

**APLICAÇÕES UTILIZANDO SISTEMAS DE  
LOCALIZAÇÃO EM GSM**

**TULIO JOSE LIMA CUSSIOLI  
WILLIAM RIBEIRO FILHO**

**DISSERTAÇÃO DO PROJETO FINAL DE GRADUAÇÃO  
EM ENGENHARIA ELÉTRICA**

**FACULDADE DE TECNOLOGIA  
UNIVERSIDADE DE BRASÍLIA**

**UNIVERSIDADE DE BRASÍLIA  
FACULDADE DE TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**APLICAÇÕES UTILIZANDO SISTEMAS DE  
LOCALIZAÇÃO EM GSM**

**TULIO JOSE LIMA CUSSIOLI  
WILLIAM RIBEIRO FILHO**

**ORIENTADOR:  
LEONARDO R. A. X. DE MENEZES**

**DISSERTAÇÃO DO PROJETO FINAL DE  
GRADUAÇÃO EM ENGENHARIA ELÉTRICA**

**BRASÍLIA / DF JULHO / 2003**



---

## Agradecimentos

---

*Longos foram os caminhos para chegar até aqui. Muitas decisões foram tomadas, outras, não menos importantes, deixadas para trás. Nesta, caminhada, agradeço, acima de tudo à Deus e a minha Família, pelo apoio dado durante todo este tempo. Agradeço também, em especial, ao professor Leonardo, pela disposição e dedicação com que tem levado o projeto, a toda a galera do Projeto Nokia pela força, e aos amigos.*

“ O futuro não é um lugar para onde estamos indo, mas um lugar que estamos criando. O caminho para ela não é encontrado, mas construído; e o ato de fazê-lo, muda tanto o realizar quanto o destino.” ( John Schaar)

Tulio



---

## Abstract

---

The localization systems already are reality in the whole world. These use of diverse techniques to locate a device or portable mobile unit. All the techniques try, of some form, to use the characteristics of the radio waves that if propagem for the space, received or generated for the device that if it desires to locate. Enclosed in this context, system WV has the purpose to implement the services of communications between Client and Server, through the functionalities of a software of common Chat. The main idea is to make with that the user (Mobile Customer) has the possibility to use the based services of communication in groups, lists and exchange of messages. The beginning of everything gives when to the user if loga in the system. After this, it has the option to invite other users for a Chat, to catch a presence list, or either, to see if somebody of its group is on line at the moment and thus to inicializar the possible transactions. This project encloses all the base of the theory of the specification Wireless Village, and also deals with the tools of development used and the forms as they had incased themselves in the context of the project.

---

## Resumo

---

Os sistemas de localização já são realidade em todo o mundo. Estes utilizam de técnicas diversas para localizar um dispositivo ou unidade portátil móvel. Todas as técnicas tentam, de alguma forma, utilizar as características das ondas de rádio que se propagam pelo espaço, recebidas ou geradas pelo dispositivo que se deseja localizar.

Incluído nesse contexto, o sistema WV tem a finalidade de implementar os serviços de comunicações entre CLIENTE e SERVIDOR, através das funcionalidades de um software de Chat comum.

A idéia principal é fazer com que o usuário (Cliente Móvel) tenha a possibilidade de utilizar os serviços de comunicação baseados em grupos, listas e troca de mensagens.

O início de tudo se dá quando o usuário se loga no sistema. Após isso, ele tem a opção de convidar outros usuários para um Chat, pegar uma lista de presença, ou seja, ver se alguém do seu grupo está on line no momento e assim inicializar as transações possíveis.

Este projeto abrange toda a base da teoria da especificação Wireless Village, e trata também das ferramentas de desenvolvimento utilizadas e as formas como elas se encaixaram no contexto do projeto.



---

## Índice

---

1 - Introdução.....	12
2 - Sistema móvel celular.....	12
3 – Métodos de localização.....	22
4 - Sistemas Celulares .....	34
5. Wireless Village.....	51
6. Introdução ao Protocolo SSP.....	62
7. Transações.....	77
8. Transmissão do Protocolo SSP.....	81
9. Protocolo CLIENTE-SERVIDOR – Seções e Transações.....	82
10. Características.....	89
11. Características do Presence.....	93
12. Características do Instant Messaging.....	95
13. Transporte CSP através do Client-Server.....	97
14. O Transporte WV para clientes móveis.....	100
15. CLP – Command Line Protocol.....	101
16. Implementação do LoginRequest e do GetContactList.....	105
17 - Conclusão.....	122
Referências Bibliográficas.....	124
APENDICE A – Classes Java .....	126
APENDICE B – MySQL.....	142



---

## Índice de Tabelas

---

Tabela 1: Capacidade de tráfego e interferência co-canal.....	14
Tabela 2 – Transação desconhecida.....	67
Tabela 3 - Informações dos elementos do PollingRequest Primitive.....	85
Tabela 4 - Elementos do Logout Request.....	87
Tabela 5 - Elementos do Disconnect.....	87
Tabela 6 - Elementos do Keep Alive.....	88
Tabela 7 - Elementos do Get Service Provider Info.....	88
Tabela 8 - Elementos de pesquisa por usuários.....	91
Tabela 9 - Elementos de pesquisa por Grupos.....	91
Tabela 10 - Elementos de Informação de uma SearchReques.....	91
Tabela 11 - Elementos de informação de uma SearchResponse.....	92
Tabela 12 - Direções das classes primitivas do gerenciamento da Contact List.....	94
Tabela 13 - Elementos do GetListRequest.....	94
Tabela 14 - Elementos do GetListResponse.....	94
Tabela 15 Elementos do GetAttributeList.....	95



## Índice de Figuras

Figura 1 – Descrição do Sistema celular.....	13
Figura 2: Conceito de distância média de reuso.....	15
Figura 3: Componentes básicos de um sistema celular.....	16
Figura 4 – Unidade Móvel em Deslocamento.....	19
Figura 5– Time Difference of Arrival.....	24
Figura 6- Satélites em orbita.....	27
Figura 7 - Modelo de predição ponto a ponto.....	31
Figura 8 - Espalhamento do sinal sobre superfície irregular.....	32
Figura 9. TDMA: Frequência X Tempo.....	36
Figura 10 - Bandas separadas para o uplink e o downlink.....	36
Figura 11 – Arquitetura do sistema TDMA/IS-136.....	37
Figura 12 - Arquitetura do sistema CDMA.....	40
Figura 13 - Codificação da voz no GSM.....	42
Figura 14 - Arquitetura do sistema GSM.....	43
Figura 15 - Arquitetura do sistema GSM/GPRS.....	47
Figura 16 – Configurações diferentes da utilização do GPRS para o acesso à Internet.....	48
Figura 17 - Caminhos para evolução de uma rede GSM.....	49
Figura 18 - Modelo de Arquitetura do Sistema Wireless Village.....	51
Figura 19 - Elementos Funcionais do Servidor Wireless Village.....	52
Figura 20 - User Profiles.....	54
Figura 21 - Interfaces E Protocolos.....	55
Figura 22 - Modelo de mínima interoperabilidade do SSP.....	58
Figura 23 - Modelo de Máxima Interoperabilidade do SSP.....	59
Figura 24 – exemplo de comunicação no mesmo home domain.....	60
Figura 25 – Divisão do SSP.....	61
Figura 26 – Transação de login .....	70
Figura 27 – Transação de Logout.....	71
Figura 28 – Arvore representativa dos serviços SSP.....	72
Figura 29 – Transação GetAvailableService .....	77
Figura 30 – Transação ServiceIndication.....	77



Figura 31 – Transação SetServiceAgreement.....	78
Figura 32 – Transação GetUserProfileRequest.....	78
Figura 33 – Transação UpdateUserProfile.....	78
Figura 34 – Transação SearchRequest.....	79
Figura 35 – Transação StopSearch .....	79
Figura 36 – Transação de Convite .....	80
Figura 37 - Logging in.....	85
Figura 38 - Logging Out.....	86
Figura 39 – Transação Keep Alive.....	87
Figura 40 - Transação do Get Service Provider Info.....	88
Figura 41 - Transação de Busca/Pesquisa.....	90
Figura 42 – Invite Transaction.....	92
Figura 43 - Transação de GetListRequest .....	93
Figura 44 – Transação de envio de mensagem.....	96
Figura 45 – Divisão do Transporte no WV.....	98
Figura 46 – Comunicação no WV.....	99
Figura 47 – Relações entre Cliente, SMSC e SAP.....	100
Figura 48 – Emulador Wireless Toolkit.....	109
Figura 49 – Instalação do emulador.....	111
Figura 50 – instalação do emulador.....	111
Figura 51– Criação de um projeto no wireless toolkit.....	112
Figura 52– listagem dos projetos existentes.....	113
Figura 53 – Criando um projeto.....	113
Figura 54 – Construção de um aplicativo.....	114
Figura 55 – Executando um aplicativo no emulador.....	115
Figura 56 - Emulador.....	116
Figura 57 – Classes do Presence.....	118
Figura 58 – LoginRequest.....	119
Figura 59 - GerContactList.....	120
Figura 60 - Classes do Cliente.....	122
Figura 61 - SAP (Diagrama de Classes).....	122





---

## Índice de Equações

---

Equação 1.....	14
Equação 2.....	21
Equação 3.....	21
Equação 4.....	25
Equação 5.....	30
Equação 6.....	30



## Símbolos, Nomenclaturas e Abreviações

<b>3GPP</b>	<i>3rd Generation Partnership Project</i>	Organização para o Projeto da 3ª Geração
<b>AAA</b>	<i>Authentication, Authorization and Accounting</i>	Autenticação, Autorização e Contabilidade
<b>CLI</b>	<i>Command Line Interface</i>	Interface de linha de comando
<b>CLP</b>	<i>Command Line Protocol</i>	Protocolo de linha de comando
<b>CDMA</b>	<i>Code Division Multiple Access</i>	Acesso Múltiplo por Divisão de Código
<b>CSD</b>	<i>Circuit-Switched Data</i>	Dados de circuito chaveado
<b>CSP</b>	<i>Cliente Server Protocol</i>	Protocolo Cliente-Servidor
<b>DTD</b>	<i>Document Type Definition</i>	Definição de tipo de documento
<b>DHCP</b>	<i>Dynamic Host Configuration Protocol</i>	Protocolo de configuração do terminal dinâmico
<b>EDGE</b>	<i>Enhanced Data Rates for Global Evolution</i>	Aperfeiçoamento de Taxa de Dados numa Evolução Global
<b>ETSI</b>	<i>European Telecommunication Standards Institute</i>	Instituto de Padrões para Telecomunicações Europeu
<b>FA</b>	<i>Foreign Agent</i>	Agente Estrangeiro
<b>GPRS</b>	<i>General Packet Data Radio</i>	Serviço Geral de Rádio de Pacotes
<b>GSM</b>	<i>Global System for Mobile Communications</i>	Sistema Global para Comunicações Móveis
<b>HÁ</b>	<i>Home Agent</i>	Agente Local
<b>HTTP</b>	<i>Hypertext Transfer Protocol</i>	Protocolo para transferência de hipertexto
<b>IEEE</b>	<i>Institute of Electrical and Electronics Engineering</i>	Instituto de Engenharia Elétrica e Eletrônica
<b>IETF</b>	<i>Internet Engineering Task Force</i>	Força Tarefa de Engenharia para Internet
<b>IMPS</b>	<i>Instant Messaging and Presence Services</i>	Serviço de presence e mensagem instantânea
<b>IP</b>	<i>Internet Protocol</i>	Protocolo da Internet
<b>IPv4</b>	<i>Internet Protocol version 4</i>	Protocolo da Internet Versão 4
<b>IPv6</b>	<i>Internet Protocol version 6</i>	Protocolo da Internet Versão 6
<b>ISP</b>	<i>Internet Service Provider</i>	Provedor de Serviços de Internet
<b>ISSO</b>	<i>International Standards Organizations</i>	Organização Internacional de Padrões
<b>LAN</b>	<i>Local Area Network</i>	Rede de Área Local ou Rede Local
<b>MCN</b>	<i>MóBILE Core Network</i>	Núcleo de rede móvel
<b>MIP</b>	<i>Mobile IP</i>	IP Móvel
<b>OSI</b>	<i>Open System Interconnection</i>	Interconexão de Sistemas Abertos
<b>PCS</b>	<i>Personal Communication Service</i>	Serviço Pessoal de Comunicação de Dados
<b>RF</b>		Radio Frequência
<b>RTPC</b>		Rede de Telefonia Pública Comutada
<b>SAP</b>	<i>Service Access Point</i>	Service Access Point
<b>SIM</b>	<i>Subscriber Identity Module</i>	Módulo de Identificação do Assinante
<b>SMCNP</b>	<i>Server to MóBILE Core Network Protocol</i>	Serviço de Protocolo de núcleo de rede móvel
<b>SMS</b>	<i>Short Message Service</i>	Serviço de Mensagens Curtas



---

<b>SSP</b>	<i>Server Server Protocol</i>	Protocolo Servidor Servidor
<b>SMC</b>	<i>Serviço Móvel Celular</i>	Serviço Móvel Celular
<b>TCP</b>	<i>Transmission Control Protocol</i>	Protocolo de Controle de Transmissão
<b>TDMA</b>	<i>Time Division Multiple Access</i>	Acesso Múltiplo por Divisão de Código
<b>UMTS</b>	<i>Universal</i>	Sistema de Telecomunicações Móveis
	<i>Telecommunications System</i>	Universal
<b>VPN</b>	<i>Virtual Private Network</i>	Rede Privada Virtual
<b>WV</b>	<i>Wireless Village</i>	Wireless Village
<b>WAP</b>	<i>Wireless Application Protocol</i>	Wireless Application Protocol
<b>WAN</b>	<i>Wide Area Network</i>	Rede de Área Dispersa
<b>WLAN</b>	<i>Wireless Local Area Network</i>	Rede Local sem Fio
<b>XML</b>	<i>Extensible Markup Language</i>	Linguagem de marcação de texto



## 1 - Introdução

Um sistema móvel é definido como uma rede de comunicações por rádio que permite mobilidade contínua por meio de muitas células. A comunicação sem fio, por outro lado, implica em comunicação por rádio sem necessariamente requerer a passagem (*handover*) de uma célula a outra durante a conversação .

O oferecimento de comunicações móveis por rádio à população tem se tomado uma área de intensa pesquisa e desenvolvimento acelerado nos últimos anos. Isto deve ser combinado com o interesse crescente em evitar os custos elevados associados à instalação e relocação em ambientes interligados por fios.

Diversos sistemas foram propostos e alguns já estão em operação para manusear o controle e tráfego de informação em sistemas móveis. Dentre outros, é possível citar: acesso múltiplo por divisão em código (*code division multiple access* - CDMA), acesso múltiplo por divisão em tempo (*time division multiple access* TDMA) e acesso múltiplo por divisão em frequência (*frequency division multiple access* - FDMA).

Um sistema de telefonia móvel convencional seleciona um ou mais canais de RF para utilização em áreas geográficas específicas. A área de cobertura é planejada para ser a mais ampla possível, o que implica em uma potência de transmissão elevada.

No sistema de telefonia móvel celular, a área de cobertura é dividida em regiões chamadas células, de modo que a potência transmitida seja baixa e as frequências disponíveis venham a ser reutilizadas.

## 2 - Sistema móvel celular



## 2.1 Descrição do Sistema Celular

O projeto de um sistema celular consiste na divisão da área a ser coberta por telefonia móvel em áreas menores, permitindo a utilização de transmissores de baixa potência e um emprego eficiente do espectro por meio do reuso de frequência.

Nesse contexto, diz-se que os sistemas celulares são limitados pela interferência, em contraposição aos sistemas móveis convencionais (potência elevada) que são limitados pelo ruído térmico.

### 2.1.1 Estrutura Celular

A princípio, uma determinada região ou área de cobertura geográfica a ser atendida pelo serviço móvel celular é dividida em sub-regiões, que são chamadas *células*.

Desse modo, célula é a área geográfica iluminada por uma estação rádio dentro da qual a recepção do sinal atende às especificações do sistema. Em sua concepção básica, o sistema celular divide essa área em células hexagonais de igual tamanho, de tal forma que as estações radio-base (ERB) estejam localizadas no centro destas.

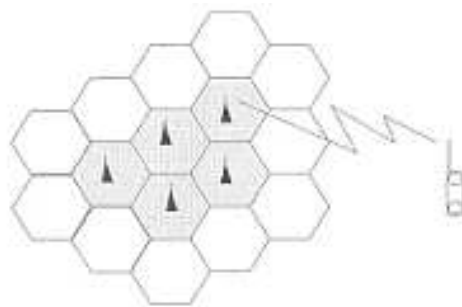


Figura 1 – Descrição do Sistema celular

- **Células Omnidirecionais** - Nas células omnidirecionais, a ERB é equipada com uma antena omnidirecional que transmite a mesma potência em todas as direções, formando assim uma área de cobertura circular, cujo centro é a própria ERB. Para efeito gráfico, essa célula é representada por um hexágono;
- **Células Setorizadas** - Nas células setorizadas, a ERB é equipada com antenas diretivas, de



tal forma que cada uma cubra determinada área.

## 2.2 Reutilização de Frequência

O reuso de frequência corresponde à utilização de uma mesma frequência em dois conjuntos distintos. A distância de reuso é limitada pela máxima interferência co-canal permitida no sistema.

No caso de um sistema homogêneo, formado por conjuntos de  $N = i^2 + ij + j^2$  células (em que  $i$  e  $j$  são inteiros positivos), a distância de reuso  $D$  é dada por

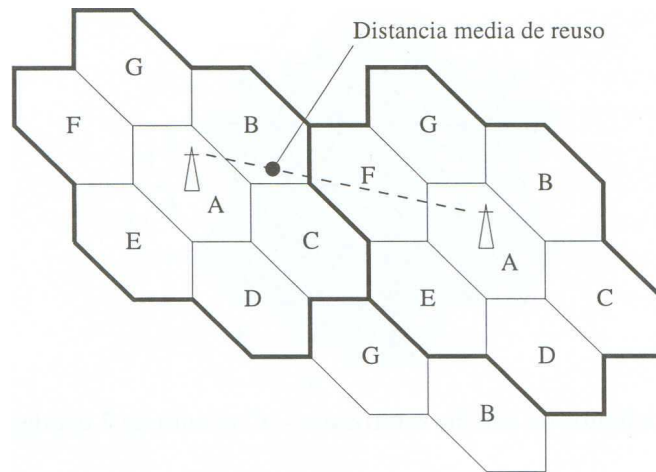
$$\frac{D}{R} = \sqrt{3N} \quad \text{em que } R \text{ é o raio da célula.}$$

### Equação 1

Evidentemente, a qualidade de transmissão cai à medida que os *clusters* ficam menores, mas a capacidade de tráfego aumenta em virtude da possibilidade de distribuir toda a canalização em poucas células. A Tabela abaixo mostra a relação existente entre a capacidade de tráfego e a interferência co-canal, para diversas dimensões do conjunto de células.

Dimensão	D/R	Canais/célula	Capacidade de tráfego	Qualidade de transmissão
1	1,73	360	Mais alta	Mais baixa
3	3,00	120	↑	
4	3,46	90		
7	4,58	51		↓
12	6,00	30	Mais baixa	Mais alta

Tabela 1: Capacidade de tráfego e interferência co-canal.



**Figura 2: Conceito de distância média de reuso.**

Os conjuntos de frequências devem ser alocados às células vizinhas, visto que para uma total cobertura de rádio as células apresentam superposição. A interferência co-canal é produzida pela utilização de frequências idênticas em células adjacentes ou próximas.

Esta interferência leva a utilizar uma distância substancial entre duas células de mesma frequência. Em virtude da técnica de reutilização, pode-se concluir que o número máximo de canais de voz passa a ser um múltiplo de 333 (416 no AMPS estendido) ou, mais precisamente, um múltiplo de 312 (395 no AMPS estendido).

### **2.3 Divisão de Células**

Quando a demanda de tráfego cresce dentro de uma determinada célula, existem duas soluções possíveis: adição de novas células ou setorização das células já existentes.

. **Adição de Novas Células** - A potência dos transmissores das células existentes é diminuída para cobrir aproximadamente metade da área original e novas células são adicionadas para cobrir as remanescentes;

. **Setorização de Células** - O conjunto de antenas omnidirecionais da célula é substituído por antenas direcionais (de  $60^\circ$  ou  $120^\circ$ ) e a célula é dividida em setores.

A dimensão da célula deve adequar-se à densidade de tráfego telefônico. Quanto maior for o tráfego, menor será a célula correspondente, uma vez que o número de canais disponíveis



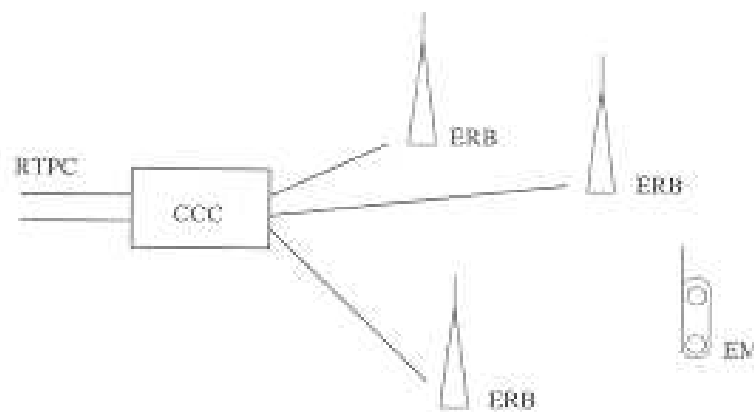
por célula é limitado. Isto implica, por exemplo, que em áreas centrais de uma cidade as células são menores do que aquelas das áreas suburbanas.

A adição de novas células é a maneira mais flexível de expandir o sistema, mas também a mais dispendiosa. Ela envolve a instalação de novas estações radio-base, torres e antenas. A aquisição de terrenos para a instalação das ERBs em áreas centrais ficou proibitiva para as operadoras, que preferem alugar pequenas áreas no topo de prédios, em terrenos ou em áreas comerciais. A setorização de células tem um custo menor para a operadora, entretanto, impõe restrições às expansões futuras.

## 2.4 Constituição do Sistema Celular

Um sistema celular típico é constituído de três elementos, além das conexões entre esses elementos. Os componentes básicos do sistema celular são:

- . Centro de Comutação e Controle (CCC);
- . Estação Radio-base (ERB);
- . Terminal Móvel (TM).



**Figura 3: Componentes básicos de um sistema celular**

A estação ou terminal móvel (EM ou TM) contém uma unidade de controle, um transceptor e uma antena. Transmite e recebe sinais de voz, possibilitando a conversação. Transmite e recebe sinais de controle, permitindo o estabelecimento da chamada. Os terminais móveis são produzidos por um grande número de fabricantes, o que faz com que o projeto e





as facilidades de assinantes implementadas variem bastante. Podem ser utilizadas em diferentes aplicações como:

- . Instaladas em carro (telefone celular veicular);
- . Transportáveis (telefone celular transportável);
- . Portáteis (telefone celular portátil).

A potência máxima de saída da unidade montada em carro é 3 watts, enquanto da unidade portátil é 0,6 W. Quando a EM acessa a CCC, envia a sua classe de estação na qual está indicada a sua potência máxima de saída.

A estação radio-base compreende as seguintes unidades funcionais:

- . Grupo de Canais de Rádio (GCR); . Interface Rádio Central (IRC);
- . Combinador de Antenas;
- . Antenas e Fontes de Alimentação.

O grupo de canais de rádio compreende canais de voz e canais de controle. A interface rádio central (IRC) trabalha como um adaptador dos sinais entre a CCC e a estação radio-base. Assim, esse equipamento recebe dados e voz das unidades de canais e envia essas informações para a CCC por meio do enlace dedicado ERB-CCC. Já na direção oposta o equipamento recebe dados e voz da CCC por meio dos enlaces de comunicação CCC-ERB e os envia para a unidade de canal ou controle correspondente. Além disso, a ERB:

- Fornece a interface entre a central de comutação de controle e as estações móveis;
- Contém uma unidade de controle, de transceptores rádio, antenas, plantas de alimentação e terminais de dados;
- Transmite e recebe sinais de controle para o estabelecimento e supervisão das chamadas;
- Transmite e recebe sinais de voz e de várias estações móveis dentro de sua área de cobertura.

A central de comutação de controle (CCC) é a central de comutação de serviço celular



considerada o coração do sistema móvel celular. Dentre suas funções, pode-se mencionar:

- É o elemento de coordenação central de toda a rede celular, pois administra todas as estações de radio-base dentro de sua área de controle, ou seja, comuta e controla um aglomerado celular;
- Estabelece a interface com a rede de comutação pública;
- Comuta chamadas originadas ou terminadas na estação móvel. Permite que a estação móvel tenha à sua disposição os mesmos serviços e facilidades fornecidos pela rede pública aos assinantes fixos;
- Dependendo do modo de transmissão (ERB-CCC), a velocidade do enlace para dados no canal de controle pode ser de 2,4 , 4,8 ou 9,6 kbit/s em enlaces analógicos ou de 64 kbit/s se for digital;
- Cada ERB é conectada a uma CCC via conexão do tipo quatro fios ligados a um multiplex PCM. Usualmente, a ligação é feita por cabo óptico;
- As antenas são responsáveis pela irradiação e recebimento dos sinais pelas estações móveis.

## **2.5 Funções Características da Rede Celular**

Um sistema celular completo pode conter várias CCC, as quais podem ser consideradas interfaces funcionais para a Rede Telefônica Pública Comutada (RTPC), e a sinalização empregada para o estabelecimento das chamadas dependerá da sinalização definida pela Rede Pública. Cada EM é conectada por software a uma única CCC, que normalmente é aquela em que o assinante é residente. Essa CCC recebe o nome de central domicílio (CCC-D).

### **2.5.1 Handoff**

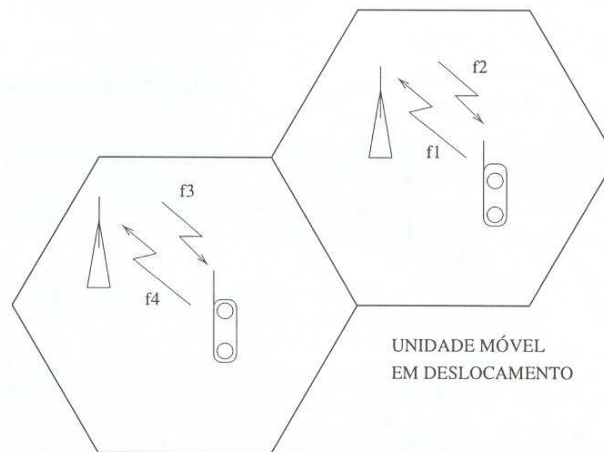
O *handoff* é uma função que permite manter a continuidade de uma conversação quando o usuário passa de uma célula para outra. O *handoff* está centralizado no CCC e causa uma interrupção na comunicação inferior a 0,5 s.

Quando a estação móvel entra na área de controle de outra CCC, essa central é conhecida como central visitada (CCC-V) e o assinante, como visitante. As chamadas para essa categoria de assinante serão roteadas e comutadas pelo CCC- V. Esse conceito de uma EM



deslocando-se de uma área de controle para outra é conhecido por *roaming*. A sinalização entre CCCs pode ser implementada, por exemplo, de acordo com o protocolo de sinalização por canal comum número 7 (SCC-7) do ITU-T, por meio de enlace direto que una as duas CCC ou ainda por um enlace da RTPC.

A Figura abaixo representa o deslocamento de uma unidade móvel.



**Figura 4 – Unidade Móvel em Deslocamento**

### 2.5.2 *Roaming* Automático

A função *roaming* automático age da seguinte forma:

1. Uma EM, ao entrar em uma nova área de controle, registra-se automaticamente na CCC que controla essa área;
2. A CCC visitada irá verificar se essa EM não havia se registrado anteriormente. Caso esse procedimento não tenha sido efetuado, a CCC visitada irá informar à CCC domicílio sobre sua nova posição;
3. Esta última, por sua vez, registrará que área de serviço o assinante está visitando. Após esse procedimento, o assinante visitante estará então habilitado a fazer e receber chamadas, como se estivesse em sua própria área de atendimento.

A mudança de uma ERB ligada a uma CCC para outra ERB ligada a outra CCC, durante uma chamada em progresso, é chamada *handoff* entre centrais. Em alguns sistemas, as áreas de controle podem ainda ser subdivididas em áreas de localização. Quando a EM passa de



uma área de localização para outra, ela deve informar a CCC sobre sua nova localização. Essa tarefa é conhecida como registro forçado ou como registro de área de localização e, nesse caso, a busca da EM é executada somente nas células que compõem a área de localização.

## 2.6 Capacidade do Sistema Celular

A capacidade de um sistema celular está basicamente relacionada com a sua possibilidade de gerenciar um grande número de assinantes, mantendo a qualidade da conversação. A capacidade do sistema celular é diretamente proporcional à capacidade da CCC. Um aumento do número de assinantes móveis implica em aumento na carga do processador central. O aumento no número de assinantes implica em reduzir a dimensão das células, o que deve elevar o número de *handoffs* por célula, gerando assim a necessidade de maior capacidade de processamento.

### 2.6.1 Tráfego

A capacidade de escoamento de tráfego do sistema celular é dado em **Erlang/km<sup>2</sup>** e depende do número de canais disponíveis, do tamanho da célula e do modelo de reutilização de frequências empregado. Quanto menor for a célula, maior será a capacidade de escoamento de tráfego do sistema. Existe, entretanto, um espaçamento mínimo para as células, cujo limite é estabelecido pelas condições de interferência co-canal.

### 2.6.2 Alocação de Canal de Tráfego

A densidade de tráfego de veículos ou pedestres de uma área de cobertura é um parâmetro crítico e por esse motivo deve ser determinado antes do projeto do sistema. O levantamento de tráfego em horas de grandes movimentos pode ser confinado a diferentes zonas dentro da área de serviços. A escolha dos locais das ERB iniciais deve ser baseada na cobertura de sinal em zonas de alto tráfego de veículos ou pedestres. Isto significa que o melhor local para situar as ERB será o centro dessas zonas.

### 2.6.3 A Unidade de Tráfego



Define-se a unidade de tráfego *Erlang*, abreviada para Erl, como o número médio de ocupações simultâneas durante um período definido. A unidade é uma homenagem a A. K. Erlang, um dos pioneiros da Teoria de Tráfego Telefônico. Para órgãos portadores de conversação comuns, o período é normalmente de uma hora. O tráfego  $A$ , em *erlangs*, é freqüentemente calculado da seguinte forma :

$$A = \lambda.t_m$$

**Equação 2**

onde  $\lambda$  = número de chamadas por unidade de tempo, ou de novas ocupações por unidade de tempo e  $t_m$  = duração média das ocupações expressas na mesma unidade de tempo.

Podem-se utilizar outros métodos de medida de tráfego. Suponha que durante um tempo  $T$  aconteceram  $N$  ocupações. A ocupação de um número  $n$  de órgãos teve a duração  $t_n$ . A duração das ocupações que já estavam em progresso no início do intervalo  $T$  é computada desde  $t = 0$ . Da mesma forma, a duração das ocupações que estão ainda em progresso no instante  $t = T$  é computada até esse instante .

O tráfego é dado então pela fórmula

$$A = \frac{1}{T} \sum_{n=1}^N t_n$$

**Equação 3**



## 3 – Métodos de localização

### 3.1 Introdução

No início os processos de localização por ondas de rádio ocorriam somente por triangulação, ainda a base teórica para a maioria dos sistemas hoje existente. Haviam os faróis de rádio, utilizados para guiar diversos meios de transporte de massa pelo o espaço aéreo ou marítimo. Mas, com a evolução tecnológica e a busca da comodidade, a necessidade de se localizar um dispositivo forma rápida e barata tomou-se evidente. Assim, surgiu a chamada industria dos Sistema de Localização Pessoal (SLP ou PLS), mais um ramo de trabalho da industria de telecomunicações.

