo pode ser visto como uma descrição das instâncias que serão criadas para que, ao se relacionar, executem tarefas do sistema.

Para a especificação dos tipos, o SDL oferece algumas estruturas básicas. Dentre elas pode-se destacar:

**System:** É a entidade mais externa, que representa o sistema como um todo. É composto por diversos blocos que se comunicam entre si e com o ambiente através de canais de sinais. Os sinais e tipos de dados utilizados pelo sistema e que devem ser conhecidos por diversos blocos devem ser declarados neste nível.

**Block:** É a entidade que agrupa um ou mais processos que se comunicam entre si e com os canais de comunicação do sistema (podendo se comunicar com outros blocos ou com o ambiente). Os sinais e tipos de dados utilizados pelo bloco e que ainda não foram declarados no nível do sistemas devem ser declarados neste nível.

**Process:** É a estrutura responsável pela especificação comportamental do sistema, através da descrição de uma máquina de estados finita com dados, que é independente das demais. O conjunto de processos que são executados em paralelo é que definem o comportamento do sistema.

## A.2 Troca de sinais

Os sinais são transportados de um bloco para outro ou para o ambiente externo (*environment*) através de canais de comunicação. Cada um destes canais pode ser uni e bidirecional e possui listas entre processos, faz-se uso de rotas de sinais, os quais possuem uma estrutura similar aos canais de comunicação. Os processos também fazem uso de rotas de sinais para se comunicar com o nível mais externo, conectando uma das extremidades da rota de sinais à um canal de comunicação do nível de sistema. Neste caso, os sinais transportados pelas rotas de sinais devem ser os mesmos transportados pelo canal de comunicação.

## A.3 Principais elementos para a descrição de processos

As figuras A.1 e A.2 apresentam alguns dos principais elementos utilizados na especificação dos processos que compõem uma especificação em SDL.

## A.4 Herança e Especialização de tipos

No processo de desenvolvimento de um sistema é muito comum a especialização de um *tipo* (*type*) pré-existente, incluindo novas funcionalidades ou redefinindo antigas. Para isso, o SDL fornece a facilidade do reuso de especificações através do conceito de herança de *tipos*, onde todas as propriedades do *supertipo* são herdadas pelo *subtipo*, devendo ser especificadas apenas as diferenças existentes entre os mesmos.

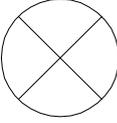
| Representação   | Descrição   |
|---|---|
|    | <i>Start</i> : utilizado apenas uma vez no processo indicando o início de sua execução  |
|    | <i>Stop</i> : utilizado para indicar o término da sua execução de um processo   |
|    | <i>State</i> : utilizado para indicar o estado em que se encontra a máquina finita que representa o processo/procedimento. Uma mudança de estado só ocorre quando se recebe um sinal. O estado pode ter um nome, ou pode ser declarado o estado <i>**</i> que se refere a todos os estados existentes, indicando que as ações a serem tomadas são comuns a todos eles. Além disso, após a recepção de um sinal (e execução das ações relacionadas), a execução pode encontrar o estado <i>^</i> que indica que o estado não se alterou devido às ações. |
|    | <i>Decision</i> : utilizado para uma tomada de decisão. Consiste de uma questão e diversas possíveis respostas e, dependendo da resposta, a execução toma um caminho distinto.  |
|    | <i>Text</i> : utilizado para a declaração de variáveis, sinal e tipos de dados.   |
|   | <i>Input</i> : utilizado para indicar um sinal de entrada. Deve vir sempre após um estado indicando uma transição devido ao recebimento deste sinal.  |
|  | <i>Output</i> : utilizado para indicar um sinal de saída. Através desta estrutura envia-se sinais e dados para outros processos. Para cada output deve haver um input associado no processo de destino. Para indicar o processo destino pode-se utilizar o endereçamento implícito - omitindo-se o endereço do destino - ou explícito - indicando-se o endereço destino através da opção TO <dest> ou indicando a rota do sinal a seguir através da opção VIA <rota>.   |
|  | <i>Task</i> : utilizado para definir uma tarefa ou um conjunto de tarefas (separados por `;`). As tarefas podem ser substituídas por um string com descrição textual das mesmas.  |
|  | <i>Return</i> : utilizado para indicar o término da execução de um procedimento. Indica um valor de retorno do procedimento, quando for o caso.   |

