UNICAMP

Contribuição na implementação de ferramentas tecnológicas para a codificação de transações e integração de redes de comunicação sem fio com redes fixas

Mamadou Moustapha Ndiaye

Tese apresentada na Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas para a obtenção do Título de Mestre em Engenharia Elétrica e de Computação

Orientador: Prof. Dr. Vitor Baranauskas
Co-orientador: Prof. Dr. Carlos Fernandes Franco Jr.

Banca Examinadora:

Prof. Dr. Ivam Ricardo Peleias .....
Prof. Dr. Ioshiaki Doi .....
Prof. Dr. Edson Moschim .....
Prof. Dr. Yuzo Iano .....
Prof. Dr. Vitor Baranauskas .....

Departamento de Semicondutores, Instrumentos e Fotônica
Faculdade de Engenharia Elétrica e de Computação
Universidade Estadual de Campinas
2003

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

N239c Ndiaye, Mamadou Moustapha
Contribuição na implementação de ferramentas tecnológicas para a condificação de transações e integração de redes de comunicação sem fio com redes fixas / Mamadou Moustapha Ndiaye.--Campinas, SP: [s.n.], 2003.

Orientador: Vitor Baranauskas.
Dissertação (mestrado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Modo de transferência assíncrono. 2. Sistemas de comunicação sem fio. 3. Processamento eletrônico de dados. I. Baranauskas, Vitor. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.

Agradecimentos

Agradeço aos professores e meus orientadores, o Prof. Dr. Vitor Baranauskas da UNICAMP e o Prof. Dr. Carlos Fernandes Franco Jr. da USP. Agradeço também em especial ao Sr. Carlos Zanvettor do UNIBANCO e a todos que me ajudaram de uma forma ou outra no desenvolvimento deste trabalho.

Resumo

Para manter o mesmo nível de qualidade de segurança na integração das redes fixas de comunicação, que operam em modos de transferência assíncrona (ATM), com as redes móveis de comunicação, que operam em códigos de divisão temporal e acessos múltiplos (CDMA), é necessário o desenvolvimento de aplicativos com protocolos de segurança adequados aos serviços que serão oferecidos. O presente trabalho descreve a implementação de um algoritmo de controle que possibilita manter o nível de qualidade da rede móvel muito próximo ao da rede fixa, com a utilização de cartões.

Palavras Chave: 1.Modo de transferência assíncrono. 2.Sistemas de comunicação sem fio. 3.Processamento eletrônico de dados.

Abstract

In order to maintain the security and quality of services in the integration of the infrastructure of wired networks, operating in Asynchronous Transfer Mode (ATM) with the wireless networks operating in ATM Code Division Multiple Access (CDMA) it is necessary to develop application and security protocols according to the services that will be provided. The present work describes the implementation of a framework with security protocols which allows to maintain the QoS (Quality of Service) level between the wired and the wireless ATM networks, using card transactions.

Keywords: 1.Asynchronous Transfer Mode. 2.Wireless Communication Systems. 3.Electronic Data Processing.

Conteúdo

Capítulo I

Preambulo

Nos procedimentos de autorização eletrônicos o tempo de resposta constitui um parâmetro muito importante. Se houver um grande número de usuários simultâneos fazendo transações em uma rede, dependendo da topologia da rede e do nível da qualidade de serviço (*Quality of Service* – QoS) pode haver degradação, aumento do tempo de resposta e conseqüentemente resultar em uma transação sem sucesso.

Neste trabalho, integrou-se a topologia da rede fixa ATM (*Asynchronous Transfer Mode*) à tecnologia de rede *Wireless* ATM implementada usando o protocolo MAC DR-TDMA (MAHMOUD, 1996, pp. 596-608) para manter o mesmo nível de QoS. Além disso, implementou-se o algoritmo de Luhn (LUHN) que permite verificar previamente o cartão do usuário antes de iniciar a transação. O algoritmo de Luhn verifica se o número do cartão digitado está válido e corresponde ao tipo de cartão em uso, se não for, a transação é encerrada. Esta solução, além de apresentar segurança, permite minimizar o tempo real de conexão e evita possível *timeout* do sistema o que resulta em aumento do desempenho. Desenvolveu-se uma infra-estrutura integrada para soluções de negócios eletrônicos podendo ser aplicadas também em ensino à distância com videoconferência de alta qualidade sem degradação.

A necessidade de conciliar as dimensões tecnológicas, modelos de gestão e estruturas organizacionais aprimorou o interesse de desenvolvimento deste projeto de pesquisa. No quadro tecnológico, a Tecnologia da Informação - TI, inicialmente teve suas mudanças aceleradas pela difusão das redes locais e microcomputadores (anos 80 e início da década de 90), agora tem as novas estruturas de comunicação como grande fator de mudança. As *internets, intranets e extranets* dão a dimensão global aos novos sistemas (FRANCO JR., 2001). As instituições financeiras de hoje em dia devem ter toda a sua plataforma baseada em redes de comunicações robustas e seguras para suportar um número cada vez maior de

transações que, em alguns casos, chegam a milhões de transações diárias. O desenvolvimento da Internet é uma preocupação das instituições financeiras e de algumas organizações que, na busca de ferramentas de plataforma robustas de *business* voltada ao B2B (*Business-to-Business*), crescem cada vez mais. Hoje em dia, as instituições financeiras utilizam cada vez mais as habilidades do engenheiro voltadas às comunicações, transações e segurança. No comércio eletrônico, uma plataforma robusta baseada em regras de negócios, *softwares* de aplicação para Internet, hardware confiável e uma segurança bem elaborada representa um conjunto de ferramentas fundamentais

O trabalho desenvolvido é uma pesquisa sobre o protocolo de segurança implementado com base no algoritmo de Luhn. Para inserir o trabalho na infra-estrutura de redes ATM [figura 31], foi necessário desenvolver um protocolo MAC para redes *Wireless* ATM CDMA/TDMA (J.F.FRIGON, 1998) e integrá-lo na rede ATM fixa, mantendo o mesmo nível de QoS. O ambiente de aplicação apresentado neste trabalho é o *Cashless Economy* (WOODFORD, 2002, pp. 41-46) que utiliza cartões de crédito para desempenhar transações entre agentes.

Este trabalho é composto de 6 capítulos organizados de acordo com a descrição que segue. No capítulo 1 apresentam-se os componentes básicos do sistema: *Cardholer, Merchant, Acquirer, Issuer,* e *Brand.* No capítulo 2 apresenta-se fluxo e processamento de transações, ATM (*Automatic Teller Machine*), *Point of Sale* (POS), *Virtual Point of Sale* (VPOS) e voz. O VPOS foi implementado com medidas de segurança que fazem com que o *Merchant* não tenha acesso aos dados do *Cardholder* disponibilizados diretamente ao *Acquirer.* O *Merchant* vai sempre receber do *Acquirer* um relatório no final de cada dia comercial que demonstra as compras aceitas e rejeitadas. No capítulo 3 apresenta-se o *switch* e o *payment gateway*. O *switch* permite rotear as transações entre as partes, permitindo que diferentes formatos de arquivos conversem entre si com alta segurança, rapidez e contingência de falhas. No capítulo 4 apresenta-se a integração de redes *wireless e wired* ATM usando o protocolo MAC DR-TDMA. No capítulo 5 apresenta-se a segurança no processamento de cartões e a segurança das transações pela Internet, considerando os sistemas integrados da plataforma tecnológica. No capítulo 6 apresenta-se a arquitetura tecnológica proposta. No capítulo 7 apresenta-se a Conclusão Geral e Considerações Finais.

Finalmente, no capítulo 8 apresenta-se a estrutura do *website* do *Merchant* para que este receba pedidos e as processe *online* utilizando o algoritmo de *Luhn*. Foram sugeridas algumas linhas de trabalhos futuros que podem ser implementadas como base da infra-estrutura.

I.1 Descrição dos componentes do sistema

Neste capítulo são descritos os principais elementos que compõem um sistema integrado de uma plataforma tecnológica de negócios. O *Switch* que compõe o elemento fundamental de suporte à conectividade com todas as redes internacionais de autorização, *settlement & clearing*, suporte às transações com cartões de crédito, débito e pagamento *online* pela Internet. O *Switch* pode ser acoplado a vários componentes como ATM (*Automatic Teller Machine*), POS (*Point of Sale*), VPOS(*Virtual Point of Sale*), Modem, WAP (*Wireless Application Protocol*), e Palmtop. O *Switch* oferece uma plataforma integrada de processamento de transações de pagamento eletrônico através da rede privada e da Internet. *Merchant, Acquirer, Issuer, Banknet* e *Cardholder* têm um papel fundamental nas transações de comércio eletrônico que será abordado neste capítulo.

I.2 Introdução

No estado da arte da Administração, os modelos de gestão compilam as tendências de qualidade e fundem os novos modelos de gestão em que uma das principais fundamentações é a integração de processos por meio de sistemas de informação. Pode-se considerar que na economia clássica e na indústria moderna sempre houve o princípio de Economia de Escala, onde, quanto maior o volume de produção, menor os custos e maior a competitividade organizacional. Graças à evolução da tecnologia de informação as organizações contemporâneas desfrutam do princípio da Economia de Integração, onde, quanto maior a informatização, maior a integração de processos, menor volume de perdas, menor custo de estoque, menores discrepâncias, pois o cliente agora pode personalizar o seu pedido "On-line" ou ainda reduzir a alocação de trabalho intermediário, de controle ou indireto, pois o cliente acessa diretamente os sistemas, por exemplo, os sistemas bancários de informações e operações ou "*home-banking*".

A evolução de processos tradicionais internos para um ambiente aberto, virtual, dirigida pela *Web*, tem mostrado uma real definição do *e-business*, que permite a integração total interna e

externa de processos de negócios, transformar operações e automatizar as relações via tecnologia da Internet.

I.2.1 Elementos ativos de transações

Em uma transação, o cliente ou *Carholder* paga ao lojista ou *Merchant* que, por intermédio de um terminal POS ou telefone, liga para o *Acquirer* do cartão para verificar a validade do cartão e se o valor da transação não ultrapassa o limite do cartão do cliente. O *acquirer* envia a requisição para o *Banknet* que, por sua vez, envia a mesma requisição para o *Issuer*, que é o banco onde o cliente detém uma conta. O *Issuer*, após verificação de limite e validade do cartão, retorna uma resposta para o *Banknet* que, por sua vez, informa ao *acquirer* que, por sua vez, informa o *Merchant*. Se a resposta for positiva, a transação é efetivada (figura 1). Qualquer instituição ou banco que fornece o cartão plástico ao cliente é um *Issuer*, ou Emissor. No cartão do cliente é registrado o PIN (*Personal Identification Number)* que é o número de conta pessoal do cliente. O PIN é composto de um BIN (*Bank Identification Number*) que é o número de identificação do banco *issuer*. O PIN informa ao *Acquirer* sobre o banco *issuer* que forneceu o cartão. O *Merchant* aceita o cartão do *Cardholder* e encaminha as transações ao provedor da Rede de Transferência de Fundos, ou EFT (*Electronic Funds Transfert)* por meio de um terminal POS (*Point of Sale*), pela Internet usando o VPOS (*Virtual Point of Sale*), pelo terminal ATM ou pelo telefone (voz). O provedor de EFT, ou brand, pode ser o *Banknet* que, por sua vez, envia a mensagem para o *issuer*, que verifica se aprova a solicitação de liberação de valor. Só depois que o *Banknet* recebe a resposta do issuer é que ela é encaminhada de volta para o *acquirer* que, por sua vez, envia para o *Merchant* (figura 1). Coube ao acquirer, como instituição financeira, os aspectos de negócios junto ao *merchant,* definidos como: fomentar o contrato entre *acquirer* e *merchant*, prover serviços de autorização adequados aos *merchants*, gerenciar os riscos, capturar as transações, manter os *logs* de autorização, assistir nas investigações de fraudes e elaborar a fatura do *merchant*. Coube ao *issuer*, como instituição financeira, os aspectos de negócios junto ao *cardholder* definidos como: processar o cartão de crédito, processar o pagamento, emitir cartões aos usuários, autorizar as transações dos usuários, prestar serviços aos usuários, elaborar planos de marketing, processar as requisições de autorização, responder aos chamados, controlar fraudes, manter arquivos de transações e elaborar a fatura do usuário.

I.2.3 Banknet

O *Banknet* é um Processador EFT, um sistema composto principalmente de redes de telecomunicações (X.25, Frame Relay, ATM e TCP/IP) e de sistemas de aplicações para o processamento de dados. Ele provê dados de diferentes formatos e tipos por meio de linhas de comunicações.

I.2.4 Switch

Um *Switch* é um dispositivo com muitas portas que provê conexões dinâmicas lógicas entre 2 segmentos sem a intervenção de um operador. Ele é um dispositivo de alta velocidade, pois, caminhos de dados podem ser estabelecidos e usados simultaneamente. O *switch* tem também as características de um *gateway* de pagamento. Um *gateway* conecta 2 ou mais redes de comunicações baseadas em diferentes protocolos. Ele provê qualquer conversão de protocolos. O *switch* rotea transações entre vários tipos de interfaces. Um roteador tem 2 ou mais interfaces de redes. Ele examina o endereço IP (*Internet Protocol*) do protocolo do Software, seleciona um caminho apropriado e encaminha os protocolos entre as redes separadas. As transações mostradas na figura 1, são encaminhadas ao *Banknet* para Autorização, *Settlement* e *Clearing* . Descrevemos a integração do *switch* na plataforma técnica do capítulo 3.

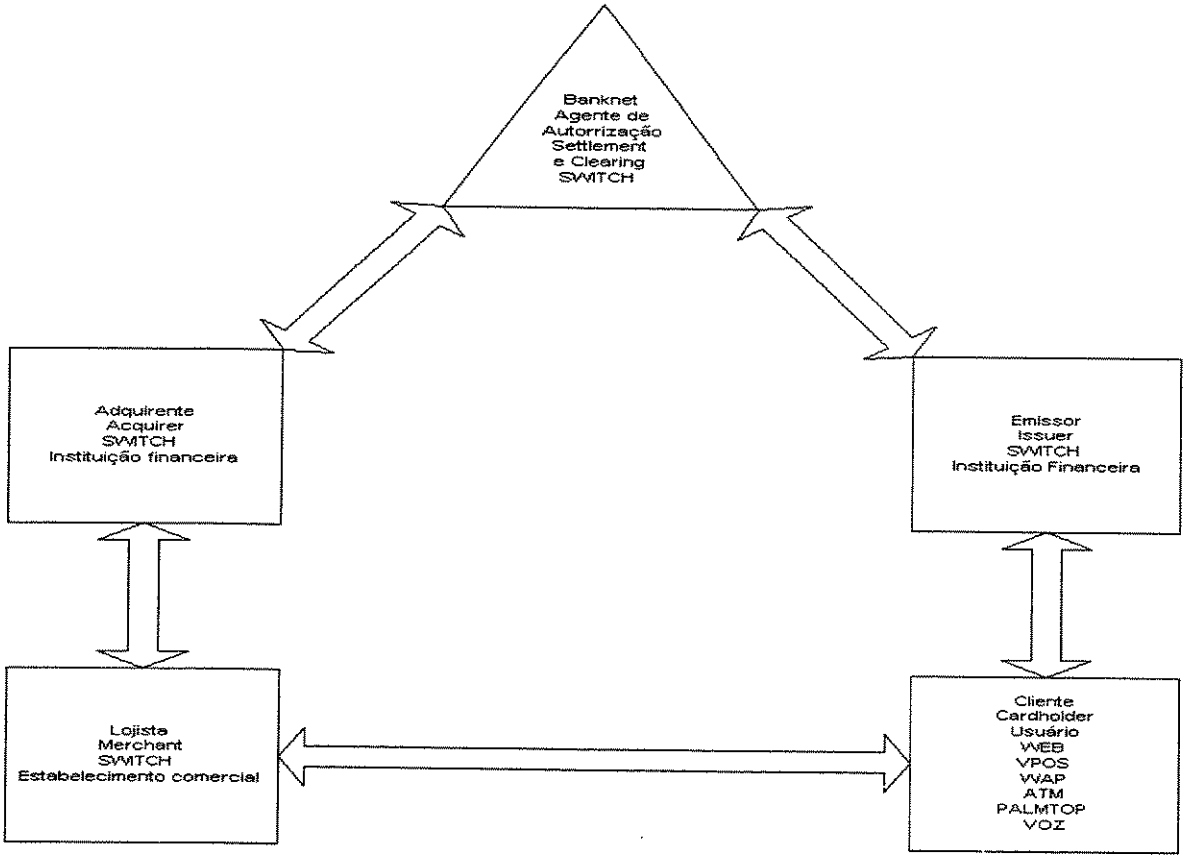


Figure 1: *Modelo Transacional*

I.2.5 Autorização

O módulo de autorização do *Switch* autoriza ou nega as requisições de transações que não podem ser roteadas para o issuer por causa de uma falha na conexão da rede, ou por causa de tempo esgotado (*Time-Out*).

Se o *issuer* falhar na resposta de uma requisição de transação, o módulo de autorização responde provendo uma autorização positiva ou negativa, ou negando esta e as demais transações, dependendo de como foi configurado.

O *Switch* é configurado para um dos seguintes níveis de autorização que são mutuamente excludentes: positivo, negativo ou proibido.

Com autorização positiva ou negativa, o *Switch* espera pela instituição *issuer* que não pode temporariamente ou permanentemente fazer suas próprias autorizações.

Autorização positiva: o módulo de autorização pode acessar todas as informações a respeito do *Cardholder* e as contas correntes dele, permitindo lhe aprovar ou negar as transações.

Autorização ON-US: o módulo de autorização ON-US permite requisição de aprovação do *Merchant* direto ao *issuer*. As informações que ele acessa incluem os extratos das contas do *Cardholder.*

Autorização negativa: o módulo de autorização negativa não acessa as informações de conta corrente do cliente. Ele pode apenas aprovar o débito que ultrapassar o limite de um cliente pré - definido. Ele não pode saber se o valor a ser autorizado é maior ou menor que o valor na conta corrente do cliente. É por isso que a autorização negativa apresenta maiores riscos para o banco *issuer* do que a autorização positiva.

Autorização proibida: o módulo de autorização é proibido aprovar qualquer transação relacionada ao banco *issuer*. Neste caso, o *Switch* se comporta como um roteador entre as instituições *acquirer, issuer* e entre as redes.

I.2.5.1 Verificações de pré-autorizações

As requisições de transações podem ser sujeitas a algumas verificações preliminares antes do módulo de autorização tomar conta delas. Isto é opcional e é definido pelos usuários do *Switch.* As verificações são como segue:

1-Verificar o limite: se a requisição excede o limite pré - fixado pelo *issuer*, a requisição é negada.

2-Expedição da data de validade: as requisições de transações de cartões com data expirada são negadas.

3-Verificar o *check digit*: o número de conta primária *PAN* (*Primary Account Number*) do cartão é verificado pelo dígito de verificação para prevenir seu uso fraudulento.

4-Verificar o valor de verificação do cartão (CVV): o CVV e a data de expiração do cartão são verificados usando rotinas que extraem dados dos *tracks* de segurança.

5-Verificar o código de serviço: verificar se o cartão é restrito ao uso local, apenas, ou pode ser usado para transações internacionais. Cartões locais não podem ser usados para transações internacionais.

6-Verificar o *status* do cartão. Verificar se o cartão é "quente". A requisição é rejeitada se exceder qualquer limite. Se a requisição é uma transação de cartão de débito, o PIN do POS é verificado.

7-Verificar entradas excessivas de PIN: o limite de número de vezes de tentativas sem sucesso quando o usuário quer acessar a conta dele pode ser configurado.

I.2.5.2 Procedimento de autorização

Depois do processo de pré - autorização, o servidor de autorização executa as seguintes tarefas:

1-Verificar o tipo de mensagem: o servidor de autorização verifica o tipo de mensagem para determinar se a transação é uma requisição de autorização para crédito ou uma transação de cartão de débito, ou, ainda, se é uma mensagem financeira de um dispositivo ATM que requer um PIN de verificação.

2-Verificar o código de processamento: O servidor de transações determina qual conta para debitar ou creditar. Ele verifica a existência da conta e se a transação é permitida. Ele também realiza a conversão de moedas, se necessário.

3-Verificar limites: ele verifica se o valor pedido está dentro do limite permitido e se o número total de transações num dado período não excedeu o permitido neste período. Ele assegura, também, que o valor total das transações não excedeu o máximo permitido por um dado período.

4-Se todas as verificações acima foram válidas, as contas apropriadas serão debitadas.

5-Transações POS: para transações POS, cartões de débitos, cartões de crédito, ele verifica o PIN, realiza o mesmo ou similar às verificações em 1 a 3, discrimina os valores de extratos de contas disponíveis, e retorna a resposta. Ele realiza, também, transações reversas.

I.2.6 Settlement Inet

O *Settlement* é a transferência de fundos decorrente das transações feitas, sujeitas a uma contabilidade. O INET é um sistema de *settlement*. Ele estabelece o balanço entre as diferentes partes (*Acquirer*, *Issuer* e *Merchant*) envolvidas na transação, para definir a quantia que cada uma deve receber nas transações diárias. O *Settlement* é o processo de reconciliação entre o débito e crédito, entre *issuers* e *acquirers*, ou seja, é o processo elaborado no fim de cada dia de negócio. Os dados de *settlement* de cada transação são armazenados e consolidados em um banco de dados que registra a atividade financeira total de cada instituição. O banco de dados de *settlement* é gerenciado pelo servidor de *settlement* que recebe as mensagens do tipo 15xx, utilizadas para extrair os dados e atualizar o banco de dados com registros recentes de transações no *cache* da memória do servidor. O servidor atualiza o banco de dados para cada um destes tipos de *settlement:* crédito, crédito reverso, débito, débito reverso, pedido, transferência, transferência reversa, autorização, e autorização reversa.

I.2.7 Procedimento

O cartão de débito permite acessar o dinheiro em conta em uma agência bancária ou em uma organização financeira. O cartão de crédito permite acessar uma conta de uma organização de cartão de crédito que paga pela transação no momento (até o limite de crédito estipulado no cartão) e, depois, cobra do cliente a soma total de todas as transações do mês em que o cartão foi utilizado e, também, a anuidade do cartão decorrente do serviço oferecido ao cliente.

Para o *cardholder* que possui um cartão de débito ou de crédito, retirar dinheiro de uma caixa ATM ou fazer compras em uma loja consiste em inserir o cartão em um dispositivo e executar algumas instruções. Para transações feitas a partir de um terminal POS ou VPOS, a cobrança é feita a cada fim do mês, enquanto que as transações feitas em *ATMs* são debitadas na conta do *cardholder* na hora e listadas no extrato da sua conta corrente. Os fundos descritos acima são enviados eletronicamente de um sistema para um outro com formato pré - determinado.