O primeiro sistema comercialmente a se popularizar foi GPS, que no início, como toda inovação, era caro. Mas, já esta se tomando comum, passando a ser parte de quase todo kit destinado a acampamentos de final de semana.

Hoje, vêm se imaginando mais aplicações para o GPS. Na maioria das vezes destinadas a agilizar o atendimento médico. Casos onde o ato de se apertar um botão de dispositivo servido pelo serviço GPS, resulta na abertura de um chamado em uma central de atendimento, que encaminha uma unidade móvel médica especializada para o local de origem do sinal a partir das coordenadas providas pelo GPS.

Todas as aplicações do GPS estão relacionadas a um dispositivo que deve estar anexo ao elemento a se localizar. Este dispositivo GPS usa de satélites em orbita da terra como faróis de rádio para encontrar as coordenadas (latitude e longitude) de onde se encontra; uma triangulação com ajuda de satélites, em que o tempo entre geração do sinal pelo satélite e recepção na unidade GPS é medido.

Como no GPS, em todos os PLS o principal elemento é o dispositivo a se localizar.

Um equipamento a mais que deve ser portado pelo assinante do serviço de localização. A tendência é que esteja embutido em um PDA com linha celular digital operando em alta taxa de transferência de dados.

Atualmente várias pesquisas estão sendo realizadas pela industria dos PLS, utilizando de teorias e técnicas diversas relacionadas neste capítulo. O objetivo de todas elas é conseguir algo prático e com menor custo, que ofereça precisão razoável, pois o objetivo não é só



localizar um dispositivo, mas também trocar informações com ele.

A maioria das técnicas a serem citadas nas próximas seções estão evoluindo. Na tentativa de agregar valor, estão se unindo a outras tecnologias como a **wireless Internet**.

Além do serviço de localização, fornecerem o serviço de Anúncios de Atendimento a Áreas Restritas. Áreas eleitas como próximas à zonas comercialmente bem atendidas por diversas lojas, que pagariam bem para ter seus *banners* chegando *on-line* em celulares de quem predispuesse a recebe-los em troca de descontos na conta do SMC mensalmente.

As técnicas disponíveis são denominadas: ângulo de chegada (AOA) ; diferença do tempo de chegada (TDoA - Time Difference of Arrival) ; impressão digital da localização (*RF fingerprint*) ; localização da ERB, GPS , Localização por potência ( Ponto a ponto e Superfícies Equipotenciais).

### 3.2 - TÉCNICA DE ÂNGULO DE CHEGADA (AOA)

A técnica AOA usa da setorização da ERB e diretividade de suas antenas como aliados. Trata-se de uma técnica que utiliza da interface de rede do sistema móvel celular para executar o serviço de localização.

A técnica pode ser aplicada em diversos serviços, contudo este texto visa mais a tecnologia móvel celular, levando a exemplos voltados mais para esta área. A técnica funciona da seguinte forma:

1. Um equipamento transmissor de ondas eletromagnéticas executa a sua função. No caso do SMC, com o uso da UM é feita uma ligação;
2. O sinal de rádio gerado por este transmissor é recebido na ERB por um conjunto de antenas diferentemente posicionadas. Cada ERB é equipada com um dispositivo extra que detecta o ângulo do qual a chamada telefônica está chegando, tendo como referencia um plano de máxima amplitude de transmissão/recepção ;
3. As ERB's mandam a chamada de voz e os ângulos para a CCC, onde os ângulos são comparados e computados para gerar a latitude e longitude da chamada;
4. A chamada segue o seu caminho normal, mas os dados de latitude e longitude são repassados à uma central de atendimento de posse de um sistema de GIS, que fornece o endereço da localização.



Uma única ERB direcional pode fornecer a direção, contudo para a posição exata da unidade móvel é necessária outra ERB. Um processo semelhante, utilizando uma única antena, é usado para localizar animais portando antenas transmissoras. Fornece-se a direção de onde se encontra o animal, e com a ajuda de informações topográficas e intensidade do sinal recebido, localiza-se o transmissor.

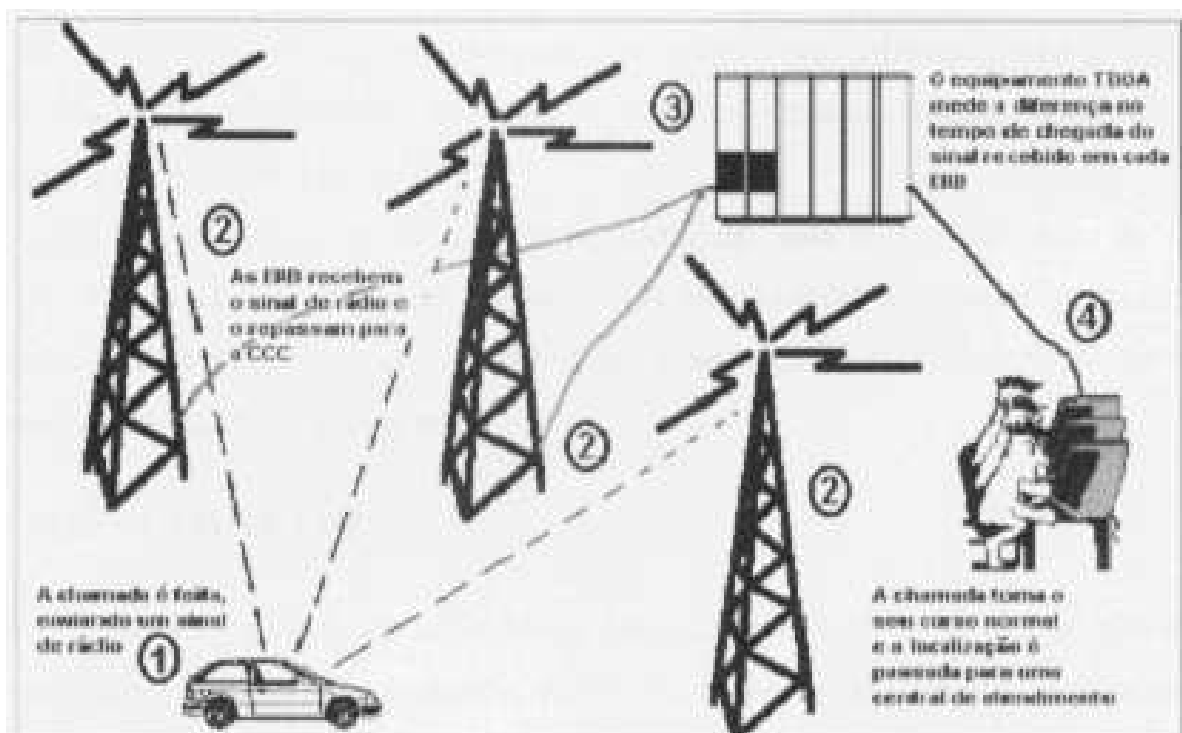
A precisão do técnica está ligada também à geometria das antenas, contando, no caso do SMC, com a ajuda de uma razoável quantidade de ERB's, que auxiliam ainda mais no processo de localização. Em áreas densamente povoadas, são raros os casos onde somente duas ERB's fornecem o ângulo de chegada.

### 3.3 - TÉCNICA DA DIFERENÇA DO TEMPO DE CHEGADA (TDOA)

Esta seção descreve outra técnica, seguindo a mesma lógica da seção anterior, que usa o SMC como objeto de trabalho. Contudo, deve-se lembrar que tal técnica também pode ser aplicada a outros sistemas que não o móvel celular.

A TDOA, baseada na rede de ERB's do SMC, utiliza da posição destas para localizar a unidade móvel. Posições disponibilizada através de um sistema GPS *on-line*, que fornece em tempo real a posição destas ERB' s e uma base de referencia de tempo precisa. O sistema funciona da seguinte forma:

Figura 5– Time Difference of Arrival







1. Uma chamada telefônica é realizada;
2. O sinal *wireless* é recebido em algumas ERB's. Desde cada ERB esta a uma distância diferente da unidade móvel geradora da chamada telefônica, o sinal chega a elas em tempos razoavelmente diferentes e imediatamente são repassados para a CCC
3. Usando de uma mesma referencia de tempo fornecida pelo sistema GPS e mais um relógio atômico, cuja precisão (número de casas decimais) determinará um menor erro na localização, encontra-se as distâncias da unidade móvel à cada ERB (pelo menos três) na CCC através da simples aplicação da equação :

$$c(\text{velocidadedaluz}) = \frac{d(\text{distanciadaERB})}{t(\text{tempofornecido})}$$

**Equação 4**

4. A chamada passada pela ERB segue o seu caminho normal, a latitude e a longitude são repassadas pela CCC a uma central de atendimento.

Esta técnica, diferente da AOA, necessita de uma ERB a mais, totalizando três, o que torna o sistema mais exigente. Mas, mesmo com três ERB, ainda podem ocorrer erros, sendo necessário as vezes uma quarta ERB.

### **3.4 - IMPRESSÃO DIGITAL DA LOCALIZAÇÃO (RF FINGERPRINT)**

Diferente da maioria das técnicas citadas acima, que apostam na rede da operadora do SMC, surgiu urna outra técnica baseada no comportamento apresentado pelas ondas eletromagnéticas em cada local da área de atendimento do serviço de localização.

Aposta-se no fato de as ondas de rádio, depois do SMC implantado, permanecerem com comportamento estacionário e assim terem características próximas das previsíveis e únicas. Urna marca registrada, sendo comparada a urna impressão digital de cada local em particular. Impressões que seriam levantadas com inúmeros *drive tests* sobre a área de atendimento. Os dados levantados são armazenados em um banco de dados a ser utilizado a



cada vez que o sistema entra em operação.

O pontos ou coordenadas geográficas aonde cada *drive* test seria realizado se distanciam em aproximadamente 50 m, preocupando-se com possíveis comportamentos anormais.

A principal vantagem deste método é a independência com relação a utilização de duas, três, quatro ou outro número de ERB' s no processo de localização. A pequenas variações devido o transito de veículos, pessoas e/ou animais não influenciam na precisão da técnica, mas mudanças na disposição de edifícios ou outras estruturas maiores sim. Mudanças que acontecem de forma lenta e previsível. O assinante, portando a unidade a se localizar, estaria fazendo uso de ondas estacionárias com características únicas. Medidas com relação ao nível do sinal recebido pelas ERB's vizinhas, comparadas as medidas às obtidas anteriormente na formação do banco de dados, fornecem a localização procurada.

### **3.5 - LOCALIZAÇÃO DA ERB**

Urna técnica que poderia ser considerada incompleta e inexata faz usa de micro/mini células, localizadas em ambientes fechados, para conseguir a informação de localização por área atendimento.

Cada urna destas pequenas células, que se assemelham a urna luminária aos olhos de pessoas leigas, podem atender a um andar de algum edifício comercial, ou urna seção de um shopping ou outra pequena área qualquer. Células que possuem, corno toda ERB, um código singular (*cell code*) para identificação. Saber qual micro ou mini célula está servindo um determinado assinante, implica também conhecer qual área restrita esta ERB atende. E, passar-se *banners* sobre promoções das lojas localizadas nesta área pelo sistema *wireless Intenet*.

Trata-se de uma serviço que pode ser vendido a partir de algumas pequenas configurações e implementações do *software* das CCC do SMC existente.

### **3.6 - SISTEMA DE POSICIONAMENTO GLOBAL (GPS)**

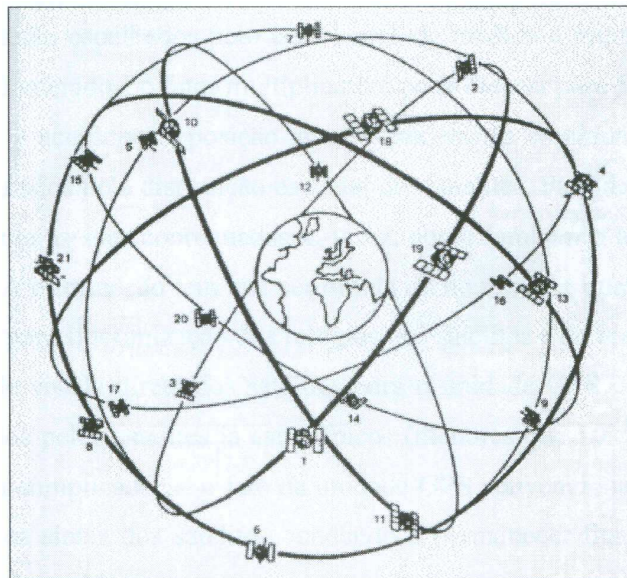
O GPS será considerado uma técnica somente pelo fato de ser um sistema que se popularizou, utilizando na realidade uma variação do TDOA, onde vários satélites e algumas referencias terrestres transmitem no lugar do dispositivo a se localizar; um comportamento



reverso.

Um cálculo semelhante ao feito no TDOA é realizado para que o receptor encontre a sua localização. Uma inversão passa a ocorrer. Antes, o objetivo era receber um sinal de rádio da unidade móvel, agora, a unidade móvel recebe o sinal de vários transmissores.

Opera com uma constelação de 24 satélites (Fig. 7), e depende que o elemento a se localizar porte uma unidade GPS, semelhante aos outros sistemas nos quais depende-se principalmente da rede disponível.



**Figura 6- Satélites em órbita**

No GPS os satélites transmitem o sinal banda larga em duas faixas de frequência denominadas L1 (1575,42 MHz) e L2 (1223,6 MHz). Os sinais são modulados por dois códigos pseudo-randômicos : o código P (*precision*) e o código CIA (*coarse/acquisition*). O sinal GPS é além disso modulado com uma mensagem de informação, conhecida com mensagem de navegação do GPS. Uma fato deve ser lembrado. Somente o código C/A na banda L1 é usado em aplicações civis. O GPS inicialmente era aplicado, como a maioria das grandes tecnologias (lembrar a Internet de hoje), em ambientes militares.

Para receber os sinais dos satélites, o receptor GPS gera uma réplica dos códigos pseudo-randômicos dos satélites. A mensagem de navegação do GPS é demodulada somente se a réplica combinar e sincronizar com o código gerado. Se a unidade portátil não conseguir este casamento, o sinal aparecerá para ele em forma de ruído. Caso contrário, é usada a mensagem de navegação dos satélites, possibilitando ao receptor calcular o tempo de



transmissão do sinal enviado pelos satélites, e, através de Eq. 4, obtêm-se as coordenadas destes.

A acurácia da posição obtida depende parcialmente das precisão das medidas e da configuração dos satélites. Erros de medida dependem de parâmetros físicos, tais como interferência da ionosfera e incertezas orbitais, e do fator de disponibilidade seletiva SA, introduzidos pelo departamento de defesa dos Estados Unidos para degradar parcialmente a qualidade das informações fornecidas pelos satélites à população civil. Estima-se erros de até 35 metros, os quais, sem este fator, são reduzidos à 8 metros.

A disposição dos satélites GPS no instante da obtenção das medidas adiciona distorções. Se eles estão espalhados pelo céu, o erro de medida é multiplicado por cerca de 1,5. Se eles estão aglomerados, o fator multiplicativo pode crescer para 5. Para estimar a acurácia da posição atual, é necessário combinar os erros de medida com os erros introduzidos pela disposição espacial dos satélites. Para determinar sua posição, a unidade GPS calcula as suas coordenadas x, y e z, corno também o tempo de chegada dos sinais dos satélites. A informação tem que ser obtida de no mínimo quatro satélites. O quarto satélite é necessário para sincronização dos relógios dos satélites e da unidade GPS.

Uma linha de visada direta dos satélites para a unidade GPS é importante, uma vez que os sinais emitidos pelos satélites já estão fracos (menores que 10-15 W) ao alcançarem a unidade. E, mais um complicador, é o fato da unidade GPS convencional necessitar de alguns minutos para obter os sinais dos satélites, tendendo a permanecer ligado permanentemente, exaurindo a sua bateria rapidamente.

### **3.7 - METODO DE LOCALIZACAO POR POTENCIA**

#### **3.7.1 - CONCEITO**

O método recebe esta denominação devido o processo de localização ser realizado a partir da medida do nível de potência dos sinais recebidos por ERB' s vizinhas à UM que os gerou.

O método da localização por potência exige mais trabalho computacional, pesquisa de modelos de propagação, ditos área a área e/ou ponto a ponto, para visualização do comportamento do sinal eletromagnético perante a topografia do terreno e a existência de edificações sobre ele. E, ainda estudos estatísticos com relação ao nível de perda aproximada em situações atenuação diferenciadas, em obstrução e efeitos multipercurso.



### 3.7.2 - MODELO PONTO A PONTO

O valor potência de um sinal transmitido está estreitamente ligado à cobertura deste sinal. Esta cobertura está ligada a modelos de predição de cobertura do sinal, que determinam o seu comportamento perante o meio onde se encontra imerso.

Para montar estes modelos de predição, tem-se que examinar a área por onde o sinal de rádio irá se propagar. Habitualmente, estas áreas são genericamente classificadas pelos aspectos que apresentam, como:

- Estruturas feitas pelo homem, ligada ao adensamento humano:
  - Em área aberta;
  - Em área suburbana; ou
  - Em área urbana;
  
- Pela natureza do terreno,
  - Se é plano;
  - Se é montanhoso;
  - Se possui áreas alagadas;
  - Se apresenta adensamento de folhagens.

O modelo de predição gerado se diferenciará dependendo de qual área foi abordada no estudo. A obtenção de um modelo de predição ponto a ponto é o ideal, contudo é trabalhosa. E, para levantamento destes modelos, são necessários três passos distintos:

1. Geração de uma condição padrão e fatores de correção;
2. Obtenção de um modelo de predição área a área;
3. Obtenção do modelo ponto a ponto, utilizando o área a área como base.

Existem muitos modelos de predição na literatura. A maioria limitando-se à obtenção do modelo área a área, onde pelos menos 68% dos valores do nível de potência são determinados com 6 a 8 dB (desvio padrão) de erro. Os modelos área a área são considerados em sua maioria bons. Mas, em grande parte das simulações, a necessidade é de um modelo



mais exato. A acurácia na determinação do valor correto necessita de um modelo final ponto a ponto, que faz uso de informações topográficas (GIS).

Com o modelo de predição ponto a ponto, conseguido a partir do modelo área a área e informações topográficas, diminui-se o erro na determinação do valor da potência recebida no local de interesse. O erro no valor da potência em determinado ponto decresce, chegando a um valor de desvio padrão de 3 dB. Contudo, conseguir o modelo ponto a ponto exige maior dispêndio de recursos e tempo, envolve maior trabalho computacional. Se a intenção é ter uma simulação aproximada e um erro razoável, o modelo área a área atende bem, principalmente em áreas razoavelmente planas.

O interesse é que aconteça o caminho em linha de visada sempre, mas o que se observa é o predomínio do outros dois casos.

A curva será levantada durante o percurso do sinal em determinada direção, quase que discretamente, ponto a ponto. O que não aconteceu com o modelo área a área, calculando uma média dos valores encontrados em *drive tests* em pontos sobre algumas circunferências equidistante do transmissor. O modelo ponto a ponto faz uma simulação matemática e não estatística; o que não o faz avesso à utilização de valores de ganho relacionados à propagação do sinal em folhagem.

A potência recebida será determinada por uma equação completa que considere os efeitos da propagação como um todo. O modelo é levantado através de uma equações mais genéricas do modelo de Lee [1] :

$$P_r = P_{ro} - \gamma \log \frac{r}{r_0} + 20 \log \frac{h_e}{h_i} + \alpha$$

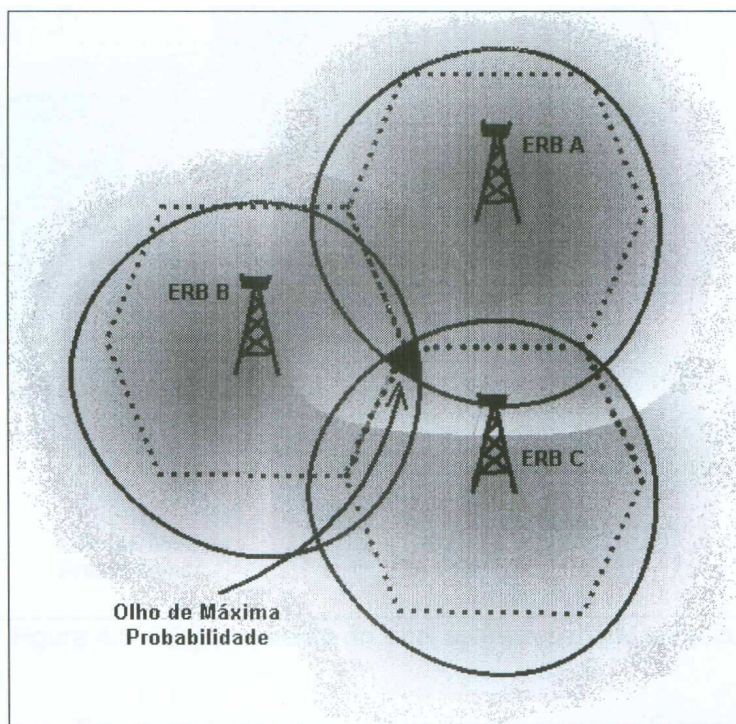
**Equação 5**

Para caminhos com obstrução, onde o valor de  $h_e$  pode ser observado :

$$P_r = P_{ro} - \gamma \log \frac{r}{r_0} + (L + \alpha)$$

**Equação 6**

Uma observação interessante é que nas equações acima o valor da altura efetiva é bem considerado.

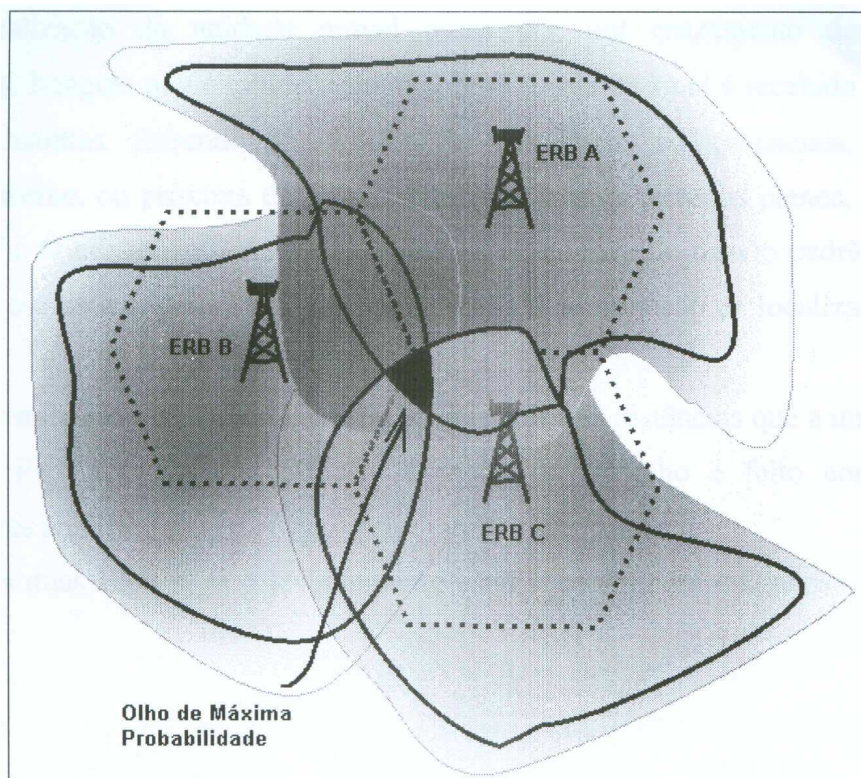


**Figura 7 - Modelo de predição ponto a ponto**

De posse das equações acima detalhadas, encontra-se o valor da potência recebida no receptor. Um trabalho braçal que pode ser evitado agregando-se as equação a um GIS. Trabalho desempenhado em diversos *softwares* à venda no mercado. Escolhe-se o ponto onde possivelmente o transmissor ficará, e com um trabalho computacional árduo levanta-se o modelo de predição ponto a ponto.

### **3.7.3 - CRUZAMENTO DE CURVAS EQÜIPOTENCIAIS**

Nas seções anteriores observamos a base do processo para determinação do valor de potência recebida em determinado ponto da área em estudo. A Fig. 9 é a representação mais aproximada devido a natureza instável do comportamento de espalhamento das ondas de rádio.



**Figura 8 - Espalhamento do sinal sobre superfície irregular**

Observando a figura nota-se uma degradação não uniforme, um espalhamento irregular. Esta irregularidade é devido à topografia do terreno. Evidencia da existência de áreas de sombra e obstruções. Em áreas planas o espalhamento é visto mais uniforme, mas em áreas de topografia irregular, as imperfeições causam reentrâncias ou uma maior dispersão. Os valores de potência do sinal, conseguido à certa distância, passam a ter, em lugares irregulares, a distância que se encontra do ponto de transmissão variada para mais ou menos com relação a um referencial, neste caso o valor levantado em propagação sobre área plana.

Em áreas irregulares somente o modelo de predição área a área, em lugar do ponto a ponto, levaria a um erro de determinação, e o valor de potência a ser obtido em distância  $x$  pelo modelo ponto a ponto, passaria a ser conseguido em  $x' = x : t z$ , sendo  $z$  um valor relativo ao comportamento topográfico.

A localização da unidade móvel passa por um cruzamento desta superfícies equipotenciais. Imagine que é sabido o quanto de potência de sinal é recebido de pelo menos três ERB' s distintas. Sabendo da distribuição das linhas equipotenciais, encontra-se a superfície disforme, ou próxima de uma circunferência para terrenos planos, para cada uma das três ERB' s. O cruzamento desta superfícies, dependendo do desvio padrão intrínseco ao levantamento estatístico, gerará um olho de máxima determinação da localização da unidade





móvel.

A determinação do olho de localização determina as distâncias que a unidade móvel se encontra das ERB's e em que direção. O resto do trabalho é feito com informações complementares ao GIS.

Apenas duas ERB' s, se o levantamento estatístico foi bem cuidadoso, podem gerar a resposta.



## 4 - Sistemas Celulares

### 4.1 Introdução

A Primeira Geração de Redes de Telefones Celulares comerciais iniciou a sua operação em 1978 em New Jersey. Essas redes utilizavam a modulação analógica para que o sinal de voz fosse transmitido pelo espaço livre. A recepção e a transmissão do sinal de voz era feita pela mesma frequência portadora. Esse sistema era chamado de AMPS (*Advanced Mobile Phone Service*). O espectro de frequência era limitado, como foi determinado pelas agências reguladoras. Somente uma faixa finita do espectro era disponível para o sistema celular funcionar. Uma das dificuldades encontradas pelo sistema neste momento, além do aumento da demanda por parte dos usuários, foi a de se otimizar o sistema, codificar e comprimir o sinal de voz. Portanto era necessária a digitalização do sinal de voz.

A Segunda Geração de Redes de Telefones Celulares comerciais foi desenvolvida em quatro padrões diferentes. Um primeiro foi desenvolvido a partir da evolução do sistema AMPS, que foi chamado de DAMPS ou TDMA/IS-54. Esse novo sistema foi desenvolvido para ser totalmente compatível com o sistema AMPS.

Já um segundo padrão totalmente incompatível com o sistema analógico foi desenvolvido na Europa durante a década de 80. Ele foi criado a partir das necessidades do *roaming* entre as regiões próximas de operadoras diferentes, de uma padronização para um sistema único e da capacidade de suportar a crescente demanda. A esse novo sistema deu-se o nome de GSM (*Global System for Mobile Communications*). O GSM rapidamente tornou-se um padrão dominante e com grandes quantidades de serviços extras como o popular serviço de mensagens, o SMS (*Short Message Service*).

Em meados dos anos 90, surgiu outro sistema chamado de CDMA (IS-95) desenvolvido pela Qualcomm. Esse sistema utiliza a codificação do sinal de voz para fazer o espalhamento desse sinal por todo o espectro de frequência, método chamado de *Spread Spectrum*. No Japão foi desenvolvido outro padrão chamado de PDC (*Personal Digital Celular*), que colocou o Japão em uma posição de isolamento com um sistema velho e incompatível com os outros padrões.

Na atualidade é comum ouvirmos também falar de sistemas 2,5G. Esse termo se refere aos sistemas celulares com serviços e taxas adicionais àquelas oferecidas pelos sistemas 2G, porém ainda não caracterizados como sistemas de terceira geração ou simplesmente 3G. Os



PCSs (*Personal Communication Services*) se enquadram nessa categoria. Esses serviços são oferecidos principalmente através de versões melhoradas dos atuais sistemas 2G sistemas operando nas bandas destinadas aos serviços PCS, bandas essas que diferem de país para país. No Brasil a banda escolhida para os serviços PCS está na faixa de 1,8 GHz. Pode-se então interpretar PCS como uma extensão dos sistemas 2G, em bandas distintas e oferecendo serviços de maior valor agregado que os sistemas 2G.

A comunicação sem fio capaz de permitir a troca de informações a altas taxas e com alta qualidade entre terminais pequenos e portáteis, localizados em qualquer parte do mundo, representa a fronteira a ser alcançada pelos sistemas de terceira geração. O início dos estudos sobre os sistemas de terceira geração foi marcado por uma indecisão mantida por duas correntes: uma defendia a criação de um único padrão mundial; a outra defendia a evolução das redes e sistemas atuais de forma a atender aos requisitos definidos a partir da visão 3G.

Atualmente, as empresas prestadoras de serviço de telefonia móvel no Brasil que tem a permissão de funcionar nas Bandas A e B oferecem seus serviços nas tecnologias TDMA/IS-136 e CDMA/IS-95. As empresas que estão em processo de inclusão no mercado brasileiro, que estão operando nas Bandas C e D, oferecem seus serviços na tecnologia GSM/GPRS. Sabendo disso, iremos descrever cada uma dessas tecnologias, suas arquiteturas, bem como suas vantagens, desvantagens e os possíveis caminhos de evolução de alguns desses sistemas dentro do contexto da geração a qual pertencem.

## **4.2 Redes de Segunda Geração (2G)**

### **4.2.1 TDMA / IS-136**

O TDMA é uma tecnologia de transmissão digital que permite o acesso de 3 usuários em um único canal de rádio frequência sem interferência, usando a mesma frequência portadora, através da alocação de um par único time slot para cada usuário, como pode ser visto na figura 10.

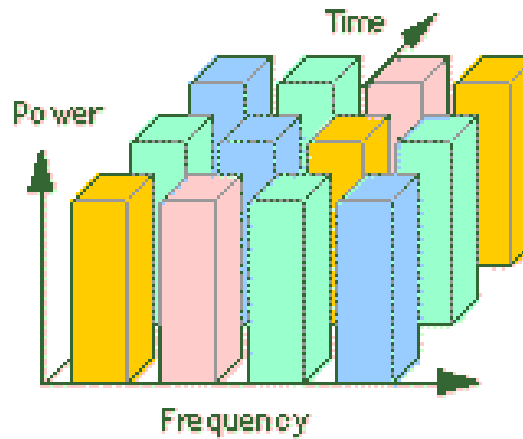


Figura 9. TDMA: Freqüência X Tempo

O *dowlink* e o *uplink* são separados em freqüências diferentes, conforme figura abaixo. No Brasil, a última versão dessa tecnologia disponível é o TDMA/IS-136 que é uma evolução do TDMA/IS-54.