Figura A.1: Descrição e representação de algumas estruturas SDL

A herança de *tipos* é feita através da construção *INHERITS*. Caso a herança seja apenas para adicionar novas funcionalidades (sem a redefinição de antigas), deve-se utilizar a construção *INHERITS jsupertipo; ADDING*. Entretanto, se além da inclusão de novas propriedades, outras existentes forem modificadas, utiliza-se a construção *INHERITS jsupertipo; e* define-se a propriedade a ser modificada como *VIRTUAL* no *supertipo*, redefinindo-a com a construção *REDEFINED* no *subtipo*.

Para que se possa utilizar este conceito de herança, alguns *tipos* devem ser utilizados:

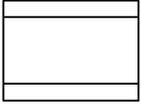
| Representação   | Descrição   |
|---|---|
|  | <i>Procedure Call</i> : utilizado para indicar uma chamada a um procedimento.   |
|  | <i>Create</i> : utilizado para criar uma instância de um processo, desde que esteja no mesmo bloco do processo criador. |
|  | <i>Comment</i> : utilizado para a inserção de comentários na especificação.   |

Figura A.2: Descrição e representação de algumas estruturas SDL

**System Type:** Define um sistema a ser reusado por outro através do mecanismo de *Package*.

**Block Type:** Define um conjunto de blocos com as mesmas características (ou seja, um tipo de bloco). Cada *Block Type* pode ser instanciado inúmeras vezes.

**Process Type:** Define um conjunto de processos com as mesmas características (ou seja, um tipo de processo). Cada *Process Type* pode ser instanciado inúmeras vezes e cada instância se comporta como uma máquina finita de estados que trabalha de forma independente e concorrente com as demais.

Sendo assim, podem ser especializados apenas os *tipos System Type, Block Type e Process Type*, sendo que o *tipo System Type* apenas pode ser reutilizado por outro sistema, não podendo ser redefinido. Para a especialização de processos, novas funcionalidades podem ser adicionadas através da inclusão de novos estados e novas transições e estados pré-existentes. Além disso, pode-se redefinir transições de *Input* e *Start*, declarando-as como *VIRTUAL* no *supertipo* e *REDEFINED* no *subtipo*. Este *subtipo* pode ainda dar origem a um outro *tipo* que herde todas as suas características, e por consequência todas as características de seu ancestral. Neste caso, para este novo *tipo*, todas as propriedades *REDEFINED* podem ser novamente redefinidas. Para que isso seja evitado, ao invés de definir as propriedades como *REDEFINED* pode-se utilizar *FINALIZED*, proibindo os *tipos* herdeiros deste *subtipo* de redefinir estas propriedades.

## A.5 Package

O *Package* permite a criação de uma biblioteca que pode ser utilizada em um sistema. No *package* podem ser definidos tipos de dados, listas de sinais, *System Types, Block Types, Process Types*, dentre outros, podendo ser utilizado integralmente ou parcialmente

pelo sistema que o inclua através da construção *USE*.

Além de poder ser incluído em um sistema, um *package* pode ser incluído em outro *package*, criando assim uma hierarquia de *packages*.

## A.6 *SDL TAU Suite*

O *SDL TAU Suite* [37, 38] é uma ferramenta *CASE* (*Computer Aided Design Software Engineering*), distribuída pela Telelogic, que utiliza a linguagem SDL para a especificação, validação e simulação dos sistemas. Dentre os componentes da versão utilizada, o *SDL TAU Suite 4.2*, pode-se destacar os seguintes:

**Organizador:** utilizado para a organização geral das especificações. Fornece uma visão de todos os diagramas e documentos que compõem o sistema, os quais podem ser livremente agrupados em capítulos e módulos de forma a facilitar a organização.