Estas transações são roteadas por meio de comunicações através de diferentes redes para as instituições financeiras de onde seus registros financeiros são acessados. Cada instituição financeira é identificada por um número de identificação do banco, chamado BIN, que mostra sua aliança com diversas redes globais como *Visa* ou *MasterCard.* Com as mudanças do ambiente global de negócios, a conectividade destas redes diversas, por um *switch* central, se torna uma necessidade fundamental para muitas instituições financeiras.

O *switch* que foi implementado nesta plataforma tecnológica provê todas estas necessidades e apresenta uma funcionalidade modular que lhe permite crescer com as oportunidades futuras de negócios.

I.3 Considerações

Neste primeiro capítulo foram apresentados os componentes fundamentais que compõem a plataforma tecnológica de *e-business* orientado ao *Business-to-Business.* Introduziu-se, também, o procedimento de autorização nas transações financeiras tendo em vista a utilização de cartões de crédito e débito. No segundo capítulo serão focalizados, com mais detalhes, os fluxos de transações, tipos de autorizações, *settlement* e *clearing.*

Capítulo II

Fluxo e Processamento de Transações

II.1 Cartões

O primeiro cartão de crédito chamado *Charge-It* foi apresentado no FNBB *(Flatbush National Bank of Brooklyn)*, em 1946, por *John Biggins*. O programa usava um *script* em papel para substituir a moeda que era aceita para os *Merchants* para compras de pequenos valores. Depois da venda, o *merchant* depositava o *script* em uma conta de um banco, que cobrava do cliente o valor total do *script*. Em 1951, o primeiro cartão de crédito é introduzido pelo FNBB. Na mesma época, o *Diners Club* introduziu o primeiro cartão de viagens e entretenimento, cartão *Travel and Entertainment* (T&E). O interesse a respeito de cartão de crédito cresceu espetacularmente quando o *Bank of* América introduziu, em 1960, o *Bank-Americard* conhecido hoje como Visa.

II.2 Tipos de Cartões

II.2.1 Cartão de débito

O Cartão de débito é geralmente emitido por uma instituição financeira. O Cartão de débito permite ao detentor acessar suas contas corrente e poupança. Dependendo das operações de negócios do issuer, o cartão de débito pode ser utilizado em ATMs, em terminal POS no site do *Merchant* equipado com *PinPads*. No terminal ATM, o cartão de débito é usado para retirar dinheiro, fazer depósitos, transferir fundos entre contas corrente, obter saldo e extrato e, em alguns casos, pagar contas. No *site* do *Merchant*, o cartão de débito é utilizado para fazer compras e transferir fundos imediatamente da conta do *cardholder*.

II.2.2 Cartão de crédito

O Cartão de crédito é usado por usuários aprovados com um limite definido. O Cartão de crédito é usado para fazer compras no estabelecimento do *Merchant*. Compras realizadas com um cartão de crédito são acumuladas como um crédito. Um extrato é enviado ao *cardholder*, sendo que todas as compras de um período definido e a taxa anual devida ao banco *issuer* são listadas. O cartão de crédito pode ser utilizado no terminal POS (Point Of

Sale) *PinPad* do site do *Merchant.* O número de cartão de crédito pode também ser utilizado no VPOS (*Virtual Point Of Sale*). O VPOS é um *Virtual POS* do *Merchant* acessível ao *cardholder* a partir de um *browser.* Em vez de um *PinPad* no site do *Merchant*, o VPOS permite ao *cardholder* acessar o *website* do *Merchant* via *internet* e realizar suas compras.

O VPOS desenvolvido neste trabalho é descrito com mais detalhes no capítulo 8 e ainda pode ser acessado a partir de um *Palmtop* ligado na Internet. Um outro canal de contato (*touch point)* desenvolvido para o acesso ao *Site* virtual do *Merchant* é o telefone celular ou um dispositivo *wireless*. Assim o VPOS permite a integração da tecnologia *WAP* à plataforma tecnológica de rede fixa.

II.2.3 Cartão com propósito múltiplo

Um cartão pode ser utilizado com propósito múltiplo, isto é, ele pode ser utilizado como cartão de crédito e como cartão de débito ao mesmo tempo, dependendo do tipo de conta escolhida pelo usuário.

II.2.4 Cartão de Viagens e de entretenimento

O Cartão de Viagens e entretenimento é emitido pelas companhias como *American Express* e *Diners Club.* Geralmente, não há crédito pré - definido, desde que o valor devido seja pago todo mês.

II.3 Fluxo de Transações

II.3.1 Fluxo de Transação ATM

Em uma transação típica de ATM, o *cardholder* requer um serviço colocando o cartão em uma máquina ATM, digitando sua senha ou PIN e selecionando o serviço que deseja. Dependendo da locação do Issuer e do Acquirer, a transação é processada por muitas redes antes de chegar ao *Issuer.*

II.3.2 Fluxo de Transação do POS

Em uma transação típica de POS, o *cardholder* faz compras em um site de *Merchant.* O *Merchant* passa o cartão do *cardholder* no terminal de POS chamado de *PinPad* e digita o valor da compra. O terminal POS é programado para se conectar ao centro de processamento

do *acquirer* que aceita a requisição e encaminha para a instituição financeira proprietária do cartão que, na sua volta, encaminha ao *issuer* do cartão. Dependendo da locação do *acquirer*, da instituição proprietária da marca do cartão, *brand* e do *issuer*, a transação é processada por muitas redes antes de chegar ao *issuer*.

II.3.3 Fluxo de transação por voz

Em uma transação típica por voz, o *cardholder* faz compras no *website* do *Merchant*. Se o *Merchant* não tiver um terminal POS, ou caso uma resposta indicando, chamar o centro de processamento do *acquirer* para efeito de autorização foi recebida, então o *Merchant* chama o *acquirer*. O centro de processamento aceita a requisição e a encaminha ao *brand* que, por sua vez, encaminha ao *issuer* do cartão. Dependendo da locação do *acquirer*, do *brand* e do *issuer*, a transação é processada por muitas redes antes de chegar ao *issuer*.

II.3.4 Fluxo de transação típica

A figura 2 mostra o fluxo de transação comum.

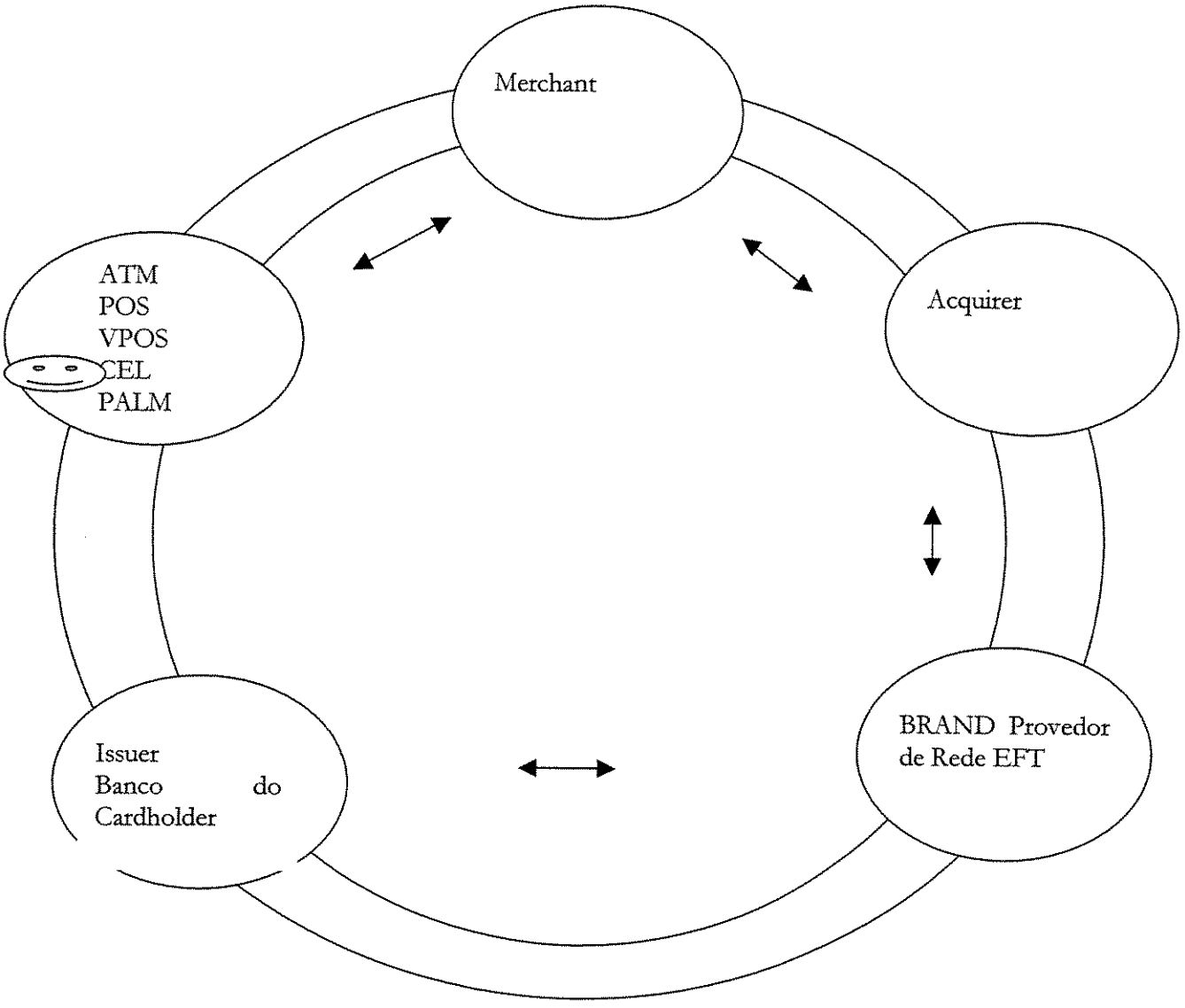


Figura 2: *Fluxo de transação convencional*

Em um ATM, em um terminal POS, em um VPOS, em um Celular ou *Palmtop*, por exemplo, um *cardholder* usa um número de cartão 65677120000111227 e solicita um valor de R$1000,00. O Banco que possui o ATM ou que contratou o *merchant* para adquirir as transações aceita a requisição e rotea a requisição de autorização para o agente de autorização (o provedor de rede de Transferência de Fundos Eletrônicos, *EFT* ou *BRAND*), como definido pelo BIN. O Provedor de Rede aceita a requisição de autorização e a rotea pela rede global ao *issuer*. A requisição de autorização do cartão 65677120000111227 de um valor de

R$1.000,00 com número do *acquirer* 4502221 é enviada pela rede de comunicação internacional ao Centro de Processamento da rede do provedor pelo outro lado do continente. O valor é convertido em moeda local. O centro de processamento da instituição financeira do *issuer* recebe a requisição de autorização, verifica a conta do *cardholder* e envia a resposta ao *acquirer* pelo provedor de rede EFT (*Eletronic Fund Transfer*).

II.4 Processamento de Transações

Transações consistem em várias classes de mensagens e funções específicas que fluem através de dispositivos, redes, bancos e *switch*. Nesta seção serão descritas as autorizações 01xx, as transações financeiras 02xx, e as transações reversas 04xx.

II.4.1 Fluxo de transações através de rede

A figura 3 mostra como as transações são roteadas para o *issuer*.

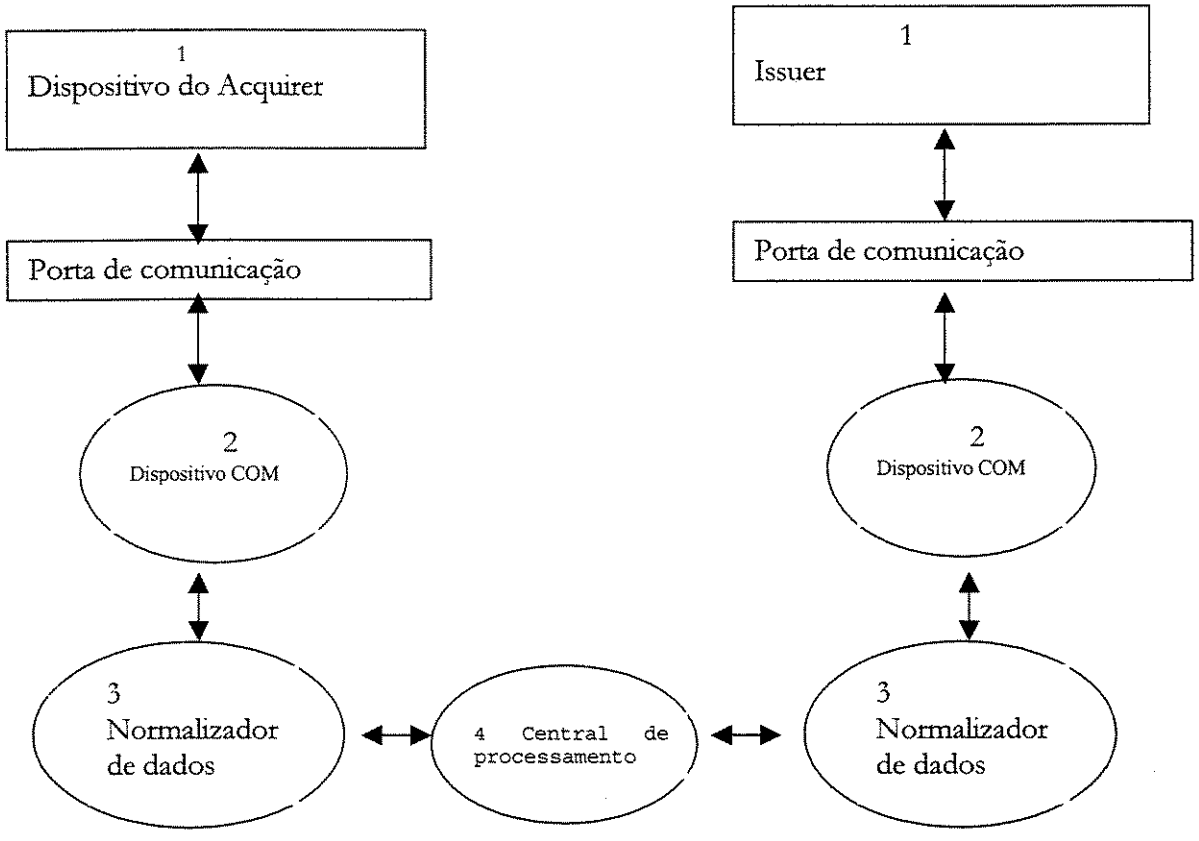


Figura 3: *Fluxo de transações*

A seqüência é definida da seguinte forma:

1. O *cardholder* inicia uma requisição à transição no dispositivo/servidor;

2. A requisição é enviada ao *Switch.*

3. A requisição é encaminhada para o dispositivo de comunicações pela porta.

4. A requisição é traduzida de um formato externo(xmf) para um formato interno (imf) pelo normalizador de dados.

A mensagem é encaminhada para ser processada.

Se a autorização não for realizada pelo *switch*, a requisição é passada para o normalizador de dados para efeito de transação de (imf) para xmf.

Uma vez em xmf, a requisição é enviada para o dispositivo de comunicação e depois para o servidor do *issuer* ou rede via porta de comunicação.

Depois do servidor do *issuer* ou da rede aprovar ou negar a requisição, a mensagem é enviada de volta pelo *switch* como uma mensagem de resposta, então o processo é feito em sentido reverso.

II.4.2 Autorização 01xx e Mensagem de Transações Financeiras 02xx

A classe de mensagens 01xx é geralmente usada para requisição de autorização nas transações com cartão de crédito. Para a transação proceder, a requisição deve ser aprovada.

Código da mensagem	Descrição da mensagem
0100	Requisição de autorização
0110	Resposta de autorização
0120	Aviso de autorização
0130	Resposta de aviso de autorização

Tabela 1: Classe de mensagens de Autorização

A classe de mensagens 02xx é geralmente usada para requisição financeira. A aprovação desta requisição pode afetar imediatamente a conta do *cardholder*. A maior parte deste tipo de requisição é feita com cartão de débito.

Código da mensagem	**Descrição da mensagem**
0200	Requisição financeira
0210	Resposta financeira
0220	Aviso financeiro
0230	Resposta de aviso de financeiro

Tabela 2: Classe de mensagens Financeiras

A mensagem 0100 é uma requisição de Autorização. A mensagem 0200 é uma Requisição Financeira.

A mensagem 0110 aceita ou nega a requisição 0100. A mensagem 0210 aceita ou nega a requisição 0200.

A mensagem 0120 é um aviso de uma requisição de autorização. A mensagem 0220 é um aviso de uma requisição financeira.

A mensagem de aviso é gerada - caso o *switch* realize um processamento *stand-in* no servidor que foi configurado de maneira a armazenar e encaminhar mais tarde a operação - depois que a comunicação seja restabelecida.

A mensagem 0130 é enviada à resposta de uma mensagem 0120. A mensagem 0230 é enviada à resposta de uma mensagem 0220.

II.4.3 Fluxo padrão de mensagens 01xx / 02xx

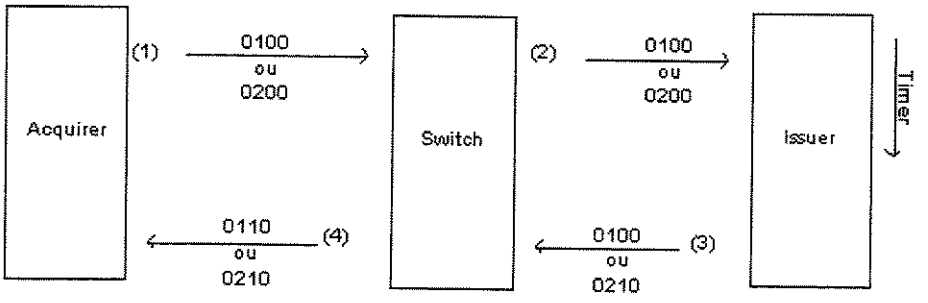


Figura 4: *Fluxo padrão de mensagens* 01xx/02xx

A seqüência é definida da seguinte forma:

(1)-O *acquirer* inicia uma requisição de autorização 0100 ou uma requisição financeira 0200 para o *switch.*

(2)- O *switch* por uma vez encaminha a requisição para o *issuer.*

(3)- O issuer envia uma resposta financeira 0210 para o *switch.*

(4)- O *switch* encaminha a mensagem 0110 ou 0210 para o *acquirer.*

II.4.4 Fluxo de Mensagens 01xx / 02xx com falha de comunicação de exceção

A figura mostra uma instância onde o *issuer* não pode conceder uma autorização devido à falha de comunicação.

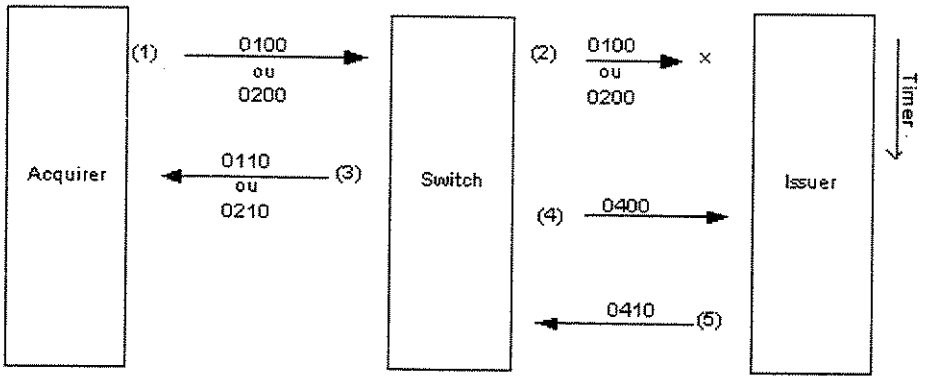


Figura 5: *Fluxo de mensagem de exceção para requisição* 01xx *e* 02xx.

A seqüência é definida da seguinte forma:

(1)-O *acquirer* inicia uma requisição de autorização 0100 ou 0200 para o *switch.*

(2)-O *switch* tenta encaminhar para o issuer a mensagem 0100 ou 0200,

(x)-mas o issuer não pode responder devido a uma falha de comunicação. Se o *switch* não entra em processamento *stand-in*, ele nega a autorização.

(3)- Mesmo se o *switch* entra em processamento *stand-in*, ele define que a transação deve ser negada.

(4)-O *switch* gera uma resposta de autorização 0400,

(3)- ou gera uma resposta financeira 0210 informando ao *acquirer* que a autorização é negada.

(3)- Se for configurado para fazer isto, o *switch* inicia uma requisição reversa 0110 e encaminha a mensagem para o *issuer.*

(5)-O *issuer* responde com uma resposta de requisição reversa 0410 apropriada para o *switch.*

Observação: As mensagens reversas devem ser enviadas como mensagens de aviso reversas 0420.

II.4.5 Processamento *stand-in:* fluxo de mensagens 01xx / 02xx

Se o *switch* pode realizar a autorização, ele também provê o processamento *stand-in.* A figura 6 mostra uma instância onde o *issuer* não pode prover autorização devido à falha de comunicação.

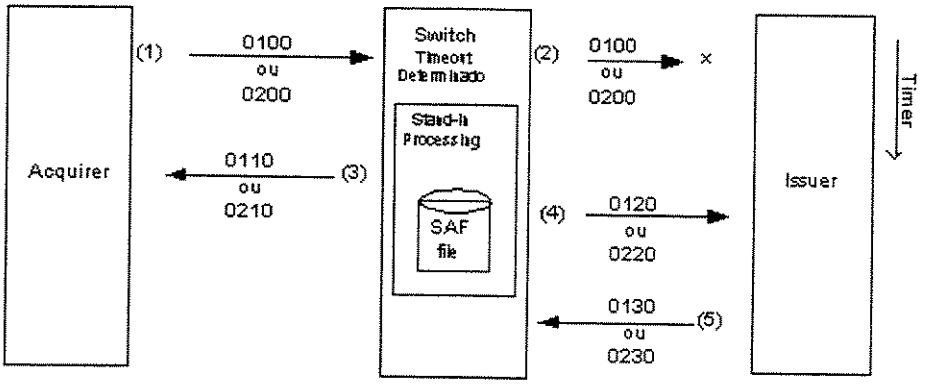


Figura 6: *Processamento no switch de mensagens* 01xx ou 02xx

A seqüência de mensagens é como segue:

(1)-O *acquirer* inicia uma requisição de autorização 0100 ou requisição financeira 0200 para o *switch.*

(2)-O *switch* tenta encaminhar para o issuer a mensagem 0100 ou 0200, mas, devido a uma falha de comunicação, o *issuer* não pode responder esta mensagem.