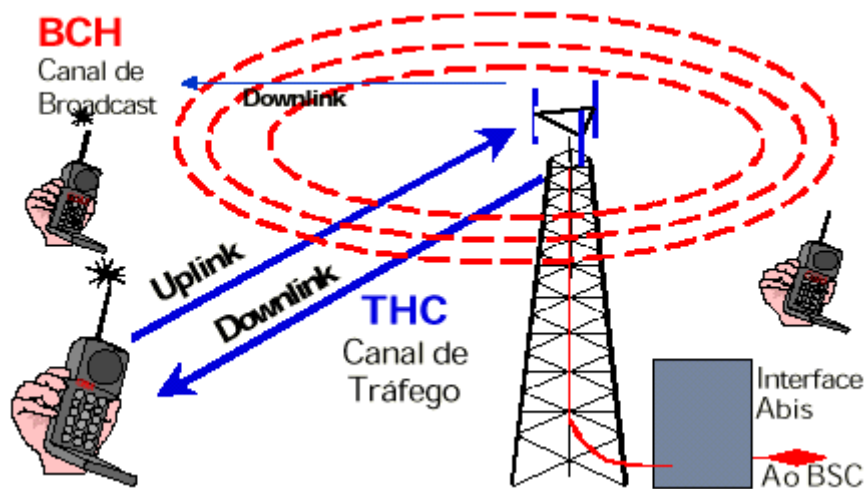
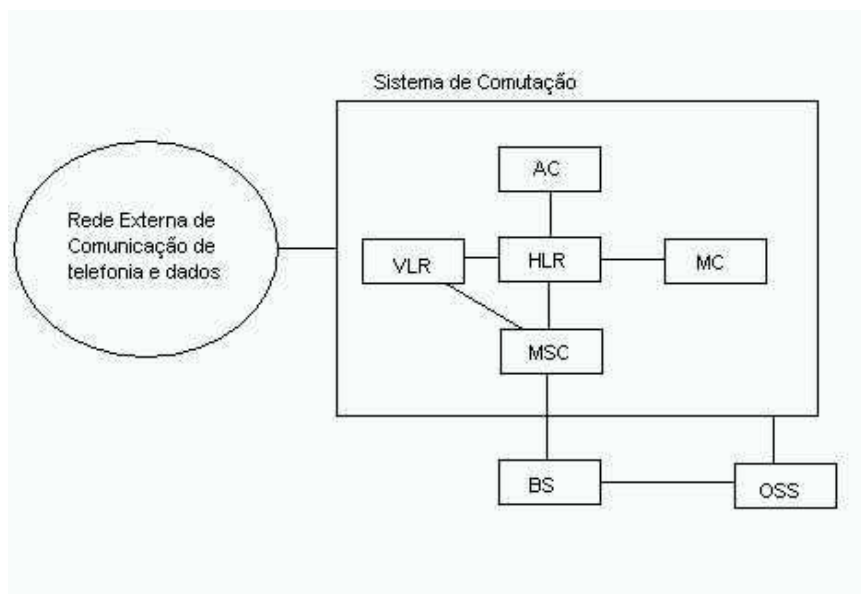


Figura 10 - Bandas separadas para o uplink e o downlink

A vantagem do TDMA/IS-136 é a qualidade de voz superior as outras versões dessa tecnologia. O TDMA/IS-136 se diferencia do TDMA/IS-54 no tipo de modulação utilizada nos canais de controle: a modulação  $\pi/4$  DQPSK. Ela eleva a taxa de transmissão nesses canais e provê serviços adicionais, tais como *paging* e o envio de mensagens curtas (SMS). A arquitetura do TDMA/IS-136 tanto para as freqüências de 800MHz e 1900MHz consiste em quatro módulos principais, conforme a figura abaixo:



**Figura 11 – Arquitetura do sistema TDMA/IS-136**

Os módulos principais são: o sistema de comutação (CCC), a estação rádio base (ERB), o sistema de operação e suporte (OSS) e a estação móvel (Aparelho).

O sistema de comutação é quem controla o processamento de chamadas e as funções relacionadas aos assinantes, contém cinco sub-módulos:

- MSC (Mobile Switching Center) - Geralmente chamada de CCC, é o módulo que realiza as funções de roteamento do telefone para a rede, fazendo o controle das chamadas para outras redes de comunicações de telefone e dados como a PSTN (*Public Switched Telephone Networks*), a ISDN (*Integrated Services Digital Networks*) e várias redes privadas.

- VLR (Visitor Location Register) - Módulo que armazena todos os dados dos assinantes temporários necessários para que os assinantes visitantes utilizem os serviços do MSC.

- HLR (Home Location Register) - Banco de dados que armazena e gerencia todos os dados dos assinantes permanentes incluindo o perfil de serviços oferecidos aos assinantes, informação de localização e *status* de atividades, sendo considerado o módulo mais importante do sistemas de comutação.

- AC (Authentication Center) - Módulo que provê os parâmetros de autenticação e encriptação que verificam a identidade dos usuários e assegura a segurança de cada ligação



e do sistema. Esta funcionalidade protege as operadoras de rede de fraude como clonagens de usuários.

- MC (Message Center) - Módulo que suporta vários serviços de mensagem, por exemplo, o voice mail e o SMS.
- ERB (Estação Rádio Base): é o módulo que conecta o usuário ao sistema de comutação, fazendo toda a comunicação de radiofrequência com o sistema de comutação e suporte (OSS). Ela é composta basicamente de equipamentos de rádios. O OSS auxilia na operação e na manutenção das atividades da rede permitindo a otimização dos serviços oferecidos.

No sistema TDMA/IS-136 contem 4 tipos de canais, o canal de controle analógico (ACCH), o canal de controle digital (DCCH), o canal de voz analógico (AVCH) e o canal de tráfego digital (DTC).

O TDMA/IS-136 traz como principal aprimoramento o canal de controle digital (DCCH) que substitui o canal de controle analógico (ACCH). O DCCH, que forma o centro das especificações do TDMA/IS-136, possibilita as operadoras serviços como o short message e sleep mode, que aumenta muito a funcionalidade como a carga da bateria dos celulares, além disso aumenta a capacidade do sistema.

Algumas outras das características importantes do TDMA/IS-136 são os serviços de pacotes de dados, o handoff móvel assistido, o SMS aperfeiçoado e o roaming inteligente. O handoff móvel assistido é usado para diminuir o número de ligações não completadas e garantir a qualidade do sistema móvel.

Para a comunicação de dados o TDMA/IS-136 é limitado, o que o ele oferece é a preparação do canal DCCH para aplicações de SMS que necessariamente requer uma plataforma externa e uma taxa de 9,6 Kbps fazendo o uso da tecnologia CSD (*Circuit Switch Data*) por canal de 30kHz.

#### **4.2.2 CDMA/IS-95**

O padrão CDMA/IS-95 é fundamentado em tecnologia com espalhamento espectral e foi projetado com o objetivo de possibilitar a transição suave da tecnologia analógica para a digital. O CDMA/IS-95 tem tido grande aceitação internacional entre os outros sistema de telefonia móvel por melhorar alguns aspectos relevantes das outras tecnologias como a interferência, sinais refletidos, baixa potências e planejamento de frequências. Os usuários



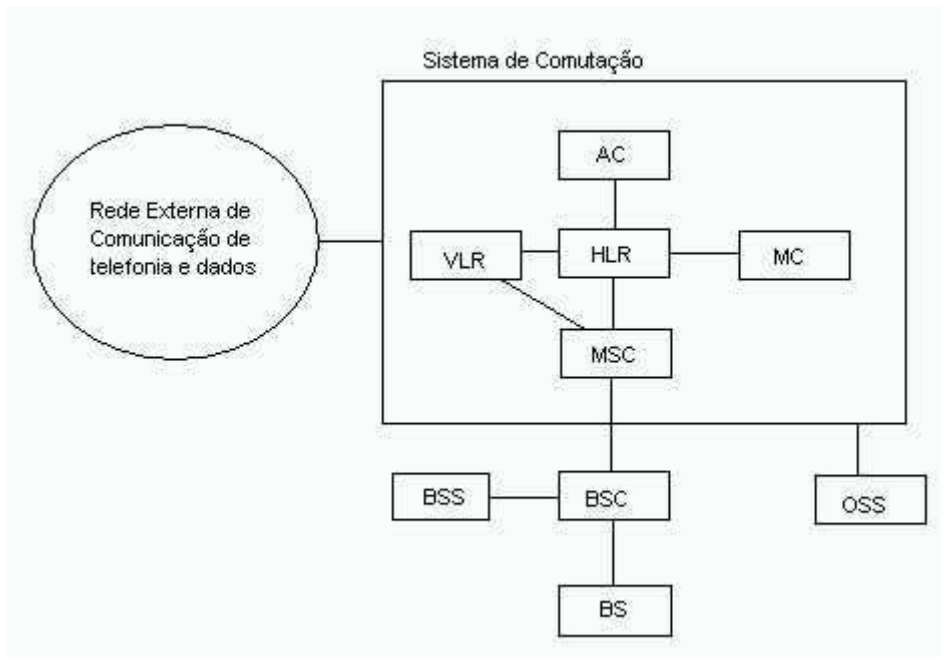
compartilham a mesma faixa de frequência, mesmo aqueles em células adjacentes onde o canal no CDMA/IS-95 ocupa 1,25MHz.

O uso da tecnologia CDMA foi liberado para a utilização para aplicações civis de telefonia móvel somente em 1989, mas foi proposto teoricamente na década de 40. Era usado nas aplicações militares, por exemplo, para esconder o sinal transmitido de espões e vencer os efeitos de forte interferência.

Além de basear-se no espalhamento espectral, otimiza a largura de banda por meio da equalização do SNR (*Signal to Noise Ration*) entre todos os usuários, dividindo com maior precisão os recursos de potência do sistema em relação às tecnologias TDMA e GSM. Assim os usuários que estão perto da estação base transmitem com menor potência, mantendo a mesma SNR que usuários mais distantes, por meio de uma menor potência a interferência entre canais é controlada. O IS-95, como o AMPS e o TDMA/IS-136 também usa o padrão IS-41 para gerenciar a mobilidade e protocolo de roaming.

A arquitetura do CDMA/IS-95 é muito parecida com a arquitetura do TDMA/IS-136, diferenciando-se somente na inclusão da BSC (*Base Station Controller*) e da BSS (*Base Station Support*). Tanto para as frequências de 800MHz e 1900MHz a arquitetura é composta por seis módulos principais, como podemos ver na figura abaixo:

- Sistema de comutação (CCC) - Controla as funções de interconexão e relacionadas aos assinantes;
- BSC - Controla o estabelecimento e processamento de chamadas;
- BSS - Auxilia as atividades de operação e manutenção das estações rádio base;
- BS ou ERB - Realiza a conexão do usuários ao BSC e as funções relacionadas a rádio comunicação;
- OSS – O Sistema de operação e suporte auxilia as atividades de operação e manutenção necessárias da central de comutação;
- Estação Móvel - Dispositivo de comunicação de voz e dados do usuário final.



**Figura 12 - Arquitetura do sistema CDMA**

As características técnicas principais do padrão CDMA/IS-95 são:

- Uso da mesma faixa espectral de operação do AMPS e IS-136 ( 800MHz );
- Largura de faixa de transmissão igual a 1.25MHz;
- No *downlink* (ERB para celular) são utilizados 64 códigos de Walsh-Hadamard combinados para ortogonalidade entre os sinais dos usuários, utilização de uma seqüência PN do tipo m para um espalhamento efetivo e robustez contra os efeitos dos multipercursos;
- No *uplink* (celular para ERB) são utilizados seqüências PN tipo m que apresentam melhores propriedades para sincronismo;
- Codificação de canal convolucional com taxa de 1/2 no *downlink* e taxa 1/3 no *uplink*;
- Uso da portadora piloto para sincronismo e detecção coerente nos terminais móveis;
- *Soft handoff* (mesma portadora para todo o sistema) - tem como principal objetivo melhorar o processo de *handoff*.

O CDMA/IS-95 ainda apresenta alguns recursos que podem ser considerados como vantagens frente aos outros sistemas:

- *Vocoder* variável: melhora a qualidade de voz , pois são introduzidas maiores redundâncias na codificação de voz, ficando menos susceptível ao ruído;





- Apresenta nas suas especificações telefones móveis duais (AMPS/IS-95). Quando usado como CDMA pode operar com um *vocoder* de 8kbps ou 13kbps.
- *Rake receiver*: é um receptor de rádio capaz de diferenciar multipercursos com uma resolução de até 0.81 $\mu$ s. O *rake receiver* é um dispositivo de diversidade que gera um ganho de recepção melhorando a qualidade e o desempenho do sistema.
- *Soft capacity* – degradação suave da qualidade do *uplink* quando há sobrecarga de usuários do sistemas.

Como o TDMA/IS-136, o CDMA/IS-95 se apresenta como uma tecnologia limitada à comunicação de dados com taxas de transmissão de 14.4kbps fazendo o necessário emprego do CSD. Também requer uma plataforma externa adicional para prover o SMS.

### 4.2.3 GSM

O sistema GSM teve suas especificações publicadas em 1990 pela ETSI (*European Telecommunication Standards Institute*). Ele foi introduzido na Europa em 1991 e rapidamente adotado em vários países do mundo.

Esse sistema, assim como o TDMA/IS-136, combina tanto o TDMA como o FDMA (*Frequency Division Multiple Access*). Nele a portadora de frequência é dividida em 8 *times slots*, sendo que a taxa de codificação de voz é de 13 Kbps, e é modulada usando GSMK (*Gaussian Minimum Shift Keyed*) em uma portadora de 200kHz.

O controle de erro e equalização contribuem para a robustez do sinal de rádio contra interferência e atenuação na transmissão. Este sistema usa um codificador de voz para definir um modelo de geração de tons e ruídos na garganta humana e a filtragem acústica feita pela boca e língua. Estas características são usadas para produzir coeficientes, que são enviados pelo TCH.

O codificador de conversação é baseado em um codificador preditivo linear com excitação residual (RELTP); este codificador é aperfeiçoado com a inclusão de um dispositivo preditivo de longo prazo (LTP). O LTP melhora a qualidade da conversação removendo a estrutura dos sons das vogais antes de codificar os dados residuais. A saída do codificador fornece 260 bits para cada bloco de conversação de 20 ms. Isto resulta em uma taxa de 13 Kbps. Os bits da saída são ordenados, conforme a sua importância, em grupos de 182 e 78 bits. Os 182 bits mais importantes são subdivididos, com a separação dos 50 bits muito importantes. A taxa de dados de 13 Kbps é consideravelmente menor que a digitalização



direta da conversação, como a feita no PCM. No futuro, codificadores de voz mais avançados reduzirão esta taxa a até 6,5 Kbps (codificação em meia taxa).

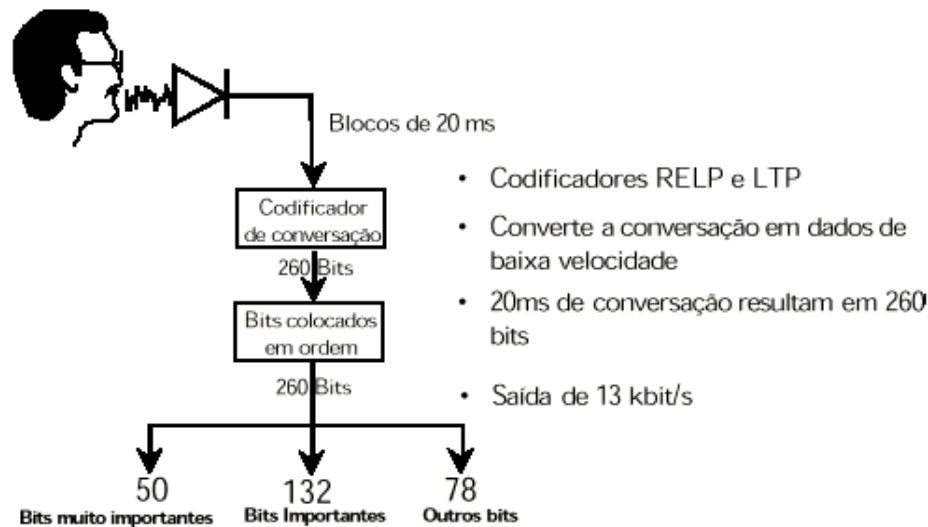


Figura 13 - Codificação da voz no GSM

A arquitetura do sistema GSM é similar a dos sistemas descritos anteriormente, exceto pela estação móvel (celular) que contém o elemento SIM (*Subscriber Identity Module*). A rede GSM é subdividida em dois subsistemas, cada qual com suas respectivas unidades funcionais. O primeiro módulo é o sistema de comutação, composto pela CCC, bases de VLR, HLR, MSC, AUC e EIR. O segundo módulo é denominado de Estação Base que é composta pela BSC e BTS que têm a mesma função dos sistemas explicados anteriormente.

Os três módulos que compõem a arquitetura do GM e não existem nos demais descritos acima são:

- **AUC** - Base de dados conectada ao HLR que é denominado centro de autenticação, cuja função é autenticar os assinantes que tentam utilizar uma rede, por meio de perfis de autenticação e chaves de cifragem. Este recurso permite a proteção das operadoras dos serviços contra fraude.
- **EIR** - Base de dados denominada registradora de identidade de equipamento, inclusive o número de série eletrônico de cada aparelho (ESN), e tem como finalidade bloquear as chamadas não autorizadas, “clonadas” ou com defeito.
- **MSC** - *Gateway* que permite a conexão para efeito de roteamento ou conexão de chamada entre o sistema de comutação GSM e outras redes, como por exemplo a rede fixa comutada.

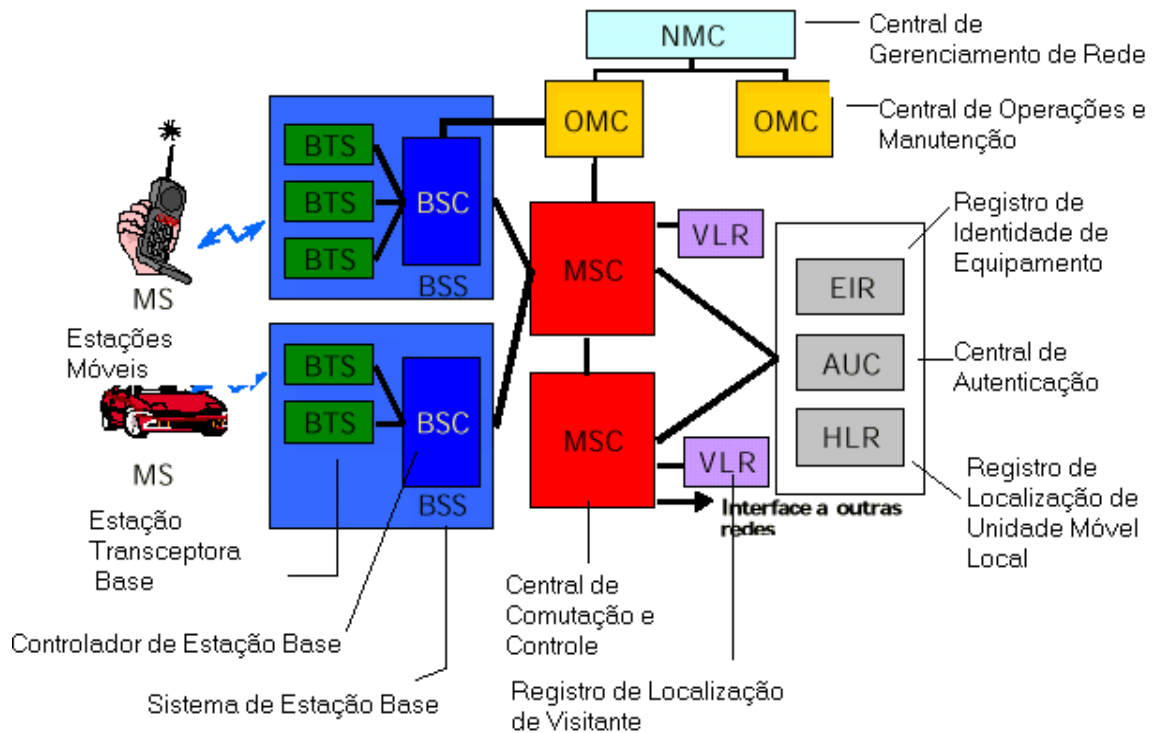


Figura 14 - Arquitetura do sistema GSM

O terminal móvel opera em várias frequências (900,1800 e 1900MHz) e modos de compatibilização com o sistema GSM. Esse sistema separa o *uplink* e o *downlink* em bandas de frequência distintas, da mesma forma que o TDMA/IS-136. Dentro de cada banda, o esquema de numeração de canais usado é o mesmo. Na verdade, um canal do GSM é formado por um *uplink* e um *downlink*.

É interessante observar que, enquanto o TCH usa um canal de frequência no *uplink* e no *downlink*, o BCH somente ocupa um canal no *downlink*. O canal correspondente no *uplink* é, na verdade, deixado desocupado. Este canal pode ser usado pela unidade móvel para canais não programados ou canais de acesso aleatório (RACH). Quando a unidade móvel quiser chamar a atenção da estação base (para fazer uma chamada, por exemplo), ela poderá fazê-lo usando este canal de frequência desocupado para enviar um RACH. Como mais de uma unidade móvel pode querer chamar a atenção da estação ao mesmo tempo, é possível que haja uma colisão de canais RACH, e talvez seja necessário que as unidades móveis façam diversas tentativas para serem ouvidas.

Os terminais móveis são compostos por duas partes principais, a primeira composta pelo equipamento em si e a segunda o “*Subscriber Identity Module*”, o SIM Card. Ele contém as informações de identificação e dados usados pelo assinante, por exemplo:



- Seu número exclusivo de assinante ou IMSI (Identificação Internacional de Assinante Móvel);
- As redes e países em que o assinante pode receber o serviço (MCC e MNC);
- Quaisquer outras informações específicas do usuário, como números de discagem rápida e memórias.

O *SIM Card* trouxe mobilidade pessoal e mobilidade para o terminal, sendo uma das vantagens do GSM.

O SMS é o atual padrão de comunicação de textos identificado pelos serviços das redes GSM e permite ao usuário consultar a base de informações da operadora em formato “texto”, sem estar conectado em banda larga e sem usar o padrão de mensagem da Internet. A velocidade dessa comunicação está entre 9.6kbps e 14kbps.

O sinal TDMA do GSM por ser de natureza digital permite a utilização de vários processos para melhorar a qualidade de transmissão, o tempo de vida útil da bateria, e a eficiência espectral em relação ao sistema AMPS. Outra vantagem do GSM é o controle de potência que permite minimizar a potência de transmissão das estações móveis e da BS, reduzindo assim a interferência gerada nos canais e o consumo.

Como o TDMA/IS-136 e o CDMA/IS-95, o GSM é excelente para comunicação de voz, mas se apresenta uma tecnologia limitada à comunicação de dados com taxas de transmissão de 9.6kbps fazendo o necessário emprego do CSD, ainda requerendo uma plataforma externa adicional para prover o SMS.

## **4.3 Redes da geração 2,5G**

### **4.3.1 CDMA 1XRTT**

A tecnologia CDMA 1XRTT é o primeiro passo na evolução do padrão CDMA/IS-95 para a terceira geração (3G), nas frequências de 800MHz e 1.9GHz. Essa versão também é conhecida como CDMA2000-1X.

As especificações atuais do CDMA são às vezes referenciadas como IS-95C e correspondem à fase 1 de implementação do CDMA2000, também conhecida por CDMA 1XRTT. As atuais implementações do padrão têm como principais características:

- Melhorias no quesitos capacidade, cobertura e eficiência espectral;



- Maiores taxas de dados (64 a 144 Kbps), incluindo transmissão de dados por comutação de pacotes;

- Maior tempo de duração da bateria no modo *stand-by*;
- Processos de *handoff* mais eficientes.
- Internet móvel, acesso a à Intranet e à Internet, vídeo conferência;
- Para acesso fixo sem fio usa-se Walsh 128 *versus* o 64, usado no CDMA;

As principais alterações de hardware de forma a atender as características anteriormente citadas são: no *downlink*

- Controle de potência mais rápido e preciso;
- Substituição da modulação BPSK por QPSK, com 128 funções Walsh;
- Redução no *overhead* por *soft-handoff*.

E no *uplink*:

- Redução de retardo no processo de controle de potência;
- Uso de modulação BPSK com detecção coerente auxiliada por código piloto;
- Uso de codificação de canal com taxa 1/4 ao invés de 1/2 e 1/3.

O CDMA2000-1XEV-DO(DATA ONLY) é o primeiro passo de evolução da tecnologia CDMA2000-1X suportando a comutação de voz por pacotes e dados a altas velocidades também comutadas em portadoras separadas, sendo que necessariamente uma portadora deve ser dedicada para dados. O canal de dados possibilita o roteamento flexível e as vantagens de transmissões de baixo custo de uma rede baseada na transmissão de pacotes. O CDMA2000-1X EV-DO fornecerá velocidades de dados de até 2,4Mbps na mesma portadora.

A promessa da tecnologia CDMA 1X-EV é de fornecer 2,4Mbps em picos de dados, média de 600kbps com alta qualidade de voz e dados em tempo real com a mesma alocação de espectro de 1,25MHz.

### **4.3.2 GPRS**

O **GPRS** é um novo serviço de valor agregado, não baseado em voz, que permite o envio e recepção de informações através de uma rede telefônica móvel.

Esse padrão de rede é uma evolução das redes GSM existentes que introduzem transmissão de dados por pacotes, permitindo que os terminais móveis operem em modo



sempre conectados (*always on*). Essa estrutura de comunicação permite que os usuários possam sempre receber e-mail, manter-se acessando aplicações pela Internet e outros serviços.

O GPRS é considerado um meio de caminho do GSM até a terceira geração de redes de serviços móveis, que terá comutação de pacotes em conexão virtual permanente junto com as conexões de circuitos de circuito. O ETSI homologou esse sistema como a segunda fase da rede GSM. O modelo de roteamento e transporte de pacotes viabiliza a cobrança mensal por serviços e/ou volume de dados trafegados em vez de cobrança por tempo e distância geodésica.

Esse padrão viabiliza várias aplicações novas que não eram possíveis em redes GSM devido às limitadas velocidades na comunicações de dados, baseadas em comutação por circuito e modems, que permitiam atingir taxas de apenas 9,6kbps e suportavam apenas a troca de mensagens de pequenos comprimentos.

Com o GPRS, os pacotes podem ser enviados e recebidos em taxas de até 115kbps. A máxima taxa teórica permitida no GPRS é de até 171,2kbps. É realizável com o uso de todos os oito time slots do rádio canal ao mesmo tempo, sem qualquer proteção de erro no quadro GSM, para um único canal físico para transporte de diversos canais lógicos IP. Evoluindo para o sistema EDGE (*Enhanced Data for GSM Evolution*), que será abordado no próximo item, essas taxas de dados poderão aumentar para até 384kbps.

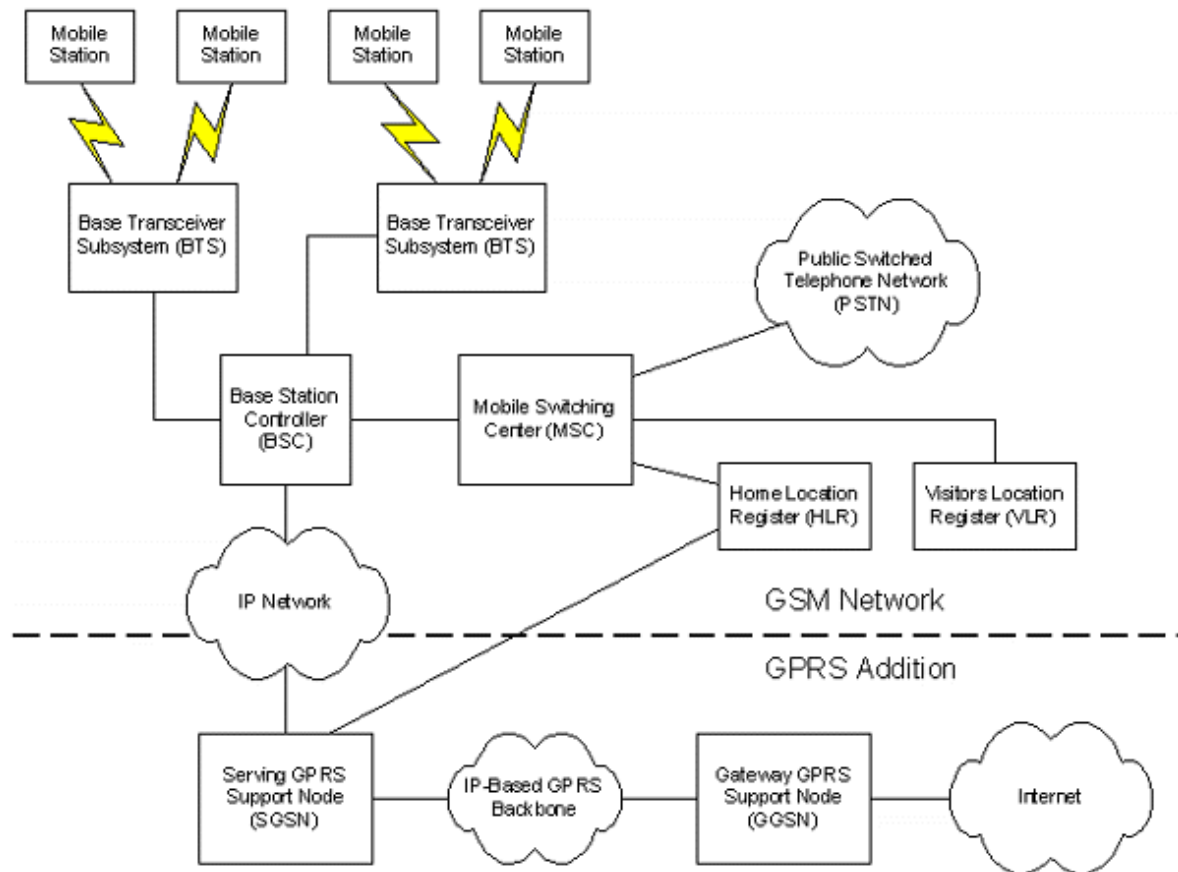


Figura 15 - Arquitetura do sistema GSM/GPRS

A estrutura GPRS, figura 16, é implementada acrescentando nós de pacotes à rede GSM por meio de “gateways”, que permitem a interconexão dos roteadores GPRS com os computadores CCC/GSM. O acesso via “gateway” permeia o *interworking* com acesso a aplicações Internet e Intranets. As características do sistema são:

- Prover acesso à Internet por meio de protocolos TCP/IP;
- Utilizar roteamento IP com rotas fixas ou dinâmicas;
- IP móvel utiliza pacotes de dados CDMA como padrão;

Como os protocolos em uso são os mesmos, as redes GPRS podem ser encaradas como sub-redes da Internet e os telefones GPRS compatíveis podem ser vistos como nós móveis dessa rede. Isso significa que cada terminal GPRS pode potencialmente ter seu próprio endereço IP e ser endereçável, provendo o acesso à Internet de várias maneiras como é descrito na figura 17 abaixo.