**Editor SDL:** possibilita a edição comportamental do sistema através do uso da representação gráfica da linguagem SDL.

**Visualizador de Tipos:** permite a visualização do impacto dos mecanismos de herança e especialização no sistema.

**Analisador:** realiza a análise sintática e semântica da especificação desenvolvida verificando a existência de erros e gerando código para os processos de simulação e validação.

**Simulador:** permite a simulação das características do sistema, tendo como saída um diagrama de troca de mensagens (*MSC - Message Sequence Chart*) que mostra o comportamento do sistema ao longo do tempo.

**Validador:** permite a validação do sistema, ou seja, percorre todos os possíveis estados em que o sistema pode se encontrar auxiliando assim na detecção de erros, como por exemplo o não tratamento de algum sinal em determinados estados.

**Editor MSC:** utilizado pelo simulador para a geração dos diagramas de troca de mensagens.

**Visualizador de área coberta:** utilizado pelo simulador e pelo validador para exibir uma árvore com todos os estados do sistema destacando aqueles alcançados pelo processo em questão (simulação ou validação).

**Geradores de código:** possibilitam a geração de código C, Cmicro e CHILL a partir das especificações em SDL.

A presença dos componentes Simulador e Validador são importantes para a realização de testes da especificação, garantindo assim a corretude do sistema. Através do uso destes componentes pode-se encontrar erros na especificação através da simulação de casos críticos, bem como através da análise dos estados alcançados pelo Validador com o uso do visualizador de área coberta (*Coverage Viewer*).

Além disso, o *SDL TAU Suite* permite a integração do desenvolvimento em SDL com outras linguagens como o ASN.1 (*Abstract Syntax Notation One*) e UML (*Unifying*