(3)-Se o *issuer* permite ao *switch* prover uma autorização, o *switch* gera uma resposta de autorização 0110 ou uma resposta financeira 0210 que envia a mensagem para o *acquirer.*

(4)-O Switch encaminha o aviso de autorização 0120 ou o aviso financeiro 0220 para o *issuer.* Se continuar a falha de comunicação, o *switch* escreve a mensagem no arquivo *log* do SAF (*Store And Forward*) e certifica periodicamente o log do SAF em busca de mensagens que foram retransmitidas ao *issuer* em ordem referencial. O processo de transmissão continua uma vez que a comunicação foi restabelecida. Assim que o servidor recebe todas mensagens de aviso no *log* do SAF, o arquivo *log* é removido e o *issuer* recebe novamente as transações de autorização.

(5)-Assim que o *issuer* recebe a mensagem 0120 ou 0220, ele responde em uma resposta de aviso de autorização 0130. O *issuer* usa também a informação 0120 ou 0220 para atualizar os *logs* de transação.

II.4.6 Fluxo de Mensagens Reversas 04xx

A classe de mensagens 04xx realiza requisições reversas parciais ou completas. O *switch* pode gerar estas transações sob as seguintes condições:

-Falha no terminal;

-Atraso da resposta do *issuer;*

-Falha de sistema.

O *switch* usa as seguintes mensagens reversas:

Código da mensagem	**Descrição da mensagem**
0400	Requisição Reversa
0410	Resposta Reversa
0420	Aviso Reverso
0430	Resposta de aviso Reverso

Tabela 3: *Classe de mensagens Reversas*

A figura 7 mostra o fluxo das mensagens 04xx

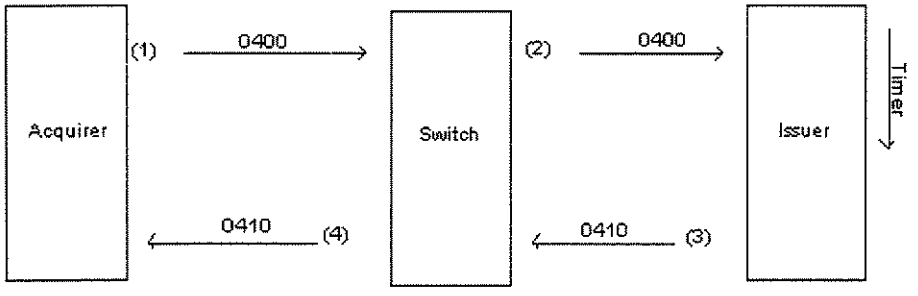


Figura 7: *Fluxo padrão de mensagens* 04xx

A seqüência de mensagens é descrita como segue:

(1)-O *acquirer* inicia uma requisição reversa 0400 para o *switch.*

(2)-O *switch* encaminha a mensagem 0400 para o *issuer.*

(3)-O issuer responde à mensagem 0400, enviando uma resposta reversa 0410.

(4)-O *switch* encaminha para o a resposta reversa 0410.

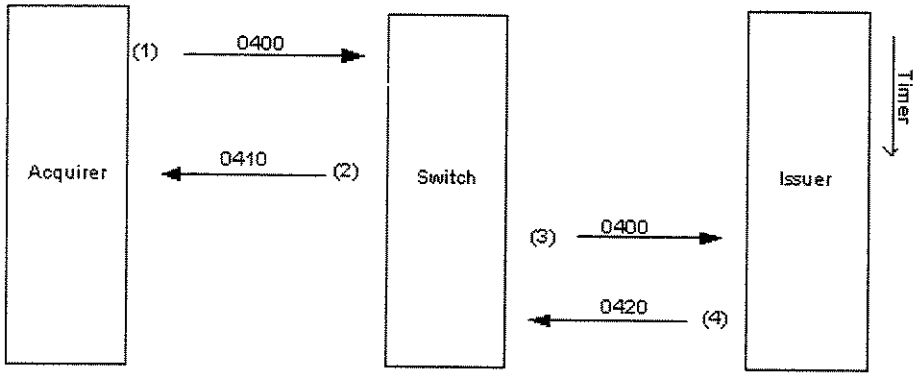


Figura 8: *Fluxo alternativo de mensagens* 04xx

A seqüência de mensagens é como segue:

(1)-O *acquirer* envia uma requisição reversa 0400 para o *switch.*

(2)-O *switch* responde ao *acquirer* com uma resposta reversa 0410.

(3)-O *switch* então encaminha a requisição para o *issuer* como mensagem de aviso 0420.

(4)-O *issuer* responde como um aviso reverso 0430 a mensagem 0420.

II.4.7 Fluxo de Mensagens Reversas completas ou parciais

A figura mostra um fluxo de mensagens que ocorre quando a transição requer uma mensagem reversa completa ou parcial.

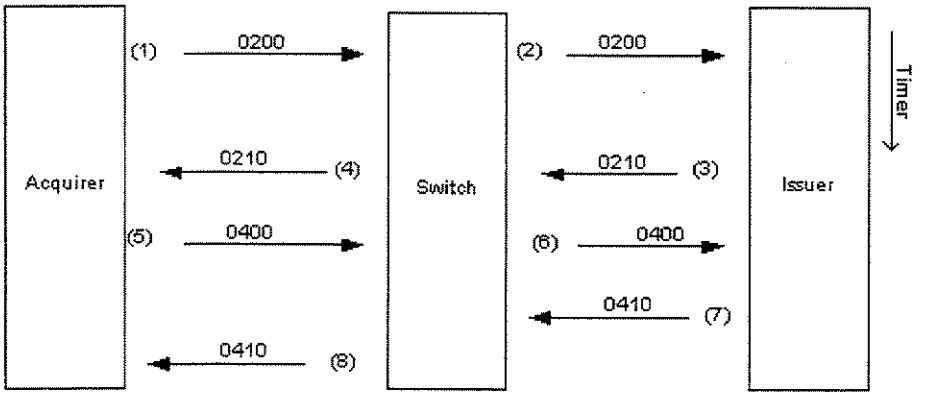


Figura 9: *Exceção de mensagem para mensagens* 04xx

A seqüência de mensagens é como segue:

(1)-O *acquirer* envia uma requisição financeira 0200 para o *switch.*

(2)-O *switch* encaminha a mensagem 0200 para o *issuer.*

(3)-O issuer gera então uma resposta financeira 0210 e responde ao *switch.*

(4)-O *switch* encaminha a mensagem 0210 para o *issuer.*

Na mensagem reversa completa ou parcial definida, o *acquirer* gera e envia uma requisição reversa 0400 para o *switch*

(5)-O *switch* encaminha a mensagem 0400 para o *issuer.*

(6)-O *issuer* gera, então, uma resposta reversa 0410 e responde para o *switch.*

(7)-O *switch* encaminha a mensagem 0410 para o *acquirer* .

Todas as transações que passam através do *switch* são gravadas em um *log* de transações localizadas no banco de dados do *switc*

Capítulo III

Switch e Payment Gateway

Neste capítulo são abordados o *switch* transacional e o *switch* ATM (*Asynchronous Transfer Mode*) que constituem os elementos fundamentais da arquitetura da plataforma tecnológica proposta neste trabalho [figura 31] em termos de segurança, interoperabilidade, velocidade, capacidade e volume de transações. Estes *switches* integrados além de serem tecnologias novas, são também tecnologias do futuro das mega-redes rápidas e inteligentes. Os equipamentos, terminais, dispositivos, estações de trabalho, servidores e outros componentes e dispositivos ligados a eles, a configuração, e a introdução de novos canais interligados ao *switch* como o *WAP* e o *Palmtop* são também abordados. A arquitetura completa está descrita no capítulo 5.

III.1 Switch Transacional

O *switch* transacional financeiro é um dispositivo de roteamento entre origens e destinos de informações chamados de nós. O *switch* suporta redes EFT (*Electronic Funds Transfer*) como: *Cirrus*, *Mastercard, Plus*, *Visa*, *Maestro*. O *switch* é portável em plataforma *Unix,* ele é capaz de rotear informação financeira a partir de terminais POS, dispositivos ATMs para bancos e redes financeiras. O *switch* provê roteamento de transações de Transferência de Fundos Eletrônicos (EFT) e Intercâmbio de Dados Eletrônicos (EDI – *Electronic Data Interchange*).

O *switch IST (Information Switching Technology)* foi considerado devido à sua arquitetura modular, à sua capacidade de poder suportar um grande número de dispositivos, dependendo do *hardware*; comutar transações entre redes de servidores e realizar autorização, dependendo da instituição financeira.

III.1.1 Arquitetura modular

A arquitetura do *switch* é composta de camadas modulares. Cada camada oferece um serviço específico. Estes serviços desempenham as seguintes funções:

- Rotear transações nas redes;

- Gerenciar e sincronizar eventos;
- Gerenciar tarefas e operações das transações;
- Comunicar com dispositivos externos;
- Comunicar entre processos;
- Acessar banco de dados.

O diagrama abaixo ilustra o desenho modular do *switch* (ver figura 10, Fonte: IST(*Information Switching Technology*))

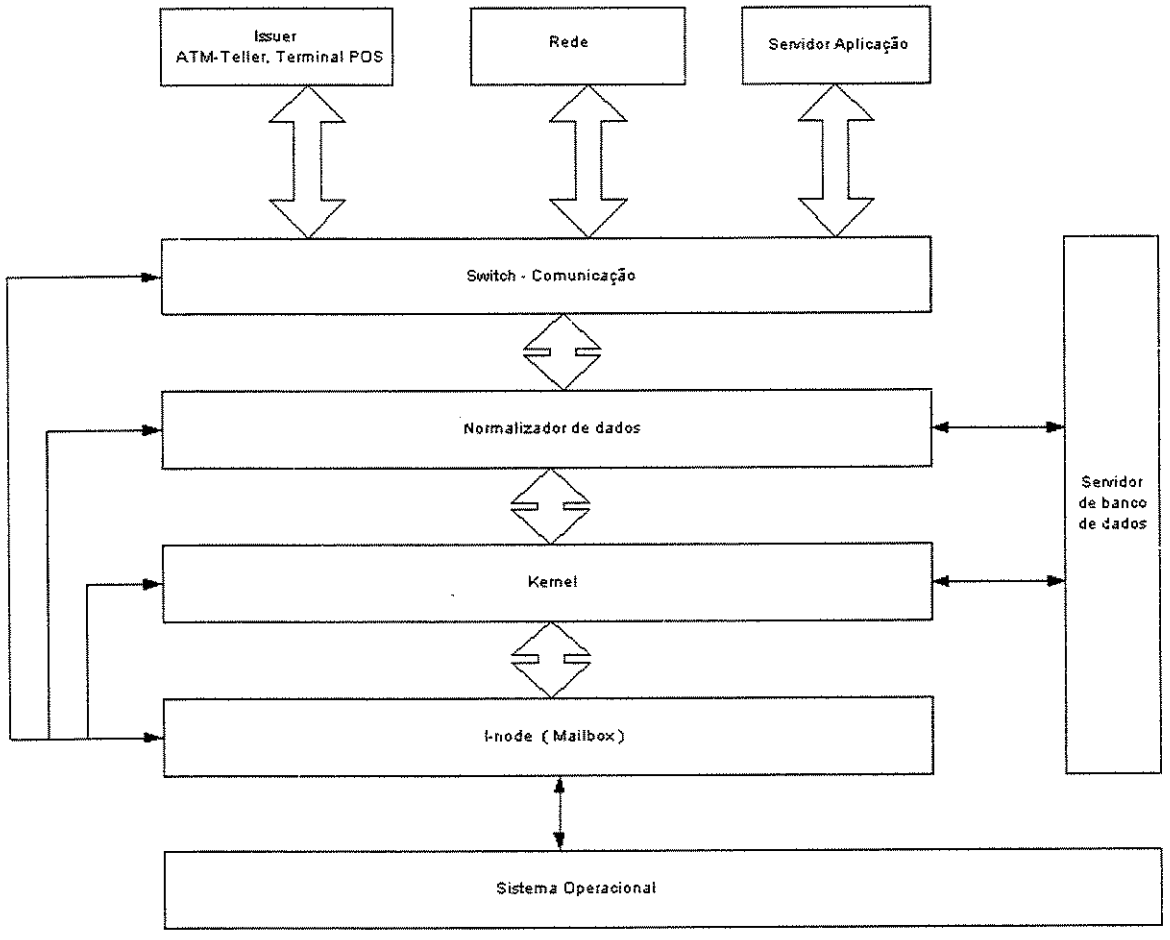


Figura 10: *Arquitetura modular do switch*

III.1.2 Determinação do issuer

O *switch* deve realizar um número de operações para processar uma transação. Quando a transação chega, a primeira coisa a fazer é determinar o ***issuer***. O processo básico

é pegar o PAN da transação e utilizá-lo para achar o BIN do *acquirer*. O *Switch* usa o banco de dados para achar o *acquirer*. O índice de pesquisa no banco de dados é o BIN. Desde que o BIN tenha um valor maior que 10 dígitos, o *switch* começa considerando os 10 primeiros dígitos do PAN. Se não achar, ele recomeça a pesquisa com 9 dígitos, assim, continua decrescendo o número de dígitos até achar o BIN. O BIN resultante da pesquisa será utilizado para definir a lógica de negócio para processar a transação. Assim, isto vai informar qual porta ou normalizador que deve ser utilizado para enviar a transação ao *acquirer* e qual medida deve ser tomada se não obtiver resposta.

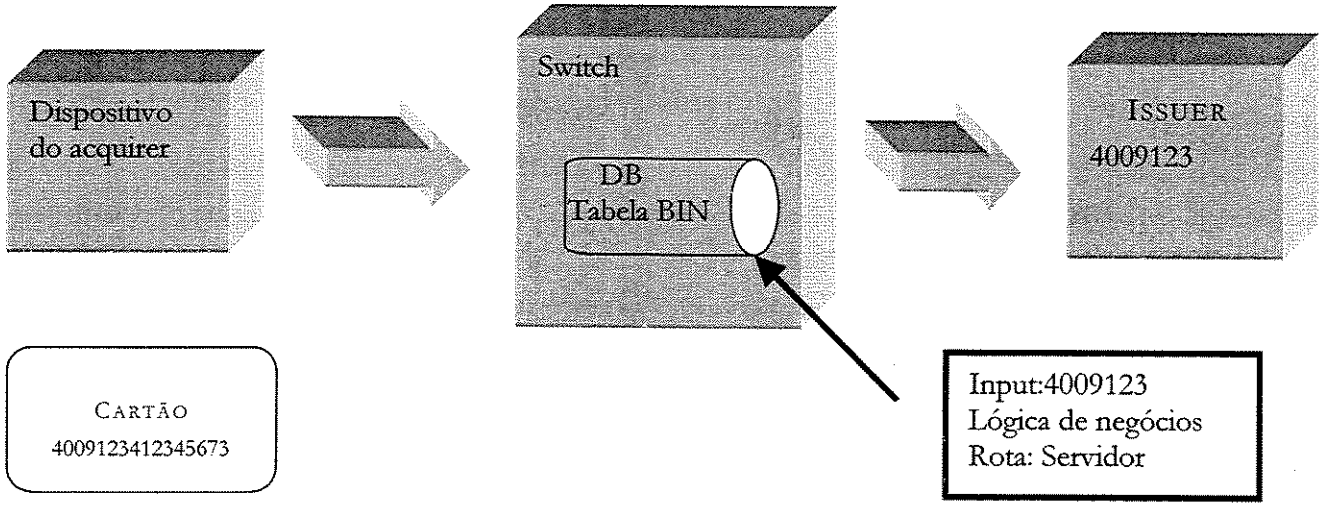


Figura 11: *Operação do switch para determinar o issuer da transação*

Se o *switch* tem uma conexão com a rede , não fica muito prático ter uma entrada BIN para cada *acquirer* que faz parte da rede. Além da dificuldade de se criar todas essas entradas, a manutenção disto seria muito complexa . Felizmente, as redes internacionais têm tendências de ter os mesmos prefixos – exemplos: os cartões da *Visa* começam pelo dígito 4 e os cartões da *Mastercard* começam pelo dígito 5. Para definir uma rede como um *issuer*, uma entrada é feita na tabela BIN utilizando o prefixo da rede considerada. Este prefixo deve ser apenas o primeiro dígito de BINs que pertence à rede.

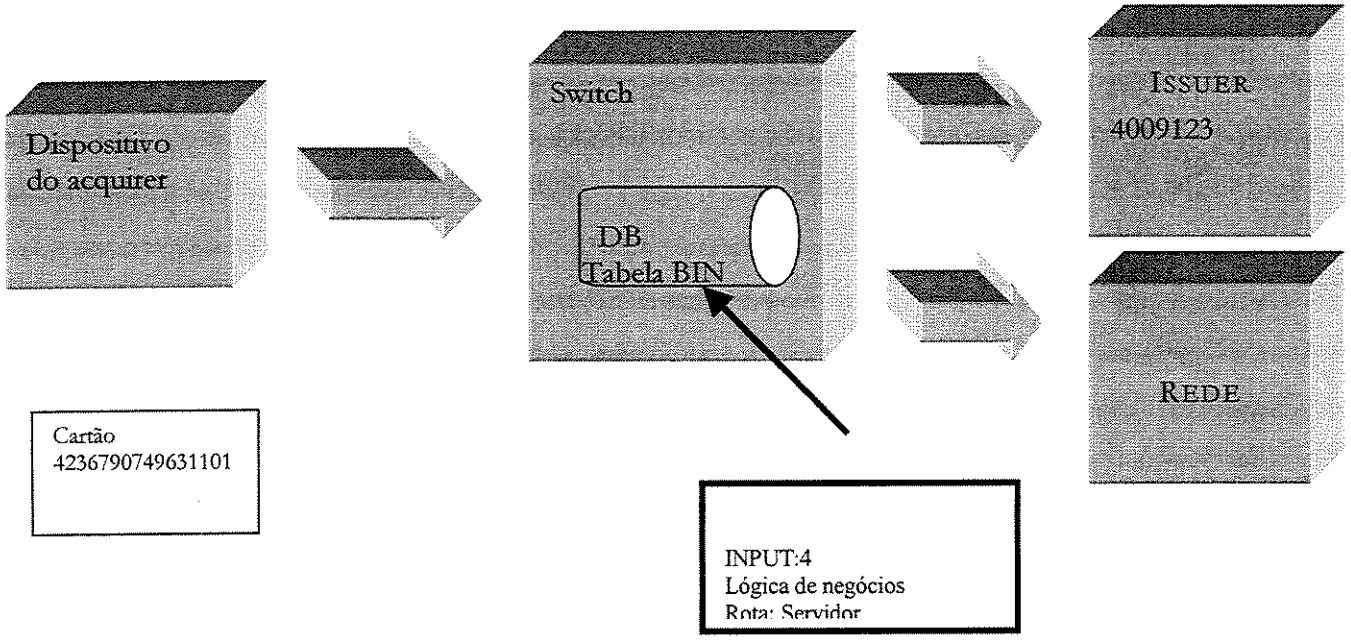


Figura 12: *Definindo a rede de transação do issuer*

No diagrama acima, se o cartão começando com 4009123 for usado, os sete primeiros dígitos vão corresponder a uma entrada BIN de um *issuer* para que a transação seja roteada. Outros cartões começando pelo prefixo 4 vão coincidir com o registro e a pesquisa continuará até que o BIN que compuser apenas o primeiro dígito seja achado. Este registro vai resultar na transação que está sendo enviada pela rede Visa.

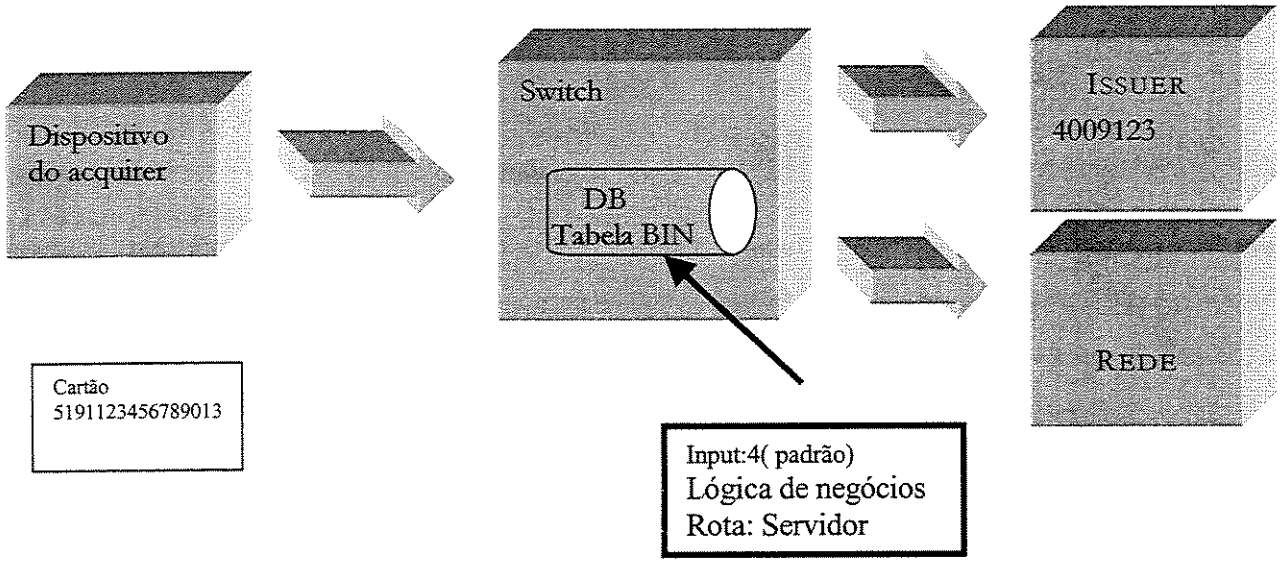


Figura 13: *Definindo o issuer padrão da transação.*

Na figura 13, se o cartão começando com 5191123 for usado, os sete primeiros dígitos vão corresponder a uma entrada BIN de um *issuer*. O primeiro número do Cartão é 5 diferente da rede de input 4, padrão. A pesquisa da rota pela tabela BIN é feita utilizando o prefixo 5 da rede MasterCard.

III.3 Características do *Switch*

Algumas características do Switch são:

-Suporte ATM (*Automatic Teller Machine*)

-Gerenciamento centralizado de ATMs ;

-Configuração de instituições financeiras;

-Monitoramento de ATMs;

-Transações e *Cash;*

-Suporte terminais POS (*Point Of Sale*);

-Conexões com muitos dispositivos de POS;

-Gerenciamento de *Merchants;*

-Transações débito e crédito;

-EDC (Electronic Draft Capture).

III.3.1 Suporte VPOS, WAP, *Palmtop*

O VPOS (*Virtual Point of Sale*), O WAP (*Wireless Application*) e o *Palmtop* constituem os novos canais integrados ao *switch*. O VPOS é o ponto de venda virtual do *website* do *merchant* disponível pela Internet. O WAP possibilita o acesso pelo telefone celular. O *Palmtop* também, conectado na Internet, constitui um canal de acesso.