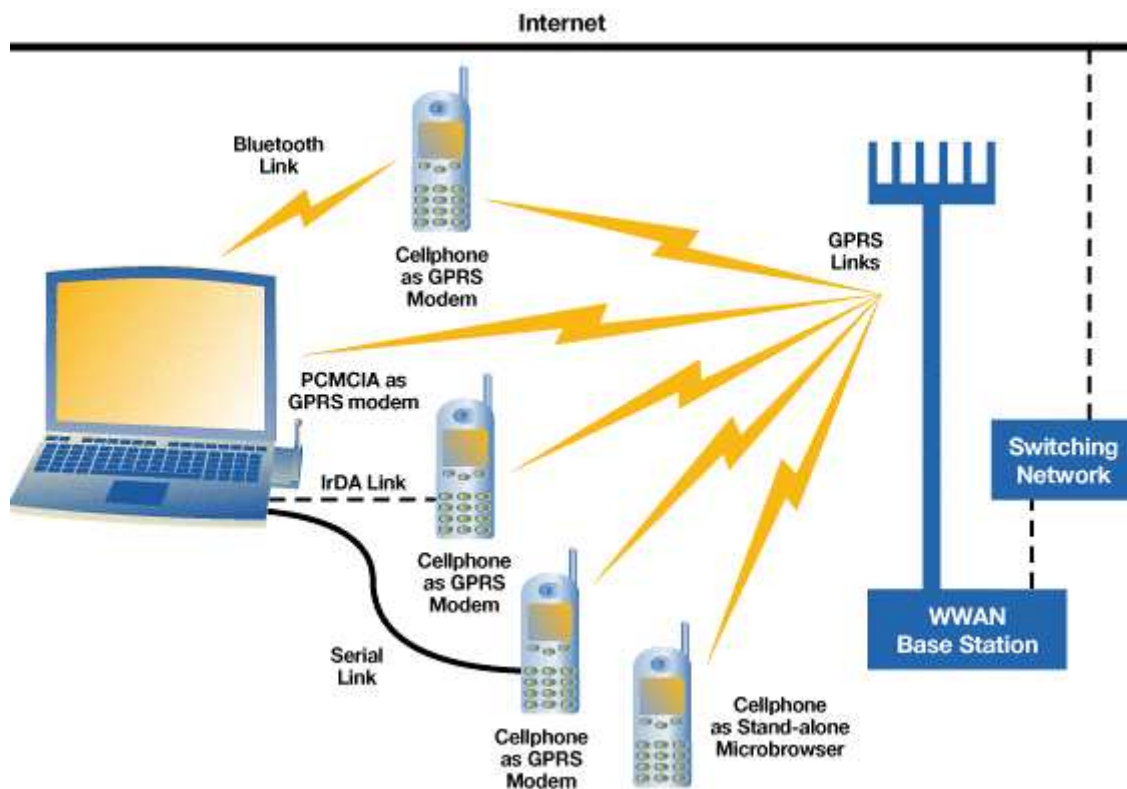


Figura 16 – Configurações diferentes da utilização do GPRS para o acesso à Internet.

O GPRS mantém a largura de faixa de canais GSM com os 200kHz atuais, 8 *timeslots* por quadro e taxa de símbolos de 270,833kHz. No caso da utilização da taxa de dados mais altas (permite uma taxa de bit bruta de 22,8kbps por *timeslot*), os dados são transportados em diversos canais, usando diversos *timeslots* com uma taxa final agregada. A taxa máxima teórica do GPRS é de 171,2kbps, usando todos os oitos *timeslots* simultaneamente, sendo aproximadamente dez vezes mais rápida que os serviços atuais do CSD nas redes GSM.

Apesar dos melhoramentos na eficiência espectral, na capacidade e na funcionalidade se comparada com os serviços de dados oferecidos atualmente é importante notar algumas limitações como a limitação na capacidade da célula em prover serviço para todos os usuários por causa do compartilhamento dos rádios entre o sistema de dados GPRS e de voz GSM, a taxa de transmissão na prática é muito menor.

A primeira rede GSM/GPRS no Brasil foi ativada pela operadora Oi/Telemar em junho de 2002. A tecnologia inicialmente disponibilizada permite a transmissão de voz e dados a uma velocidade de até 64kbps. O assinante da telefonia móvel celular baseada na tecnologia GSM/GPRS poderá escolher o seu provedor de telefonia de longa distância, diferente do que ocorre hoje no serviço móvel celular tradicional.





## 4.4 EDGE

O EDGE (*Enhanced Data for GSM Evolution*) representa a evolução final de comunicações de dados dentro do padrão de GSM. Esse esquema utiliza uma nova modulação disponível nas redes GSM, de forma a ampliar a taxa de transmissão para até 384kbps (teoricamente até 473kbps), praticamente aproveitando muito da infra-estrutura GSM atualmente existente.

O GPRS prevê a implementação de um novo Sistema de Estação Base (BSS), o que facilitaria a migração para o EDGE. Desta forma, para o EDGE será necessário a substituição da Unidade Transceptora EDGE (TRU), acrescida de atualizações de software. A adição de hardware de transceptores é comum ao ampliar a rede. A necessidade adicional será a instalação do TRU EDGE, em vez do TRU GSM no BSS, ao implementar o EDGE. As unidades transceptoras EDGE podem controlar simultaneamente voz comutada em circuitos e roteamento de pacotes GPRS.

O EDGE mantém o espaçamento de canais GSM de 200kHz existentes, 8 timeslot s por quadro e taxa de símbolos de 270,833kHz. O que muda é a modulação no enlace de rádio comunicação, no modo GSMK a 8PSK. Essa modulação permite uma taxa bruta de bits de até 69,2kbps por timeslot.

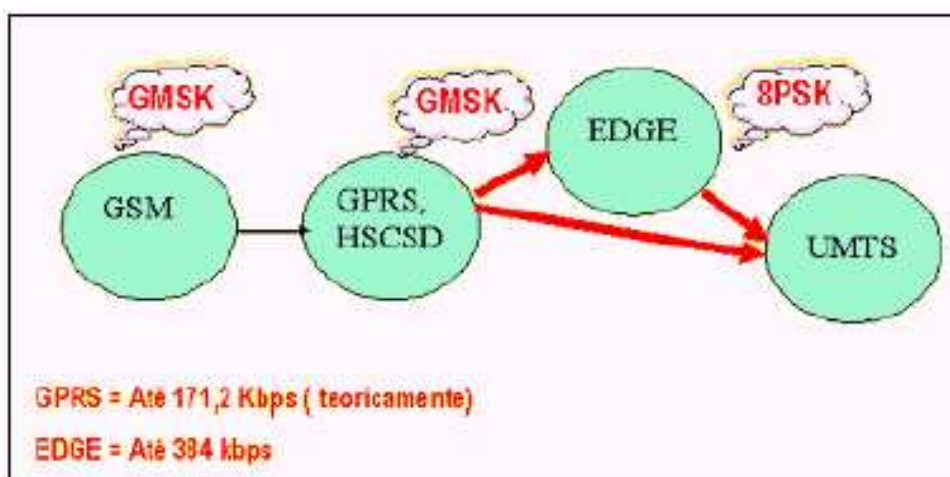


Figura 17 - Caminhos para evolução de uma rede GSM



O 8PSK é usado para uma rotação de  $3\pi/8$  entre símbolos (constelação 16PSK) com 3 bits por símbolo. Entre as suas desvantagens está uma redução na imunidade ao ruído, o que requer maiores níveis de potência.

Os provedores de serviços precisarão bloquear canais específicos para entrar no EDGE que não encaixa diretamente no sistema existente, mas exige uma reutilização das bandas na rede, níveis de potência e um novo cálculo do sistema de *link budget*.

O EDGE está baseado em um esquema de modulação na interface aérea usando o 8PSK que também é usado para o 3GSM (3ª geração do GSM). As operadoras de rede que incorporassem este sistema estariam a meio caminho para a transição definitiva para a terceira geração de sistemas móveis. Com EDGE a taxa de sinalização de dados na interface aérea poderia chegar a 59,2kbps.



## 5. Wireless Village

### 5.1 INTRODUÇÃO

O Wireless Village é um Serviço de Presença e Mensagens Instantâneas (IMPS), fundado pela Ericsson, Motorola e Nokia, a fim de definir uma série de especificações universais para o desenvolvimento de aplicativos para comunicação entre as unidades móveis em sistemas GSM. Estas especificações seriam utilizadas para troca de mensagens e informações de presença entre as unidades móveis, com o intuito de que fosse possível se desenvolver aplicativos (serviços) baseados em localização para estas unidades.

A arquitetura do WV descreve o sistema IMPS e a relação entre a rede móvel e a internet. Este é um Sistema baseado em Cliente-Servidor, onde o servidor é o IMPS Server, e os clientes são cada unidade móvel, ou outras serviços/aplicações, como PC-clients por exemplo. Para uma melhor interoperabilidade entre os dispositivos, os Servidores IMPS e os Gateways são conectados via Protocolo Servidor-Servidor (SSP), como mostra a figura abaixo:

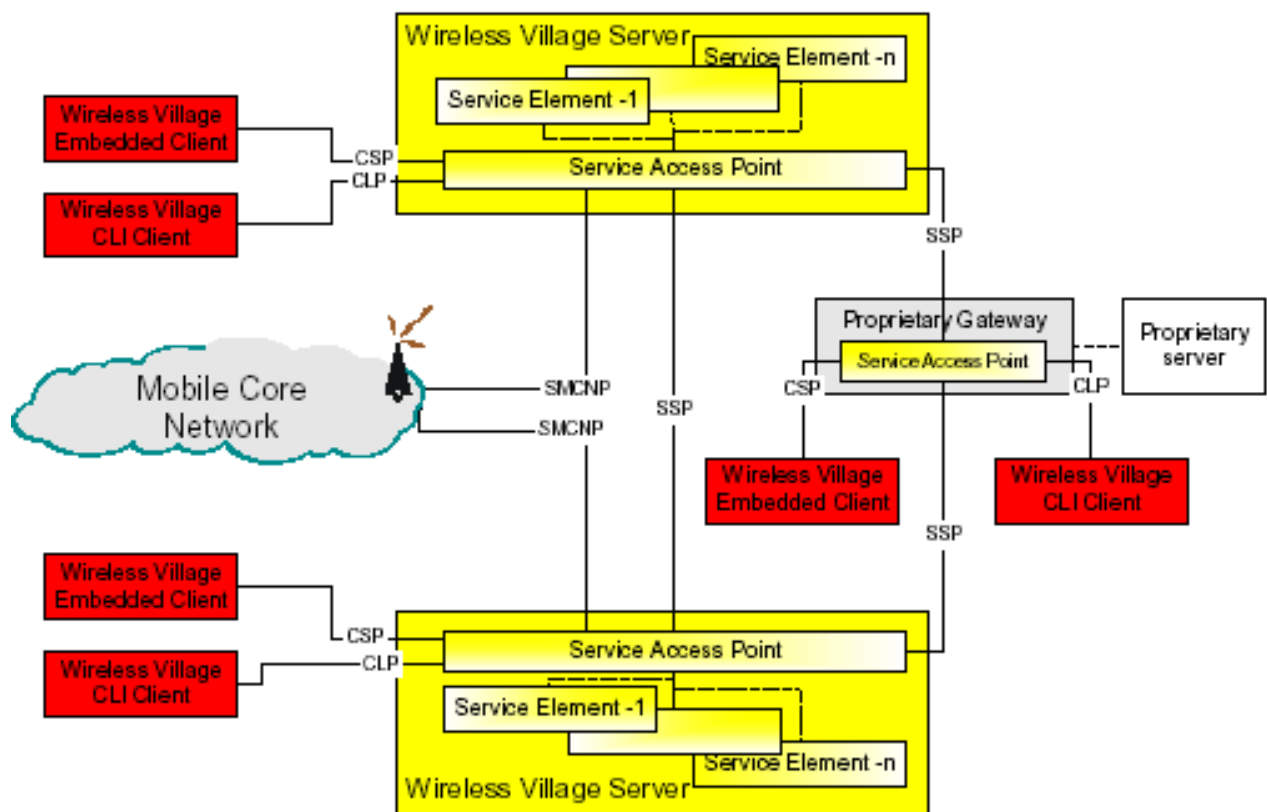


Figura 18 - Modelo de Arquitetura do Sistema Wireless Village



O LCS não faz parte do Servidor. Portanto, entra no Mobile Core Network (núcleo de rede móvel).

Podemos dizer que o Servidor é formado pelo SAP mais o Presence.

Dentre os elementos do Servidor WV, podemos destacar os seguintes, como mostra a tabela abaixo:

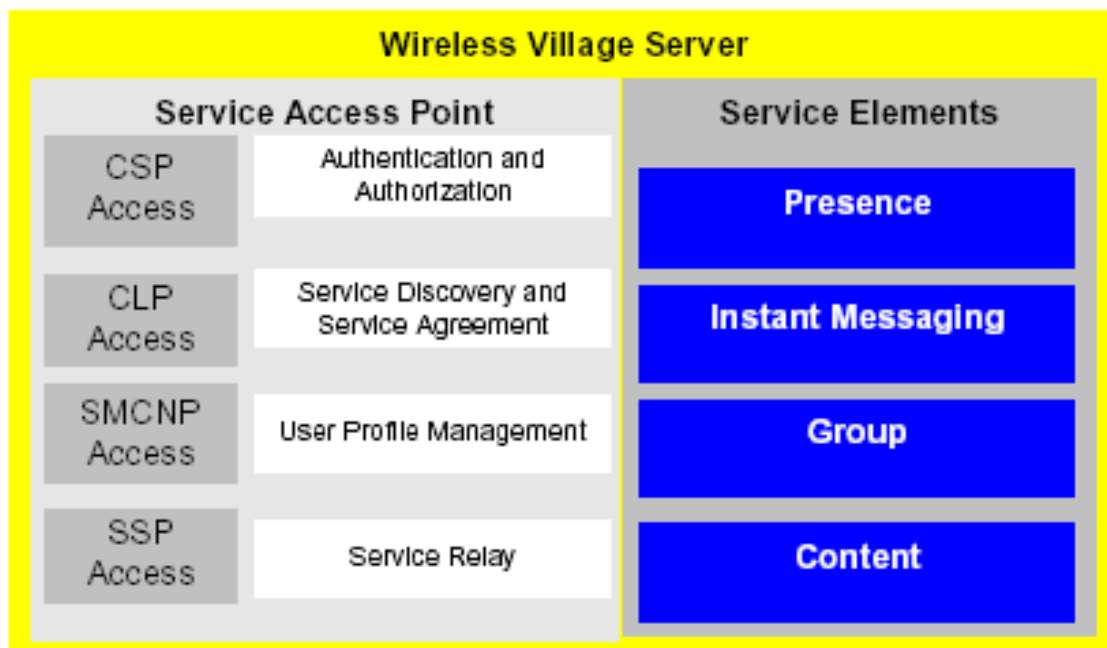


Figura 19 - Elementos Funcionais do Servidor Wireless Village

## 5.2 ARQUITETURA DOS ELEMENTOS WV – DESCRIÇÃO FUNCIONAL

### 5.2.1 SERVIDOR WV

O Servidor WV é o ponto central do Sistema WV. É composto por quatro elementos principais, os quais são acessados via SAP. São eles:

1. **Presence Service Element:** provê ao sistema informações de presença. Isto inclui Updates, resposta à requisições de informações, obter e guardar informações sobre localização e presença. Um ponto importante a ser citado é que as informações de



presença podem ser manipuladas implicitamente pelo sistema ou explicitamente pelos usuários.

Por exemplo, um usuário pode escolher por receber informações de presença sobre algum outro usuário que esteja na sua Lista de Contatos.

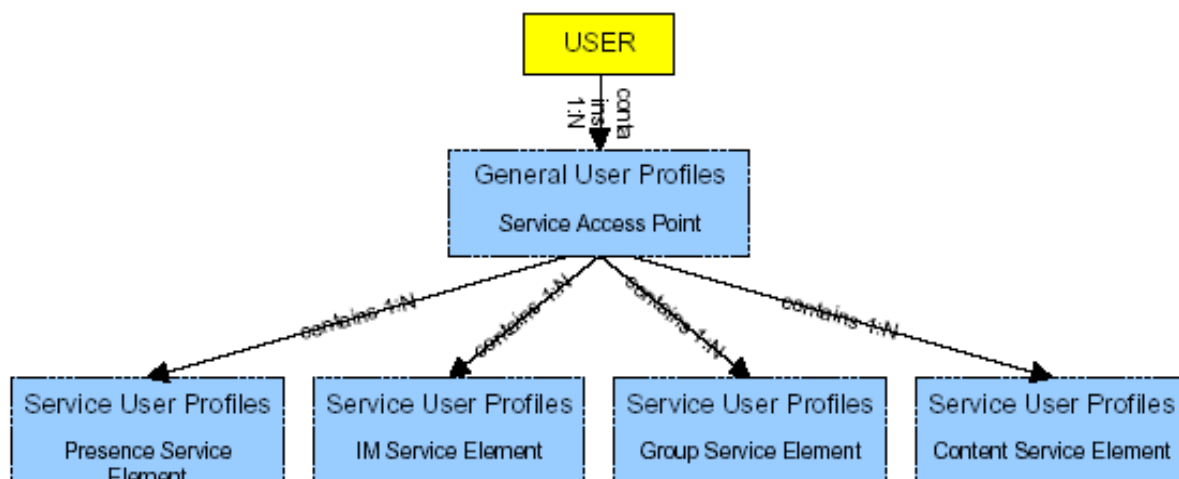
2. **Instant Messaging Service Element:** provê ao sistema a possibilidade de enviar e receber mensagens instantâneas. Estas mensagens podem ser enviadas ou recebidas de um usuário WV ou até mesmo um usuário de outros sistemas de Instant messages. O WV suporta vários tipos de Instant messages, como Texto, Vídeo, Gráfico ou Som.
3. **Group Service Element:** provê ao usuário a possibilidade de gerenciar e configurar as características de cada grupo da sua Lista de Contatos. Estes grupos podem ser Públicos ou Privados.
4. **Content Service Element:** provê a possibilidade de compartilhar fotos, documentos ou figuras entre os usuários WV.

### 5.2.2 SAP

O SAP tem a função de interface entre o WV Presence e os usuários.

Dentre as funções do SAP, podemos citar:

1. **Autenticação e Autorização:** a autenticação é usada para verificar a identidade de uma entidade (usuário, rede ou alguma aplicação). Já a função da autorização será verificar o quê a entidade autenticada anteriormente está autorizada a fazer.
2. **Service Discovery e Service Agreement:** o Service Discovery identifica o total de serviços que o usuário pode utilizar Depois, o Service Agreement provê o cliente destes serviços.
3. **User Profile Management:** descreve como o usuário deseja gerenciar e interagir com seus serviços de comunicações. O User Profile Management permite que uma aplicação recupere e atualize o user profile, conforme modelo abaixo:



**Figura 20 - User Profiles**

### 5.2.3 CLIENTE WV

O Cliente Wireless Village nada mais é do que um usuário embutido no sistema, com um terminal móvel.

Cientes de diferentes fornecedores tem diferentes aparências, e não tem exatamente a mesma funcionalidade. A vantagem do Cliente WV é que apesar da diferença entre os clientes, estes poderão estar completamente interoperantes um com o outro, através do PROTOCOLO CLIENTE SERVIDOR.

### 5.2.4 CLIENTE CLI (INTERFACE DE LINHA DE COMANDO)

É uma interface de linha de comando onde o cliente utiliza mensagens de texto para se comunicar com o Servidor WV. Um exemplo de Cliente CLI é o serviço de SMS.



## 5.3 INTERFACES E PROTOCOLOS

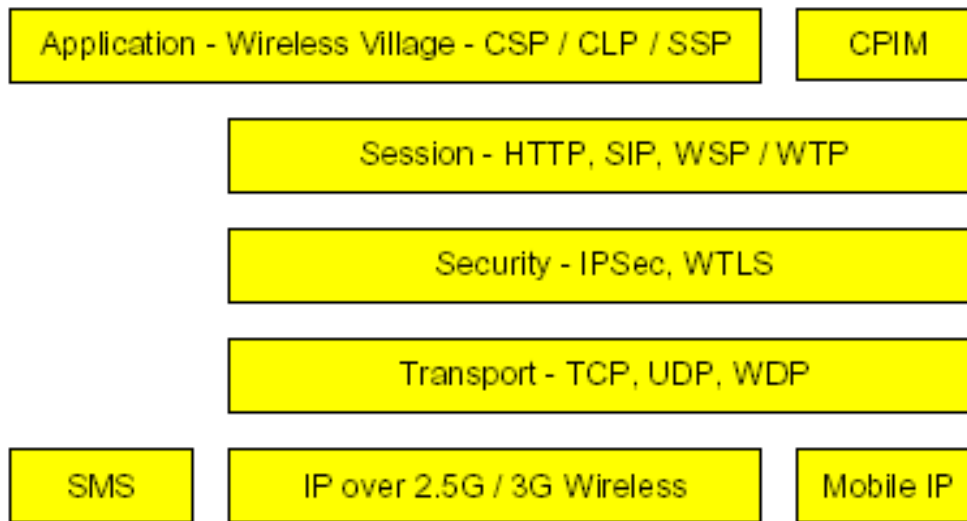


Figura 21 - Interfaces E Protocolos

### 5.3.1 PROTOCOLO CLIENTE SERVIDOR (CSP)

Permite que os clientes dos terminais móveis e fixos acessem o Servidor WV.

### 5.3.2 PROTOCOLO DE LINHA DE COMANDO (CLP)

Provê acesso ao Servidor WV através da linha de comando dos Clientes WV.

### 5.3.3 PROTOCOLO SERVIDOR-SERVIDOR (SSP)

Tem a função de conectar os Servidores WV.

## 5.4 IMPS

As quatro principais características do IMPS são:

- Presença
- Instant Messaging
- Grupo
- Conteúdo Compartilhado



Instant Messaging Service é um serviço onde os usuários podem, através de um software de conversação, se comunicar com a internet utilizando mensagens de texto curtas, ou até mesmo alguma aplicação multimídia.

A grande utilidade do Instant Message está na mesclagem deste com o Serviço de Presença. Aí, é possível que um usuário tenha um grupo de pessoas, amigos ou parceiros cadastrados, e também é possível que o seu status (on line, off line, etc) e sua disponibilidade para diálogo sejam alterados para um grupo todo ou somente para algumas pessoas do grupo.

A combinação do Mensagem Instantânea, Grupos, Presença e Conteúdo Compartilhado é chamada de IMPS.

#### **5.4.1 PROTOCOLOS**

Muitos produtos IMPS ao redor do mundo utilizam diferentes protocolos de comunicação. Estas diferenças são complicadas pela necessidade de uma solução comum para a utilização do IMPS em redes wireless.

A necessidade de um protocolo de comunicação comum para utilizações IMPS está impedindo o crescimento de serviços para unidades móveis (celulares), como serviços de mensagens móveis e informações sobre presença.

A solução do WV é uma simetria de comunicação para IMPS é a comunicação entre um dispositivo móvel e o IMPS de outra rede qualquer.

Nenhum protocolo existente hoje em dia proporcionam tamanha conectividade.

#### **5.4.2 ESPECIFICAÇÕES WV PARA COMUNICAÇÃO**

As especificações WV são abertas à indústria que suportam uma variedade de protocolos de transporte e características robustas para o IMPS. Estas características são baseadas em XML (Extensible Markup Language).

A escolha por XML está no fato do XML ser a linguagem que está crescendo mais rapidamente em termos de representação de dados estruturados através da Web.

Além disso, XML é uma linguagem que tem sido largamente implementada e bem testada. Isso irá assegurar uma fácil implementação e teste de interoperabilidade.





Com a solução WV para IMPS, será possível que qualquer unidade móvel, por mais antigo que seja o protocolo IMPS utilizado, consiga se comunicar com tecnologias mais recentes, e vice versa.

### **5.4.3 OBJETIVOS WV**

Enfim, o objetivo da WV é a criação de um protocolo aberto, comum e que torne possível a interoperabilidade do IMPS entre as unidades móveis e qualquer outra estação, seja ela fixa ou móvel, de qualquer outra rede.

## **5.5 SSP (PROTOCOLO SERVIDOR-SERVIDOR)**

O WV SSP permite a comunicação entre diferentes domínios de serviços IMPS.

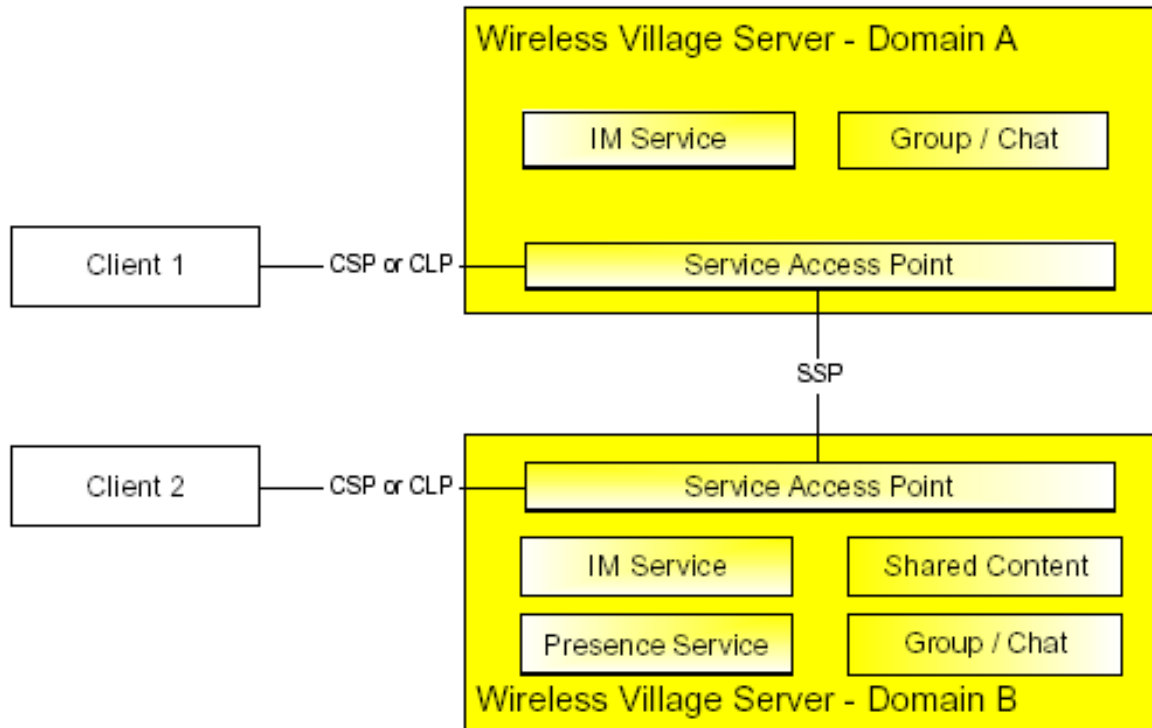
### **5.5.1 MODELO DE INTEROPERABILIDADE DO SSP**

O termo Home Domain é o domínio onde o cliente está cadastrado, autenticado e autorizado a utilizar os serviços IMPS.

O termo “Primary Service Element” (PSE) é o primeiro elemento de serviço do serviço IMPS para um cliente.

O PSE pode estar no Home Domain do cliente ou em um domínio remoto.

O Modelo de Mínima interoperabilidade pode ser definido conforme o diagrama abaixo:



**Figura 22 - Modelo de mínima interoperabilidade do SSP**

No modelo acima, o cliente 1 está localizado no Home Domain A e o cliente 2 está localizado no Home Domain B. O Domínio A implementa o IM e o Serviço de Grupo, enquanto que o Domínio B implementa todas as configurações possíveis para o elementos WV.

O conjunto comum de serviços do Servidor WV é IM e Grupo. Então, na configuração/modelo acima os clientes 1 e 2 estão interagindo pelos domínios através da mínima configuração possível de interoperabilidade entre ambos.

Já no modelo abaixo, esta interoperabilidade ocorre de uma maneira um pouco diferente.

Neste modelo, chamado de máxima interoperabilidade, o cliente 1 está localizado no Home Domain A, e o cliente B está localizado no Home Domain B.

O Domínio A implementa os Serviços de Presença e Grupo, enquanto o domínio B implementa o IM e o Conteúdo Compartilhado.

Este modelo de interoperabilidade do WV permite que os clientes 1 e 2 utilizem uma completa configuração das suas características e interajam entre si através do protocolo SSP.

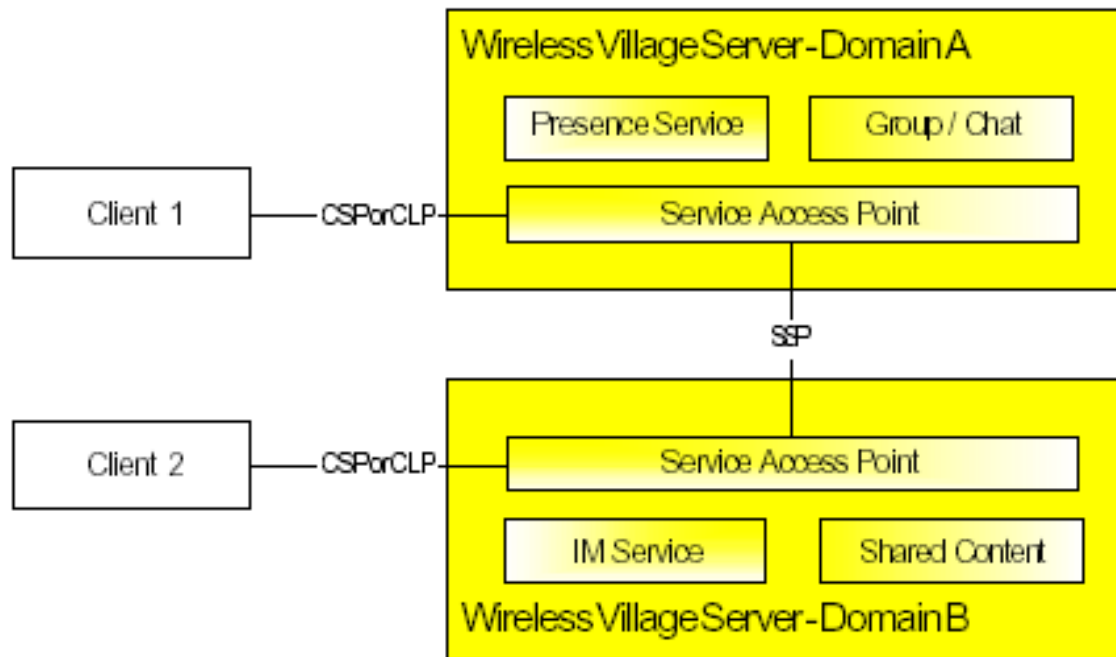


Figura 23 - Modelo de Máxima Interoperabilidade do SSP

### 5.5.2 ROTEAMENTO NOS SERVIÇOS SSP

A troca de informações ou mensagens entre os Domínios WV é, normalmente feita através de uma conexão direta via SSP.

Entretanto, WV suporta o roteamento de mensagens entre os Home Domains e o PSE.

O roteamento SSP entre os domínios é baseado nas regras de IOP (Interoperabilidade) e na “negociação” de permissões entre os domínios.

Depois que as negociações de permissões são feitas entre os domínios, cada domínio deve estar apto para rotear e retransmitir os serviços ao longo do caminho.

A tabela de roteamento é criada baseada nessa “negociação de permissões” entre os Servidores WV.

Cada WVS possui uma lista estática de conexões diretas de outros WVS's. Esta lista especifica o que foi acordado (negociado) entre os domínios para poder ser passado adiante para uma das conexões diretas dos WVS's.