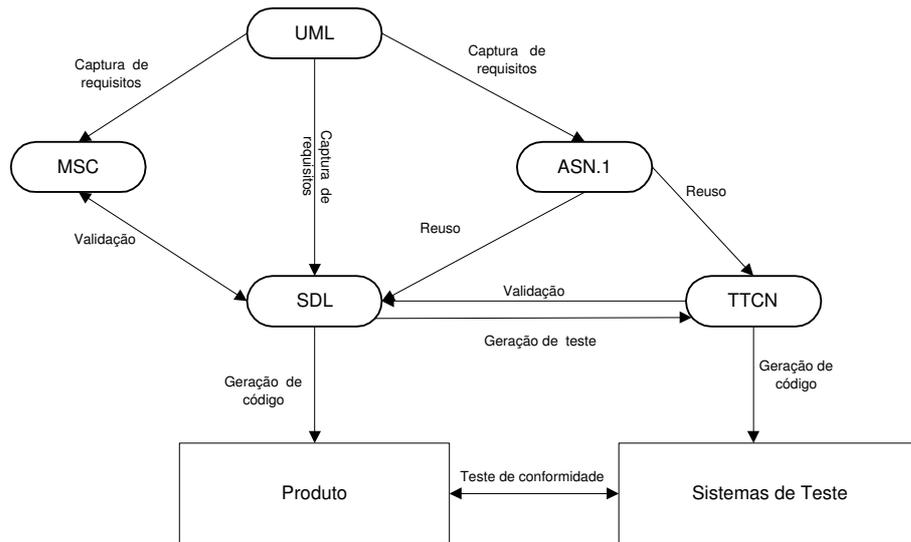


Figura A.3: Relação entre linguagens que fazem interface com o SDL TAU Suite

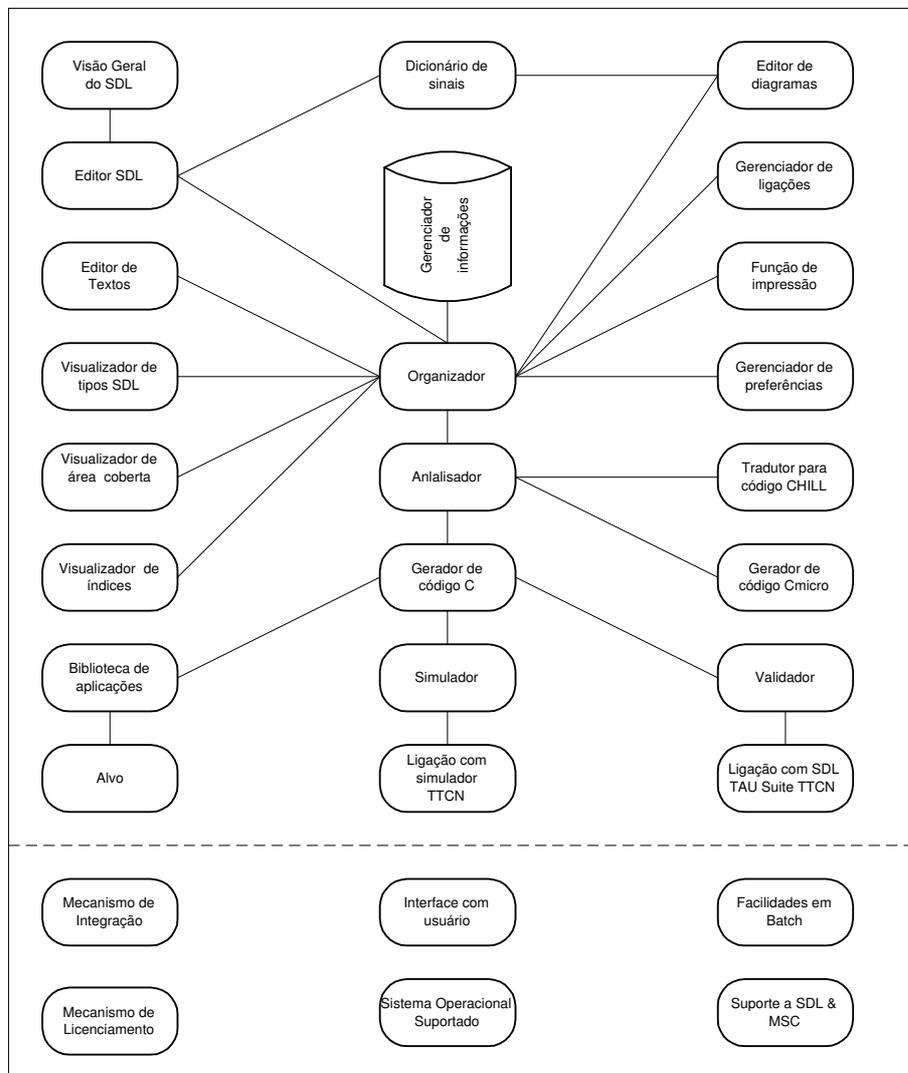


Figura A.4: Visão geral dos módulos que compõem o SDL TAU Suite

*Modelling Language*), garantindo, assim, uma maior flexibilidade no processo de desenvolvimento. Existe também a possibilidade da realização de testes de casos de uso através do uso da linguagem TTCN (*Tree and Tabular Combined Notation*). A relação entre estas linguagens é apresentada na figura A.3.

Uma visão geral dos módulos componentes da ferramenta e suas relações pode ser visto na figura A.4.

O Organizador é responsável por gerenciar quase todos os componentes do pacote *SDL TAU Suite*. Através dele tem-se o acesso a todas as funcionalidades disponíveis. Tanto para a realização da simulação quanto da validação, as especificações passam primeiramente por uma análise (através do Analisador) que garante a corretude sintática das especificações. Após esta análise, é gerado o código C, utilizado pelos módulos responsáveis pela realização das simulações e da validação. Ainda é disponibilizada uma biblioteca de aplicações que, quando integrada ao gerador de código C, permite a geração de uma implementação em C do sistema especificado e a execução do mesmo.