III.3.2 Suporte operações EFT

Os elementos listados a seguir constituem o suporte de operações EFT:

-Tradução de mensagens;

-Sistemas de pagamento;

-Transferência entre instituições;

-Processamento Seguro;

-Verificação do PIN;

-Chaves de autenticações de mensagens (MAC);

-Troca de chaves;

-Geração de PIN;

-Dispositivos de segurança como *Attala* ou *Excryt*

III.3.3 Processamento distribuído em redes

As redes de processamento distribuídos são redes:

-LAN (*Local Area Networks*);

- MAN (*Metropolitan Área Networks*);

-WAN (*Wide Área Networks*);

-GAN (*Global Area Networks*).

III.3.4 Suporte dispositivo de comunicações e protocolos

O switch suporta os seguintes protocolos de comunicação: *TCP/IP, FRAME RELAY, X.25, 3270 SNA LU2, 3780 Bisync* e *ATM*. Neste trabalho, o protocolo MAC DR-TDMA (MAHMOUD, 1996) foi implementado por objetivo de desenvolver um protocolo *Wireless* ATM e integrar as redes ATM fixas com as redes *Wireless* ATM.

III.3.5 Gerenciador de Relatórios

Os relatórios são gerados sobre: *settlement*, detalhes das transações, execuções, estatísticas, taxas, autorizações, processamento *Stand-in* SAF (*Store And Forward*) definido de acordo com a lógica do negócio, autorização positiva, autorização negativa,

processamentos *online* e *batch*, atualizações em tempo real, capturas de transações, gerenciamento de mensagens, temporizador, login de usuários, roteamento *delivery*, formatação, controle, monitoramento de redes, monitoramento do processamento e monitoramento de protocolo que inclui identificação, resolução, *tracking*, interface do operário, estado da rede, atividade da rede, controle do *Switch* e processamento em tempo real.

III.4 Solução de Infra-estrutura integrada

A solução de infra-estrutura integrada oferece a integração de novos canais como a tecnologia WAP, as soluções VPOS e *Palmtop* pela Internet. Todas estas soluções pressupõem que o *Merchant* tenha um *website* na Internet que o *Cardholder* possa acessar para realizar suas operações. Assim, no capítulo 8, tem um exemplo de estrutura de um website do *merchant*. A tecnologia baseada no *switch* ATM (*Asynchronous Transfer Mode*) descrito abaixo oferece mais segurança, interoperabilidade, robustez, volume de transações e alta confiabilidade. Será implantado, no capítulo 6, um ambiente seguro baseado em Servidores paralelos e redundantes que, se um servidor falhar, o outro pode continuar a operar e assumir os serviços do servidor que falhou.

III.4.1 Switch ATM e Redes de processamentos distribuídos

O *switch* ATM permite interligar diferentes pontos da rede. O objetivo é permitir a ligação apropriada entre os equipamentos, servidores, *workstations*, concentrando ou expandindo as ligações físicas.

O *switch* ATM pode realizar comutação especial e temporal. No caso da comutação temporal, o *switch* ATM altera a posição na escala de tempo ocupada por um determinado conjunto de informações. No caso da comutação especial, toda informação apresentada em uma determinada porta de entrada será dirigida a uma porta de saída específica.

O *gateway* aqui permite a integração entre uma rede não ATM e uma rede ATM. Freqüentemente a *gateway* incorpora funções típicas de *switches* como comutação especial e temporal, armazenamento temporário ao lado da conversão de protocolos de comunicação.

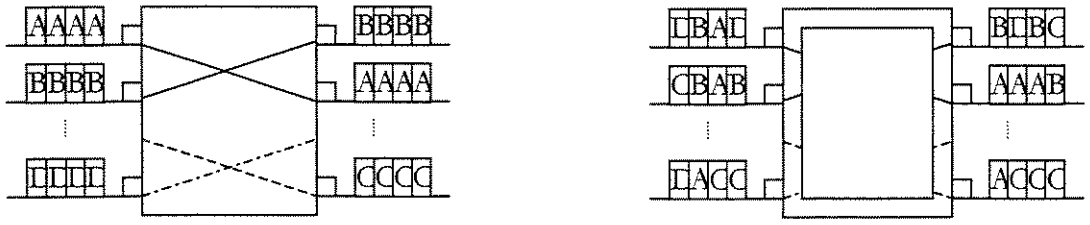


Figura 14: *Tipos de comutação*

Os *switches* são responsáveis por dirigir aos seus destinos finais as informações que trafegam pela rede. Assim, eles usam recursos de comutação e de armazenamento temporário dessas informações. A quantidade de portas que os diversos modelos de *switches* suportam vão de duas até centenas de portas. Cada porta corresponde a uma entrada e a uma saída independente operando no modo *full-duplex*. A capacidade de um *switch* em termos de número de portas depende da taxa de transmissão adotada em cada porta. Por exemplo, um mesmo *switch* de grande porte pode suportar até 64 portas SONET OC-3 (155,52 Mb/s) ou até 256 portas DS3 (44,736Mb/s). Quanto ao tipo das portas, a variedade também é muito grande. São oferecidos tipos correspondentes aos vários padrões nativos para camada física ATM, tais como: DS3, TAXI, SONET OC-3 (ou STM-1). Outros tipos de portas suportam a interoperabilidade entre redes ATM e redes diversas, viabilizando a ligação direta de um *switch* ATM com equipamento *Ethernet, Fast Ethernet, Gigabit, FDDI, Frame Relay* e *SMD*.

III.4.1.1 ATM versus Frame Relay

O tamanho dos pacotes do *Frame Relay* é variável, enquanto que o tamanho de pacotes da ATM é fixo em 53 bytes. A tecnologia ATM garante que em uma dada conexão, a ordem de transferência das células é preservada de um extremo ao outro da comunicação. A utilização de células fixas traz algumas vantagens importantes num ambiente onde convivem tipos de informação como: previsão de retardo na rede; previsibilidade de projeto de hardware de suporte de protocolos de comunicações (especificações de *softwares* e das suas estruturas de controle); encapsulamento; projeto modular de *switch*; paralelismo e

escalabilidade dos recursos internos de *switch* nas aplicações em tempo real — no nosso caso há garantia de desempenho e de Qualidade de Serviço QoS

III.4.1.2 QoS - Qualidade de serviço

O QoS é um conjunto de parâmetros de desempenho tais como: atraso, variação do atraso, taxa de perda de células, que pode ser negociado durante o estabelecimento de uma conexão. Durante o estabelecimento de uma conexão, o usuário pode solicitar uma classe de OoS. Os parâmetros requisitados para a atribuição de uma QoS são:

-Taxa de erros de células: razão entre o número de células entregues com erro e o número total de células transferidas (adimensional);

-Taxa de blocos de células com erros graves: razão entre o número de blocos com erros graves e o número total de blocos transmitidos. Um bloco é o conjunto seqüencial formado por um número arbitrário de células (adimensional);

-Taxa de perda de células: razão entre o número de células perdidas e o número total de células transmitidas (adimensional);

-Atraso na transferência de células: tempo decorrido entre a transmissão de uma célula e sua recepção pelo destinatário final (em segundos);

-Variação do atraso de células: também conhecido por *jitter*, mede a flutuação dos atrasos de transferência de células (em segundos).

As classes de serviços de transporte definem se o serviço referido pela conexão pode ser completamente fornecido pelas camadas físicas de ATM da rede ou não, enquanto as classes de QoS definem conjuntos quantificados e predefinidos dos parâmetros de finalidade escolhidos a priori pelo usuário. Não há nenhuma quantificação numérica pelo usuário para qualquer dos parâmetros.

III.4.2 Interligação na Rede e Classes de QoS

A figura 15 mostra dois tipos de interconexão conhecidos como UNI (Interface Cliente-Servidor) e NNI (Interface Servidor-Servidor). A UNI define a interface entre um

equipamento da rede e o *switch.* A NNI define a interface entre *switches*. A UNI e a NNI estabelecem regras e protocolos para viabilizar a interconexão e a interoperabilidade de sistemas na rede.

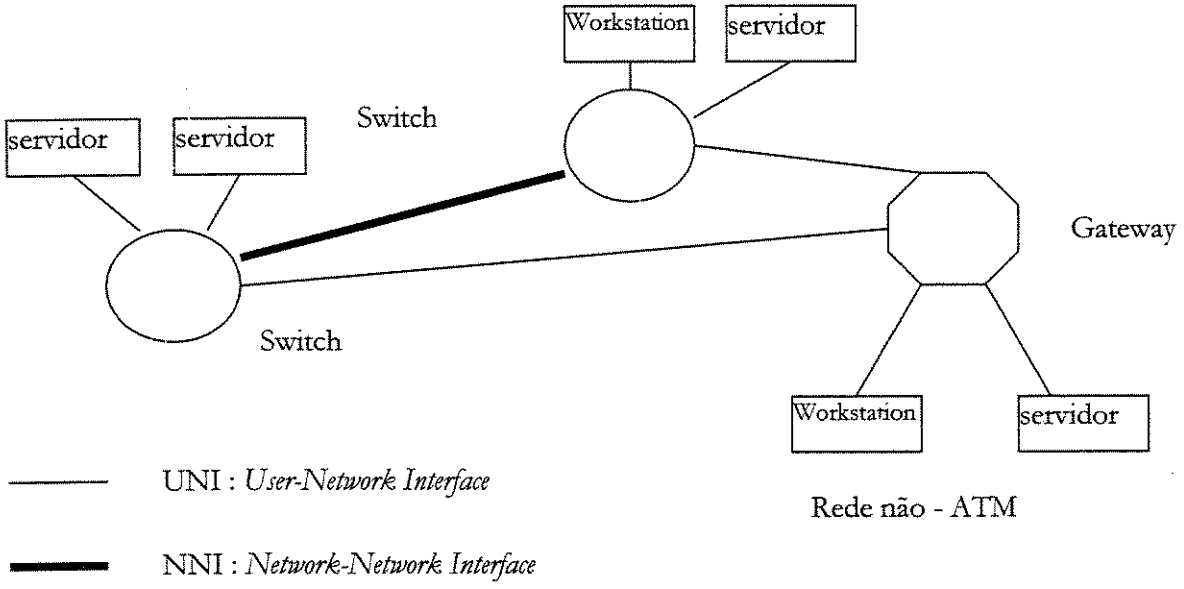


Figura 15: *Modelo de arquitetura de rede* LAN ATM

As classes de serviços de transportes suportadas são:

-Classe A: serviço orientado à conexão, com taxa fixa de bits e requisitos de tempo entre os pontos finais de comunicação;

-Classe C: serviço orientado à conexão, com taxa variável de bits e sem requisitos de tempo entre os pontos finais de comunicação;

-Classe X: serviço orientado à conexão, no qual todos os parâmetros de tráfego são definidos pelo usuário.

As classes de *QoS* definidas são:

-Classe Zero: suporta um serviço do tipo "melhor esforço", também chamada de classe de QoS não especificada, pois não há quantificação de qualquer dos parâmetros de qualidade;

-Classe 1: suporta um serviço que emula um circuito com taxa de bits constantes com restrições de atraso estabelecidas para cada célula. Este serviço emula um circuito digital convencional. Pode ser utilizado, por exemplo, em aplicações que usem uma fonte de sinais de vídeo sem compressão de dados;

-Classe 2: suporta um serviço que emula um circuito de taxa variável de bits e com restrições de atraso, como o utilizado em aplicações de teleconferência;

-Classe 3: suporta os requisitos de tráfego de um serviço orientado à conexão, com taxa variável de bits e sem restrições de atraso aplicado às células. Esta classe foi criada para dar suporte à interoperabilidade com protocolos orientados à conexão do *Frame Relay*;

-Classe 4: suporta os requisitos de um serviço não-orientado à conexão, com taxa variável de bits e sem restrições de atraso.

Estas duas classificações não são independentes nem todas as combinações são permitidas ou fazem sentido prático. A escolha da classe de transporte precede e condiciona a escolha da classe de QoS. Por exemplo, não há sentido em especificar uma conexão como classe C (para transporte) e classe 1 (para QoS). É viável, entretanto, atribuir a uma conexão classe X quaisquer das classes de QoS.

A seleção da classe de transporte para uma conexão depende da expectativa do usuário ao processamento pela rede das células de sua conexão. Se há a necessidade da manipulação de pacotes de protocolos de nível superior à camada ATM (por exemplo, pacotes AAL), a escolha deve recair sobre as classes A ou C. Caso contrário, a classe X deve ser especificada. Para ilustrar este ponto, deve ser considerado o caso em que um usuário deseja estabelecer uma conexão que simule um circuito digital (taxa constante de bits com atrasos entre os dispositivos controlados) e que este circuito tenha interconexão com dispositivos não - ATM (um multiplexador DS1). Neste caso, a rede terá de analisar os pacotes AAL da conexão para executar esta função de interconexão e, conseqüentemente, o

usuário deve estabelecer a classe de transporte A. Caso esta função de interconexão não seja necessária, as classes X (transporte) e 1 (QoS) devem ser configuradas.

Finalmente a parametrização e a definição de tais classes de QoS que estão disponíveis para cada classe de transporte em uma rede específica são prerrogativas do fornecedor do serviço de rede. A única restrição é a obrigatoriedade do provimento da classe de QoS zero (classe não especificada) para todas as classes de serviço de transporte.

III.4.3 Emulação de rede LAN

A emulação de rede LAN tem como finalidade emular serviços de LAN existentes como IEEE 802.3 *Ethernet* e IEEE 802.5, sobre uma rede ATM. Deste modo, as aplicações executadas nos dispositivos ATM podem interagir como se estivessem em um ambiente de LAN tradicional. Além disso, o ambiente de Emulação de LAN permite a interconexão de Redes ATM com as LANs tradicionais, por meio do uso de técnicas conhecidas como *bridging*. Consegue-se, assim, que aplicações residentes em dispositivos ATM operem com aplicações residentes nos equipamentos de usuários ligados às redes LANs. Duas características de LAN tradicional que diferem de características das redes ATM são:

- Não é orientada à conexão dos serviços oferecidos pelas LANs existentes;

- Tem capacidade de broadcast e *multicast* típica dessas LANs. Estas características opõem-se à natureza orientada à conexão das redes ATM e a falta de suporte dessas redes às funções *broadcasting* e *multicasting*. A emulação de LAN viabiliza as chamadas LANs virtuais. O administrador de rede pode estabelecer várias ELANs (*Emulated LAN*) sobre uma rede ATM. Dispositivos ATMs e LANs tradicionais podem ser associadas às diferentes ELANs definidas, independentemente de suas localizações físicas. Estabelecem-se deste modo, LANs virtuais que podem ser dinamicamente criadas e re-configuradas para refletir, por exemplo, arranjos organizacionais, sem que sejam necessárias alterações físicas na rede. Os benefícios trazidos pelas LANs virtuais representam um impulso significativo para a adoção em larga escala de ambientes de Emulação de LAN no qual convivem LANs virtuais, identificadas por VLAN. Dois tipos de dispositivos ATM suportam a implementação da camada de Emulação de LAN:

- As chamadas NICs (*Network Interface Cards*) ATM, que são as placas adaptadoras por meio das quais os dispositivos ATM se ligam à rede ATM. Através destes dispositivos, os protocolos da camada de rede dos dispositivos se comunicam como se estivessem ligados a uma LAN tradicional; a diferença está na possibilidade de uso da maior largura de faixa oferecida pelas redes ATM;

- Os *switches* de LAN com interface ATM. Estes sistemas intermediários funcionam como ele entre uma rede ATM e uma LAN tradicional. Eles possibilitam a interoperabilidade de uma ELAN sobre a infra-estrutura ATM com uma LAN verdadeira, desde que o mesmo protocolo MAC seja utilizado em ambos os lados. Com isto, permitem que estações da LAN e dispositivos ATM compartilhem uma mesma LAN virtual.

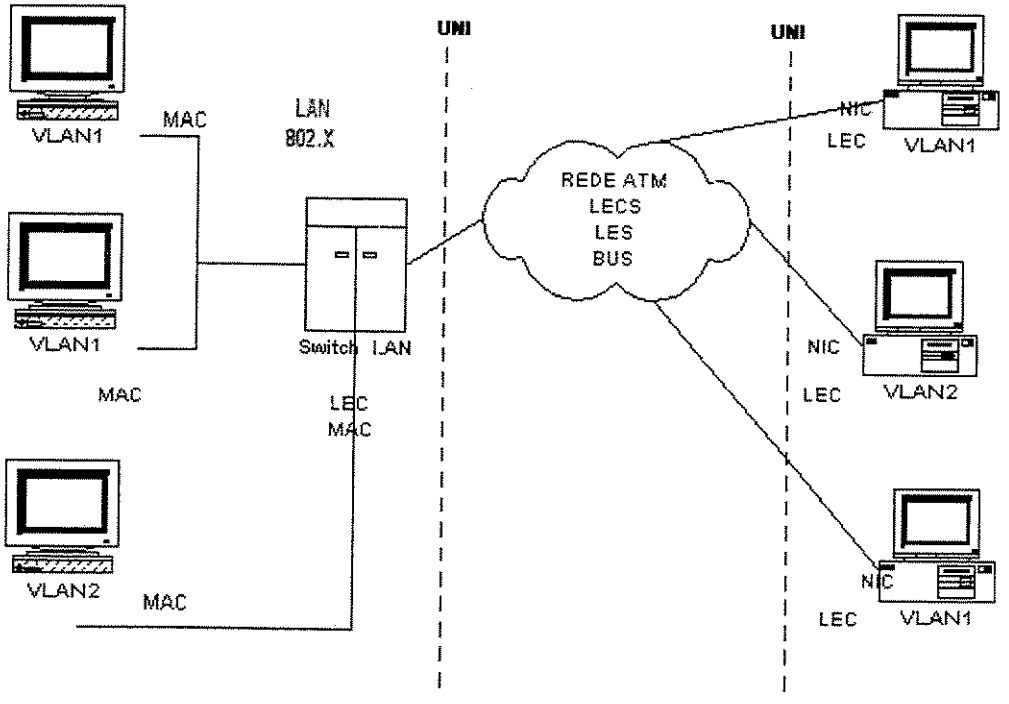


Figura 16: *Emulação* LAN

O ELAN é composto pelas seguintes entidades:

LEC (LAN *Emulation Client*): LEC é a entidade que, nos dispositivos ATM, executa as funções de transferência de dados, resolução de endereços e outras funções de controle. O

LEC fornece às camadas superiores as interfaces de serviço que emulam a subcamada MAC das ELANs.

Cada LEC é parte de um dispositivo ATM e, assim sendo, é identificado por um endereço ATM único. Um LEC está associado com um ou mais endereços MAC, acessíveis através do seu endereço ATM: no caso de um *switch* de LAN com interface ATM, o LEC residente estará associado a todos os endereços MAC alcançáveis através de suas portas e que estejam configurados numa particular ELAN; no caso de uma estação de trabalho com um NIC ATM, o LEC residente poderá estar associado a apenas um endereço MAC. Tipicamente, uma ELAN é composta por um conjunto de LECs. Por outro lado, um dispositivo ATM pode integrar múltiplas ELANs, mas deve suportar um LEC para cada ELAN de que faça parte.

LES (LAN *Emulation Server*): LES é a entidade responsável pelas funções de coordenação e controle da ELAN. Ele fornece recursos para o registro e a resolução de endereços MAC (a resolução de endereços, neste caso, refere-se á associação entre endereços MAC e seus correspondentes endereços ATM). Há apenas um LES por ELAN, identificado por um endereço ATM único.

BUS (*Broadcast and Unknown Server*): Bus é a entidade que manipula todo o tráfego de *broadcast e multicast* na ELAN. Além disso, é responsável pelo tráfego com endereço de destino desconhecido, em situações em que se aguarda o término dos procedimentos de resolução de endereços. Cada LEC "enxerga" apenas um BUS por ELAN, identificado por um endereço ATM único. No LES, este endereço está associado ao endereço MAC de *broadcast.* A especificação do ATM Fórum permite implementações distribuídas ou redundantes do BUS dentro de uma ELAN.

LECS (LAN *Emulation Configuration Server*): LECS é a entidade responsável pela atribuição dos LECs individuais a diferentes ELANs. A atribuição é feita pela associação do LEC requisitante com o LES que corresponde a uma dada ELAN.

III.5 Considerações

A tecnologia de modo de Transferência Assíncrono (ATM- *Asynchronous Transfer Mode*) se propõe prover dutos de alta velocidade para transferência de qualquer tipo de informação. O impacto da tecnologia ATM decorre de sua flexibilidade em adaptar-se nas plataformas e nos protocolos com alto fator de interconectividade, interoperabilidade e de gerenciamento. A tecnologia ATM constitui a grande unificadora dos ambientes LAN-MAN MAN-GAN *(Global Area Network)* em maior volume informação por unidade tempo. A ATM suporta o TCP/IP e o maior número de protocolos. O principal enfoque nas pesquisas sobre ATM é voltado para a construção de interfaces de gerenciamento e de protocolos de comunicações. Ela exporta os serviços de banda larga. A ATM é a única tecnologia que reúne ao mesmo tempo os seguintes atributos: alta taxa de velocidade de transmissão, aplicações isócronas (voz), dados, multimídia e gerenciamento dinâmico de banda. O gerenciamento dinâmico de banda fornece:

- Qualidade de serviços;
- Forma comutada;
- Orientado à conexão;
- Transmissão e comutação de dados, voz e vídeo;
- Segurança;
- Rede distribuída;
- LANS Virtuais.

A tecnologia de comutação de pacotes (*packet switching*) como X.25 explora características de comunicação de dados tais como a não-sensibilidade ao retardo, o que as diferenciam da comunicação de voz e também da utilização da largura de faixa.

A comutação de pacotes provê multiplexação estatística para acomodar as transmissões em rajadas típicas das aplicações de dados, ou seja, a largura de faixa é alocada e liberada dinamicamente na medida da necessidade, o que evita eventuais desperdícios associados à locação estatística, conseqüentemente, se consegue ganhos em eficiência.

Diferente da comutação por circuitos (PVC) utilizada nas comunicações clássicas telefônicas por voz, a comutação de pacotes (SVC) não se refere a uma conexão física dedicada durante o período da comunicação. Neste caso, a comunicação é lógica e não física. Isto permite o compartilhamento de várias comunicações dos mesmos recursos físicos, otimizando o seu

uso. No caso da tecnologia de *Frame Relay*, apesar de ter comutação de pacotes, a multiplexação é estatística.

III.6 Proposta de trabalho futuro

Como implementar WAP-ATM, isto é como implementar a tecnologia ATM em comunicação sem fio (*wireless)*? Considerando o fato que a ATM é desenhada para **pequena *BER*** *low Bit Error Rate* (Pequena Taxa de Erro de Bit) associada à fibra óptica.