Exemplificando, podemos citar como exemplo o seguinte caso:



Imaginemos dois usuários localizados no mesmo Home Domain, como mostra a figura abaixo:

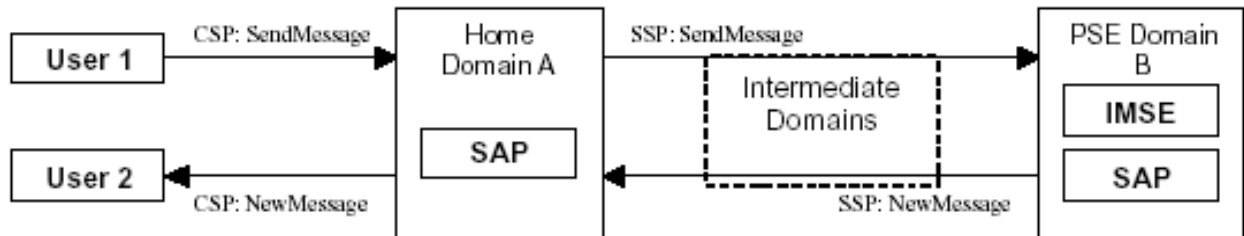


Figura 24 – exemplo de comunicação no mesmo home domain

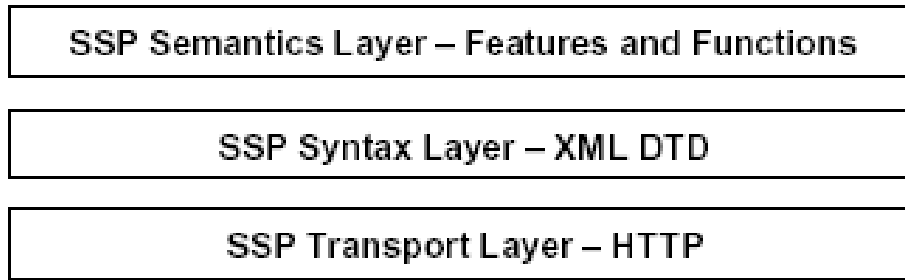
Os clientes 1 e 2 estão localizados no Domínio A. O IM PSE está localizado no domínio B. Os domínios A e B estão conectados através de alguns domínios intermediários. A transação de uma mensagem do cliente 1 para o cliente 2 é feita da seguinte forma:

```
C1 -> DA: CSP-SendMessage
DA -> DB: SSP-SendMessage (through intermediate domains via routing)
DB -> DA, SSP-NewMessage (after checking block list etc.)
DA -> C2, CSP-NewMessage
```

Cada domínio intermediário irá retransmitir a SSP-SendMessage para o próximo host, baseados na lista de negociações que foram criadas entre os mesmos.

### 5.5.3 CARACTERÍSTICAS DO PROTOCOLO SSP

O protocolo SSP é dividido em três camadas principais.



**Figura 25 – Divisão do SSP**

SSP Semantics Layer define a configuração das características e funções que o SSP pretende endereçar no modelo de total interoperabilidade entre os WV domains.

SSP Syntax Layer define a linguagem de comunicação para que os WV SAP's compreendam a informação entre os WVS's e conclua a interoperabilidade das características e funções definidas no SSP Semantics Layer. Nesta parte do protocolo SSP serão definidas as especificações para o uso do XML DTD.

SSP Transport Layer define o método de comunicação que transporta a “linguagem de comunicação” entre os WV SAP's para concluir a interoperabilidade. Por especificação, este método é através de HTTP.



## 6. Introdução ao Protocolo SSP

### 6.1 DEFINIÇÕES BÁSICAS

**Seção:** É o período em que os servidores WV's interagem entre si, para que o provedor de serviços forneça ao requisitante o que foi pedido. Cada "Provider Server" mantém uma seção para cada "Requestor Server". Então, existem duas seções entre 2 domínios, uma para negociar quais serviços do IMPS serão requisitados, e outra para fornecer os serviços.

**Transação:** É uma seqüência de operações para completar uma específica função ou característica do SSP. As transações SSP incluem transações simples, transações duplas e transações multi-percurso.

**Message:** É definida como uma sintaxe de unidades em uma interação. Uma mensagem SSP pode conter algumas Meta-informações, incluindo informações sobre o protocolo (versão) , sobre a seção (Session-ID), sobre a transação (transaction-ID), e informações de atributos (one-way / two-way, request / response). Todas as transações durante uma seção devem conter o mesmo Session-ID.

### 6.2 PARES DE SEÇÕES VS. CONEXÕES

Existem duas seções entre dois domínios. Cada domínio mantém uma seção para prover um serviço e outra onde são negociados os serviços do IMPS.

Existem, pelo menos, duas conexões físicas, chamadas pares de conexão, que levam os serviços para a seção correta.

As conexões físicas levam os serviços requisitados do Servidor Requisitor para o Servidor Provedor somente em uma direção, ou a notificação do Servidor Provedor para o Requisitor, na outra direção.

As conexões são reutilizáveis, ou seja, cada seção pode utilizar algumas ou todas as suas conexões para transportar as suas transações.

### 6.3 ENDEREÇAMENTO

No WV, as entidades endereçáveis são:



- Usuário
- Lista de Contatos
- Grupo (public or private)
- Conteúdo (public or private)
- Message
- Service (SSP)

**User Addressing and Global-User-ID:** Global-User-ID é um número único que o SSP usa para identificar um usuário no WV. Refere-se tanto ao endereço eletrônico na internet como ao número do celular do usuário.

Se fizer referência ao número do telefone móvel do usuário, este deverá sempre começar com a string '+', como no exemplo abaixo:

Local-User-ID:   wv:Jon.Smith

wv:+358503655121	→	Número do telefone móvel
wv:0503655121	→	Endereço na Internet

Global-User-ID:   wv:Jon.Smith@imps.com

wv:+358503655121@imps.com

wv:0503655121@imps.com

Os usuários podem ser identificados de acordo com os três seguintes parâmetros:

- **ScreenName:** É a combinação do nome que o usuário escolhe numa seção de grupo com o Group-ID. O usuário pode ter diferentes ScreenNames em diferentes ocasiões bem como em diferentes grupos.
- **NickName:** É um nome que é usado internamente em um cliente para esconder o UserID dos contatos.
- **Alias:** É um nome que um usuário sugere a outros usuários que seja utilizado como nickname.



**Contact List Addressing and Contact-List-ID:** Contact-List-ID é o número que o SSP usa para identificar cada Lista de Contatos de cada usuário. A sintaxe do Contact-List-ID é mostrada abaixo:

ww:john/colleagues@imps.com

ww:/managers@imps.com

**Group Addressing and Group-ID:** Group-ID é um número que o WV utiliza para identificar cada grupo. Exemplos:

ww:john/mygroup@imps.com

ww:/technical\_forum@imps.com

**Client Addressing and Client-ID:** O Client-ID identifica o Cliente WV como uma aplicação, permitindo o múltiplo acesso para o mesmo usuário e também a comunicação direta de aplicação para aplicação.

## 6.4 TIPOS DE DADOS

O SSP define quatro tipos básicos de dados: String, Char, Integer e Boolean, e três tipos de estruturas de dados, Enum, DateTime e Structure.

### String

De acordo com as especificações WV, todo elemento de informação é uma String, a menos que seja especificado.

Pela definição, uma String é um elemento composto por uma seqüência de caracteres.

### Char

Um elemento tipo Char é um caractere simples codificado em UTF-8.





## **Integer**

Um elemento inteiro é um número decimal de 32-bit que pode variar de 0 a  $(2^{32} - 1)$ .

## **Boolean**

É um tipo de elemento que pode assumir os valores “True” ou “False”.

## **Enum**

É um elemento que possui valor pré definido.

## **DateTime**

É um elemento que segue as especificações do ISO-8601 e é expresso como string. Este elemento pode estar sem indicação de zona, mas também pode estar no formato Coordinated Universal Time (UTC) ou hora local, como por exemplo:

Local time: 20011019T125031

UTC: 20011019T095031Z

## **Structure**

É a combinação de todos os outros elementos como especificado.

## **6.5 CARACTERÍSTICAS E FUNCIONALIDADES**

### **6.5.1 SEGURANÇA**

Quando se fala de segurança no WV, estamos falando não só da comunicação servidor-servidor, mas também, da certeza de que o dado enviado e/ou recebido pelo usuário final em um dado domínio IMPS foi, de fato, originado do servidor daquele domínio.

### **6.5.2 MODELOS DE CONFIANÇA**

Um modelo de confiança é criado entre o domínio A e o domínio B se e somente se ambos tiverem se autenticado mutuamente.



### **6.5.3 CONTROLE DE ACESSO**

A autenticação e autorização entre servidores de domínios diferentes é realizada pelo controle de acesso entre cada servidor.

O controle de acesso está presente quando falamos de três tipos de comunicação:

- **ON LINE SESSION MANAGEMENT:** trata-se de uma **CALLBACK** inicial somente para dizer que a conexão foi feita, ou seja, é como se fosse um Mensagem de Status.
- **TRANSACTION MANAGEMENT:** a transaction management utiliza o controle de acesso em cada autenticação de transação. Se baseia na informação dos elementos da transação em cada serviço requisitado e em cada serviço respondido.

### **6.5.4 SEGURANÇA NO TRANSPORTE**

Sempre que possível, o transporte dos dados é feito com criptografia, como por exemplo a utilização de protocolos SSL / TLS, PGP, PKI e outros certificados digitais.

### **6.5.5 SEGURANÇA NOS DOMÍNIOS INDIVIDUAIS**

Um simples domínio de IMPS é, geralmente, protegido com Firewalls.

## **6.6 TRANSACTION MANAGEMENT**

### **6.6.1 TRASAÇÃO INVÁLIDA**

Uma transação é considerada válida se ela se completa dentro de um determinado período de tempo.

O tempo de validade de uma transação é a soma do atraso da rede, mais o tempo de processamento da transação mais o ajuste de offset.

Estes três elementos devem ser configurados em cada domínio pelo operador responsável.

O operador configura e define os valores destes parâmetros baseado nas características da rede, do hardware e do software que estão sendo utilizados.

Portanto, uma transação pode ser considerada inválida se ela não é completada dentro deste período pré-determinado.

### **6.6.2 TRANSAÇÃO DESCONHECIDA**

Uma transação pode ser considerada desconhecida por um dos seguintes motivos:



- A mensagem de requisição tem um erro de sintaxe, ou seja, o XML não está bem formatado, XML inválido ou erro dos valores nos dados;
- Um dos elementos de Meta-informação é inválido. Dentre os elementos de Meta-Informação mais importantes, podemos citar:

<b>Informação dos elementos</b>	<b>Tipo</b>	<b>Descrição</b>
Originados do Cliente	Booleano	Indica se a Requisição é originada do Cliente ou do Elemento de Serviço
ID da Seção	String	Identifica a Seção gerenciada pelo Servidor Provedor
ID da Transação	String	Identifica se a transação é originada do Servidor Requisitor ou Provedor
ID do Serviço	String	Identifica o domínio do inicializador
ID do Usuário	String	Identifica o usuário representado pelo Domínio do Servidor Requisitor. Está presente se o requisitor é originado de um cliente
ID do Cliente	String	Identifica o ID do cliente do usuário.

**Tabela 2 – Transação desconhecida**

- O serviço requisitado refere-se a algo que não corresponde ao que havia sido negociado entre o Requisitor e o Provedor;
- O serviço a ser cedido não pode ser associado com o serviço requisitado.

### **6.6.3 GENERAL STATUS CODE**

Dentre todas as transações SSP, os códigos de status retornados são os seguintes:



- Continue (100) – for all complementary transactions
- Queued (101) – for all complementary transactions
- Started (102) – for all complementary transactions
- Server queued (104)
- Bad Request (400)
- Service not supported (405) – for all complementary transactions
- Service Unavailable (503)
- Invalid Timeout (504)
- Service not agreed (506) – except transactions required for the service agreement
- Internal Server Error (500)
- Invalid server session (620) – except transactions allowed outside of a session
- Multiple errors (900)
- Not logged in (604)
- Bad parameter (402)
- Forbidden (403)
- Not found (404)

## **6.7 GERENCIAMENTO DE SEÇÕES**

O Gerenciamento de Seções do SSP inclui Estabilização de Seções, Término de Seções e Manutenção de Seções. O CALLBACK é utilizado para a Estabilidade de uma Seção. Em todo o restante do Gerenciador de Seções é utilizado o Controle de Acesso (ACCESS CONTROL).

## **6.8 ESTABILIDADE DE UMA SEÇÃO**

Os pré requisitos para que ocorra uma Estabilidade de Seção são:

- O Host-ID A representa um único ponto de acesso no Domínio A;
- O Host-ID B representa um único ponto de acesso no Domínio B;
- Uma negociação para Offline Configuration deve ter sido feita entre os domínios A e B;



- No Servidor A, a identidade do Servidor B é registrada com, no mínimo { B-Host-ID, B-Service-ID, B-password }, sendo que um password vazio para B é aceitado;
- No Servidor B, a identidade do Servidor A é registrada com, no mínimo { A-Host-ID, A-Service-ID, A-password }, sendo que um password vazio para A também é aceitado;

Os pontos básicos de uma conexão são:

- O Servidor A inicializa uma conexão 1 com o Servidor B baseada nos registros que A possui de B, contendo os parâmetros { A-Service-ID, A-secret-token }.
- O Servidor B analisa o parâmetro { A-Service-ID } enviado por A e compara com os dados gravados no seu registro. Se estes não forem encontrados ou se não houver simetria, a conexão é fechada.
- O Servidor B inicializa a conexão 2 com o Servidor A, contendo os parâmetros { B-Service-ID, Bsecret-token }.
- O Servidor A por sua vez, analisa o parâmetro enviado por B nos seus registros. Se este não for encontrado, ou se houver falta de simetria, a conexão é fechada.
- O Servidor A envia uma Requisição de Login para o Servidor B através da conexão 1, contendo { A-Service-ID, A-password-digest }. O “A-password-digest” é gerado através do A-password e do B-secret-token baseado num esquema de agrupamento contido nos registros de ambos os servidores.
- O Servidor B envia então uma Requisição de Login para o Servidor A através da conexão 2, contendo { B-Service-ID, B-password-digest }. O “B-password-digest” é gerado através do B-password e do A-secret-token baseados num esquema de agrupamento, como descrito no item anterior.
- O Servidor B verifica então o A-password-digest. Se a verificação falhar, a conexão é cancelada por B.
- O Servidor B responde ao Servidor A com a Requisição de Login através da conexão 2, contendo o Status da transação e algumas informações da nova seção mantida por B. O LoginResponse pode conter uma lista opcional de Host Names, chamada de **Redirect List**.
- O Servidor A verifica o B-password-digest. Se a verificação falhar, a conexão é cancelada por A.



- O Servidor A responde ao Servidor B com o LoginResponse através da conexão 1, contendo o Status da transação e informações da nova seção mantida agora por A. O LoginResponse pode conter uma lista opcional de Host Names, chamada de **Redirect List**.

Em termos ilustrativos, o Login Transaction é representado pelo diagrama abaixo.

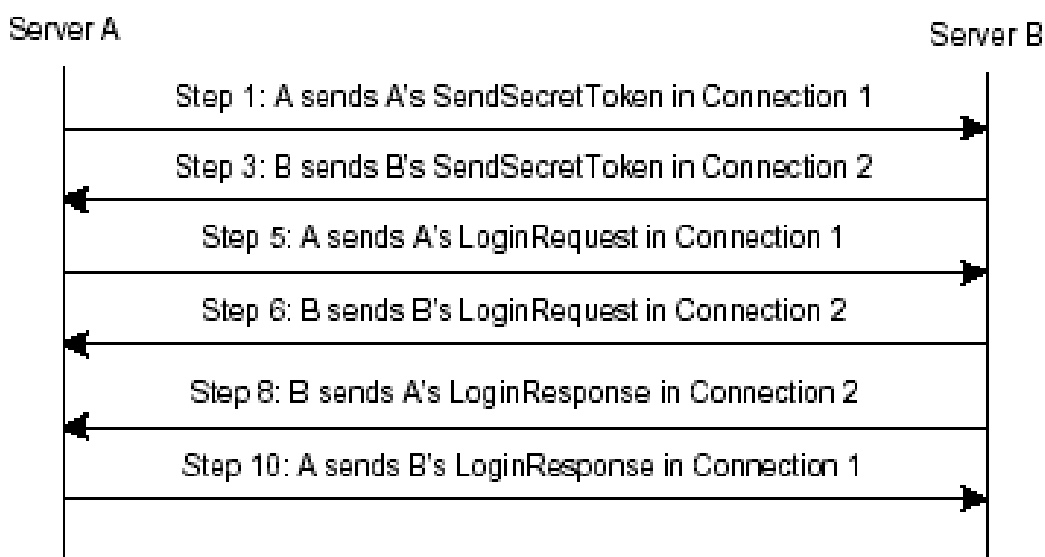


Figura 26 – Transação de login

Uma secret-token é uma String Randômica gerada por cada servidor durante uma conexão. Após os 10 passos descrito acima, se tudo ocorreu com sucesso, os dois domínios estarão autenticados entre si.

É formado então o que chamamos de “Connection Pair” entre os Servidores.

Agora, os dois domínios estão prontos para negociar serviços entre si.

## 6.9 MANUTENÇÃO DE UMA SEÇÃO

A seção é mantida aberta de acordo com a necessidade de ambos os Servidores. Inicialmente, durante a Estabilização de uma Seção, é negociado um período de duração da seção entre os 2 domínios. Mas este período pode ser renegociado para um novo intervalo de tempo durante a comunicação.

Se, durante um determinado intervalo de tempo, nenhuma comunicação ocorrer entre os domínios, a seção é fechada.



## 6.10 FINALIZAÇÃO DE UMA SEÇÃO

A seção pode ser finalizada, tanto pelo Servidor A como pelo Servidor B, a qualquer instante. Qualquer um dos Servidores podem pedir um Logout. Neste caso, o término de uma seção é feito por duas transações, uma de Logout e outra de Disconnect, como mostra o diagrama abaixo:

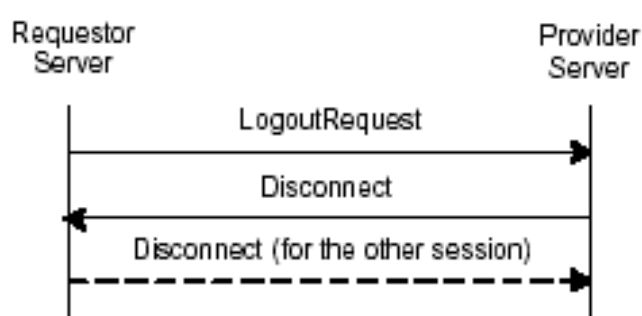


Figura 27 – Transação de Logout

Isto pode ser feito para garantir a segurança das informações que estão trafegando entre os domínios.

Vários podem ser os motivos para que uma seção seja finalizada. Uma seção pode ser terminada normalmente ou anormalmente.

Dentre os motivos normais, podemos destacar:

- A seção expira;
- O serviço expira;

Dentre os motivos anormais, destacamos:

- Queda na conexão;
- Seção inválida;



Depois que a seção terminou, os Servidores podem restabelecer uma nova seção, desde que sejam seguidos os mesmos 10 passos descritos anteriormente.

## 6.11 GERENCIAMENTO DOS SERVIÇOS

A árvore diagrama que melhor representa os serviços SSP pode ser definida conforme abaixo:

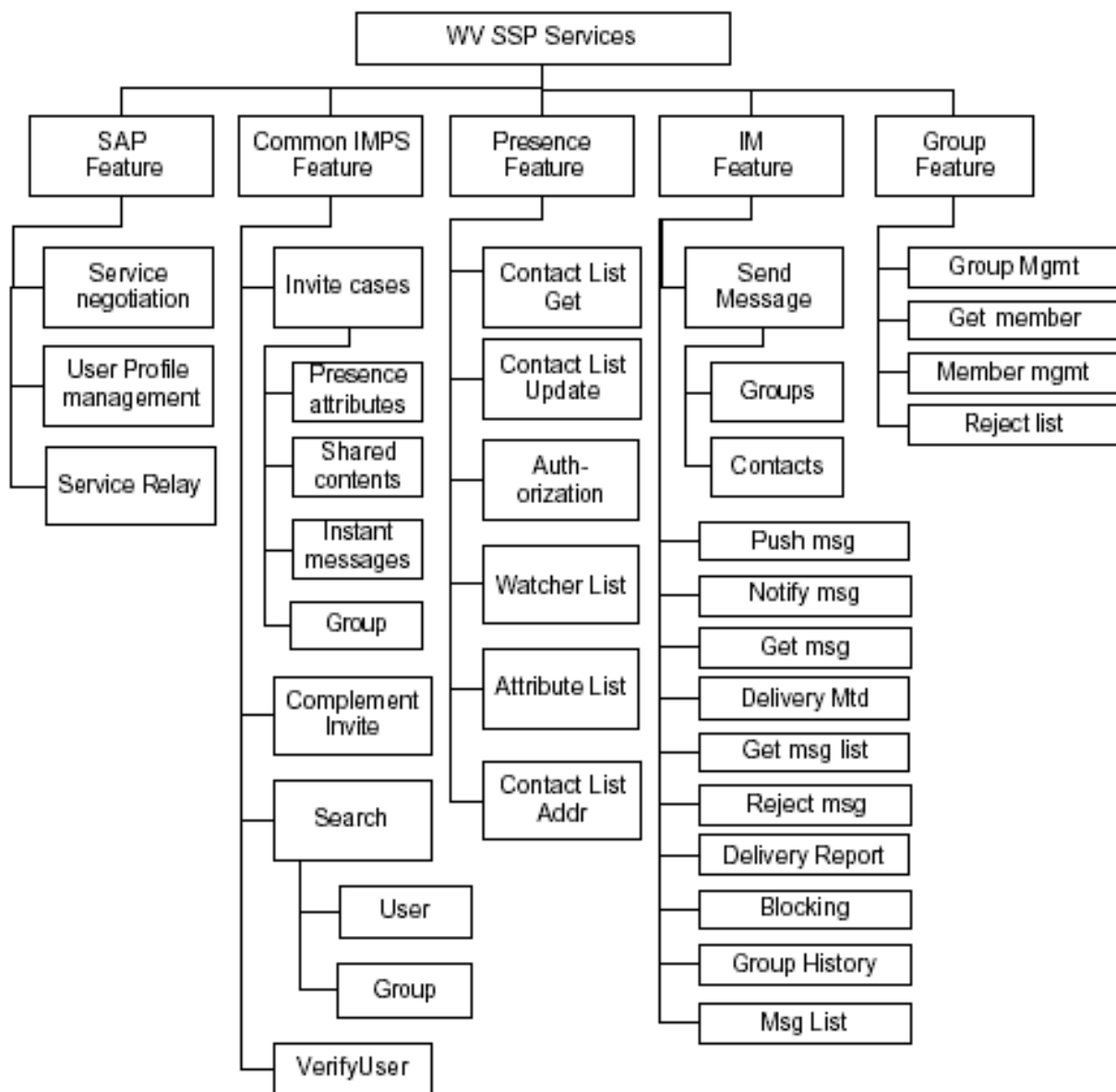


Figura 28 – Árvore representativa dos serviços SSP





## 6.12 ESTRUTURA DOS SERVIÇOS WV

O WV Services estão organizados na seguinte estrutura:

- **Features:** configuração associada às funcionalidades;
- **Functions:** define uma configuração relacionada a cada Transação para cada Feature;
- **Transactions:** define uma configuração relacionada às primitivas de cada função;
- **Information Elements:** é o mais baixo nível de construção dos blocos nas transações.

## 6.13 CARACTERÍSTICAS DO SAP

- O **Service Negotiation** inclui as seguintes transações:
  - ✓ GetAvailableService
  - ✓ ServiceIndication
  - ✓ SetServiceAgreement
- O **User Profile** inclui as seguintes transações:
  - ✓ GetUserProfile
  - ✓ UpdateUserProfile
- **Service Relay:** O SAP pode suportar Service Relay, inclusive roteamento.

## 6.14 CARACTERÍSTICAS COMUNS AO IMPS

- **Invite** inclui as transações de Invite/Cancel-Invitation. Todos os tipos de Invite suportados pelo WV podem ser inclusos na Lista de Serviços. Estes tipos de Invite são Presença, IM, Conteúdo Compartilhado e Grupo.
- **Complementary Invite** inclui as transações de Complementary Invitation / Cancel-Invitation.



- **Search** inclui as escolhas opcionais para a Pesquisa Geral. Todos os tipos de busca suportados pelo WV SSP podem ser inclusos na Lista de Serviços. Os tipos são Usuário e Grupo.
- **VerifyUser** inclui a transação de VerifyUserID.

## 6.15 CARACTERÍSTICAS DO PRESENCE

- **Contact List Get** inclui as seguintes Transações:
  - GetContactList
  - GetListMember
  - GetListProperties
- **Contact List Update** inclui as seguintes Transações:
  - CreateContactList
  - DeleteContactList
  - AddListMember
  - RemoveListMember
  - SetListProperties
- **Authorization** inclui as seguintes transações:
  - ReactiveAuthorizarion
  - CancelAuthorization
- **Watcher List** inclui a seguinte Transação:
  - GetWatcherList
- **Attribute List** inclui as seguintes Transações:



- CreateAttributeList
  - DeleteAttributeList
  - GetAttributeList
- **Contact List Addr** inclui as seguintes Transações:
    - Subscribe
    - UnSubscribe
    - GetPresence
    - UpdatePresence

## 6.16 CARACTERÍSTICAS DO IM

- **Send Msg** inclui as opções de escolha nas transações de SendMessage e ForwardMessage. Todos os tipos suportados pelo WV SSP ( Group-ID e ContactList-ID ) podem ser inclusos na Service List.
- **Push Msg** inclui as transações de PushMessage.
- **Notify Msg** inclui as transações de MessageNotification.
- **Get Msg** inclui as transações de GetMessage.
- **Delivery Mtd** inclui as transações de SetMessageDeliveryMethod.
- **Get Msg List** inclui as transações de GetMessageList sem as funcionalidades de grupo.
- **Reject Msg** inclui as transações de RejectMessage.
- **Delivery Report** inclui as transações de NotifyDeliveryStatusReport.
- **Blocking** inclui as transações de BlockUser e GetBlockedList.
- **Group History** indica se o IM Service Element suporta Group chat.
- **Msg List** inclui algumas opções de escolha para a transação de GetMessageList.

## 6.17 CARACTERÍSTICAS DE GRUPO



- **Group Mgmt** inclui as seguintes transações:
  - CreateGroup
  - DeleteGroup
  
- **Get Member** inclui as transações de GetJoinedMember.
  
- **Member mgmt** inclui as seguintes transações:
  - AddGroupMember
  - GetGroupMember
  - RemoveGroupMember
  - MemberAccess
  
- **Reject List** inclui as transações de RejectList.



## 7. Transações

Nas transações SSP dos servidores WV, sempre que um servidor ou cliente fizerem uma requisição, seja de serviço ou qualquer outra coisa, haverá uma resposta (STATUS) por parte do outro elemento receptor indicando que a requisição foi recebida.

Para ilustrar este fato, abaixo estão alguns tipos de transações e como estas são realizadas entre os servidores.

### 7.1 TRANSAÇÃO “ GetAvailableService ”



Figura 29 – Transação GetAvailableService

### 7.2 TRANSAÇÃO “ ServiceIndication ”



Figura 30 – Transação ServiceIndication



### 7.3 TRANSAÇÃO “ SetServiceAgreement ”



Figura 31 – Transação SetServiceAgreement

### 7.4 TRANSAÇÃO “GetUserProfile”

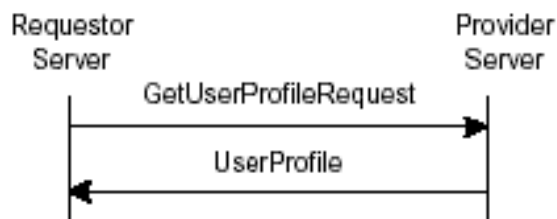


Figura 32 – Transação GetUserProfileRequest

### 7.5 TRANSAÇÃO “ UpdateUserProfile ”

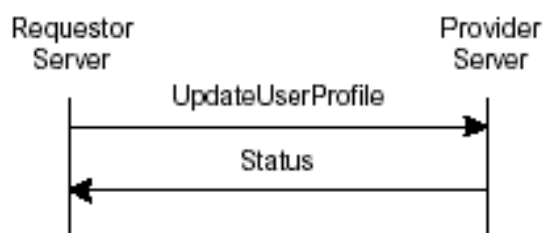


Figura 33 – Transação UpdateUserProfile



## 7.6 TRANSAÇÃO “ GeneralSearch “

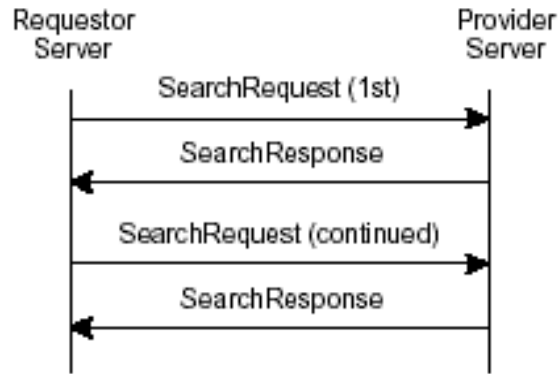


Figura 34 – Transação SearchRequest

## 7.7 TRANSAÇÃO “ StopSearch “

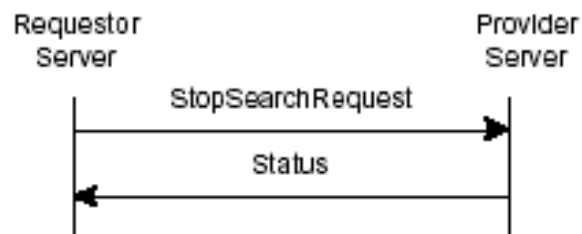
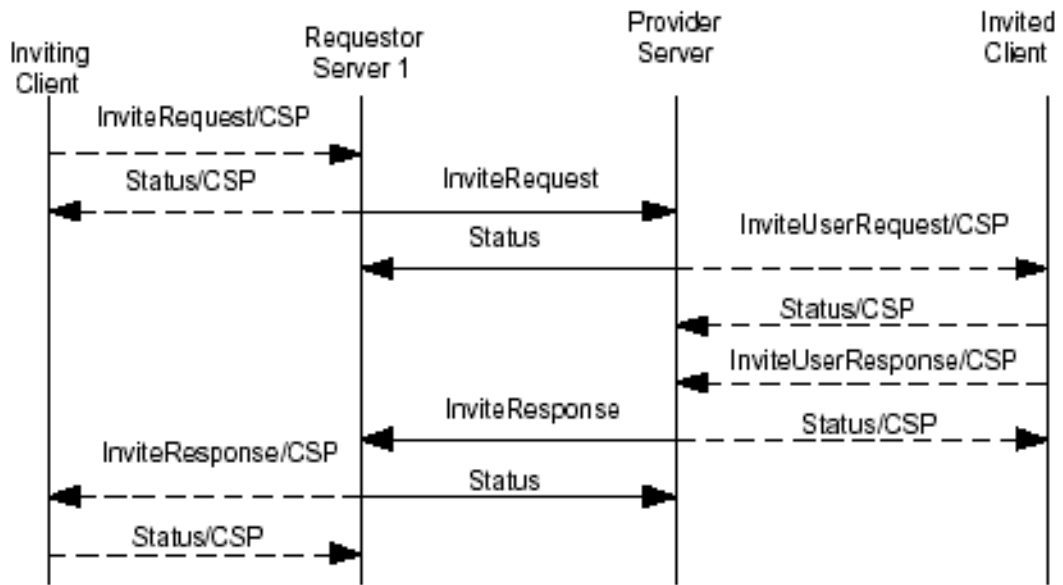


Figura 35 – Transação StopSearch

## 7.8 TRANSAÇÃO “ Invitation “



**Figura 36 – Transação de Convite**





## 8. Transmissão do Protocolo SSP

As mensagens SSP são carregadas e transmitidas via HTTP / HTTPS através do protocolo de transporte TCP.

Através de conexões físicas, as requisições de serviços são transportadas até o Servidor Requisitante, e as requisições de notificação até o Servidor Provedor.

Num SSP, a identificação de uma transação é sempre gerada pelo requisitor, ou seja, por quem está iniciando uma comunicação.

Concomitantemente, a Resposta SSP incluirá, na transação, o mesmo identificador que foi transmitido pelo requisitor.

Dessa forma, no SSP, uma transação de requisição e de resposta terão o mesmo identificador.

### 8.1 PARES DE CONEXÃO

O HTTP/HTTPS é um protocolo assimétrico, ou seja, ele sozinho não faz a comunicação. Ele precisa de duas conexões físicas TCP para realizá-la.

Dessa forma, uma conexão TCP é originada, como se fosse um cliente HTTP / HTTPS, do Servidor Requisitor para o Servidor Provedor. Essa conexão física seria então a conexão 1.

Similarmente, outra conexão TCP do Servidor Provedor para o Servidor Requisitor, também como se fosse um cliente HTTP/HTTPS, sendo então a conexão 2.

Quando há uma requisição de algum serviço, a transação SSP é iniciada pelo Servidor Requisitor, é transmitida pela conexão 1, e a transação de resposta a essa requisição vem pela conexão 2, originada do Servidor Provedor.

A transação SSP de requisição, bem como a resposta a essa requisição, são entregues apenas pelo HTTP/HTTPS POST request.

A SSP request em si é transmitida pelo HTTP/HTTPS body.

A HTTP/HTTPS POST reply é uma transação onde o body da mensagem é vazio, ou seja, ela existe apenas para cumprir as especificações.

Da mesma forma, quando o Servidor Provedor requisita uma transação de notificação, esta é transmitida pela conexão 2, e a resposta a esta transação de notificação é devolvida pelo Servidor Requisitor através da conexão 1.

Um Connection Pair pode ser usado para duas ou mais seções.



## 9. Protocolo CLIENTE-SERVIDOR – Seções e Transações

### 9.1 SEÇÃO WV

Uma Seção é a estrutura por onde os serviços WV são fornecidos ao Cliente WV.

Uma Seção WV é estabelecida quando o cliente faz o login no Servidor WV, e é finalizada quando o cliente desloga ou quando o SAP decide desconectá-lo. A Seção é identificada pelo Session-ID.

A autenticação do usuário é feita na fase do login, e é considerada válida por toda a seção.

No entanto, o Servidor WV pode, a qualquer momento, desconectar o Cliente, e pedir para que o mesmo faça novamente o login.

Durante uma Seção WV, tanto o Cliente como o SAP tem que manter o que se chama de “session context”, que contém informações dinâmicas dos serviços que o Cliente está utilizando.

### 9.2 ENDEREÇAMENTO

As entidades endereçáveis do WV são:

- Usuário
- Lista de Contatos
- Grupo de Usuários (private e public)
- Conteúdo (private e public)

### 9.3 FORMATO GENÉRICO PARA ENDEREÇAMENTO

De acordo com as especificações WV, a sintaxe genérica para os endereços são de acordo com os exemplos abaixo:

```
Address = "wv:" [User-ID] ["/" Resource] ["@" Domain]
Resource = Group-ID | Contact-List-ID | Content-ID
Domain = sub-domain * ( "." sub-domain)
```

Dos endereços acima, podemos dizer que o User-ID refere-se à identificação do usuário no WV. O Domain identifica o domínio do Servidor WV, e o Resource identifica se o domínio é público ou privado.



Quando o User-ID está presente sem o Resource, o endereço refere-se ao usuário. Se o User-ID está presente com o Resource, o endereço refere-se ao private resource do usuário. E se o User-ID não estiver presente, o endereço refere-se ao public resource do usuário dentro de um domínio.

Já o domínio é opcional. Se não estiver presente, o endereço irá se referir implicitamente ao home domain.

#### **9.4 CODIFICAÇÃO DO ENDEREÇAMENTO**

Se tivermos o usuário “\$albert” no domínio “bsb.com”, um endereço WV válido para ele seria:

wv:[%24albert@bsb.com](mailto:%24albert@bsb.com)

#### **9.5 USER ADDRESSING**

O WV utiliza o User-ID unicamente para identificar o usuário dentro do WV. O User-ID é sintaticamente equivalente a um endereço de e-mail

O User-ID pode se referir tanto ao tipo de endereço do usuário na internet como ao número do aparelho celular do mesmo.

Como foi visto anteriormente, se o User-ID referir-se ao número do aparelho celular do usuário, ele deverá começar com a string “+”. Caso contrário, se ele se referir ao tipo de endereço na internet, ele começa sem nenhum sinal.

#### **9.6 CONTACT LIST ADDRESSING**

Assim como no SSP, o CSP utiliza o Contact List ID unicamente para identificar alguma lista de contato de algum usuário. As sintaxes são definidas de acordo com os exemplos abaixo:

```
wv:john/colleagues@imps.com
wv:/managers
wv:john/friends
wv:/managers@imps.com
```

O mesmo ocorre para o Grupo de Usuários.

#### **9.7 CLIENT ADDRESSING**



O Client-ID é a única identificação do Cliente WV que o usuário costumeiramente utiliza. É esperado que o client-ID permita:

- Múltiplo acesso do mesmo usuário
- Comunicação direta de aplicação para aplicação.

## **9.8 GERENCIAMENTO DE TRANSAÇÕES**

Uma Transação WV é o mecanismo de comunicação entre o Cliente WV e o WV SAP.

O propósito de uma transação é a troca de dados entre as entidades ou a requisição de uma operação.

As transações podem ser originadas do Cliente WV ou do WV SAP.

Uma transação consiste de uma mensagem de requisição e de uma mensagem de resposta.

A resposta para uma request message pode ser obtida dentro de 20 segundos após inicializada uma transação.

Após este período a entidade requisitora pode reenviar a request message utilizando o mesmo identificador da transação.

OBS:- As transações são seqüenciais, ou seja, uma transação tem que ser completada antes que outra seja inicializada.

Múltiplas transações significam que duas ou mais transações são abertas no mesmo período de tempo.

Neste caso, estas transações poderão ser realizadas utilizando-se a mesma conexão para a transmissão, ou em conexões separadas.

## **9.9 STATUS PRIMITIVE**

Se algum erro de processamento ocorrer em alguma parte da transação, a outra parte receberá uma resposta de Status Primitive, em vez da Response Primitive esperada.

O Status Primitive também pode ser utilizado em caso de sucesso na transação, e não somente em caso de erro.

O resultado da estrutura sempre contém um dos Status Code contidos na especificação WV. Ele deve conter uma String de Descrição ou, se necessário, uma descrição detalhada explicando o erro ocorrido.



Este Resultado Detalhado pode ser usado com qualquer código de erro, mas o principal é quando o erro retorna algum dado específico sobre MessageID, UserID, GroupID, ou ScreenName.

Por exemplo, vamos supor uma mensagem que é enviada para vários recipientes, dentre os quais, um UserID é inválido. Neste caso, o Resultado pode ser um StatusCode 201 (Sucesso parcial). Já o DetailedResult para a mesma mensagem poderá retornar um Bad UserID juntamente com o StatusCode 501 (usuário desconhecido).

No caso de nenhuma parte de transação ter sucesso, o erro gerado não poderá ser descrito apenas por um StatusCode. Então, o código 900 (múltiplos erros) é retornado.

Algumas informações dos elementos do StatusPrimitive estão na tabela abaixo.

Informação dos elemntos	Tipo	Descrição
Tipo de mensagem	Status	Identifica a mensagem
ID da transação	String	Identifica a transação
ID da sessão	String	Sessão para ID da sessão
ID do Cliente	Structure	Identifica a requisição IM do cliente
Resultado	Structure	O resultado de um transação

Tabela 3 - Informações dos elementos do PollingRequest Primitive

## 9.10 LOGGING IN

As transações de Login podem ser representadas pelo diagrama abaixo.

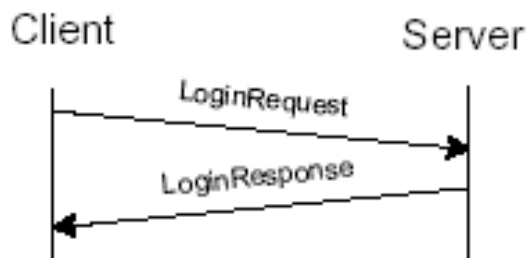


Figura 37 - Logging in



Após os Servidores processarem a requisição, ele envia um Login Response para o Cliente, contendo detalhes da operação Login.

Quando o Login não é bem sucedido no SAP, uma Mensagem de Status é enviada indicando que o login falhou, em vez de um Login Response Message.

O Cliente pode escolher entre acesso de controle two-way ou four-way.

Se a escolha for por two-way, o LoginRequest contém o elemento “Password-String” com o password em plain text. O Servidor irá responder com Sucesso, Falha ou um Pedido de Nova autorização.

Se a escolha for por four-way, o LoginRequest não contém nem o “Password-String” nem o “Digest-Bytes”. Em lugar destes, o LoginRequest irá conter o elemento “Supported Digest Schema”. O Servidor responde com uma objeção “nonce”, baseada no Digest Schema.

O Servidor também pode escolher por não usar nenhum esquema de autenticação. Neste caso, o elemento Digest Schema indica “PWD”, e o elemento Nonce não está presente no LoginRequest Primitivo.

## 9.11 LOGGING OUT AND DISCONNECTING

O usuário pode se deslogar dos serviços WV usando o Log out Request Message. Dessa forma, o Servidor irá responder com uma Disconnect Message.

A transação de Desconexão é representada como segue:

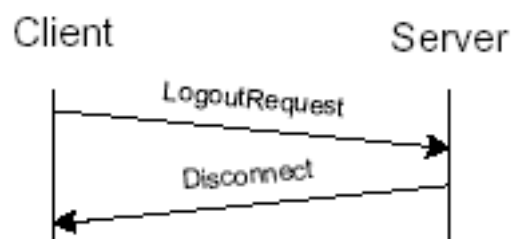


Figura 38 - Logging Out

Caso o usuário esteja participando de um ou mais grupos de discussão quando o Logout for completado, o Servidor irá automaticamente remover o usuário deste(s) grupo(s).

O Server pode desconectar o Cliente por qualquer razão que o mesmo ache necessário. O Servidor envia então Disconnect message contendo o Session-ID, e o Result contendo o código e o texto descritivo.



A partir das tabelas abaixo é possível visualizar os tipos de elementos utilizados nas transações de LogoutRequest e Disconnect.

Informação dos elementos	Tipo	Descrição
Tipo de mensagem	LogoutRequest	Identifica a mensagem
ID de transação	String	Identifica a transação
ID da sessão	String	Sessão ID para sessão

Tabela 4 - Elementos do Logout Request

Informação dos elementos	Tipo	Descrição
Tipo de mensagem	Disconnect	Identifica a mensagem
ID de transação	String	Identifica a transação
ID da sessão	String	Sessão ID para sessão
Resultado	Structure	Indica o código e a descrição do porquê da desconexão

Tabela 5 - Elementos do Disconnect

Da mesma forma que as transações de Logging In e LogoutRequest, as outras possíveis transações entre Cliente e Servidor são:

- **KEEP ALIVE**

É enviada pelo Cliente para zerar o timer da conexão, quando não houver tráfego de mensagens na rede.



Figura 39 – Transação Keep Alive



Informação dos elementos	Tipo	Descrição
Tipo de mensagem	KeepAliveRequest	Identifica a mensagem
ID de transação	String	Identifica a transação
ID da sessão	String	Sessão ID para sessão
Tempo de vida	Integer	Indica o novo tempo de vida para a sessão, em segundos

Tabela 6 - Elementos do Keep Alive

- GET SERVER PROVIDER INFO**  
 Requisita do Servidor informações sobre os tipos de Serviços que aquele Servidor pode fornecer.



Figura 40 - Transação do Get Service Provider Info

Informação dos elementos	Tipo	Descrição
Tipo de mensagem	GetSPInfoRequest	Identifica a mensagem
ID de transação	String	Identifica a transação
ID da sessão	String	Sessão ID para sessão
ID do cliente	Structure	Identifica a requisição do cliente

Tabela 7 - Elementos do Get Service Provider Info





## 10. Características

### 10.1 TRANSAÇÃO GERAL DE BUSCA

Um usuário pode procurar por usuários e grupos, podendo limitar o número de resultados enviados pelo Servidor.

Essa pesquisa é realizada utilizando-se uma lista com um ou mais Search-Pairs.

Um Search-Pair consiste em um Search-Element e um Search-String.

O Search-Element diz por quais propriedades o usuário ou grupo poderá ser encontrado pelo Search-String.

Esta transação funciona da seguinte forma.



O Cliente envia uma SearchRequest message ao Server incluindo o Search-Pair-List, o tipo de busca e o Search-Limit (número máximo de resultados de cada vez). O Servidor responde com o SearchResponse message, que inclui o resultado da pesquisa.

Se a pesquisa obtiver sucesso, o SearchResponse message poderá incluir também o Search-ID, o Search-Index, o Search-Findings (numero de itens encontrados) e o Search-Results (os dados atuais).

O diagrama que representa as transações está abaixo.

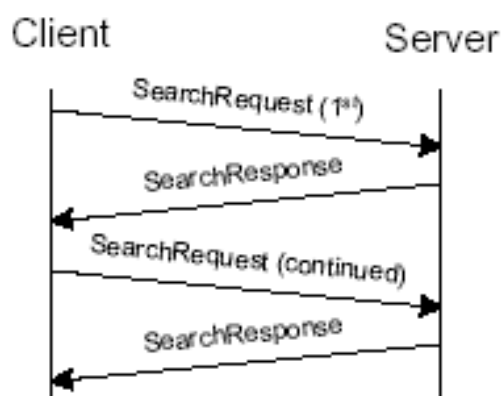


Figura 41 - Transação de Busca/Pesquisa

O resultado de uma pesquisa de usuário é sempre um User-ID. Da mesma forma, o resultado de uma busca por grupos é um Group-ID.

Os elementos de pesquisa por usuários estão listados abaixo.

Elemento de busca	String de busca
USER_ALIAS	A string de busca é uma substring do user alias
USER_ONLINE_STATUS	A string de busca dá um status online
USER_EMAIL_ADDRESS	A string de busca é uma substring de user email address
USER_FIRST_NAME	A String de busca é uma substring de user first name
USER_ID	A String de busca é uma substring de user ID
USER_LAST_NAME	A String de busca é uma substring de user last name
USER_MOBILE_NUMBER	A String de busca é uma substring de user mobile number



**Tabela 8 - Elementos de pesquisa por usuários**

Para pesquisa de grupos, os elementos de pesquisa são:

<b>Elemento de busca</b>	<b>String de busca</b>
GROUP_ID	A string de busca é uma substring do <u>group_ID</u>
GROUP_NAME	A string de busca é uma substring do <u>group_name</u>
GROUP_TOPIC	A string de busca é uma substring do <u>group_topic</u>
GROUP_USER_ID_JOINED	A String de busca é uma substring de user ID
GROUP_USER_ID_OWNER	A String de busca é uma substring de user ID

**Tabela 9 - Elementos de pesquisa por Grupos**

Após feita a pesquisa, o usuário pode optar por realizar uma nova pesquisa, ou então por parar com as buscas.

Quanto aos elementos de informação em uma busca e de uma resposta a uma busca, os nomes e tipos estão descritos nas tabelas abaixo.

<b>Informação dos elemntos</b>	<b>Tipo</b>	<b>Descrição</b>
Tipo de mensagem	Status	Identifica a mensagem
ID da transação	String	Identifica a transação
ID da sessão	String	Sessão para ID da sessão
Limite de busca	Integer	Indica aquantidade máxima de resultados que podem ser recebidos na 1º requisição
Indice de busca	Integer	Indica que os resultados podem estar iniciando em numa lista particular
Busca de ID	String	Apenas identifica o resultado da transação

**Tabela 10 - Elementos de Informação de uma SearchReques**



Informação dos elemntos	Tipo	Descrição
Tipo de mensagem	SearchResponse	Identificador de mensagem
ID da transação	String	Identifica a transação
ID da sessão	String	Identifica a seção
ID de Busca	String	Apenas identifica o resultado da transação
Resultado	Integer	Indica o número atual de resultados encontrados

Tabela 11 - Elementos de informação de uma SearchResponse

## 10.2 TRANSAÇÕES DE CONVITES

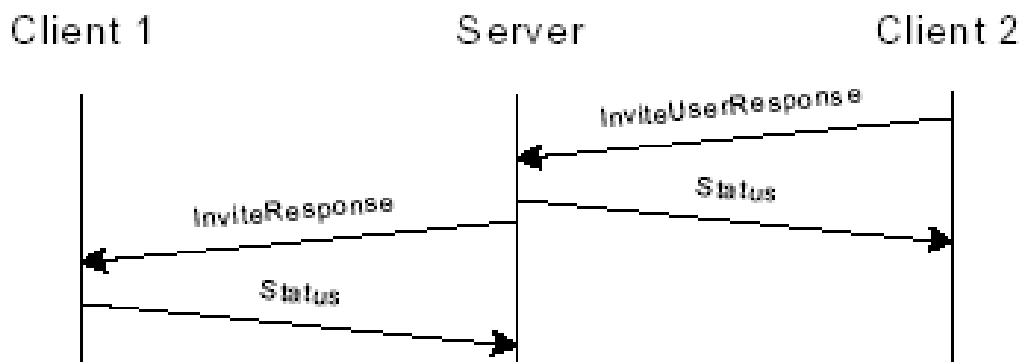


Figura 42 – Invite Transaction

O usuário convidado pode aceitar ou rejeitar o convite, através de uma InviteUserResponse message para o Servidor, contendo o ID do convite, o indicador de aceptância, o ID do assunto, e um pequeno texto de resposta opcional ao usuário que o convidou.



## 11. Características do Presence

### 11.1 LISTA DE CONTATOS

Dentre as propriedades da ContacList, as principais são:

- **DisplayName:** É uma String de texto fornecida pelo usuário que pode estar presente na interface do Cliente.
- **Default:** É uma String setada pelo usuário onde T (True) significa que uma determinada ContacList é default do usuário, e F (False) significa que não.

A transação do GetContactList é dada pelo diagrama abaixo.

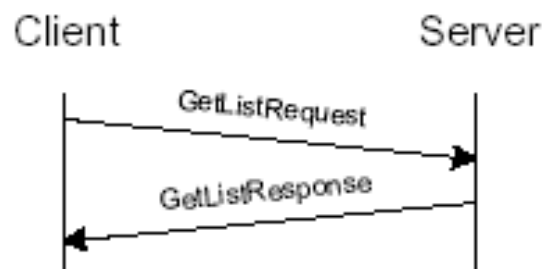


Figura 43 - Transação de GetListRequest



As transações de ContactList possíveis são:

<b>Primitive</b>	<b>Direction</b>
GetListRequest	Client → Server
GetListResponse	Client ← Server
CreateListRequest	Client → Server
Status	Client ← Server
DeleteListRequest	Client → Server
Status	Client ← Server
ListManageRequest	Client → Server
ListManageResponse	Client ← Server

Tabela 12 - Direções das classes primitivas do gerenciamento da Contact List

Quanto aos elementos da transação de GetListRequest e GetListResponse, estão descritos na tabela abaixo:

<b>Informação de Elementos</b>	<b>Tipo</b>	<b>Descrição</b>
Tipo da Mensagem	GetListRequest	Identificador de Mensagem
ID da Transação	String	Identifica a Transação
ID da Seção	String	Identifica a Seção

Tabela 13 - Elementos do GetListRequest

<b>Informação de Elemento</b>	<b>Tipo</b>	<b>Descrição</b>
Tipo de Mensagem	GetListResponse	Identificador da Mensagem
ID da Transação	String	Identifica a Transação
ID da Seção	String	Identifica a Seção
ID da Lista de Contatos	Structure	Lista de todos os ID's da lista de contatos

Tabela 14 - Elementos do GetListResponse

## 11.2 LISTA DE ATRIBUTOS



Através da lista de atributos o usuário configura os atributos de presença.

O usuário pode criar atributos específicos para um usuário ou para uma lista de contatos inteira.

A tabela abaixo mostra os elementos de uma transação de GetAttributeList.

<b>Informação de Elemento</b>	<b>Tipo</b>	<b>Descrição</b>
Tipo da Mensagem	GetAttributeListRequest	Identificador da Mensagem
ID da Transação	String	Identifica a Transação
ID da Seção	String	Identifica a Seção
Lista Default	Boolean	Indica se a lista de Atributos default é requisitada
ID da lista de contatos	Structure	Identifica a lista de contatos e devolve os atributos listados

**Tabela 15 Elementos do GetAttributeList**

### **11.3 GET PRESENCE TRANSACTIONS**

Se autorizado, um usuário pode obter informações de presença de outro usuário, através de uma GetPresenceRequest message ao servidor, contendo o User-ID ou o nome da lista de contatos.

O Servidor então responderá com uma GetPresenceResponse message, contendo o resultado da requisição e os atributos de presença.

## **12. Características do Instant Messaging**



## 12.1 TRANSAÇÃO DE ENVIO DE MENSAGENS

Um usuário pode mandar uma mensagem para outro usuário, para um grupo ou para um usuário de outro grupo.

Para isso, o Cliente manda uma `SendMessageRequest` para o Servidor que detém o serviço de Instant Messaging.

Esta mensagem contém o nome do recipiente e o nome do próprio usuário que está enviando a mensagem.

O Servidor responde com uma `SendMessageResponse`, que contém o Message-ID, para que o envio da mensagem seja verificado mais tarde.

Esta `SendMessageRequest` contém o elemento de `Message-Info`, que inclui o tipo de conteúdo, o tipo de criptografia usado na transação, o tamanho da mensagem, o remetente, a data e a hora.

O resumo da transação pode ser ilustrado pelo diagrama abaixo:



Figura 44 – Transação de envio de mensagem

## 12.2 DETERMINAÇÃO DO MÉTODO DE ENVIO DAS MENSAGENS

O usuário que irá receber a mensagem, indica através da `Capability-List` o método de entrega de mensagens que ele deseja que seja utilizado na sua configuração.

Um ponto importante deste método é o `AcceptedContentLength`, que indica o tamanho máximo da mensagem.

Após iniciada uma transação, o Cliente receptor pode alterar o método, a qualquer momento desejado.





---

### **12.3 MESSAGE DELIVERY TRANSACTIONS**

Após receber uma mensagem, o usuário pode retribuí-la através do NewMessage Transaction, retornando o Transaction-ID e o Message-ID da nova mensagem.

## **13. Transporte CSP através do Client-Server**



### 13.1 MODELO LÓGICO DAS COMUNICAÇÕES

O Transporte WV é dividido em dois canais: **Canal obrigatório de dados** (onde toda a troca de primitivas no CSP é feita) e um canal opcional CIR (Communication Initiation Request), utilizado para ativar o canal de dados quando o mesmo a conexão com o mesmo não estiver estabilizada.

Isto pode ser visualizado no diagrama abaixo:

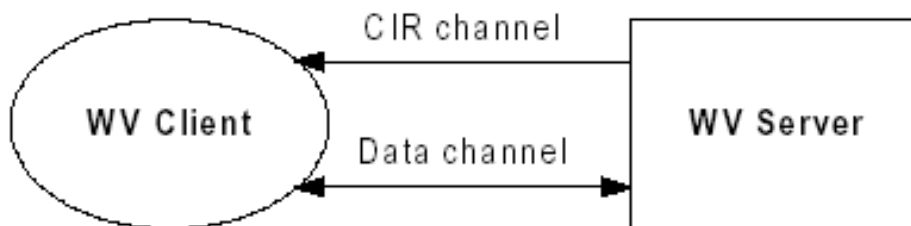


Figura 45 – Divisão do Transporte no WV

A necessidade e a utilidade do canal CIR depende do protocolo e do transportador utilizado no canal de dados.

Os protocolos utilizados no canal de dados são WSP, HTTP, HTTPS e SMS. No caso de WSP, HTTP e HTTPS a comunicação é assimétrica, ou seja, ela sempre se originará do Cliente para o Server.

Caso o Servidor necessite iniciar uma comunicação, existem duas possibilidades:

- O Servidor insere uma requisição de transação dentro de uma response message;
- O Servidor envia uma communication initiation request message através do canal CIR ao Cliente;

Na tecnologia SMS, tanto o Cliente como o Servidor podem originar as transações, e o canal de dados está sempre disponível. Com isso o canal CIR não é necessário.

### 13.2 SEÇÕES WV E GERENCIAMENTO DE CANAIS

As transações e as seções no modelo WV são independentes do WV Transport Binding. Isto porque uma seção WV não requer uma conexão permanente do canal de dados.



A conexão TCP/IP ou a seção WSP podem ser desconcertadas durante uma seção por razões de performance ou por qualquer outro motivo.

O canal CIR é do tipo connection-oriented, ou seja, a conexão tem que estar o tempo todo estabilizada.

O diagrama abaixo ilustra esta comunicação.

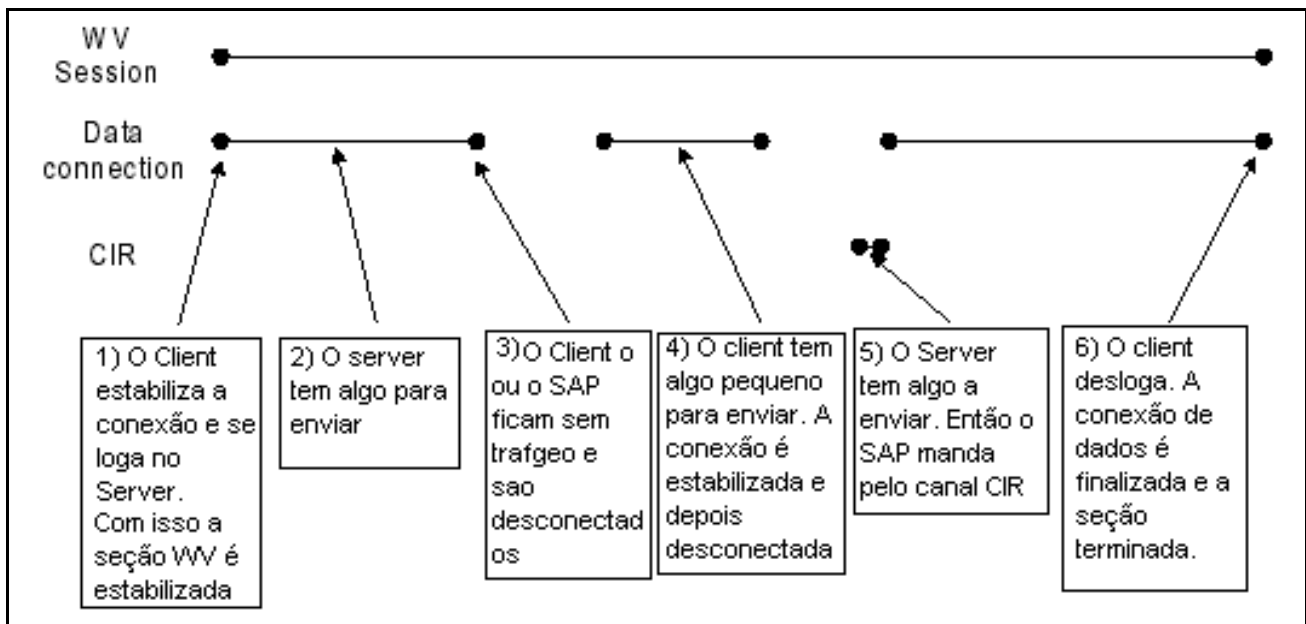


Figura 46 – Comunicação no WV



## 14. O Transporte WV para clientes móveis

O transporte SMS utiliza a tecnologia GSM short message para facilitar as transações WV.

Nesta tecnologia, tanto o Cliente WV como o Serviço Central de Mensagens Curtas devem suportar tanto MO (Mobile-Originating) short messages como o MT (Mobile Terminating) short messages.

Devido à simetria da tecnologia SMS, o canal CIR não é necessário.

Então, quando o Cliente deseja enviar uma mensagem, ele se comunica com o SAP através da SMSC.

A SMSC é apta a rotear as mensagens do WV Cliente para o WV SAP.

Quando o SAP envia uma mensagem para o Cliente, a SMSC encaminha a mesma, quando o recipiente é identificado como um número móvel.

O diagrama abaixo demonstra as relações entre o Cliente, o SMSC e o SAP.

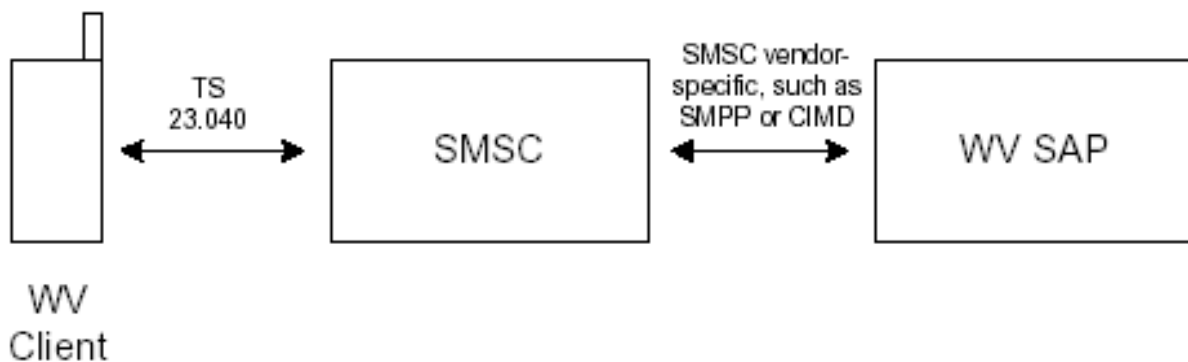


Figura 47 – Relações entre Cliente, SMSC e SAP

Os protocolos utilizados para a comunicação entre a SMSC e o SAP estão definidos no SMSC-vendor.

Como exemplo, é possível citar SMPP e CIMD.



## 15. CLP – Command Line Protocol

Antes de falar sobre o CLP, é necessário descrever um pouco o CLI (INTERFACE DE LINHA DE COMANDO).

O cliente CLI é uma herança de dispositivos wireless como por exemplo os telefones GSM ou qualquer outro dispositivo que tenha possua o serviço SMS two-way.

O Cliente CLI utiliza mensagens de texto para se comunicar com o Servidor WV.

O CLP foi projetado de tal forma que fosse possível a comunicação entre o Servidor WV e o Cliente CLI, de tal forma que, da interação entre ambos fosse possível o Cliente CLI suportar os serviços CLI.

O CLP torna possível a interação entre o Cliente CLI e o Servidor WV via mensagens de texto.

O CLP consiste de comandos que o usuário digita em um dispositivo como teclado ou keypads, e envia uma mensagem SMS utilizando o Servidor WV para que outro usuário receba a mensagem.

O Servidor, após enviar a mensagem, responde com um Status Message indicando que o usuário recipiente leu a mensagem.

As mensagens CLP são trafegadas e transmitidas via mensagens de texto two-way, por default.

Para simplificar o endereçamento das mensagens SMS é recomendado um pequeno alias de 4 dígitos ou menos para identificar o usuário remetente da mensagem.