O problema é que a ATM é desenhada tomando consideração que taxas de erro de bit de transmissão do meio são baixas e distribuídas randomicamente, que é o caso da fibra óptica baseada em sistemas de transmissões. Como conseqüência, se a célula da ATM e os mecanismos de comunicações com fio são aplicados ao canal de comunicação sem fio, a ATM vai perder muitos dos seus ganhos. Uma das razões da ATM conseguir sua alta velocidade de comutação é devida à falta de controle de erros. A ATM assume que o BER é baixo para o meio de transmissão. Ela permite, então, que os protocolos das camadas superiores realizem o controle de erros. Na maior parte dos casos, o protocolo utilizado é o TCP (*Transmissão Control Protocol)*. O TCP usado para *winsock* e para Internet assume que todos os erros são devidos a congestionamento. No caso da fibra óptica isto é verdade. A prova disso é a lentidão de *download* de uma página da WEB durante o horário de intenso uso da internet. No caso de comunicação sem fio, há muitos erros. Assim, o TCP vai assumir que os erros de transmissão sobre o ar são erros de congestão e começa a seqüência do mecanismo de controle de congestão. No caso da ATM em comunicação sem fio, isto vai baixar a eficiência de uso da ATM.

A solução é implementar uma forma de detecção, correção e retransmissão de erros nas células ATM ou mecanismos. Nas camadas superiores, como TCP, isto vai reduzir erros, mas pode reduzir os efeitos do TCP. Assim, o efeito colateral deveria ser analisado neste caso. A implementação do WAP-ATM vai agregar a nossa plataforma tecnológica um caminho de redundância pelo ar tão eficiente quanto pela comunicação com fio.

Capítulo IV

Integração de redes wireless e wired ATM usando o protocolo MAC DR-TDMA

O tempo de resposta para autorização de uma transação eletrônica constitui um parâmetro muito importante. Se houver grande número de usuários simultâneos fazendo transações em uma rede, dependendo da topologia da rede e do nível da QoS, pode haver degradação, elevar o tempo de resposta, conseqüentemente, gerar uma transação sem sucesso.

Integramos a topologia de rede fixa elaborada neste trabalho à tecnologia de rede W*ireless* ATM implementada usando o protocolo MAC DR-TDMA (MAHMOUD, 1996, pp. 596-608) sobre TCP para manter a mesma qualidade de serviço QoS de usuários de redes ATM fixas e redes ATM móveis.

IV.1 Wireless sobre TCP

O TCP é o protocolo de transporte mais utilizado em computadores, especificamente desenvolvido para transmissão segura de dados em rede. O TCP é freqüentemente usado com o protocolo de Internet IP ambos desenvolvido pela ARPANET. O TCP e o IP são diferentes tipos de protocolos e podem ser adaptados a outros protocolos como UDP/IP ou TCP/MAC. Como o TCP foi especificamente concebido para redes fixas onde a congestão é a primeira causa de perda de pacotes, o TCP desempenha bem as tarefas de controle de fluxo e de congestionamento. Mas, o desempenho do TCP em redes *Wireless* pode sofrer degradação devido às características de transmissão do meio do *Wireless*. O meio do *Wireless* não está de acordo com as premissas de controle de erro e de congestionamento implementado no protocolo TCP para redes fixas, o que ocasiona a degradação do desempenho.

IV.2 O protocolo de controle de transmissão segura TCP

No baixo como no alto nível de transmissão de dados, pode ocorrer perda de pacotes de dados devido a várias causas, como: erro de transmissão, falha de hardware, rede muito

carregada, uso de sistema não seguro. O protocolo TCP foi desenvolvido para suprir todos estes problemas.

A tecnologia TCP permite ao destinatário do pacote transmitir de volta uma mensagem ACK de recebimento do pacote de dados enviado ao remetente. O remetente inicia, também, um timer quando envia um pacote e, se ele não receber um ACK depois do tempo de expiração do timer, ele retransmite o mesmo pacote.

Para evitar a confusão causada por atraso ou duplicação de ACK, o protocolo envia de volta uma seqüência de resposta, desta forma, o destinatário pode associar corretamente cada pacote recebido do remetente ao ACK correspondente. Mas, usando ACK positivo, apenas, não é eficiente e gasta banda, porque pode atrasar o envio de um novo pacote até receber um ACK do pacote anterior. A técnica de "*sliding window*" descrita em seguida, é utilizada para corrigir este problema. Esta técnica permite ao remetente enviar muitos pacotes antes de esperar por um ACK para melhorar o uso da banda. Ela consiste em colocar os pacotes em uma seqüência de janelas de tamanhos fixos e transmitir os pacotes dentro das janelas.

Assim que o remetente recebe um ACK do primeiro pacote dentro da janela, ele coloca a janela ao lado e envia os pacotes novos. Quando um pacote é perdido e o *timer* expira, o remetente retransmite este pacote. Este protocolo guarda e atualiza os pacotes com ACK. O TCP permite a janela variar em função do tempo. Cada ACK que especifica a quantidade de bits recebidos especifica também a quantidade adicional de bits que couber. Isto permite controlar o fluxo e restringir a transmissão até ter um *buffer* suficiente que agüente o volume de dados.

Concluímos que a técnica da segurança do TCP é baseada no ACK positivo e na retransmissão. Como o TCP é um protocolo de Internet, o atraso de transmissão depende do tempo de chegada do ACK em cada roteador transmissor. O TCP usa o algoritmo de retransmissão adaptativa para resolver a variação dos atrasos na Internet baseada na seguinte equação:

$$RTT = (\alpha * RTT_{anterior}) + (1-\alpha)*RTT_{novo}$$

Timeout = β * RTT

Onde RTT é o tempo de transação estimado (Round Trip Tim). Geralmente, β=2.

As retransmissões são geralmente sinais de congestionamento devidos à sobrecarga de datagramas em um ou mais pontos de comutação na rede. Os pontos terminais geralmente desconhecem onde e porque a congestão ocorreu.

Para os pontos terminais, a congestão significa apenas aumento de atrasos. A maioria dos protocolos de transporte usa o *timeout* e a retransmissão para resolver o problema do aumento do atraso, retransmitindo os datagramas. Senão fosse, aconteceria um congestionamento que travaria toda a rede. Para evitar o colapso da congestão, o TCP reduz a taxa de transmissão quando ocorre retransmissão. Isto é conhecido como algoritmo de "*slow start*".

Enquanto começa uma nova conexão ou depois de uma congestão, o tamanho da janela fica um único segmento e é incrementado por segmento cada vez que chega um ACK. Quando chega o ACK, a janela de congestão fica 2 e 2 segmentos são enviados. Quando chegam os 2 ACKs que correspondem aos 2 segmentos anteriores, o tamanho da janela fica 4 e 4 segmentos são enviados. Assim por diante. Logo que o tamanho da janela atinge a metade do seu tamanho original antes da congestão, o TCP entre em uma fase de "evitar congestão" e diminui a taxa de incremento. Durante esta fase, a janela de congestão é incrementada por 1 apenas, se todos os segmentos na janela responderam ACK.

Resumindo, janela de tamanho variável, retransmissões com t*imeout*, *timer*, algoritmo de "*slow start*" e técnicas de "evitar congestão" são os principais pontos de desempenho do protocolo TCP em redes fixas. Mas, estas técnicas não são aplicáveis em redes *wireless* devido à natureza do meio.

IV.3 Degradação de TCP e Soluções

Para rede *wireless*, a congestão não é a causa de perdas de dados, ao contrário, as perdas ocorrem por causa de desvanecimento e apago do meio de transmissão. Quando uma perda de dados ou atraso inesperado ocorre, o TCP interpreta isto como congestionamento e diminui o tamanho de transmissão da janela antes de retransmitir os pacotes, inicia o

mecanismo de controle de congestão e "reseta" o *timer* de retransmissão. Estas medidas resultam a uma redução desnecessária de uso de banda, o que provoca uma degradação de desempenho sob forma de saída degradada e grandes atrasos. Para evitar a degradação de desempenho de TCP, foi proposto o TCP/MAC.

IV.4 Integração de rede ATM fixa com rede wireless ATM

A integração de rede ATM fixa com *wireless* ATM apresenta um certo número de problemas. O protocolo MAC de ATM usando apenas CDMA ou TDMA tem os seguintes problemas:

- Protocolo ineficiente para gerenciar *slots* de controle de acesso e reduzir o atraso de acesso de pacotes de controle de tempo sensitivo;

- Algoritmo de alocação ineficiente para integrar o compartilhamento da banda com controle de tráfego CBR, UBR, e ABR.

No caso de CDMA puro, quando o número de transmissões simultâneas aumenta, a interferência aumenta causando degradação no *link*. Neste caso, não se pode garantir QoS. De fato, para garantir QoS aos usuários, a alocação de banda deve ser feita na estação base por um sistema híbrido TDMA/CDMA, onde acesso simultâneo de diferentes usuários é feito de *slot* em *slot*.

Para ser compatível com a rede ATM fixa, a rede *Wireless* ATM deve suportar classes de serviços CBR, VBR e ABR. Assim, a seleção do protocolo de controle de acesso do meio MAC é fundamental para suportar estes serviços no *wireless*.

Propomos o protocolo MAC DR-TDMA (J.F. Frigon, 1998) para a rede *Wireless* ATM que suporta as classes de serviços ABR, CBR e VBR, enquanto assegura a qualidade de serviço, QoS requisitada.

O protocolo TDMA de Reservação Dinâmica (DR) é usado para multiplexar conexões multimídia em um único canal de rádio. Uma descrição completa do protocolo MAC de DR-TDMA pode ser encontrada na (J.F. Frigon, 1998).

A figura 18 mostra uma estrutura do frame do protocolo MAC DR-TDMA.

IV.4.1 Algoritmo de alocação de banda para Wireless ATM

O principal objetivo do algoritmo de alocação de banda do ATM é explorar a estatística de multiplexação enquanto negociar o QoS de cada conexão da rede.

Nesta implementação o seqüenciador decide baseado no conhecimento atual do estado da rede e dos parâmetros de conexão, como atribuir os *slots* para cada conexão e para efeito de controle de tráfego. No *Wired* ATM, o *switch* tem um acesso direto aos *buffers* para uma multiplexação estatística. No entanto, no *Wireless* ATM os *buffers* estão distribuídos nos terminais móveis, assim o *switch* não tem acesso direto. Além disso, o MAC do *Wireless* ATM deve lidar com a estrutura do frame do TDMA ver figuras 17 e 18 a seguir (J.F.FRIGON, 1998).

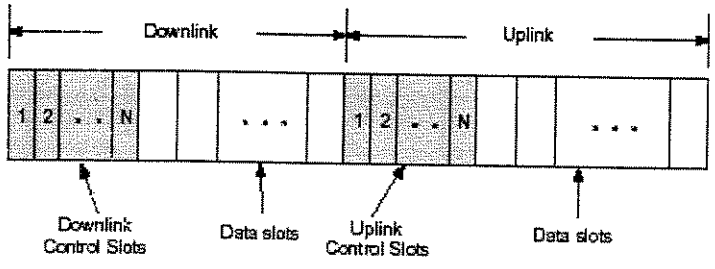


Figura 17: *Estrutura do frame* DR-TDMA

A figura 18 mostra o problema de alocação de banda para *Uplink* e *Downlink* na *frame* t+1. Para alocação de *Uplink*, o móvel deve observar seus parâmetros dinâmicos na *frame* t-1, o transmitir na estação base na *frame* t e receber a alocação do *frame* t+1.

Figura 18: *Problema de alocação de banda*

Qualquer atraso de transmissão de informação do móvel para a estação base vai degradar o desempenho do algoritmo de alocação de banda. De outro lado, pela alocação de *downlink*, a estação base tem um acesso direto aos *buffers* do *downlink*. A partir do *frame* t, a estação base pode iniciar uma transmissão de *downlink* do *frame* t+1. No entanto, desde que a alocação de ambos *uplink* e *downlink* no *frame* pode ser decidida apenas no início do *frame* MAC, qualquer *slot* não utilizado não poderia ser re-alocado baseado nos requisitos instantâneos dos outros fluxos. Assim, usou-se algoritmo Pseudo-Bayesiano para resolver o problema de alocação de banda.

IV.4.2 Algoritmo Pseudo-Bayesiano

No algoritmo Pseudo-Bayesiano (BAYES, 1998) os pacotes que chegam são registrados e logo depois tentam ser transmitidos em cada *slot* seguinte até conseguir a transmissão com a probabilidade q_r. Assim, se houver n pacotes registrados, inclusive os novos chegados no início do *slot*, a taxa de tentativas é expressa como: $G(n) = nq_r$.

O objetivo do algoritmo é maximizar o *throughput* (saída ou débito) que é aproximadamente igual a $G(n)e^{-G(n)}$, mantendo a taxa de tentativas G(n) igual a 1. O algoritmo opera atualizando o número de pacotes estimado $\hat{n}$ sobre o número de pacotes registrado n.

Cada pacote registrado é transmitido independentemente em um *slot* t com a seguinte probabilidade, $q_r t = \min(1, 1/\hat{n}^t)$. Assim, se tivermos uma estimativa exata ($n = \hat{n}$), a taxa de tentativas G(n) será igual a 1. O número de pacotes registrado estimado $\hat{n}^{t+1}$ no início de cada *slot* t+1 é atualizado a partir do número de pacotes registrado estimado $\hat{n}^t$ do *slot* t, de acordo com a seguinte regra:

$$\hat{n}^{t+1} = \begin{cases} \max(\lambda, \hat{n}^t + \lambda - 1) & \text{Para pacotes ociosos e sucedidos} \\ \hat{n}^t + \lambda + (e-2)^{-1} & \text{Para pacotes colidos} \end{cases} \qquad (1)$$

Onde λ é a taxa de chegada estimada por *slot* (intensidade).
Podemos então escrever o tráfego gerado pela classe i como:

$$G_i(n_i) = n_i q_i \qquad 1 \le i \le p \qquad (2)$$

o número total de tentativas é:

$$G(n_1,...,n_p) = \sum_{i=1}^{p} G_i(n_i) = \sum_{i=1}^{p} n_i q_i \tag{3}$$

A probabilidade de um pacote de prioridade i seja transmitida com sucesso é:

$$\begin{aligned} P_{SUCC}^{i} &= n_i q_i (1-q_i)^{n_i-1} \prod_{\substack{j=1 \\ j\neq i}}^{p} (1-q_j)^{n_j} \\ &\approx G_i(n_i) e^{-G(n_1,...,n_p)} \end{aligned} \tag{4}$$

e a probabilidade de um pacote de qualquer classe seja transmitida com sucesso é:

$$\begin{aligned} P_{SUCC} &= \sum_{i=1}^{p} P_{SUCC}^{i} = \\ &= \sum_{i=1}^{p} \left[n_i q_i (1-q_i)^{n_i-1} \prod_{\substack{j=1 \\ j\neq i}}^{p} (1-q_j)^{n_j} \right] \\ &\approx G_i(n_{1,...,} n_p) e^{-G(n_1,...,n_p)} \end{aligned} \tag{5}$$

Se $G(n_1, ..., n_p)$ igual 1, a saída máxima fica 1/e.

Seja γ_i o parâmetro de prioridade da classe i, o parâmetro de prioridade de cada classe é dado por:

$$G_i(n_i) = n_i q_i = \gamma_i \qquad 1 \le i \le p \tag{6}$$

Se forem impostas restrições nos parâmetros de prioridade da forma:

$$\sum_{i=1}^{p} \gamma_i = 1 \tag{7}$$

Obteremos, assim a saída ótima desejada:

$$P_{SUCC} = G_i(n_1,...,n_p) e^{-G(n_1,...,n_p)} = \left(\sum_{i=1}^{p} \gamma_i \right) e^{-\sum_{i=1}^{p} \gamma_i} = 1/e \tag{8}$$

e cada classe i tem saída γ_i/e.

Agora, assumindo que antes do *slot*, o número de pacotes registrados n_i de cada classe de prioridade i estatisticamente independente é dada pela equação de distribuição de Poisson com :

$$\hat{n}_i \geq \gamma_i \tag{9}$$

Temos, então:

$$p(n_i) = \frac{\hat{n}_i^{n_i}}{n_i!} e^{-\hat{n}_i}$$
$$p(n_1,...,n_p) = \prod_{i=1}^{p} p(n_i) \tag{10}$$

Cada pacote de classe i ($1 \leq i \leq p$) é transmitido independentemente no próximo *slot* com a probabilidade $q_i = \gamma_i / \tilde{n}_i$ ($\tilde{n}_i \geq \gamma_i$) e com os parâmetros de prioridade definidos.
Derivando, agora, a probabilidade de distribuição do número de pacotes registrados de cada classe de tráfego com prioridade de cada um dos estados dos *slots* seguintes: ocioso, sucesso, colisão.

Caso de sucesso:

Usando a regra de Bayes:

$$p(n_1,...,n_p \mid succ_j) = \frac{p(succ_j \mid n_1,...,n_p)\, p(n_1,...,n_p)}{p(succ_j)} \tag{11}$$

onde,

$$p(succ_j \mid n_1,...,n_p) = n_j \frac{\gamma_j}{\hat{n}_j} \left(1 - \frac{\gamma_j}{\hat{n}_j}\right)^{n_j - 1} \prod_{\substack{i=1 \\ i \neq j}}^{p} \left(1 - \frac{\gamma_i}{\hat{n}_i}\right)^{n_i}$$
$$p(succ_j) = \sum_{n_1=0}^{\infty} \cdots \sum_{n_p=0}^{\infty} p(succ_j \mid n_1,...,n_p)\, p(n_1,...,n_p) = \frac{\gamma_i}{e} \tag{12}$$

logo,

$$p(n_1,...,n_p \mid succ_j) = e \frac{(\hat{n}_j - \gamma_j)^{n_j - 1} e^{-\hat{n}_j}}{(n_j - 1)!} \prod_{\substack{i=1 \\ i \neq j}}^{p} \frac{(\hat{n}_i - \gamma_i)^{\hat{n}_j} e^{-\hat{n}_i}}{n_i!} \tag{13}$$

$$p(n_j \mid succ_j) = \sum_{n_1=0}^{\infty} \cdots \sum_{n_{j-1}=0}^{\infty} \sum_{n_{j+1}=0}^{\infty} \cdots \sum_{n_p=0}^{\infty} p(n_1,\ldots,n_p \mid succ_j)$$
$$= \frac{(\hat{n}_j - \gamma_j)^{(n_j-1)}}{(n_j-1)!} e^{-(\hat{n}_j-\gamma_j)} \tag{14}$$

$$p(n_i \mid succ_j) = \sum_{n_1=0}^{\infty} \cdots \sum_{n_{i-1}=0}^{\infty} \sum_{n_{i+1}=0}^{\infty} \cdots \sum_{n_p=0}^{\infty} p(n_1,\ldots,n_p \mid succ_j)$$
$$= \frac{(\hat{n}_i - \gamma_i)^{n_i}}{n_i!} e^{-(\hat{n}_i-\gamma_i)} \qquad i \neq j \tag{15}$$

$$p(n_1',\ldots,n_p' \mid succ_j) = p(n_1 = n',\ldots,n_j = n_j'+1,\ldots,n_p = n_p' \mid succ_j)$$
$$= e \prod_{j=i}^{p} \frac{(\hat{n}_i - \gamma_i)^{n_j} e^{-\hat{n}_i}}{n_i'!} \tag{16}$$

$$p(n_j' \mid succ_j) = p(n_j' = n_j'+1 \mid succ_j) = \frac{(\hat{n}_j - \gamma_j)^{n_j'}}{n_j'!} e^{-(\hat{n}_j-\gamma_j)}$$
$$p(n_i' \mid succ_j) = p(n_i = n_i' \mid succ_j) = \frac{(\hat{n}_i - \gamma_i)^{n_j}}{n_i'!} e^{-(\hat{n}_i-\gamma_i)} \qquad i \neq j \tag{17}$$

Caso de colisão:
Seguimos o mesmo raciocínio do caso de sucesso, temos:

$$p(n_1,\ldots,n_p \mid coll) = \frac{p(coll \mid n_1,\ldots,n_p)\, p(n_1,\ldots,n_p)}{p(coll)} \tag{18}$$

$$p(coll \mid n_1,\ldots,n_p) = 1 - p(idle \mid n_1,\ldots,n_p) - \sum_{i=1}^{p} p(succ_i \mid n_1,\ldots,n_p)$$
$$= 1 - \prod_{i=1}^{p} (1 - \frac{\gamma_i}{\hat{n}_i})^{n_i} \tag{19}$$
$$- \sum_{i=1}^{p} \left[n_i \frac{\gamma_i}{\hat{n}_i} (1 - \frac{\gamma_i}{\hat{n}_i})^{n_i-1} \prod_{\substack{j=1 \\ j\neq 1}}^{p} (1 - \frac{\gamma_j}{\hat{n}_j})^{n_j} \right]$$

$$p(coll) = 1 - p(idle) - \sum_{i=1}^{p} p(succ_i) = \frac{e-2}{e} \tag{20}$$

$$p(n_1,...,n_p \mid coll) = \frac{e}{e-2}\prod_{i=1}^{p}\left(\frac{e^{-\hat{n}_i}}{n_i!}\right)\left(\prod_{i=1}^{p}\hat{n}_i^{n_i} - \prod_{i=1}^{p}(\hat{n}_i^{n_i} - \gamma_i)^{n_i}\right.$$
$$\left. - \sum_{i=1}^{p}\left[n_i\gamma_i(\hat{n}_i - \gamma_i)^{n_i-1}\prod_{\substack{j=1 \\ j\neq 1}}^{p}(\hat{n}_j - \gamma_j)^{n_j} \right]\right) \tag{21}$$

$$p(n_i \mid coll) = \sum_{n_1=0}^{\infty}...\sum_{n_{i-1}=0}^{\infty}\sum_{n_{i+1}=0}^{\infty}...\sum_{n_p=0}^{\infty} p(n_1,...,n_p \mid coll)$$
$$= \frac{e}{e-2}\frac{e^{-n_i}}{n_i!}\left(\hat{n}_i^{n_i} - e^{\gamma_i - 1}\left[(2-\gamma_i)(\hat{n}_i - \gamma_i)^{n_i} + n_i\gamma_i(\hat{n}_i - \gamma_i)^{n_i-1}\right]\right) \tag{22}$$

$$p(n_1',...,n_p' \mid coll) = p(n_1 = n_1',...,n_p = n_p' \mid coll)$$
$$= \frac{e}{e-2}\prod_{i=1}^{p}\frac{e^{-\hat{n}_i}}{n_i!}\left(\prod_{i=1}^{p}\hat{n}_i^{n_i'} - \prod_{i=1}^{p}(\hat{n}_i - \gamma_i)^{n_i'} - \sum_{i=1}^{p}\left[n_i\gamma_i(\hat{n}_i - \gamma_i)^{n_i'-1}\prod_{\substack{j=1 \\ j\neq 1}}^{p}(\hat{n}_j - \gamma_j)^{n_j'} \right]\right) \tag{23}$$

$$p(n_i' \mid coll) = p(n_i = n_i' \mid coll)$$
$$= \frac{e}{e-2}\frac{e^{-\hat{n}_i}}{n_i'!}\left(\hat{n}_i^{n_i'} - e^{\gamma_i - 1}\left[(2-\gamma_i)(\hat{n}_i - \gamma_i)^{n_i'} + n_i'\gamma_i(\hat{n}_i - \gamma_i)^{n_i'-1}\right]\right) \tag{24}$$

Depois da colisão, o número de pacotes registrados em cada classe de prioridade não fica independente nem fica uma equação de distribuição de Poisson. Entretanto, podemos calcular a média, a variância da distribuição marginal obtida.

$$E\left[n_i' \mid coll\right] = \hat{n}_i \div \frac{\gamma_i}{e-2} \tag{25}$$

$$Var\left[n_i' \mid coll\right] = \hat{n}_i + \frac{\gamma_i}{e-2} - \left(\frac{\gamma_i}{e-2}\right)^2 \tag{26}$$

Assim, obtemos a equação de correlação:

$$Corr\left[n_i', n_j' \mid coll\right] = \frac{E\left[n_i' n_j' \mid coll\right] - E\left[n_i' \mid coll\right] E\left[n_j' \mid coll\right]}{\sqrt{Var\left[n_i' \mid coll\right] Var\left[n_j' \mid coll\right]}}$$

$$= \frac{-\gamma_i \gamma_j}{(e-2)^2 \left[\left(\hat{n}_i + \frac{\gamma_i}{e-2} - (\frac{\gamma_i}{e-2})^2 \right) \left(\hat{n}_j + \frac{\gamma_j}{e-2} - (\frac{\gamma_j}{e-2})^2 \right) \right]^{1/2}} \tag{27}$$

Assim, o algoritmo pode ser resumido da seguinte forma:

Para (cada classe de prioridade i, para aumentar a prioridade)

$$\hat{\gamma}_i^t = \min\left(\hat{n}_i^t, 1 - \sum_{j=1}^{i-1} \hat{\gamma}_j^t - \sum_{j=i+1}^{p} \min(\hat{n}_j^t, \gamma_j) \right)$$

$$L = 1 - \sum_{i=1}^{p} \hat{\gamma}_i^t \tag{28}$$

Para (cada classe de prioridade i)

$$\hat{\gamma}_i^t = \hat{\gamma}_i^t + \frac{\lambda_i}{\sum_{j=1}^{p} \lambda_j} L \tag{29}$$

A probabilidade de transmissão é dada por:

$$q_i^t = \min(1, \frac{\hat{\gamma}_i^t}{\hat{n}_i^t}) \tag{30}$$

Usando os resultados precedentes obtidos, para um frame de K *slots* com taxa de chegada de pacotes por *frame* dado por λ.
O algoritmo opera mantendo a atualização do número de pacotes registrado estimado $\hat{n}$ do número de pacotes total registrado no sistema.
O novo pacote que chega no *frame* t vai tentar independemente transmitir em cada *frame*, começando pelo *frame* t+1. Cada pacote registrado é transmitido independemente no *frame* t com probabilidade q_r^t=min (1, K/$\hat{n}^t$) sobre um *slot* independentemente escolhido com uma probabilidade uniforme. Assim, se $\hat{n}$ =n, a taxa de tentativas por *slot* é G(n)=(n/K)q_r será igual ao valor ótimo 1.