### 15.1 USER-ID

A definição de User-ID no CLI é semelhante à utilizada no CLP, como mostram os exemplos abaixo:

Local-User-ID:           wv:Jon.Smith

Global-User-ID: wv:Jon.Smith@imps.com

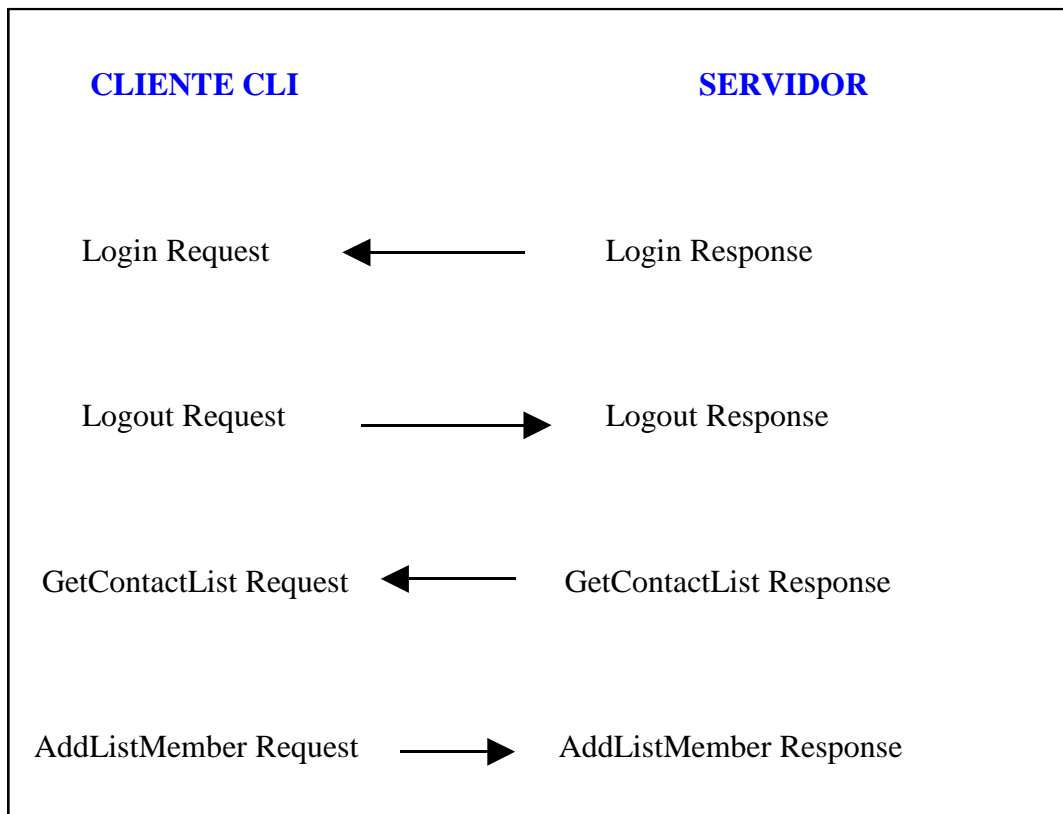
                  wv:+123456789@imps.com

                  wv:09876543@imps.com



Da mesma forma, várias transações podem ser feitas do Cliente CLI para o Servidor, e vice versa.

Alguns exemplos de transações estão representadas abaixo.



**Exemplos de transações**

## 15.2 REQUISIÇÃO DE LOGIN

O LoginRequest é a forma com que o cliente inicia uma nova seção com o servidor.

A sintaxe para o login request é dada abaixo.

Syntax: <UserID> <Password>

Send To: WV-Login

Por exemplo:

[john@im.mot.com](mailto:john@im.mot.com) inst1

**AddListMemberRequest - Comando: WV-Add**



A sintaxe para este tipo de requisição é:

Syntax: <UserID>

Send To: WV-Add

### 15.3 EXEMPLO DE UMA SEÇÃO IMPS

Os comandos necessários e suas respectivas sintaxes para que seja estabelecida uma seção de IM estão demonstrados abaixo, desde o login, até o termino da seção.

- Criar um Seção

```
SEND TO WV-LOGIN: "john 1234"
```

```
RECV FROM WV-LOGIN: "IMPS: User john is logged in to imps.wv.com domain"
```

- Gerenciamento de Lista de Contatos

```
SEND TO WV-CONTACTS: ""
```

```
RECV FROM WV-CONTACTS: "IMPS: your contact List is empty"
```

```
SEND TO WV-ADD: "mike"
```

```
RECV FROM WV-ADD: "IMPS: mike is added to your contact list as alias 9801"
```

```
SEND TO WV-ADD: "mark"
```

```
RECV FROM WV-ADD : "IMPS: mark is added to your contact list as alias 9802"
```

- Obter Presença



SEND TO WV-CONTACTS: "mark, mike"  
RECV FROM WV-CONTACTS: "1-O-mike 2-A-mark"  
SEND TO WV-SUBSCRIBE: "mike"  
RECV FROM WV-SUBSCRIBE: "IMPS: Subscription to mike is complete"  
RECV FROM WV-PRESENCE: "IMPS: User mike is Offline"  
.....  
RECV FROM WV-PRESENCE: "IMPS: User mike is Available"  
SEND TO WV-UNSUBSCRIBE: "mike"  
RECV FROM WV-UNSUBSCRIBE: "IMPS: Unsubscribed from mike"

- Instant Messging

SEND TO WV-mike (e.g. alias 9801): "Hi Mike, this is John, how are you"  
RECV FROM WV-mike: "IMPS: From mike: I'm fine, John, how are you?"  
Respond to Message: "Fine, thank you. I need to talk to you"  
RECV FROM WV-mike: "IMPS: From mike: I'm busy now. Let's talk tomorrow"

- Término da Session

SEND TO WV-LOGOUT: ""  
RECV FROM WV-LOGOUT: "IMPS: User john is logged out"





## 16. Implementação do LoginRequest e do GetContactList

O propósito do projeto foi realizar a comunicação entre os diversos componentes do Sistema WV comentados anteriormente.

A prática foi realizada no lemom, onde foram necessárias 4 máquinas da rede interna para que fosse possível a simulação da comunicação entre o LCS, PRESENCE, CLIENT e SAP.

Pela especificação, o Presence, o Cliente, o LCS e o IM não se comunicam diretamente entre si. Toda a comunicação entre as partes passa primeiro pelo SAP.

Dessa forma, toda a comunicação entre o SAP e o CLIENT, e entre o SAP e o LCS foi implementada para que fosse realizada via XML.

Em poucas palavras, XML ( Extensible Markup Language ) pode ser definida como uma linguagem de marcação de dados extensível - ao contrário do [HTML](#) - que foi projetada para permitir o uso do SGML ( Standard Generalized Markup Language ) na World Wide Web.

Ela provê um formato para descrever dados estruturados que facilita declarações mais precisas do conteúdo.

O XML não é uma simples linguagem de marcação pré-definida: ela é uma metalinguagem - uma linguagem usada para descrever outras linguagens - que permite que o seu usuário defina a sua própria marcação. Uma linguagem de marcação pré-definida como o HTML especifica um modo de descrever informação em apenas uma classe específica de documento. O XML permite que o usuário defina as suas próprias linguagens de marcação para atender à inúmeras classes de documentos diferentes. Isto é possível porque o XML é escrito em [SGML](#), a metalinguagem padronizada internacionalmente para sistemas de marcação de texto.

Daí a necessidade da comunicação SAP-CLIENT e SAP-LCS ser em XML.

SAP e Presence são partes bastante próximas, ou seja, estão dentro de Server. Para este caso, não há especificação de como deve ser a comunicação entre ambos. Então, foi adotado a comunicação entre o SAP e o Presence feita em RMI (Remote Method Invocation), que é a transmissão de objetos entre duas classes java remotamente.

A comunicação entre as unidades WV, como descrito anteriormente, pode ser feita através de vários protocolos.

O adotado para este projeto foi o protocolo HTTP, devido a fácil comunicação com o servidor TomCat.



As ferramentas de desenvolvimento utilizadas no decorrer do projeto foram:

- Java2ME
- Java2SE
- UML
- XML e HTTP
- RMI

Um tópico importante de ser abordado é o J2ME.

Esta classe do java é bastante importante, pois é ela quem gera o aplicativo virtual que foi utilizado como sendo Cliente.

## 16.1 O QUE É J2ME

A plataforma Java 2 Micro Edition (J2ME) implementa a linguagem Java para vários dispositivos tais como telefones celulares, PDAs, Internet screenphones, settop boxes digitais para TV, sistemas de navegação automotiva, comutadores e roteadores de rede, componentes para automação residencial, etc.

A plataforma J2ME consiste de máquinas virtuais (VM) e APIs especificadas em "documentos de configuração" ("Configuration" – CDC e CLDC) em conjunto com perfis ("Profiles" - MIDP) para uma determinada aplicação ou mercado.

A versão suportada na maioria dos dispositivos celulares no Brasil é a CLDC 1.0 e o MIDP 1.0. O suporte comercial ao MIDP 2.0 está previsto para o final de 2003, quando deverão ser lançados os primeiros modelos baseados nessa nova versão do perfil.

O **CLDC** (Connected Limited Device Configuration) contém uma API mínima para poder rodar aplicativos em dispositivos móveis com limites de processamento e memória, tais como, telefones celulares, smartphones, pagers e PDAs.

Em oposição temos a **CDC** (Connected Device Configuration) que supõe dispositivos com mais capacidade tais como settop boxes digitais, screen-phones, Nokia Communicator e alguns PDAs mais poderosos.

No que diz respeito ao CLDC para aparelhos celulares, temos as seguintes características:



- Hardware requerido: mínimo de 160KB de memória para Java, um processador de no mínimo 16 bits com baixo consumo (adequado a baterias típicas de um celular) e conexão de rede (neste caso wireless – 9.6Kbps, 144Kbps ou 2Mbps).

- Software que inclua suporte a um subconjunto da linguagem Java e a um subconjunto da máquina virtual Java que definam um núcleo de funções que permitam o desenvolvimento de aplicações móveis.

A segurança, que a CLDC define, está baseada no chamado “Sandbox Security Model”, isto é, a máquina virtual terá um espaço de memória independente do restante das aplicações do celular (tais como agenda, tons, imagens, configuração, browser WAP, etc).

Nesse modelo de segurança restringe-se as operações que podem ser executadas a um conjunto de funções da API tão somente. Nenhuma outra operação é permitida. Com isso a aplicação não pode acessar a área de memória do calendário ou agenda de contatos do aparelho celular. Outro ponto importante no que tange ao Java é o fato que não é permitido carga de classes definidas pelo usuário de forma a garantir a segurança de qualquer aplicação desenvolvida com a API CLDC.

Essa é uma diferença significativa em relação ao BREW que não define limites à aplicações que por sua vez podem acessar áreas de memória de qualquer parte do celular, compartilhar dados com outros aplicativos, alterar áreas de código também.

O fato do C/C++ permitir acesso via apontadores a memória, o que o Java não permite, ajuda o BREW a ser mais flexível, porém exige que o programador saiba o que está fazendo.

Por esse motivo o custo de desenvolvimento no BREW é mais alto, pois toda aplicação tem que ser testada e certificada pela Qualcomm para garantir que não se introduza na rede aplicações “mal intencionadas ou simplesmente mal feitas” e com isso ponha-se em risco toda a credibilidade da plataforma BREW.

O J2ME corta esse risco pela raiz, o que o faz ser mais limitado, porém seguro e com custos de desenvolvimento menores. Em cada plataforma há suas vantagens e seus “trade-offs”.

No topo da camada de CLDC temos a camada de software chamada Profile. O Profile, ou Perfil, especifica um conjunto de APIs que moldam-se a um determinado tipo de dispositivo.

A diferença entre Configuração e Perfil é que a Configuração descreve de forma geral uma família de dispositivos, enquanto o Perfil fica mais específico para um tipo particular de aparelho em uma dada família.



O Perfil não somente acrescenta funcionalidades àquele particular aparelho.

O Perfil MIDP contém, por exemplo, APIs para aparelhos celulares enquanto o CLDC aborda APIs para dispositivos móveis de pouca capacidade de processamento, dentre os quais temos celulares, pagers etc.

O CLDC define em sua API os pacotes `java.io`, `java.lang`, `java.util` e `javax.microedition.io` (conexão e interfaces). Aqui, é possível observar que não há suporte a ponto flutuante, suporte completo à classe `Error` do J2SE, referências fracas, verificação de arquivos de classes (há no lugar uma ferramenta de “pré-verificação”), finalização - `Object.finalize()`, JNI (Java Native Interface), reflection, thread groups/ daemons e user-defined class loaders.

O MIDP define tudo o que há no CLDC com a adição aos pacotes `javax.microedition.lcdui` (interface com o usuário), `javax.microedition.rms` (sistema de gerência de registros para persistência de informações), `javax.microedition.midlet` (suporte para aplicações MIDP, os chamados midlets).

Vários fabricantes adicionam ao MIDP algumas outras APIs, tais como suporte a uma forma simplificada de AWT (Abstract Window Toolkit) e controles de vibração, som e display.

Cada fabricante adiciona livremente o que considerar importante. Isso torna difícil muitas vezes portar aplicações entre fabricantes, a menos que se use sempre o conjunto mínimo especificado no MIDP 1.0.

Com o MIDP 2.0, haverá um conjunto mínimo que incluirá controles de vibração, som e display entre outras capacidades. Dessa forma haverá mais portabilidade entre os vários modelos e fabricantes.

O J2ME Wireless Tool Kit foi a ferramenta utilizada para gerar o simulador de aparelho celular.

Este simulador tem a aparência definida abaixo:



Figura 48 – Emulador Wireless Toolkit

Para se utilizar o J2ME toolkit, vamos pegar com exemplo um programa como o seguinte código.

```
import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;

public class HelloJ2ME extends MIDlet implements CommandListener
{
    private Display display;
    private TextField t;
    private Command comandoDeSaida;
```



```
private Form formularioPrincipal;

public HelloJ2ME() {
    display = Display.getDisplay(this);
    formularioPrincipal = new Form("HelloJ2ME");
    comandoDeSaida = new Command("Exit", Command.SCREEN,1);
    t = new TextField("Text","Hello World!",15,TextField.ANY);
    formularioPrincipal.addCommand(comandoDeSaida);
    formularioPrincipal.append(t);
    formularioPrincipal.setCommandListener(this);
}

public void startApp() {
    display.setCurrent(formularioPrincipal);
}

public void pauseApp() {
}

public void destroyApp(boolean incondicional) {
}

public void commandAction(Command comando, Displayable s) {
    if (comando == comandoDeSaida) {
        destroyApp(false);
        notifyDestroyed();
    }
}
}
```

O primeiro passo é iniciar o KToolbar que é onde criaremos um projeto. A figura abaixo ilustra onde pode ser localizada a aplicação em ambientes Windows.



**Figura 49 – Instalação do emulador**

Uma vez iniciado o aplicativo KToolbar, o mesmo se apresentará como a seguir.



**Figura 50 – instalação do emulador**

Após iniciar o KToolbar devemos abrir um novo projeto. Na tela da aplicação KToolbar existe o botão "New Project ...". Aperte esse botão para que seja aberto um formulário que solicitará que seja dado um nome de projeto (qualquer) e o nome da classe MIDlet que rodará o programa J2ME. No nosso exemplo, o nome do projeto é "Hello World!" e conforme descrito no código fonte do programa, a classe é chamada "HelloJ2ME".



**Figura 51– Criação de um projeto no wireless toolkit**

Ao apertar o botão "Create Project" o sistema automaticamente criará uma estrutura de diretórios referentes ao projeto dentro do diretório de instalação do J2ME Wireless Toolkit. Essa estrutura estará abaixo do diretório "Apps\<Nome do Projeto>" que fica dentro da estrutura de diretórios do Toolkit (por exemplo "C:\J2MEwtk\Apps\Hello World!\").

Nela temos os seguintes diretórios:

- \Src - nele colocaremos o código fonte de nossa aplicação, ou seja, o arquivo "HelloJ2ME.java".
- \Bin - nele o Toolkit criará os arquivos .JAR e .JAD que são necessários para poder executar a aplicação em um celular ou mesmo no simulador.
- \Res - nele são colocados todos os arquivos adicionais que são necessários para que a aplicação seja executada. Um exemplo típico é uma imagem no formato PNG com o ícone da aplicação. Todos os arquivos que forem colocados nesse diretório serão empacotados dentro do .JAR e serão acessíveis pela aplicação J2ME. No nosso exemplo não colocaremos nada lá.

Ao pressionar o botão "Create Project", o KToolbar criará os diretórios acima descritos e também o próprio diretório do projeto "Hello World!". Porém antes o Toolkit pedirá para que sejam confirmadas as informações do descritivo do Projeto/aplicação. Essas informações ou atributos serão o conteúdo do arquivo .JAD (Java Descriptor) a ser criado. No nosso caso não nos preocuparemos com esses dados e simplesmente apertaremos o botão "OK". O resultado aparece a seguir.



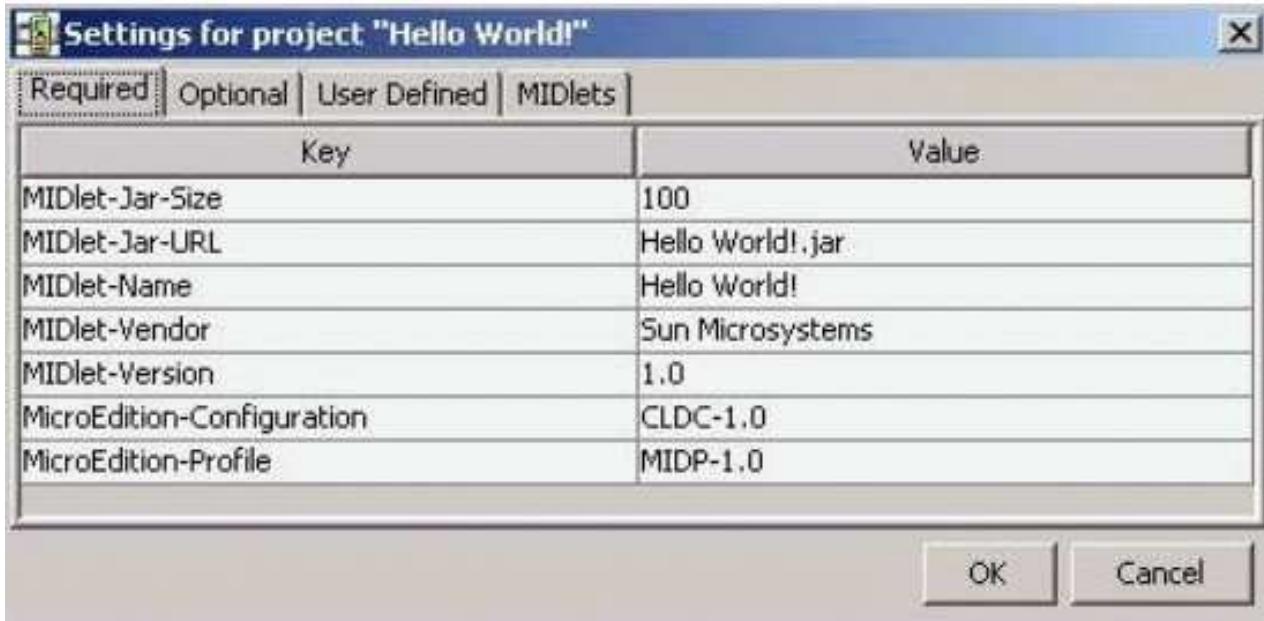


Figura 52– listagem dos projetos existentes

Após apertar o "OK", o Toolkit indicará no seu console que os diretórios do projeto foram criados conforme a figura abaixo.

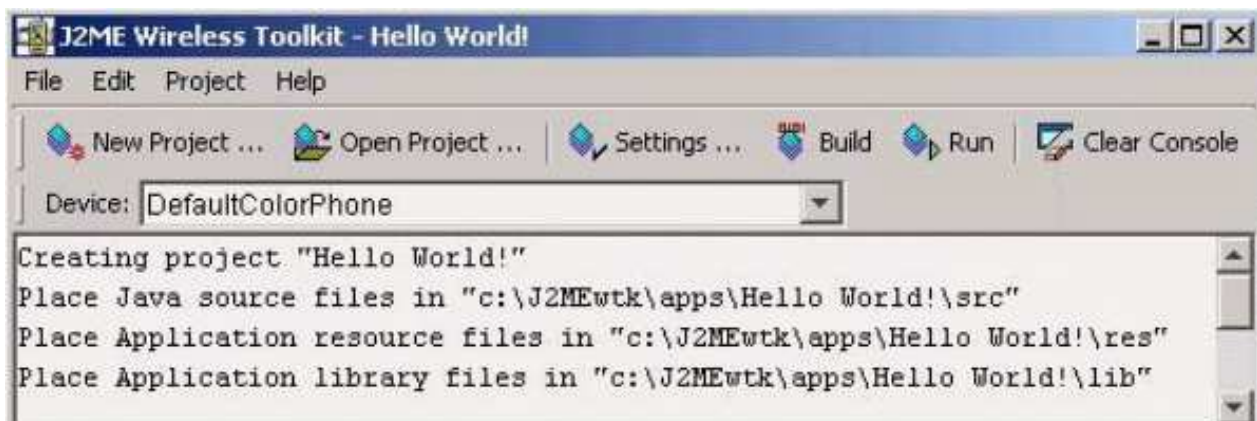


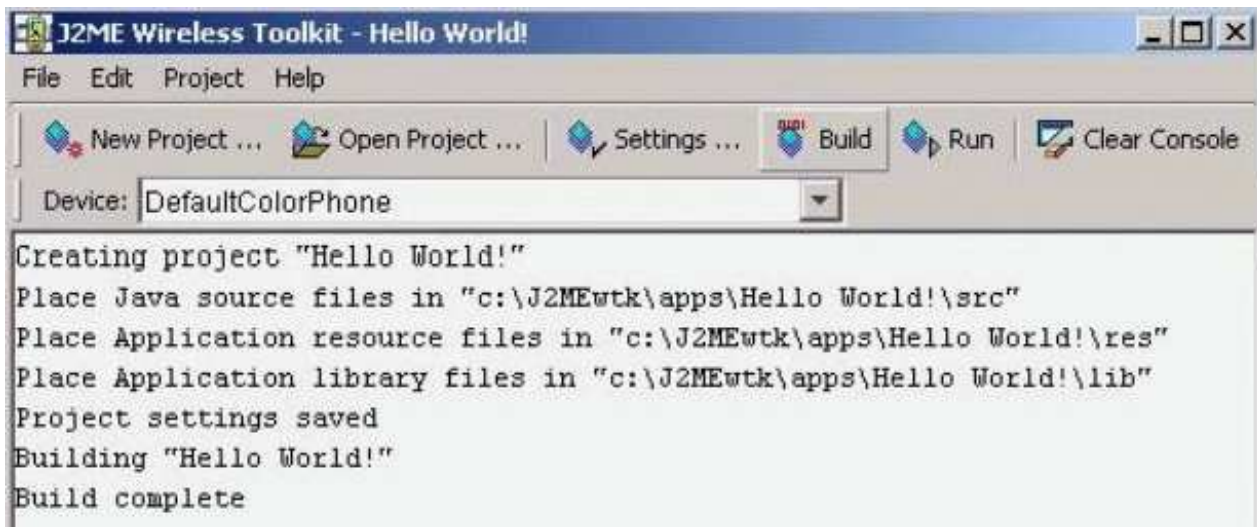
Figura 53 – Criando um projeto

Nesse momento o Toolkit nos sugere que os arquivos fonte sejam colocados no diretório "...apps\Hello World!\src". Nesse momento devemos copiar o arquivo "HelloJ2ME.java" para esse diretório.

A partir de agora estamos prontos para construir a aplicação!



Basta apenas apertar o botão "Build" no KToolbar que o Toolkit compilará os arquivos fonte e criará os arquivos .class que serão usados para rodar o simulador. Abaixo segue a Figura da construção do aplicativo "Hello World!".



**Figura 54 – Construção de um aplicativo**

Caso não ocorra nenhum problema de compilação teremos a mensagem "Build Complete", ao final das mensagens, indicando que está tudo pronto para executar a aplicação "Hello World!" no simulador. Caso ocorram erros de compilação os mesmos serão apresentados no console do Toolkit.

No nosso caso, o programa está depurado e o código fonte está correto. Portanto não há nenhum problema de compilação. Neste caso estamos prontos para executar a aplicação. Para isso basta apertar o botão "Run" no KToolbar. Ao fazer isso será aberta uma nova janela com o simulador escolhido. No nosso exemplo é o simulador genérico colorido, conforme a figura abaixo.



**Figura 55 – Executando um aplicativo no emulador**

A figura acima mostra o estágio inicial do simulador que permite que seja escolhido qual Midlet executar. Nosso caso só há um Midlet especificado no arquivo .JAD, portanto temos apenas que pressionar o SoftKey para lançar a aplicação "Hello World!" (botão em destaque logo abaixo da palavra "Launch"). O resultado da execução do "Hello World!" segue abaixo.

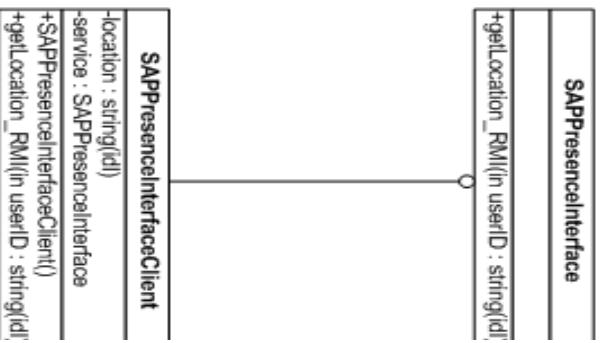
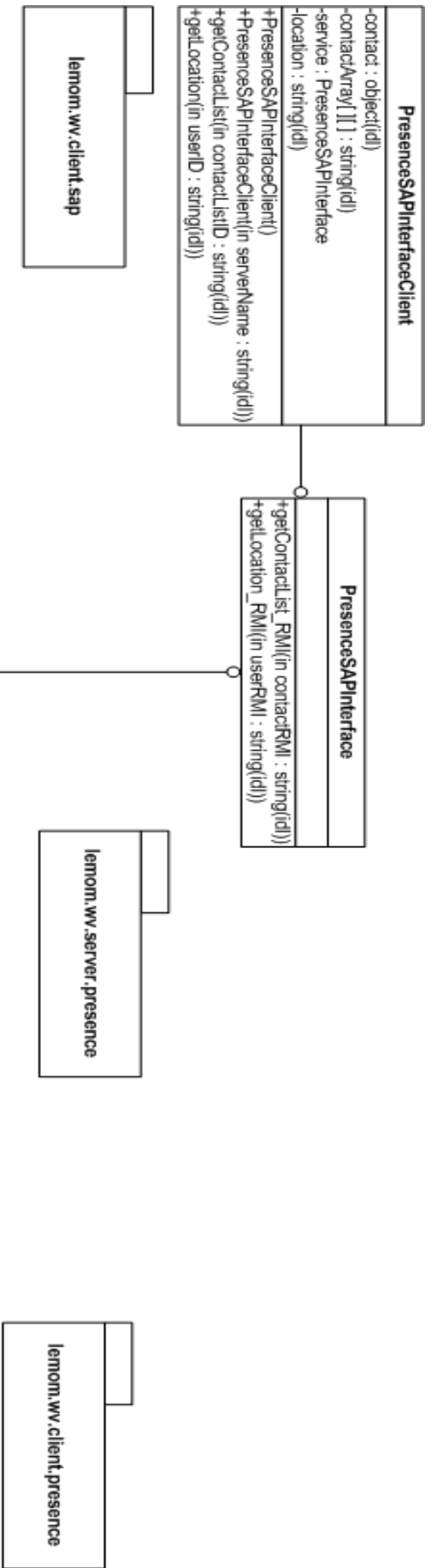


**Figura 56 - Emulador**

Então, partindo deste ponto, ou seja, partindo do ponto que o Cliente seria este Simulador Virtual do Wireless Tool Kit, e também introduzindo os conceitos WV anteriormente descritos, foi feita a modelagem do sistema.

As unidades WV (Presence, SAP, Cliente e LCS) foram modeladas, com diagrama de classes em UML e esquemas gráficos de funcionamento dos mesmos.

As classes do Presence estão abaixo. Cada classe está com os respectivos métodos e atributos. No diagrama de UML, é possível a verificação da relação entre as mesmas.





### Figura 57 – Classes do Presence

Acima, as classes mostram o formato da informação que o Server e o Cliente trocarão entre si.

A comunicação entre os dois, como dito anteriormente, é feita em RMI.

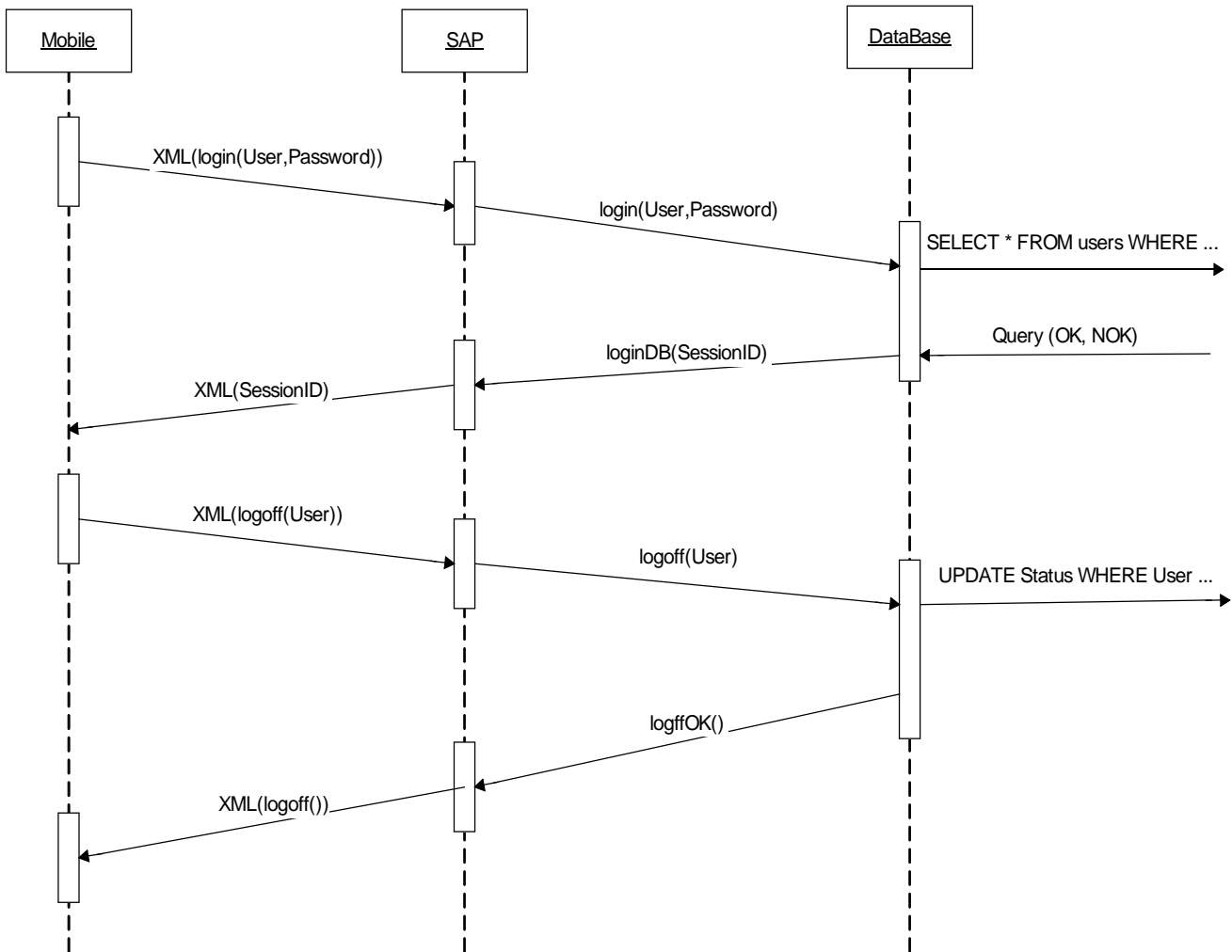
As informações dos usuários, como ContactLists, Login e Senha estarão todas no banco de dados.

Sempre que usuário necessitar obter estes dados, o SAP (intermediário em toda comunicação) se comunicará com o Server, que por sua vez irá buscá-los no Database.

O banco de dados desenvolvido foi em MySQL.

Para que fosse possível a demonstração, foi criado um banco de dados chamado ContactList e uma tabela chamada ContactList, e posteriormente foram inseridos alguns dados.