Para cada *frame*, o número total estimado de pacotes registrado ao início do *frame* t+1 e atualizado do número total de pacotes registrados ao início do *frame* t do *feedback* de cada

slot k do *frame* t (k_{nc}: número de *slots* ociosos ou sucedidos e k_c: número de *slots* em colisão no *frame* t) de acordo com a regra seguinte:

$$\hat{n}^{t+1} = \lambda + k_{nc} \max\left(0, \frac{\hat{n}^t}{K} - 1\right) + k_c\left(\frac{\hat{n}^t}{K} + \frac{1}{e-2}\right) \tag{31}$$

IV.5 Considerações

O protocolo da Reservação Dinâmica TDMA proposta neste capítulo é a tecnologia do futuro de redes *Wireless* ATM que permite conectar redes ATM fixas com dispositivos *Wireless* ATM mantendo o mesmo nível de qualidade de serviço.

Capítulo V

Arquitetura de segurança

V.1 Introdução

Neste capítulo serão abordados os procedimentos de segurança implementada na plataforma tecnológica. Será mostrada como a criptografia é usada para armazenar e transmitir informação de comércio eletrônico. Os algoritmos de criptografia serão tratados junto com a metodologia implementada no nosso caso específico.

V.2 Técnicas de criptografia

A criptografia é baseada em algoritmos matemáticos que embaralham as informações em forma não legíveis. A decriptografia é o processo de usar o mesmo algoritmo para restaurar as informações embaralhadas na sua forma original.

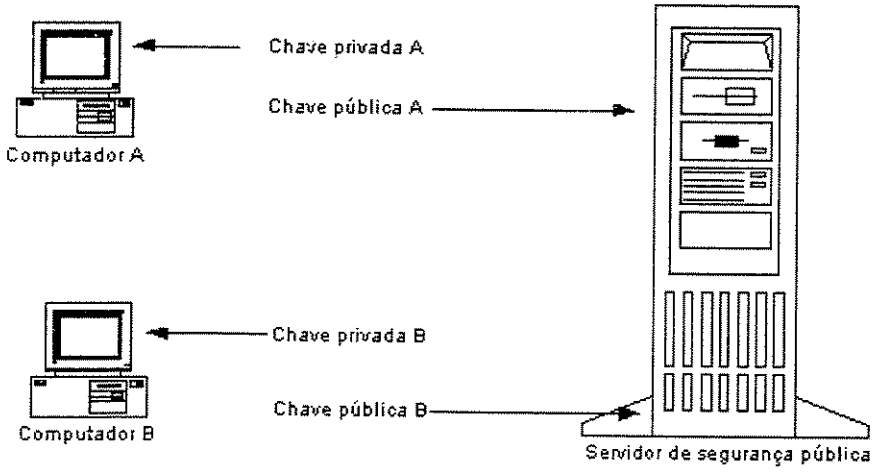


Figura 19: *A criptografia de chave pública utiliza 2 chaves, uma privada e a outra pública.*

A chave privada é guardada em segredo pelo proprietário, enquanto a chave pública é enviada a um servidor de chave pública, disponibilizada livremente.

Técnica de criptografia	Aplicação
Criptografia com chave simétrica	Usada em ambientes onde a chave única pode facilmente ser trocada. Usada também para criptografar informações no disco de armazenagem
Criptografia com chave assimétrica	Usada em redes públicas como a *Internet* onde a troca de chave pode ser normalmente difícil. Uma tecnologia para os certificados digitais e assinaturas.
Funções de sentido único	As assinaturas digitais e os certificados digitais são utilizados para provar a autenticidade do conteúdo de um documento. Geralmente são utilizados em conjunto com sistemas de chaves públicas.

Tabela 4: *Clasificação das técnicas de criptografia primárias: simétrica, assimétrica e sentido único.*

V.3 Objetivos da Criptografia

Os sistemas de criptografia provêm um alto nível de confiança, integridade, não repúdio, e autenticidade da informação que está sendo trocada nas redes, especialmente nas redes públicas como a Internet, onde pessoas e negócios estão envolvidos em comércio eletrônico. Estas áreas chaves são descritas abaixo.

V.3.1 Confiabilidade

Provê privacidade das mensagens e dos dados armazenados, escondendo a informação pelo uso de técnicas de criptografia.

V.3.2 Integridade da mensagem

A Integridade da mensagem assegura para todas as partes que a mensagem fica inalterada do momento que ela foi criada ao momento que ela foi aberta pelo recipiente. Mensagens alteradas podem causar tomadas de ações inapropriadas pelas pessoas.

V.3.3 Não repudiação

O não repúdio provê uma forma que não permite ao usuário negar uma ação da sua própria autoria, isto é o usuário não pode negar posteriormente que ele fez alguma atividade.

V.3.4 Autenticação

A autenticação provê dois serviços:

- O primeiro é identificar a origem da mensagem e prover a segurança e a autenticidade dela.

- O segundo é verificar a identidade das pessoas 'logadas' no sistema e depois disso, continuar a rastrear as identidades caso alguém tente quebrar a conexão, mascarando-se como um usuário.

V.4 Tecnologias de criptografia

Todos estes serviços de criptografia descritos previamente são disponibilizados através das seguintes tecnologias de criptografia:

Certificados (ID, Digitais) autentica os usuários, *Websites*, códigos de *Software*, e informações em geral utilizando a certificação de um terceiro.

Assinaturas digitais, mensagens "assinadas" para validar a origem e a integridade dos conteúdos.

Canais seguros: a criptografia pode ser usada para criar canais seguros sobre redes privadas ou públicas para prover privacidade. Três exemplos são ilustrados na figura 18.

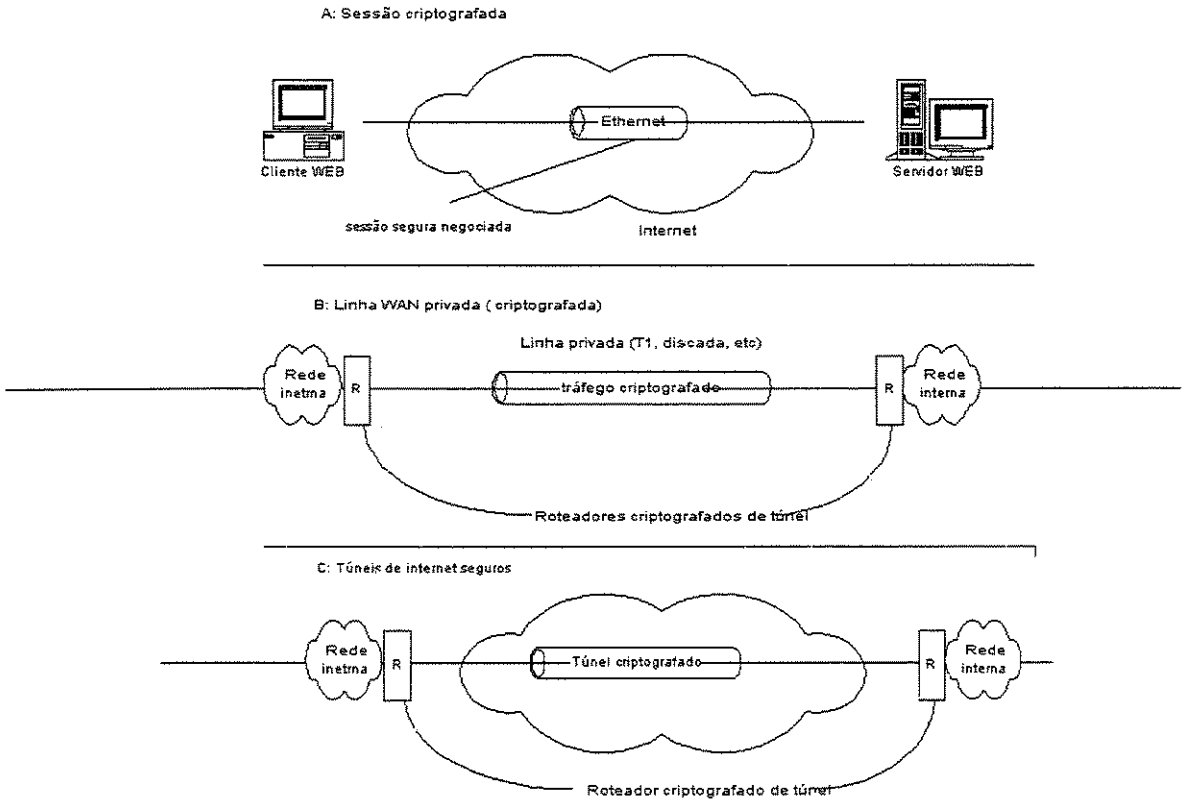


Figura 20: *Três tipos de canais de segurança que podem prover confiança em transmissões de dados.*

A figura 20 ilustra três tipos de canais de segurança que podem prover confiança em transmissões de dados como: Ids digitais, certificados e autoridades certificadas para obter uma chave pública pela *internet* que requer um certo nível de confiança. A chave pública deve ser atribuída à pessoa certa, pois alguém pode se mascarar com uma identidade falsa e utilizá-la. Com os serviços acima definidos, é possível acessar um número maior de outros serviços. Por exemplo, o administrador de sistema pode restringir o acesso das páginas do servidor *Web* para um grupo específico de pessoas. Transações legais e negócios podem ser feitos usando assinaturas digitais. Os usuários podem ser assegurados de que o *website* que eles estão visitando é autêntico e, também, de que ele não está sendo monitorado por um intruso com o propósito de coletar os números de cartão de crédito e as informações pessoais.

A seção seguinte descreve alguns dos serviços mais comuns providos por estas tecnologias. Na maior parte das vezes, estes serviços são acessíveis através de esquemas de chave pública (assimétrica) do que por meio de esquemas de chave privada.

V.4.1 IDs digitais, Certificados, e Autoridades de Certificados

A distribuição de chaves é difícil, mesmo pelo esquema de chave pública (assimétrica), quando a Internet é o canal de comunicação. No caso de rede interna, o administrador de rede pode estabelecer um servidor de segurança separado, onde cada um pode colocar chaves públicas neste servidor. Na Internet a obtenção de chave pública requer um certo nível de confiança. A solução é trabalhar com uma organização credenciada chamada de autoridades de certificado como *Verisign*.

Certificates Authorities ou CA são equivalentes a cartões de chamadas telefônicas pessoais. O certificado é um pacote que contém sua chave pública com outras informações como seu *e-mail*, seu endereço, seu nome completo, e mesmo o seu número do cartão de crédito. Certificados estão sendo muito importantes na *internet*, como passaporte e carteira de motorista estão para o viajante. Uma transação que usa uma chave pública é mostrada na figura 19 e na figura 21 são mostradas transações baseadas em certificados.

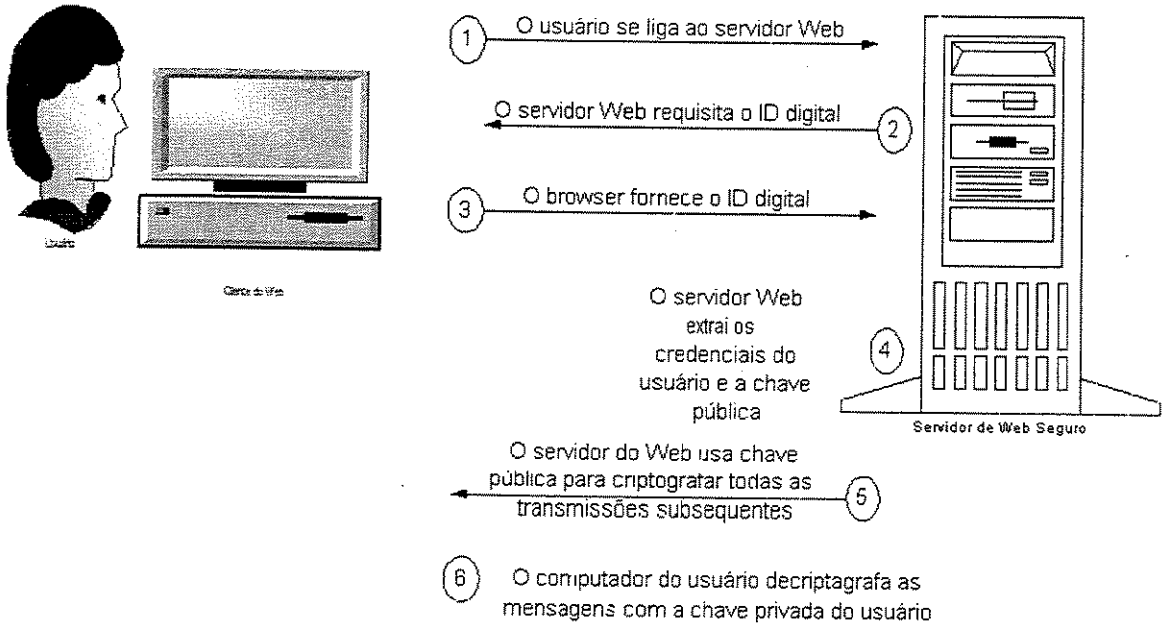


Figura 21: *Transações baseadas em certificados*

Os passos seguintes serão realizados considerando que o usuário Vitor tem um certificado. Caso contrário, o servidor de *web* não poderá validar o Vitor.

1- Vitor acessa uma rede privada.

2- O servidor de *web* peça o certificado de Vitor antes de proceder.

3- O *browser* do Vitor apresenta o certificado dele.

4- O servidor *web* verifica a validade do certificado do *issuer* a partir da autoridade de certificado do *issuer* e extrai a chave pública dele, se for válida.

5- Todas as informações subseqüentes serão criptografadas pela chave de Vitor.

6- O *Web browser* de Vitor decriptografa a transmissão com a chave privada dele.

Um canal de transmissão segura poderia ser estabelecido usando a chave pública do servidor *web* para criptografar todas as informações enviadas. Este método comum é o mais utilizado até que a obtenção de certificado por usuário seja difundida.

V.4.2 Assinaturas digitais

Assinaturas digitais permitem ao recipiente autenticar o remetente e verificar se a mensagem é de acordo com o que foi enviada. A assinatura digital de uma mensagem deve ser feita pelo remetente. Para prover a integridade da mensagem, o seguinte procedimento é realizado:

1- Criptografe o texto com sua chave privada.

2- Envie ambos, o texto criptografado e o texto normal, ao recipiente.

3- O recipiente decriptografa o texto criptografado com a chave pública.

Hoje em dia, o melhor procedimento é usar função de sentido - único.

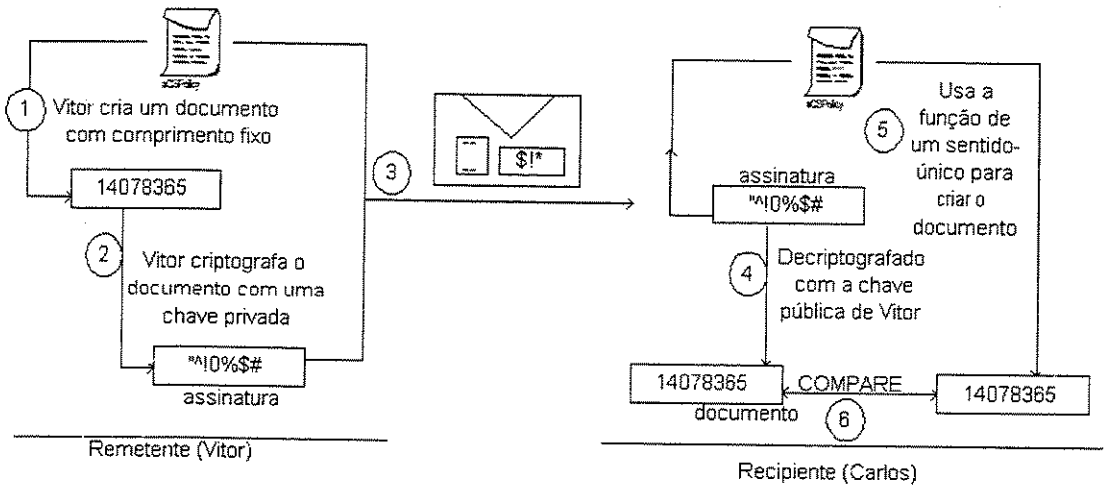


Figura 22: *Passos para usar assinatura digital que envolve criar a assinatura (o remetente) e verificar a assinatura (o recipiente).*

Segue os passos do procedimento mostrado na figura 22:

1- O remetente executa a função sentido - único para criar uma mensagem a ser enviada.

2- O remetente criptografa a mensagem com a chave privada dele para criar uma assinatura digital.

3- O remetente envia a assinatura e a mensagem para o recipiente.

4- O recipiente decriptografa a assinatura utilizando a chave pública do remetente para ler a mensagem.

5- O recipiente aplica a mesma função: sentido - único, na mensagem recebida do remetente.

6- O recipiente compara a mensagem que ele criou com a mensagem recebida do remetente. Se for idêntica, a integridade da mensagem é verificada.

V.4.3 Canais de segurança

Um canal seguro entre o cliente e o servidor *Web* é útil para transação de negócios e privacidade em geral. Ambos, clientes e servidor, devem executar esquema de segurança parecido. No primeiro contato, o cliente e o servidor negociam os parâmetros de sessão,

depois trocam a chave utilizada para criptografar todos os demais dados. Uma vez obtido um canal de segurança, os usuários podem trocar *e-mails,* acessar informações, comprar produtos e conduzir transações comerciais com segurança. Duas tecnologias são implementadas para prover conexões de canais seguros: o SSL (*Secure Sockets Layer*) e o PCT (*Private Communication Transport*). Eles provêm os seguintes serviços:

1- Autenticar usuários e servidores por permitirem ambos transmissões seguras.

2- Criptografar para esconder os dados transmitidos.

3- Integridade para prover a segurança que os dados não foram alterados durante a transmissão.

4- O SSL de 48 bits já foi quebrado. O PLT é uma melhoria do SSL que inclui e usa chaves separadas para autenticação.

V.4.4 Túneis de Internet seguros

O túnel pode prover uma forma de colocar pacotes de protocolos diferentes em pacotes IP e os transportar na *internet* de uma locação para a outra. O SNA ou IPX pode trafegar entre sites na Internet. A criptografia de dados é usada para assegurar o tráfego de protocolos de túneis [PPTP, L2F, L2TP(=PPTP+L2F)].

V.4.5 Comércio eletrônico

Para conduzir negócios na Internet, lojistas e clientes precisam de uma forma automática de coletar com segurança e processar pagamentos de cartões de credito. O protocolo SET provê esta segurança. O SET autentica usuários, estabelecimentos e bancos, provê transações seguras usando técnicas de criptografia. Certificados são requeridos para assegurar a legitimidade do envolvimento das partes. O SET é integrado no sistema de processamento de cartões de crédito.

Na criptografia simétrica, o esquema DES mais seguro usa 128-bit ou mais. O algoritmo de criptografia transforma textos simples em textos cifrados usando funções de transposição e de substituição.

V.5 Número de Identificação Pessoal (PIN)

O PIN é requerido para transação com carta de débito num POS ou para retirada de dinheiro em uma caixa ATM.O PIN fornecido pelo cliente é relacionado ao cartão dele. A exatidão do PIN é verificada na leitura da banda magnética.

V.5.1 Forma de verificação do PIN

Para segurança da conta corrente do cliente, o PIN não é codificado diretamente no cartão dele, mas um outro valor é codificado, quando o cliente digita o PIN dele. Os números digitados pelos clientes são associados com os números e verificados à medida que o cliente entra com o PIN, os dados são criptografados e transmitidos para serem verificados pelo *switch* ou pelo servidor da *brand* do banco. Os *PINs* são formados de 4 à 12 dígitos de comprimento e, como a criptografia é de 64 bits, é para transmitir o PIN. Isto é conhecido como bloco de PIN. Como cada caractere do PIN é número, cada um vai precisar de apenas 4 bits, isto permite que um bloco de 64 bits possa assegurar 16 dígitos. Isto significa que algo tem que preencher o espaço do bloco vazio. Têm alguns padrões para criar o bloco de PIN. Os dois que vamos analisar são:

V.5.2 PIN PAD

O método de PIN PAD começa com os dígitos do PIN. O resto do bloco é preenchido com dígitos hexadecimais F. Por exemplo, caso o PIN por 1234, método do PIN PAD cria um bloco de PIN da seguinte forma:

1234FFFFFFFFFF

V.5.3 ANSI PIN

O método ANSI começa criando um bloco de PIN com os primeiros dígitos sempre vazios, seguidos por um dígito que representa o comprimento do PIN (do hexadecimal 4 a C). Os dígitos do PIN seguem geralmente o PAD F e preenche as posições sobradas. O bloco do número da conta primária PAN (*Primary Account Number*) é criado, começando pelos 4 dígitos, todos zeros, seguidos pelos 12 dígitos mais para direita do PAN, excluindo o dígito de verificação (isto significa os 12 dígitos do PAN começando a partir do 13° do fim). Por exemplo, no caso de um cartão com um **PAN: 6456 0199 9900 0116**, os passos para o método ANSI são os seguintes:

Bloco formado	Código correspondente
Bloco do PIN	041234FFFFFFFFF
Bloco PAN	0000601999900011
Bloco do PIN formado	041254E6666FFFEE

Tabela 5: *Verificação do* PIN *pelo método* ANSI

Regra: 0&0=0; 0&1=1; 1&0=1; 1&1=0

Um variante é similar ao PIN PAD, exceto que o PIN e o comprimento do PIN são enviados com um caráter de preenchimento, geralmente 0.

Um variante do método ANSI é utilizar os 12 dígitos mais para esquerda do PAN no lugar dos 12 dígitos mais para direita.

O método de verificação do PIN no cartão é geralmente baseado em dois métodos:

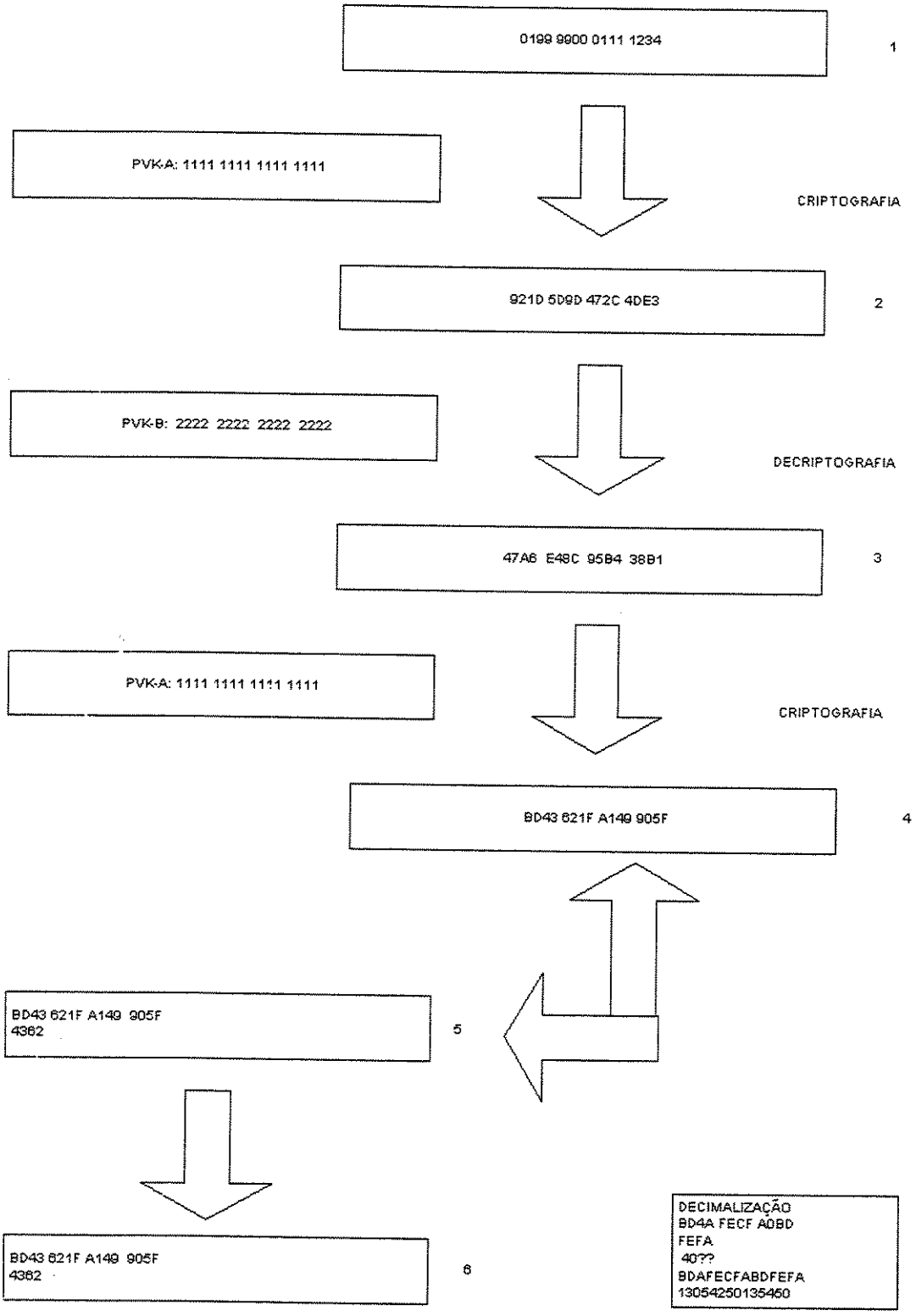
O método PVV (Valor de Verificação do PIN) e o método DES (IBM3624). Antes, estes algoritmos trabalhavam apenas em uma direção, tornando impossível o fato de pegar o dado do cartão e extrair o PIN a partir dele. Isto significa que para verificar o PIN que foi digitado pelo cliente, o PIN digitado deve ser colocado no algoritmo para obter o resultado que é comparado com o valor guardado no cartão.

V.5.4 Método do Valor de Verificação do PIN (PVV)

O método PVV envolve calcular o Valor de Verificação do PIN a partir do PAN. O **PVKI** (Índice da Chave de Verificação do PIN) com mais 2 chaves de verificação do PIN (**PVK- A** e **PVK- B**) como segue na figura 23.

PIN: 1234
PAN: 6456 0199 9900 0116

PVKI: 1
PVK-A: 1111 1111 1111 1111
PVK-B: 2222 2222 2222 2222



BD43 621F A149 905F
4362

4013

Figura 23: *Método de verificação de* PIN

Os Passos do Método de verificação de PIN

1- Dados de início começam pelas informações seguintes:

1.1 Os 11 mais para direita do PAN, excluindo o dígito de verificação.

1.2 O dígito 1 do PVKI

1.3 Os quatros dígitos do PIN mais para esquerda.

2- O dado é criptografado com a chave de verificação do 1° PIN, PVK-A.

3- O texto do passo 2 é criptografado usando a segunda chave de verificação do PIN, PVK-B.

4- O texto do passo 3 é criptografado usando novamente o PVK-A

5- Os quatros primeiros dígitos decimais (isto é de 0 a 9) do texto do passo 4 definem o valor de verificação do PIN (PVV). Neste caso 4362.

6- Se tiver menos do que quatro dígitos decimais no texto do passo 4, os dígitos não decimais são juntados (isto é de A a F) e decimalizados subtraindo 10 a partir de seus valores. Os dígitos decimais resultantes são juntados aos dígitos do passo 5 até obter 4 dígitos. O valor decimal obtido é usado como o PVV.

V.6 Processamento seguro

O *Cardholder* digita o PIN dele em um ATM ou em um terminal POS; a verificação do PIN é geralmente feita em outro lugar que não seja o próprio terminal. Assim, é importante transmitir como digitado pelo cliente. Para assegurar a confidência, o PIN digitado pelo cliente é criptografado para transmissão.

V.6.1 Conceito de zona de operação

Se o *Switch* não pode verificar o PIN, como é o caso de muitas transações *off-us,* o PIN deve continuar com a mensagem de transação para o servidor do sistema. A chave de trabalho usada para criptografar o PIN entre o *Switch* e o servidor não é a mesma chave de trabalho usada entre o *acquirer* e o *Switch.* Este conceito de usar chave de trabalho diferente

entre sistemas de processamento, ou nós em transmissão, é conhecido como zona de operação de criptografia.O conceito de zona de operação é mostrado na figura abaixo:

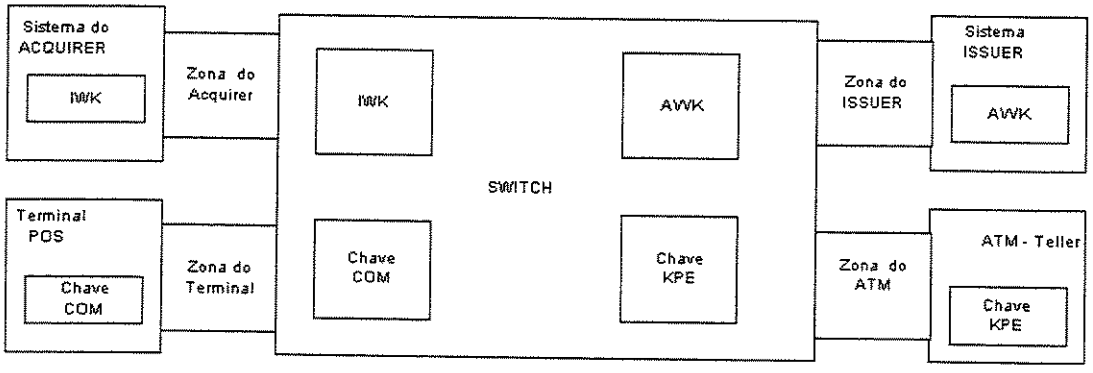


Figura 25: *Conceito de zona de operação do switch*

V.6.2 A zona de operação

Antes que o PIN seja encaminhado ao *issuer,* ele é re-criptografado usando uma outra chave em um processo chamado de tradução de PIN.

V.6.3 Chaves de zona de operação

Dentro de cada zona de operação existe uma chave para criptografar os *PINs* em uma requisição de transação. Apenas o *switch* e o processador que formam a zona de operação sabem a respeito da chave. As chaves podem ser estabelecidas usando o intercâmbio de chaves dinâmicas entre o processador e *switch*. Entre o terminal POS e o *switch,* temos as chaves COM e entre o ATM e o *switch*, temos as chaves KPE. Entre uma rede de *acquirer* e o *switch* temos as seguintes características de transações:

Vindo de	**Indo para**	**Chaves**
POS	Switch	KPE ou COM
ATM	Switch	KPE ou COM
Acquirer	Switch	IWK
Switch	Issuer	AWK

Tabela 6: *Chaves de zona de operação envolvidas nas transações*

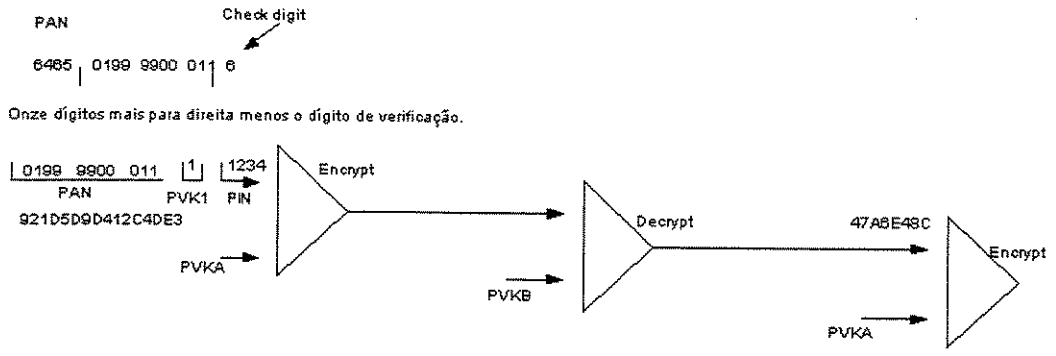


Figura 24: *Zona de operação de criptografia*

V.6.4 Chaves

Quando um *cardholder* entra com o PIN secreto dele em um terminal ATM ou POS, a certificação do PIN é geralmente feita em algum lugar diferente do terminal. Mas, é necessário transmitir o PIN como foi digitado pelo cliente. Para assegurar a segurança do cliente, o PIN digitado pelo cliente é criptografado na transmissão.

Para transmissão, a chave é usada para criptografar o PIN. A chave deve ser conhecida pelo dispositivo que envia o dado para poder ser criptografado, e pelo dispositivo que recebe o dado para poder ser decriptografado.

V.6.5 Dígito verificador

O dígito de verificação é um número de 4 dígitos obtido pela criptografia de um bloco de zeros com uma chave dada. Os primeiros 4 dígitos obtidos são dígitos de verificação da chave. Eles são usados para verificar se as cópias da chave são idênticas.

Dígitos de verificação são úteis na troca de chaves entre redes e na verificação do uso correto da chave durante a criptografia em um dispositivo seguro. O dígito de verificação não pode ser utilizado para obter os valores nítidos da chave. Os dígitos de verificação são guardados no *switch* apenas para efeito de informação

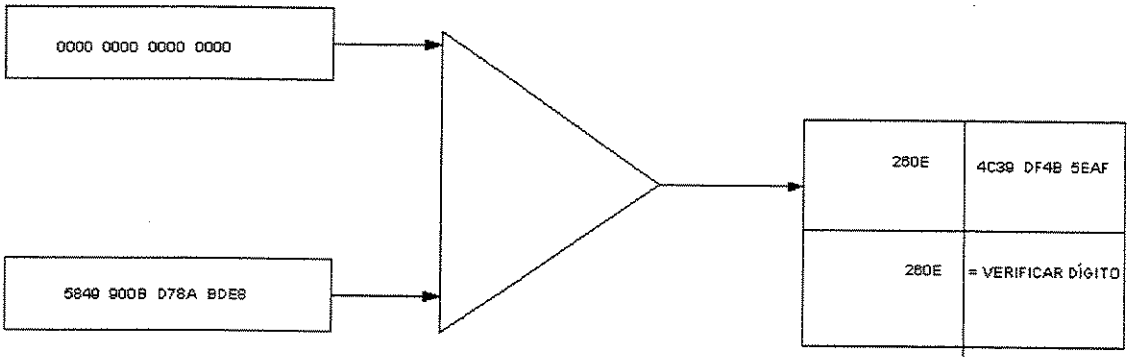


Figura 26: *Verificação de dígito*

V.7 Verificação do PIN

No caso de transações On-us, é possível ter o *switch* configurado para verificar o PIN do *cardholder*. Mesmo assim, uma instituição pode ainda querer que isto seja realizado por seu próprio servidor. Neste caso, a verificação do PIN é feita como qualquer outra transação *off-us*. O processo de verificação do PIN pelo *switch* é mostrado na figura 27.

O *acquirer* (ATM, POS ou REDE) cria e criptografa o PIN usando uma chave W e o envia ao *switch*. O *switch* coleta os parâmetros e as informações de segurança relacionadas ao *acquirer* e ao *issuer* e os envia ao módulo de segurança (HSM) junto com o PIN criptografado para efeito de verificação. O HSM verifica o PIN, este processo envolve:

a- Decriptografar o PVK com o MFK

b- Decriptografar a chave W com o MFK.

c- Decriptografar o PIN com a chave W.

d- Aplicar o procedimento de verificação de PIN apropriada.

O HSM retorna uma resposta VÁLIDA ou INVÁLIDA.

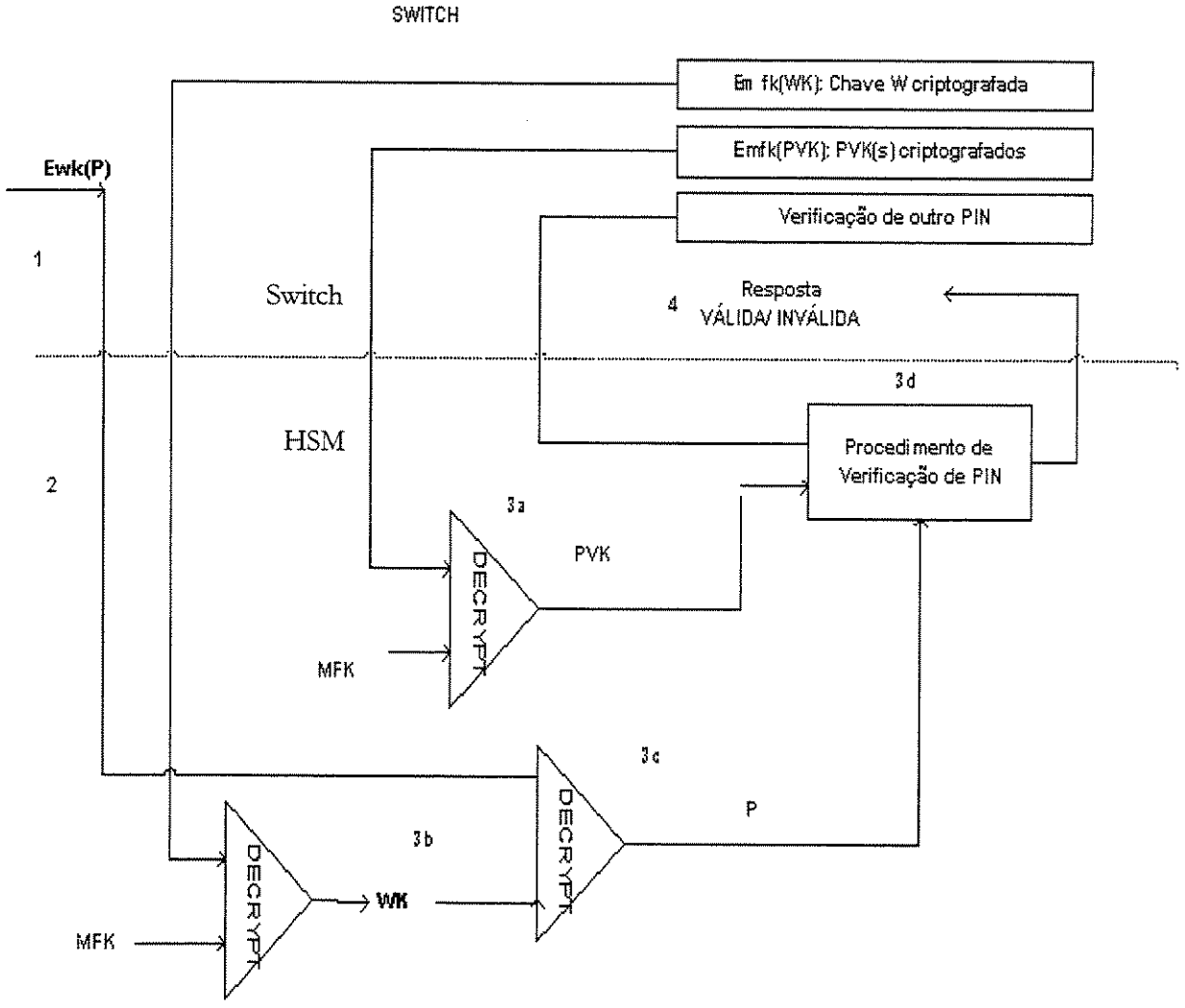


Figura 27: *Fluxo de informações para transações on-us*

V.8 Códigos de mensagens de Autenticação

O MAC é enviado ao recipiente como parte da mensagem. O recipiente tenta recriar o mesmo MAC a partir da mensagem e o compara com o MAC recebido. Se eles coincidem, então tem uma alta confiabilidade, isto é, o que foi enviado corresponde ao que foi recebido. Se os valores do MAC comparados forem diferentes, pode se chegar a seguintes conclões: ou a mensagem foi alterada, ou a linha de comunicação foi grampeada, ou as chaves MAC

não são sincronizados. Os requesitos de códigos de mensagens de autenticação variam entre cada duas instituições.

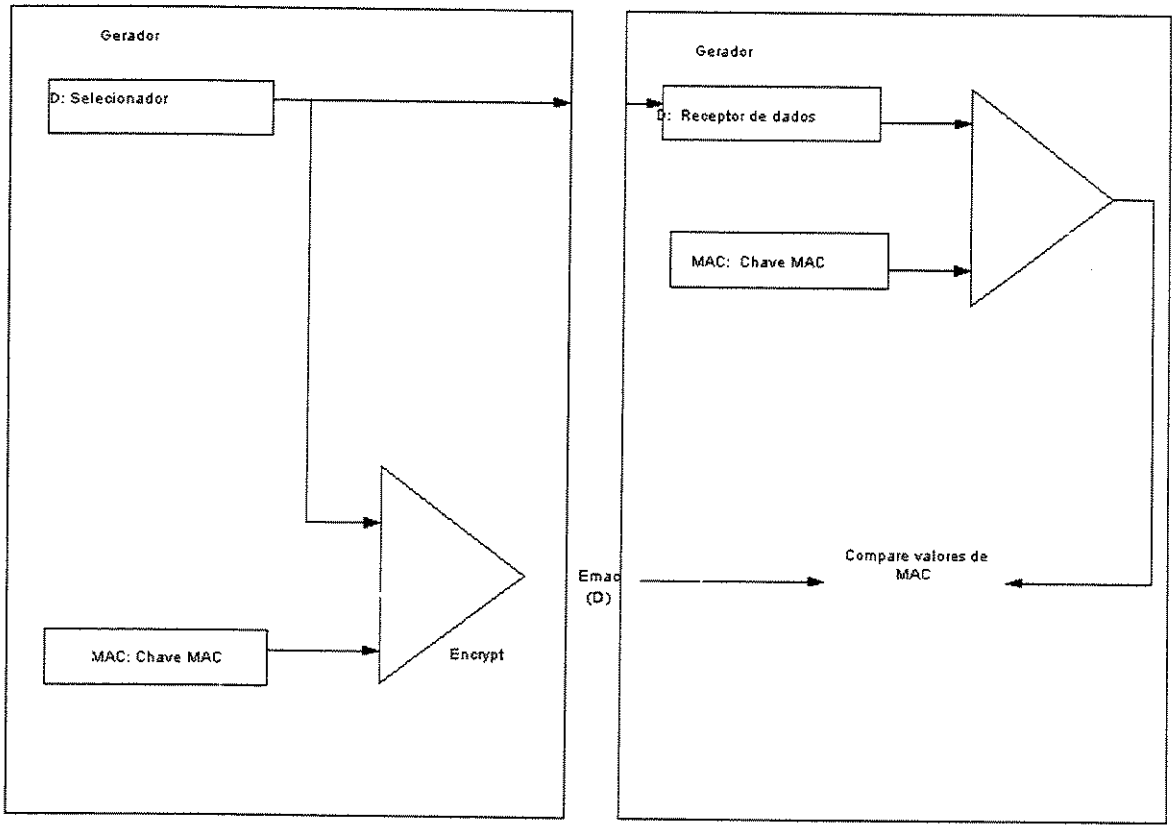


Figura 28: *Fluxo de Códigos de Autenticação de Mensagem(MAC)*

Capítulo VI

Arquitetura Tecnológica

A arquitetura tecnológica contempla toda a plataforma técnica no qual o sistema é implementado. Neste capítulo são tratadas as arquiteturas de redes LAN, WAN e de *Clusters*. A arquitetura interna do sistema é baseada nas regras de negócios mapeadas na arquitetura de aplicação.