O diagrama que melhor representa as transações de Login está abaixo:



**Figura 58 – LoginRequest**

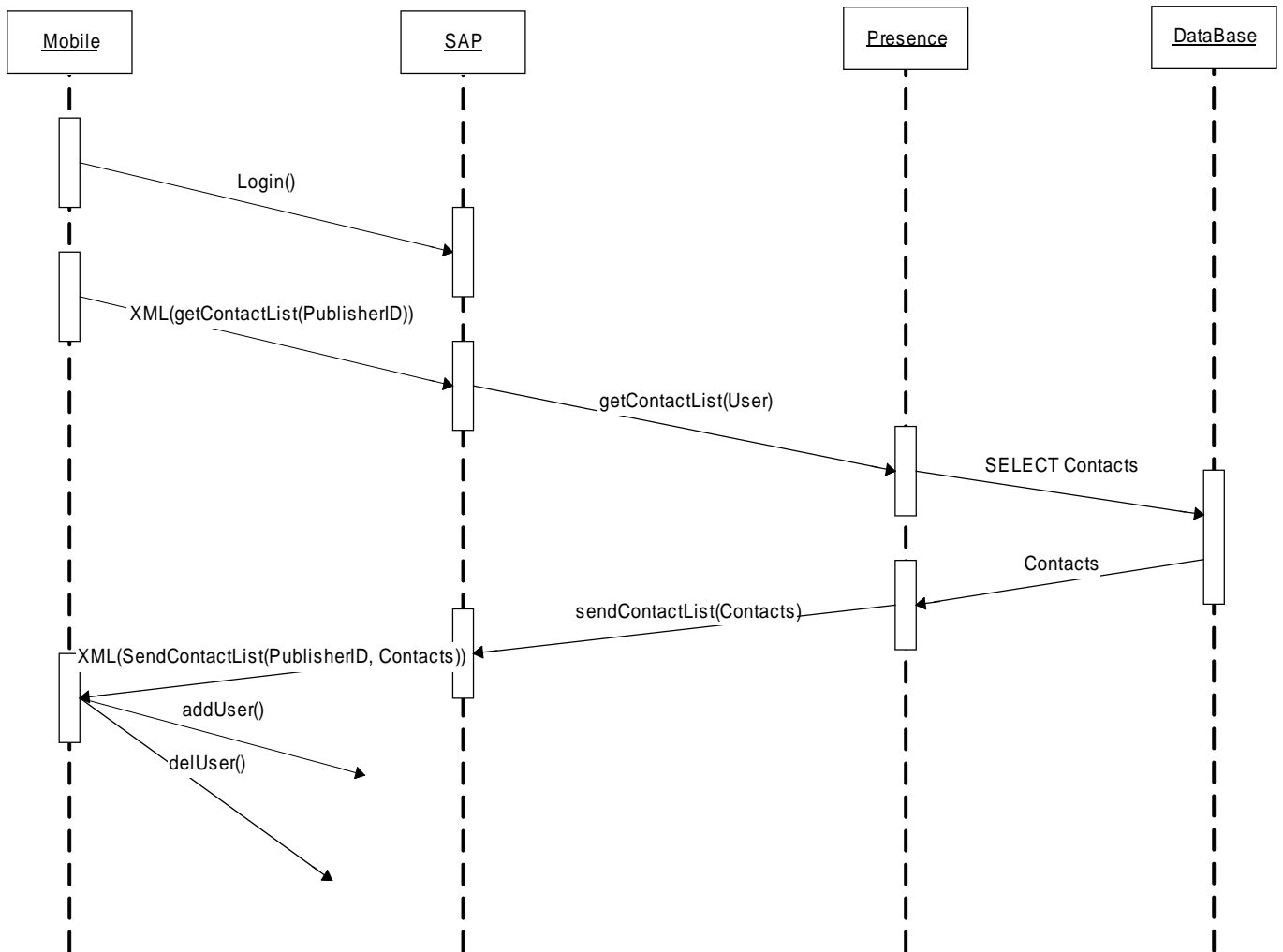
Nele, é possível observar-se que o Cliente envia o pedido de login com o User e o Password. O SAP, recebe estes dados em XML. O SAP tem uma classe Parser que transforma o dado XML em OBJETO e repassa este objeto ao Presence por RMI. O servidor por sua vez, faz a busca no banco de dados, comparando as informações recebidas com as informações gravadas, e devolvendo ao SAP a autorização ou não do processo de login.

No Logout, o procedimento é o mesmo. O Cliente envia o pedido de LogOut ao SAP em formato XML. O SAP transforma estes dados em Objeto, repassa ao Presence por RMI.

O Presence dá um UpdateStatus no Database e efetua o Logout.

Da mesma forma se procede o GetContactList.

O Diagrama para esta transação está representado abaixo.



**Figura 59 - GerContactList**

Após efetuado o Login, o usuário requisita uma Lista de Contatos ao Server.

O SAP repassa os dados para o Presence.

O Presence faz uma busca no banco de dados e seleciona os contatos para aquele usuário.

Após isso, o Presence envia os contatos ao SAP que transforma os dados em XML e repassa ao Cliente.

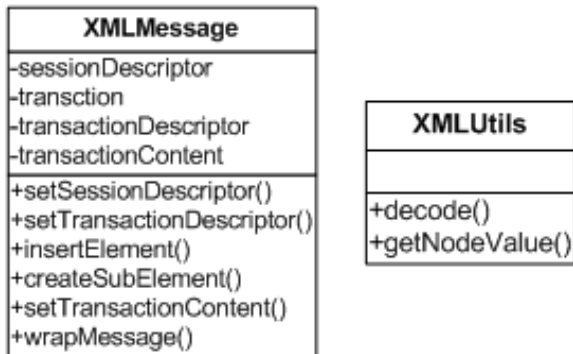
Após isso, o Cliente tem as opções de modificar, adicionar ou deletar contatos, mas estas funções ainda não foram implementadas.

Nas duas ultimas figuras abaixo, estão representados os diagramas de classes do Cliente.





### lemom.wv.client.xml



### lemom.wv.client.service

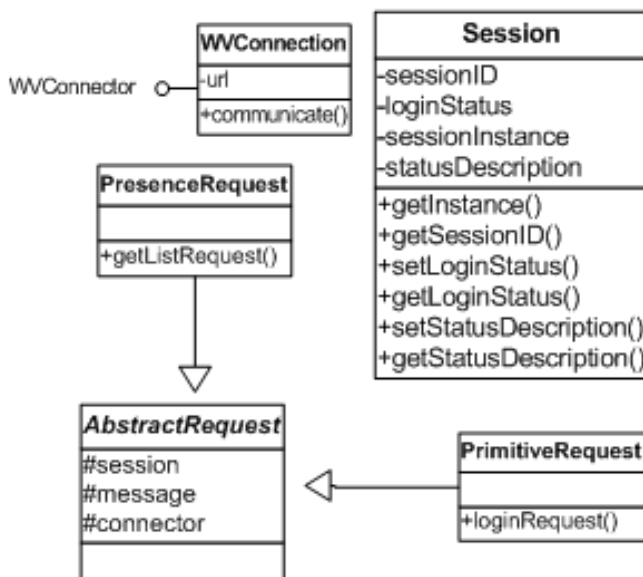




Figura 60 - Classes do Cliente

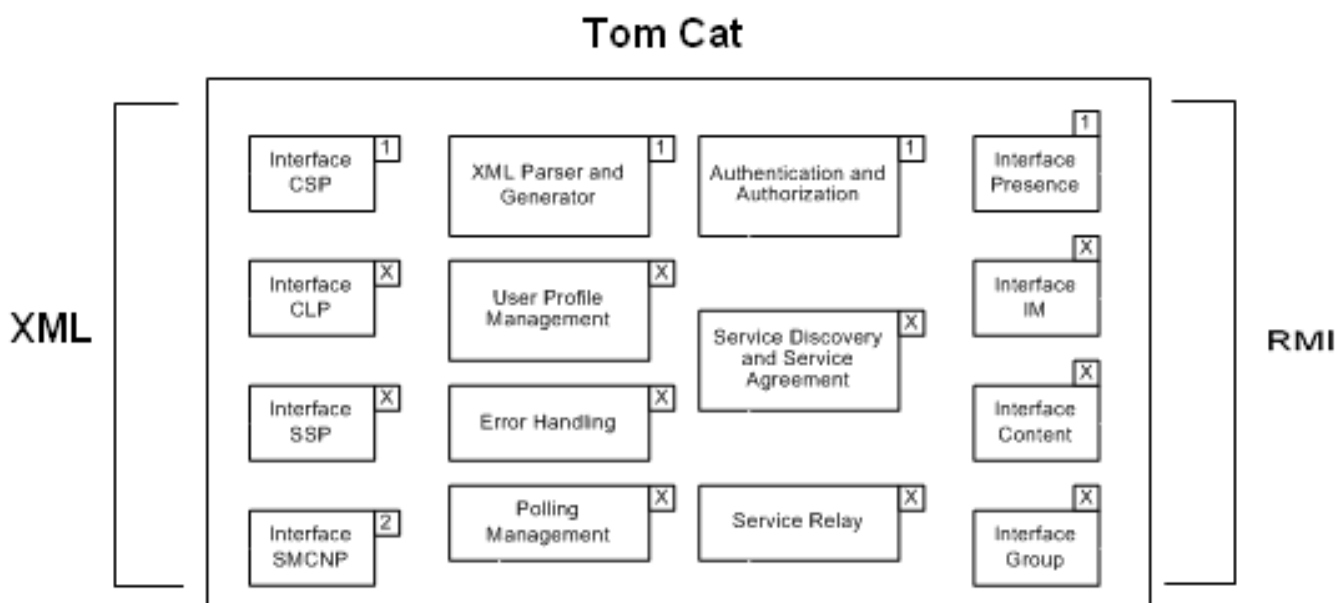


Figura 61 - SAP (Diagrama de Classes)

## 17 - Conclusão

O projeto implementado foi bastante teórico no começo, pois ainda não sabíamos como seria a comunicação, nem quais as entidades a serem utilizadas no sistema.

A documentação do Wireless Village deu um impulso bastante grande, no que se refere ao ponto de partida para a implementação em si.

Definidos os parâmetros principais do Projeto, passamos para a nova fase, que foi a implementação das classes.

O canal de comunicação foi o TCP, que é o padrão para qualquer tipo de troca de informações remotas.

Para a comunicação Cliente-SAP, utilizamos o Java Wireless ToolKit, onde há um emulador do Cliente (aparelho celular), de onde tudo se inicia.



Através deste emulador, as transações eram realizadas entre o Cliente e o SAP, e a transmissão de dados toda em XML, através do protocolo HTTP.

No SAP, tivemos que implementar 2 serviços de escuta. Um foi o Servidor TomCat, para receber os dados em XML do Cliente, e outro foi o RMI, para receber os dados do Presence.

Além destes dois, foi configurado também um sniffer simples, apenas para mostrar os momentos em que o SAP recebia e enviava requisições para o Cliente, Presence e LCS.

No Presence, o único serviço configurado foi o RMI, para ficar aguardando requisições do SAP.

A única parte do projeto onde não foi possível obter sucesso como o esperado foi com relação ao Servidor LCS, que até o momento, não foi disponibilizado para testes.

Então, foi simulado um servidor LCS, apenas para ilustrar a comunicação entre as entidades.

Este servidor também recebe e transmite em RMI.

Com estes dados, foi possível provar, pelo menos um início, do que é possível ser implementado em um aparelho celular, em termos de comunicação entre os usuários.

Foi possível mostrar também que, para tal, não há necessidade de grande infra-estrutura, já que a grande parte do projeto baseia-se em ferramentas de desenvolvimento obtidas livremente na internet.

Outro ponto importante a destacar é que a plataforma que foi utilizada para tal foi o Linux, o que acarreta numa baixa ainda maior dos custos, já que a maioria das ferramentas de trabalho utilizadas são de código livre, como o Poseidon (software utilizado para fazer diagramas em UML), todas as ferramentas java (TomCat, Wireless Toolkit, RMI), editores de texto (OpenOffice), e editores java (NetBeans).



## Referências Bibliográficas

- [1] IETF – *Internet Engineering Task Force – Mobile IP Working Group IP Routing for Wireless/Mobile Hosts* – 2002. [www.ietf.org](http://www.ietf.org)
- [2] IEEE - *IEEE Standard 802.11*.- 1997. [www.ieee.org](http://www.ieee.org)
- [3] IEEE Standards Department. - *IEEE Draft Standard for Wireless LAN – Medium Access Control (MAC) and Physical Layer (PHY) Specification* - Documento P802.11D6.1, 1997. [www.ieee.org](http://www.ieee.org)



- [4] ANDERSSON, Christoffer - *GPRS and 3G Wireless Applications: Professional Developer's Guide Series* - John Wiley & Sons, Inc, 2001.
- [5] *Plataforma CDMA2000 (1XRTT)* - Lucent Technologies, Junho, 2002 (Apresentação para TCO Centro Oeste Celular).
- [6] *A 3G Introduction to Technology, 3G Standards and 3G Spectrum* - Junho/2002 - <http://andrew100.tripod.com/3g.html>.
- [7] BERNAL, Paulo Sérgio M. - *Comunicações Móveis – Tecnologias e Aplicações* - São Paulo, Editora Érica, 2002.
- [8] CHEN, Yi-an - *A Survey Paper on Mobile IP* - <http://www.securitytechnet.com/hot-topic/mip.html>
- [9] *Mobility and Mobile IP, Introduction* – IPUnplugged, 2002. [www.ipunplugged.com](http://www.ipunplugged.com)
- [10] *IPUnplugged Solution Brief Seamless Roaming – 3G / Wireless LAN* – 2002 – [www.ipUnplugged.com](http://www.ipUnplugged.com)
- [11] *3GPP2 PR001 v1.0.0/ Wireless IP Network Architecture based on IETF protocols* – Julho, 2000 – [www.3gpp2.org/docs](http://www.3gpp2.org/docs)
- [12] KELLER, Ralf; GOLDBECK, Tomas e BOSTROM, Tomas - *Ericsson Mobile Operator WLAN Solution* – [www.ericsson.com](http://www.ericsson.com)
- [13] RAMASWAMY, Kumar - *Issues in Inter-working Models for 3G and Wireless Local Area Networks* – THOMSON Multimedia, 2002.
- [14] GEIER, Jim - *Wireless Networking Handbook* – Macmillan, 2002. <http://news.dcn-asu.ru/BOOKS/Wireless/ewtoc.html>



[15] SOARES, L. F. G. Et al. – *Redes de Computadores - Das LANs, MANs e WANs às redes ATM* - Ed. Campus – 1995.

[16] HAYKIN, Simon – *Communication Systems* – 4ª Edição - Wiley & Sons, Inc., 2002.

[17] H.M. DEITEL, P.J. DEITEL – *JAVA – Como Programar* – 3º Edição

[18] WIRELESS VILLAGE DOCUMENTATION – [www.wireless-village.org](http://www.wireless-village.org)

## APENDICE A – Classes Java

```
/******  
*  
* ContactList.java v0.1  
*  
* Servidor LCS - INdT e UnB - 23/05/2003  
*
```



**\* Implementação para a interface remota PresenceSAPInterface para prover objetos remotos entre o**

**\* Servidor de Presença e o SAP**

**\***

**\*\*\*\*\* \*/**

```
package lemom.wv.server.presence;
```

```
import java.util.*;
```

```
import java.sql.*;
```

```
import com.mysql.jdbc.Driver;
```

```
public class ContactList {
```

```
    private String query;
```

```
    private List contacts;
```

```
adduser()
```

```
deluser()
```

```
    public ContactList () {
```

```
        System.out.println("Please fill an userID.");
```

```
    }
```

```
    public ContactList (String userID) {
```

```
        Connection connection;
```

```
        Statement statement;
```

```
        ResultSet resultSet;
```

```
        try {
```



```
try {
    Class.forName("com.mysql.jdbc.Driver").newInstance();
} catch(ClassNotFoundException e) {
    System.out.println("Driver not found.");
    e.printStackTrace();
}

try {
    connection =
//DriverManager.getConnection("jdbc:mysql://192.168.200.xx/ContactList.ContactList?
user=root&password=lemom");
    connection =
DriverManager.getConnection("jdbc:mysql://localhost/ContactList.ContactList?
user=root&password=lemom");
} catch(SQLException e) {
    System.out.println("Database not found.");
    e.printStackTrace();
}

statement = connection.createStatement();
query = "SELECT contacts FROM ContactList.ContactList WHERE
username = \'' + userID + '\''";
resultSet = statement.executeQuery(query);
System.out.println(resultSet.getString("contacts"));

StringBuffer results = new StringBuffer();
ResultSetMetaData metaData = resultSet.getMetaData();
int numberOfRows = metaData.getRowCount();
while (resultSet.next()) {

    List.add(resultSet.getString(1));
    for (int i = 1; i <= numberOfRows; i++) {
        results.append(metaData.getObject(i));
    }
}
```





```
        }
        result += "\n";
    }

    statement.close();
    connection.close();
} catch (SQLException e) {
    System.out.println("Database Error");
} catch (ClassNotFoundException e) {
    System.out.println("Driver not found");
}
}

public List getContactList (String userID) {

    ContactList contacts = new ContactList(userID);
    return contacts;

}

public static void main (String args[]) {

    ContactList contacts = new ContactList(userID);

}
```

```
/**
*****

```

```
*
```

```
* PresenceSAPInterface.java v0.1
```

```
*
```

```
* Servidor LCS - INdT, UnB - 16/05/2003
```



```
*  
*   Interface que referencia os métodos RMI que serão utilizados entre o SAP e o  
Servidor de Presença,  
*   denominado Presence.  
*  
*****/  
  
package lemom.wv.server.presence;  
  
import java.rmi.*;  
import java.util.*;  
  
public interface PresenceSAPInterface extends Remote {  
  
    /* Metodo utilizado para se obter uma lista de Contatos do Servidor, onde se entra um  
usuário (String) como  
    parametro e se retorna um ArrayList com o nickname/domain dos usuarios na lista  
de contato */  
  
        public ArrayList getContactList_RMI(String contactRMI) throws  
RemoteException;  
  
    /* Metodo utilizado para recuperar a localizacao do usuario a partir de um  
requerimento para o LCS Server  
    utilizando o SAP como intermediario */  
  
        public String getLocation_RMI(String userRMI) throws RemoteException;  
  
    //    public String getOnLineStatus_RMI(String userRMI) throws RemoteException;  
  
}
```



```
/*
 *
 * PresenceSAPInterfaceImpl.java v0.1
 *
 * Servidor LCS - INdT, UnB - 23/05/2003
 *
 * Implementa a interface para a interface remota PresenceSAPInterface para
prover objetos remotos entre o
 * Servidor de Presenca e o SAP
 *
 **** */

//package lemom.wv.server.presence;

import java.rmi.*;
import java.rmi.server.*;
import java.util.*;

public class PresenceSAPInterfaceImpl extends UnicastRemoteObject implements
PresenceSAPInterface {

    private List contact = new ArrayList();
    private String location;
    private SAPPresenceInterfaceClient sapInterface;
    private static GenericGUI presenceGUI;

    public PresenceSAPInterfaceImpl () throws RemoteException, Exception {
    }

// Metodo que recupera a ContactList junto ao Banco de Dados
    private void updateContact (String contactRMI) {

        if (contactRMI.equalsIgnoreCase("wv:john/My_friends@smith.com")) {
```



```
        contact.clear();
        contact.add("Brainstorm/wv:bright@dark.com");
        contact.add("Randall/wv:randall@fairlane.com");
        contact.add("Jenny/wv:jenny@logic.com");
        contact.add("Vicky/wv:ex@wife.com");
        presenceGUI.printStatus( "Contact List found!" );
    } else
    if (contactRMI.equalsIgnoreCase("wv:paul/Work@smith.com")) {
        contact.clear();
        contact.add("Pinky/wv:pinky@world.com");
        contact.add("Cerebro/wv:cerebro@world.com");
        presenceGUI.printStatus( "Contact List found!" );
    } else {
        contact.clear();
        presenceGUI.printStatus( "Contact List not found!" );
    }
}
```

*// Metodo que recupera a Location junto ao Banco de Dados*

```
private void updateLocation (String userRMI) {

    try {
        sapInterface = new SAPPresenceInterfaceClient("192.168.200.3");
        location = sapInterface.getLocation_RMI(userRMI);
        presenceGUI.printStatus( "Location: " + location);
        presenceGUI.printStatus( "SAP was reached!" );

    } catch (RemoteException e) {
        location = "Failed";
        presenceGUI.printStatus( "Could not reach SAP!" );
    }
}
```



```
}
```

```
public ArrayList getContactList_RMI (String contactRMI) throws  
RemoteException {
```

```
    presenceGUI.printStatus( "Searching Contact List ..." );  
    presenceGUI.printStatus( "ContactListID: " + contactRMI );  
    updateContact(contactRMI);  
    for( int i = 0; i < contact.size(); i++ ) {  
        presenceGUI.printStatus((String)contact.get(i));  
    }
```

```
    return (ArrayList) contact;
```

```
}
```

```
public String getLocation_RMI (String userRMI) throws RemoteException {
```

```
    updateLocation(userRMI);  
    presenceGUI.printStatus( "Searching Location ..." );  
    return location;
```

```
}
```

```
public static void main (String args[]) throws Exception {
```

```
    //GUI Interface
```

```
    presenceGUI = new GenericGUI( "Presence SERVER" );  
    presenceGUI.printStatus( "Presence SERVER initialized!" );
```

```
    PresenceSAPInterface service = new PresenceSAPInterfaceImpl();  
    String serverObjectName = "rmi://192.168.200.11/PresenceSAPInterface";  
    Naming.rebind(serverObjectName, service);
```

```
}
```

```
}
```



```
/*  
*  
* SAPPresenceInterfaceClient.java v0.1  
*  
* Servidor LCS - INdT, UnB - 23/05/2003  
*  
* Implementação do Cliente para a interface remota PresenceSAPInterface para  
prover objetos remotos  
* relacionados a Lista de Contatos entre o Servidor de Presença e o SAP  
*  
***** */
```

```
//package lemom.wv.client.presence;
```

```
import java.rmi.*;
```

```
import java.util.*;
```

```
import java.io.*;
```

```
public class SAPPresenceInterfaceClient {
```

```
    private String location;
```

```
    private SAPPresenceInterface service;
```

```
    public SAPPresenceInterfaceClient (String serverName) {
```

```
        try {
```

```
            String remoteName = "rmi://" + serverName +
```

```
            "/SAPPresenceInterface";
```

```
            service = (SAPPresenceInterface) Naming.lookup(remoteName);
```

```
        } catch (ConnectException e) {
```

```
            e.printStackTrace();
```

```
        } catch (Exception e) {
```



```
        e.printStackTrace();
    }

}

public String getLocation_RMI (String userRMI) throws RemoteException {

    location = (String) service.getLocation_RMI(userRMI);
    return location;

}
}

/*****
*
* GenericGUI.java v0.1
*
* Servidor LCS - INdT, UnB - 23/05/2003
*
* Classe que define a interface gráfica utilizada pelo LCS Client e SAP
*
*****/

//package lemom.wv.server.presence;

import java.awt.*;
import java.awt.event.*;

import javax.swing.*;

public class GenericGUI extends JFrame {
```



```
private JLabel titleLabel;
private JTextArea transactionsField;
private JButton exitButton;

//Configura a GUI
public GenericGUI( String windowTitle )    {

    super( windowTitle );

    // cria uma instancia da classe interna ActionEventHandl
    ActionEventHandl
    er handler = new ActionEventHandl
    er();

    // configura a GUI
    Container container = getContentPane();
    container.setLayout( new BorderLayout() );

    titleLabel = new JLabel( windowTitle );
    titleLabel.setToolTipText( "Active Server" );
    transactionsField = new JTextArea( 20,40 );
    transactionsField.setEditable( false );
    container.add( titleLabel, BorderLayout.NORTH );
    container.add( transactionsField, BorderLayout.CENTER );
    container.add( new JScrollPane( transactionsField ) );

    exitButton = new JButton( "Exit" );
    exitButton.addActionListener( handler );
    container.add( exitButton, BorderLayout.SOUTH );

    addWindowListener(

        new WindowAdapter() {

            public void windowClosing( WindowEvent event )
```





```
        {
            System.exit( 0 );
        }

    }

);

setVisible( true );
pack();

} // fim do construtor

private class ActionEventHandler implements ActionListener {

    public void actionPerformed((ActionEvent event) )
    {
        if ( event.getSource() == exitButton )
            System.exit( 0 ); //termina o aplicativo;
    }

} // fim da classe interna

public void printStatus( String Status ) {

    transactionsField.append( Status + "\n" );

}

} //fim da classe GenericGUI
```



```
/*
*
* PresenceSAPInterfaceClient.java v0.1
*
* Servidor LCS - INdT, UnB - 23/05/2003
*
* Implementacao do Cliente para a interface remota PresenceSAPInterface para
prover objetos remotos
* relacionados a Lista de Contatos entre o Servidor de Presenca e o SAP
*
***** */

//package lemom.wv.server.presence;

import java.rmi.*;
import java.util.*;
import java.io.*;

public class PresenceSAPInterfaceClient {

    private ArrayList contact;
    private String contactArray[][];
    private PresenceSAPInterface service;
    private String location;

// Construtor padrao para esta classe
    public PresenceSAPInterfaceClient () {

    }

// Construtor que recebe o endereco IP do servidor a se conectar
    public PresenceSAPInterfaceClient (String serverName) {
```



```
try {
    String remoteName = "rmi://" + serverName +
"/PresenceSAPInterface";
    service = (PresenceSAPInterface) Naming.lookup(remoteName);
} catch (ConnectException e) {
    e.printStackTrace();
} catch (Exception e) {
    e.printStackTrace();
}
}

// Metodo que recebe uma ContactList e a transforma em uma matriz de contatos
public String[][] getContactList (String contactListID) throws RemoteException {

    contact = (ArrayList) service.getContactList_RMI(contactListID);
    contactArray = new String[contact.size()][2];
    if (contact.size() == 0) {
    } else if ((contact.size() == 1) && (contact.get(0).equals("Failed"))) {
    } else {
        for (int i = 0; i < contact.size(); i++) {
            StringTokenizer st = new StringTokenizer((String)
contact.get(i), "/", false);
            while (st.hasMoreTokens()) {
                contactArray[i][0] = st.nextToken();
                contactArray[i][1] = st.nextToken();
            }
        }
    }
    return contactArray;
}
```



```
// Metodo que faz a chamada da localizacao do usuario
    public String getLocation (String userID) throws RemoteException {

        location = (String) service.getLocation_RMI(userID);
        return (String) location;

    }
}

/*****
*
* SAPPresenceInterface.java v0.1
*
* Servidor LCS - INdT, UnB - 23/05/2003
*
* Interface que referencia os métodos RMI que serão utilizados entre o SAP e o
Servidor de Presença,
* denominado Presence.
*
*****/

//package lemom.wv.server.sap;

import java.rmi.*;
import java.util.*;

public interface SAPPresenceInterface extends Remote {

/* Metodo utilizado para se obter a localizacao de um usuario junto ao SAP e LCS
Server
```



```
public String getLocation_RMI (String userRMI) throws RemoteException;

}

/*****
 *
 * SAPPresenceInterfaceImpl.java v0.1
 *
 * Servidor LCS - INdT, UnB - 23/05/2003
 *
 * Implementacao para a interface remota SAPPresenceInterface para prover
objetos remotos entre o
 * Servidor de Presença e o SAP
 *
 *****/

//package lemom.wv.server.sap;

import java.rmi.*;
import java.rmi.server.*;
import java.util.*;

public class SAPPresenceInterfaceImpl extends UnicastRemoteObject implements
SAPPresenceInterface {

    private String locationString;

    public SAPPresenceInterfaceImpl () throws RemoteException {

        super();
    }
}
```



```
}

private void updateLocationString (String user) {

    LCS_Client client = new LCS_Client();
    locationString = client.getLocation(user);

}

public String getLocation_RMI (String userRMI) throws RemoteException {

    updateLocationString(userRMI);
    return locationString;

}

public static void main (String args[]) throws Exception {

    SAPPresenceInterface service = new SAPPresenceInterfaceImpl();
    String serverObjectName = "rmi://192.168.200.3/SAPPresenceInterface";
    Naming.rebind(serverObjectName, service);

}
}
```

## APENDICE B – MySQL

```
/*
*
* Contact.sql v0.1
*
* Servidor LCS - INdt, UnB - 16/05/2003
*
*/
```



```
* Script que criará a tabela com as colunas básicas utilizadas para guardar os  
dados de Contact List, além  
* de inserir alguns valores de teste nesta tabela  
*  
***** */  
  
/* Verificar "my.cnf" em relação aos parâmetros de configuração para InnoDB */  
  
CREATE TABLE ContactList (  
    id NUMBER (10),  
    username VARCHAR2 (30) CONSTRAINT user_nn NOT NULL,  
    contactname VARCHAR2 (30) CONSTRAINT contact_nn NOT NULL,  
    domain VARCHAR2 (50) CONSTRAINT domain_nn NOT NULL,  
    contacts VARCHAR2 (4000) CONSTRAINT contacts_nn NOT NULL,  
    last_update TIMESTAMP WITH LOCAL TIME ZONE DEFAULT SYSDATE,  
    CONSTRAINT id_pk PRIMARY KEY (id),  
    CONSTRAINT user_un UNIQUE (username));  
  
/* Talvez seja necessário para uma tabela em MySQL  
TYPE = InnoDB */  
  
/* CREATE INDEX user_idx ON ContactList (username); */  
  
CREATE SEQUENCE id_sequence  
    START WITH 1  
    INCREMENT BY 1  
    NOMAXVALUE  
    NOCYCLE;  
  
SELECT id_sequence.NEXTVAL from DUAL;  
  
/* SET linesize 200; */
```



```
INSERT INTO ContactList VALUES (id_sequence.NEXTVAL,  
                                'John',  
                                'My_friends',  
                                'smith.com',  
                                'paul/bob.com#ringo/bob.com',  
                                DEFAULT);
```

```
INSERT INTO ContactList VALUES (id_sequence.NEXTVAL,  
                                'Paul',  
                                'Pals',  
                                'bob.com',  
                                'john/smith.com#ringo/bob.com',  
                                DEFAULT);
```

```
INSERT INTO ContactList VALUES (id_sequence.NEXTVAL,  
                                'George',  
                                'Dudes',  
                                'smith.com',  
                                'elvis/presley.com',  
                                DEFAULT);
```

```
INSERT INTO ContactList VALUES (id_sequence.NEXTVAL,  
                                'Ringo',  
                                'Guys',  
                                'bob.com',  
                                'george/smith.com',  
                                DEFAULT);
```

```
INSERT INTO ContactList VALUES (id_sequence.NEXTVAL,  
                                'Elvis',  
                                'Rock',  
                                'presley.com',  
                                'john/smith.com#paul/bob.com',  
                                DEFAULT);
```

```
SELECT username, contactname, domain FROM ContactList;
```





```
// CREATE DATABASE ContactList;

CREATE TABLE ContactList.ContactList (
    id INT(10) PRIMARY KEY NOT NULL AUTO_INCREMENT,
    username VARCHAR(30) NOT NULL UNIQUE,
    contactname VARCHAR(30) NOT NULL,
    domain VARCHAR(50) NOT NULL,
    contacts TEXT NOT NULL,
    last_update TIMESTAMP NULL);

CREATE INDEX user_idx ON ContactList.ContactList (username);

INSERT INTO ContactList.ContactList (username, contactname, domain, contacts)
VALUES ("John", "My_friends", "smith.com", "paul/bob.com#ringo/bob.com");

INSERT INTO ContactList.ContactList (username, contactname, domain, contacts)
VALUES ("Paul", "Pals", "bob.com", "john/smith.com#ringo/bob.com");

INSERT INTO ContactList.ContactList (username, contactname, domain, contacts)
VALUES ("George", "Dudes", "smith.com", "elvis/presley.com");

INSERT INTO ContactList.ContactList (username, contactname, domain, contacts)
VALUES ("Ringo", "Guys", "bob.com", "george/smith.com");

INSERT INTO ContactList.ContactList (username, contactname, domain, contacts)
VALUES ("Elvis", "Rock", "presley.com", "john/smith.com#paul/bob.com");

DESCRIBE ContactList.ContactList;

SELECT * FROM ContactList.ContactList;
```