VI.1 Especificação técnica do processador principal

O processador principal é composto de 4 máquinas E4500 escaláveis e distribuídas da seguinte forma: 2 servidores de bancos de dados em *cluster* e 2 servidores de aplicações em *cluster* com a seguinte configuração em cada um:

- 4 CPUs de 400MHz cada
- 8GB de Memória RAM
- 16GB (2 X 8GB) de disco rígido interno: *RAID* 0+1, *Hot pluggable*
- Disco rígido EMC Symmetrix 8730 de 1Terabyte , ½ Terabyte para cada servidor
- Estrutura do sistema de armazenamento: RAW (ambiente OPS)
- *Links* ATM, 150MB
- Link de 100Mb/s (Fibra óptica)
- *Load Balancing* com Cisco *Director Locator*
- 2 roteadores e 2 *swithes* Cisco em modo de *Fail-Over*
- 2 *Proxies server e 2 Firewalls*
- Gerenciamento e monitoramento de rede
- *Switch* com *link* a principais redes INET
- *Gateway* interligado a multi-plataforma: *Mainframes*, *Ciscs e Riscs*
- 2 Servidores WEB Compaq Proliant DL 350, Dual, Pentium III, 753 MHZ, 4GB de memória RAM, (2 X 20GB) de disco rígido interno, *hot swappable RAID* 0+1

VI.2 Modelo do *Cluster*

Segue na figura 29 um modelo de cluster implementado para efeito de redundância.

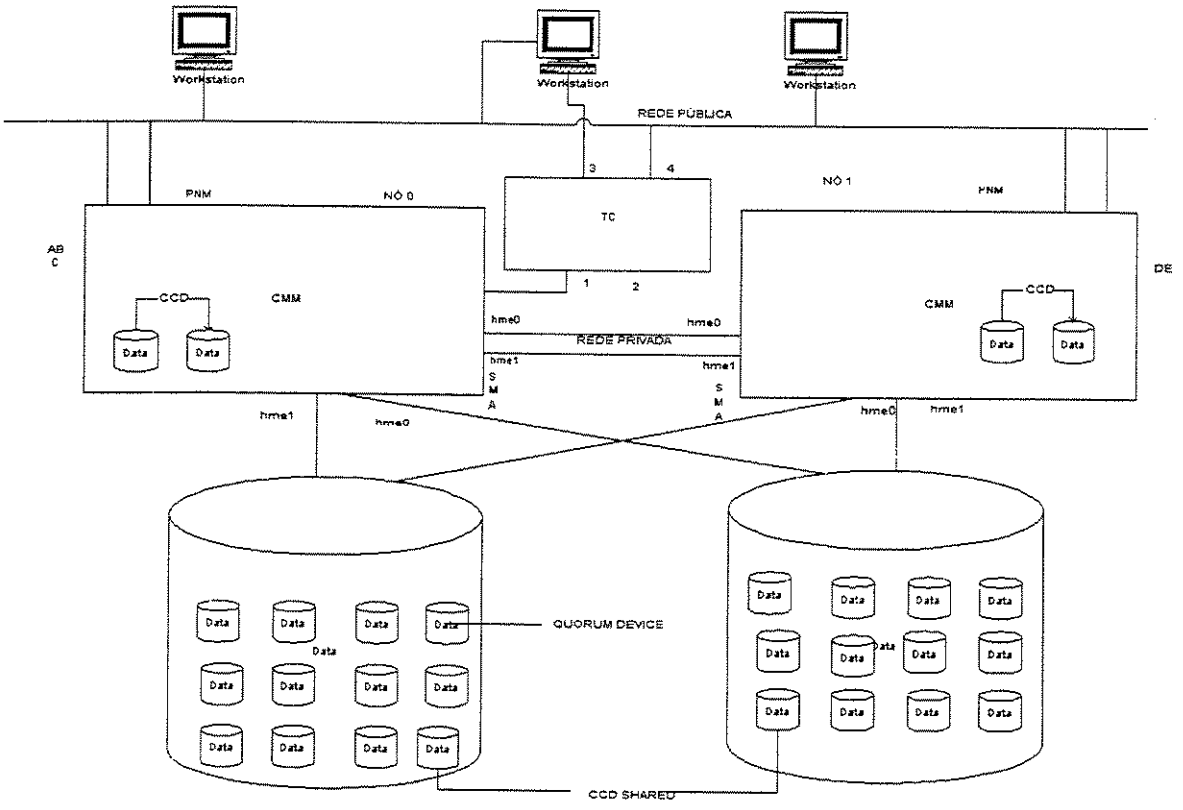


Figura 29: *Esquema do Cluster*

VI.2.1 Serviços *OPS* do *Cluster*

Os dois servidores estão alocados em dois ambientes físicos diferentes. Os dois servidores devem trabalhar em processamento paralelo distribuído com contingência. Eles estão conectados a um *firewall* ligado externamente a um VPN privado. A *firewall* é configurada de tal forma que o servidor não aceita nenhuma requisição vinda da VPN privada, mas, sim, aceita a resposta da VPN que é conseqüente de uma requisição já solicitada. Isto tem efeito de proteger o sistema contra intrusão externa. Os dois servidores rodam em um ambiente distribuído, distribuindo de forma simétrica à carga dos processos 50% para cada servidor. Em caso de falha de um deles, o outro servidor assume os 100% da carga. O sistema é configurado de forma que não haverá perda de performance. Para assegurar as medidas de segurança, os endereços IPs dos dois servidores são "*nateados*", utilizando um roteador Cisco 2610.

VI.2.2 Gerenciamento do *Cluster*

Para gerenciar o Cluster, deve-se monitorar desempenho e as falhas de seguintes elementos: nó, CPUs, Memória, Interfaces de redes interconectadas, *Arrays* de discos, Controladores, Sistema Operacional, Banco de dados. Deve-se, também, diminuir o tempo de detecção de falhas, o tempo para reinicializar os serviços. Deve-se automatizar a detecção de *Crash* e o restabelecimento dos serviços.

VI.3 Arquitetura de aplicações

O diagrama da figura 30 mostra o processo de autorização *On-line* em termos de aplicações e conectividade. Ele descreve a relação entre os principais componentes do sistema, a interação com o *website* do *merchant* descrito no Capítulo 8 e os fluxos de transações.

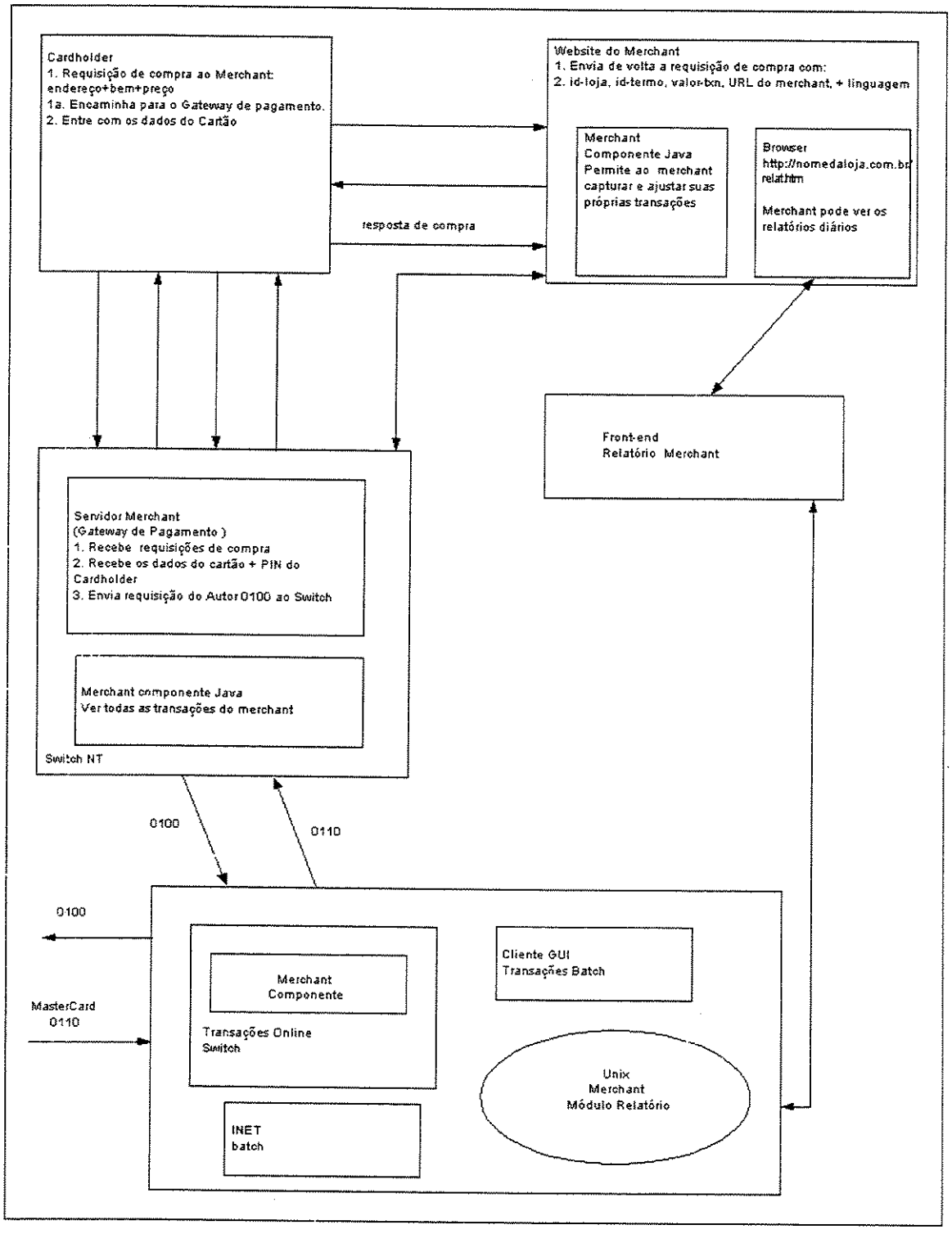


Figura 30: *Arquitetura de aplicações*

Geração de arquivo de processo *merchant*.

A transação *settlement* é encarregada de popular as tabelas. O normalizador *merchant* da transação de *settlement*. As mensagens de *settlement* recebem informações a partir da tabela e popula as tabelas *batches*.

VI.4 Plataforma integrada

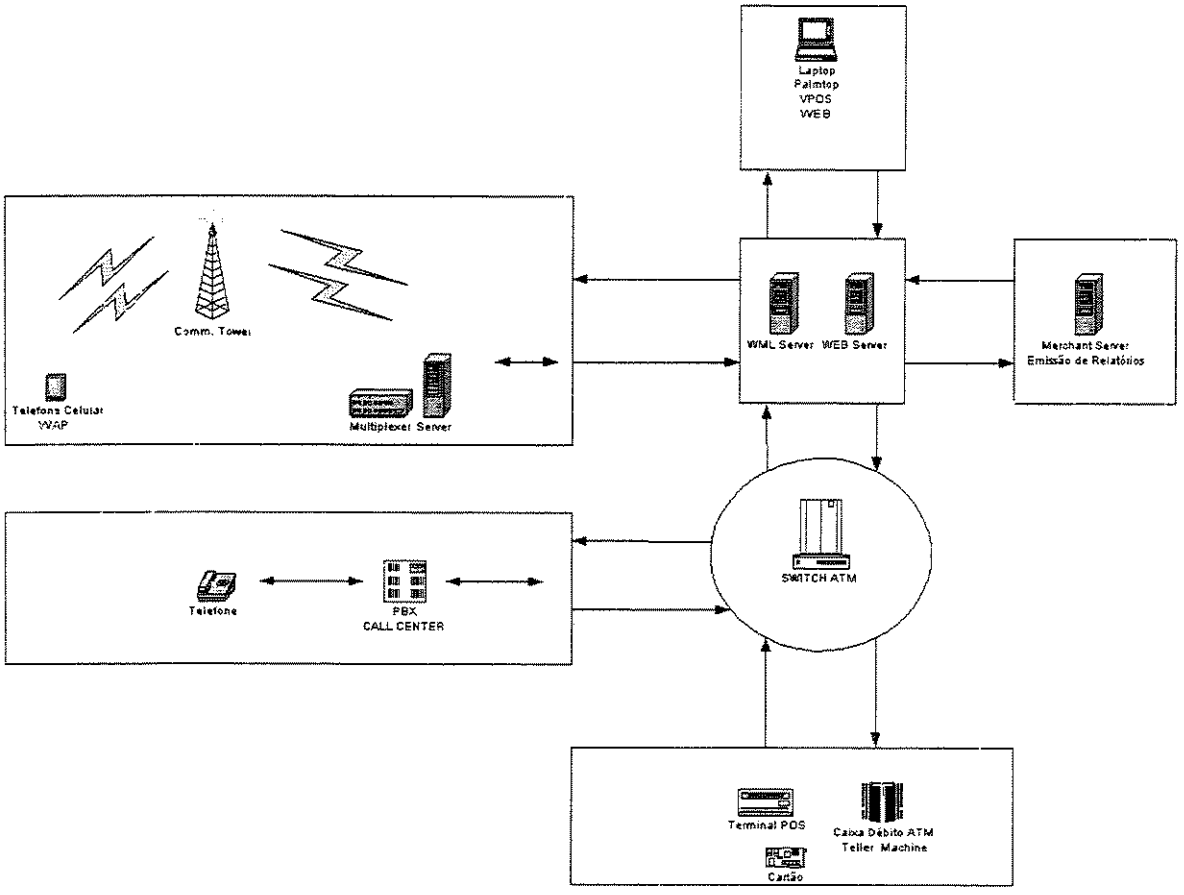


Figura 31: *Macro visão da plataforma integrada*

Capítulo VII

Considerações Finais e Conclusão Geral

Com a implantação do algoritmo de Luhn (LUHN) ganhou-se mais segurança e rapidez nas transações financeiras. Mais rapidez, no sentido de não ser mais necessário enviar os dados do cartão a serem processados até o *Brand* para verificar se estão corretos. A verificação é feita no momento da entrada dos dados. Antigamente, os dados do cartão eram recolhidos pelo *Merchant* e enviados ao *Acquirer* e depois ao *Brand*, que os verificava e mandava de volta, no dia seguinte, ao *Merchant*, que infomava ao usuário o seu estado de compra aceito ou rejeitado – isto conhecido como processo *Batch* – que agora pode ser substituído por um processo *online*, baseado na checagem utilizando o algoritmo de Luhn (ver implementação em C no capítulo VIII).

Outro ganho com a implantação desta solução é a minimização do tempo de resposta. Pelo fato de que os dados não trafegam na rede na fase de verificação, então não haverá impacto sobre as demais transações que estarão sendo realizadas naquele momento, evitando com isto prováveis *timeout* ou até mesmo que transações muito importantes sejam derrubadas.

Com a implantação do algoritmo de Luhn, o processo se torna mais seguro, no sentido de que, com a pré-autorização, se cria mais uma camada de segurança que dificulta as tentativas de fraudes e bloqueia dados falsos do acesso à rede de transações.

Considerando o fato que todos os *Brands* de Cartão estão fora do país, soluções que viabilizam rapidez e segurança nas transações financeiras são fundamentais. Além disso, com a expansão das redes de telecomunicações de milhas de quilômetros de *links* de fibra óptica, os enlaces de satélites, o advento das redes SDH e DWDM e dos protocolos ATM, se faz necessário implantar uma solução multi-plataforma sem perda de QoS. Os dispositivos *Wireless* estão crescendo no uso em transações financeiras, mas apresentam um tempo de resposta maior do que as feitas em *wired* ATM em fibra óptica. Assim, com a implantação do *Wireless* ATM baseada no MAC CDMA/TDMA (MAHMOUD, 1996) e a melhoria da performance no protocolo TCP/IP usando o algoritmo Pseudo-Bayesiano (BAYES, 1997) nas

estações de rádios bases dos dispositivos de comunicações *wireless*, pode-se integrar os ambientes e melhorar o nível de QoS do *wireless* ATM aproximadamente no mesmo nível do *wired* ATM de fibra óptica (J.F. Frigon, 1998).

A integração das redes ATM fixas e ATM *Wireless* sem perda de qualidade de serviços da plataforma tecnológica[figura 31], associada aos aspetos de segurança e rapidez providos na implementação do algoritmo de Luhn oferecem uma infra-estrutura robusta e versátil.

Esta plataforma integrada representa uma solução consolidada para Videoconferência, treinamento à distância, cirurgia remota e para vários outros tipos de aplicações e transações onde o QoS tem um fator decisivo.

O algoritmo Pseudo-Bayesiano (BAYES, 1997) aqui apresentado e integrado na plataforma tecnológica da solução, mostrou-se muito eficiente (J.F. Frigon, 1998).

Além da contribuição tecnológica da solução aqui implementada em termos de segurança, tempo de resposta menor, rapidez e utilizando toda tecnologia da Internet, temos outras características de negócios, como o Cashless Economy (WOODFORD, 2002) cuja a implantação tem sua operacionalização em solução tecnológica baseada na infra-estrutura aqui desenvolvida. O mundo do e-Business que, em soma, pode ser caracterizada como uma visão micro-econômica do mundo macro do Cashless Economy do Woodford é totalmente fundamentado sobre soluções tecnológicas eficientes e seguras.

O *Cashless Economy* aplicado no mundo global, de uma certa forma, não conheça barreiras aduaneiras de circulação de moedas, não está sujeito a especulação monetária feita sobre a moeda nos mercados de ações (onde a presença ou a escassez pode derrubar em fração de segundo uma economia), além disso, limita o poder de intervenção dos bancos centrais.

A solução tecnológica transacional está basicamente formatada neste trabalho, mas para motivação de trabalhos futuros, pode-se desenvolver regras de negócios implementadas em software de gerenciamento para formatar toda lógica transacional, como foi o caso do cartão, tendo em vista as leis e regimentos de toda a comunidade global financeira, envolvida na nova realidade de *Cashless Economy*.

Um fator limitante neste sentido é que o mundo de comércio eletrônico e de transações financeiras, com os avanços tecnológicos, pode aumentar ainda mais o *gap* entre paises desenvolvidos e sub-desenvolvidos, e acabando por isolar estes últimos no cenário econômico do comércio eletrônico. Assim, cabe fazer um grande esforço para tornar realidade a inclusão digital destes paises no ambiente de projetos, como iniciativas de ONGs (Organizações Não-Governamentais) ligadas ao ICT (*Inforamtion Communication Technology*).

Este trabalho teve como objetivo contribuir para o desenvolvimento de novos protocolos, de soluções de segurança e de plataforma tecnológica integrada e inovadora para as novas realidades de negócios, em particular e transacionais, pela Internet em geral.

Entre outros, os benefícios oriundos desta solução foram amenizar drasticamente as despesas com infra-estrutura logística, gerar maior volume de negócios diários, amenizar os riscos de fraudes nas transações financeiras, e ampliar o uso da plataforma desenvolvida para aplicações visando o desenvolvimento econômico-social.

Capítulo VIII

Arquitetura Web – Exemplo de um caso

No processamento *online*, o *cardholder* deve saber na hora se a compra dele for aprovada ou rejeitada. Isto é o processamento do cartão deve ser efetuado em tempo real, o que é diferente do processamento de lote onde o *cardholder* tem que esperar o dia seguinte para saber se a compra dele foi aprovada ou negada. No processamento de lote ou *batch*, o *Merchant* recolhe todos os números de cartões no fim de expediente e envia para o *Brand* que decide se o cartão em questão é aceito ou rejeitado. A implementação do algoritmo de Luhn, permite processar o número do cartão do *cardholder* em tempo real no *Website* do *Merchant*.

Neste Capítulo, faremos a montagem de um caso, implementando os códigos de programas do *Site* do *Merchant* para armazenar e extrair informação usando o ODBC (*Open Database Connectivity*). Serão implementados, também, os códigos de programas do carrinho de compras (*Shopping cart*) com os seus módulos: adicionar itens, remover itens, consultar novamente os itens selecionados etc.

Na seção seguinte, será mostrado o algoritmo de Luhn mod 10 para validar *online* os números de cartões de crédito digitados pelo *cardholder* na compra pela Internet.

Uma interface de comunicação baseada no protocolo TCP/IP entre o *Merchant* e a instituição *Acquirer* pode ser estabelecida por um canal VPN seguro.

VIII.1 Site virtual do Merchant

O website do *Merchant* recebe pedidos e, após receber e validar o pagamento, ele os coloca para distribuição, a qual envia diretamente para o cliente. Para um bom desempenho do *website*, sempre procure um disco rápido de acesso muito veloz para armazenar os dados. Este tipo de disco permite aos usuários buscar rapidamente os pedidos e rever o carrinho de compras.

O carrinho de compras ou *Shopping Cart* permite ao comprador adicionar, remover ou consultar itens. O próximo passo após selecionar os itens de compras, é entrar com o número do cartão de crédito.

A tabela de carrinho de compras tem duas chaves: uma que liga para cliente e outra que liga para produto ou tabela de inventário.

A tabela de produto contém o nome da loja, o IDinventario, o custo, a descrição do produto, o desconto e a quantidade em armazém.

A tabela do cliente contém todos os dados do cliente: nome, sobrenome, endereço, cidade, estado, CEP, telefone, e-mail, IDreferencia para ligar para as demais tabelas.

A tabela de pagamento precisa manter informações significativas do cartão de crédito do cliente que ele usará para pagar a compra. A tabela de pagamento contém os códigos de aprovação, dados de expiração, número do cartão até 16 dígitos, valor da compra, descrição, taxas.

Implementamos o algoritmo de *Luhn* que verifica se o tipo de cartão de crédito informado pelo cliente está correto para efeito de pré-autorização.

A tabela de pedido informa quatro itens, quais tipos de produtos. Ela é parecida com a tabela de carrinho de compras, exceto que a tabela de pedido é estática, ela muda apenas se a mercadoria foi pedida. Neste caso um *flag* é definido para que a mesma ordem não seja repetida. Define-se também um *flag* para identificar se a mercadoria foi enviada ou não.

VIII.2 Estrutura do diagrama de relacionamento

Todas as tabelas são ligadas umas às outras. Usando uma tabela, pode-se acessar outras.

As chaves usadas são IDreferencia e IDinventario.

IDreferencia liga o usuário, o conteúdo do carrinho de compras, a informação do cartão de crédito, endereço para entrega e outras tabelas do banco de dados.

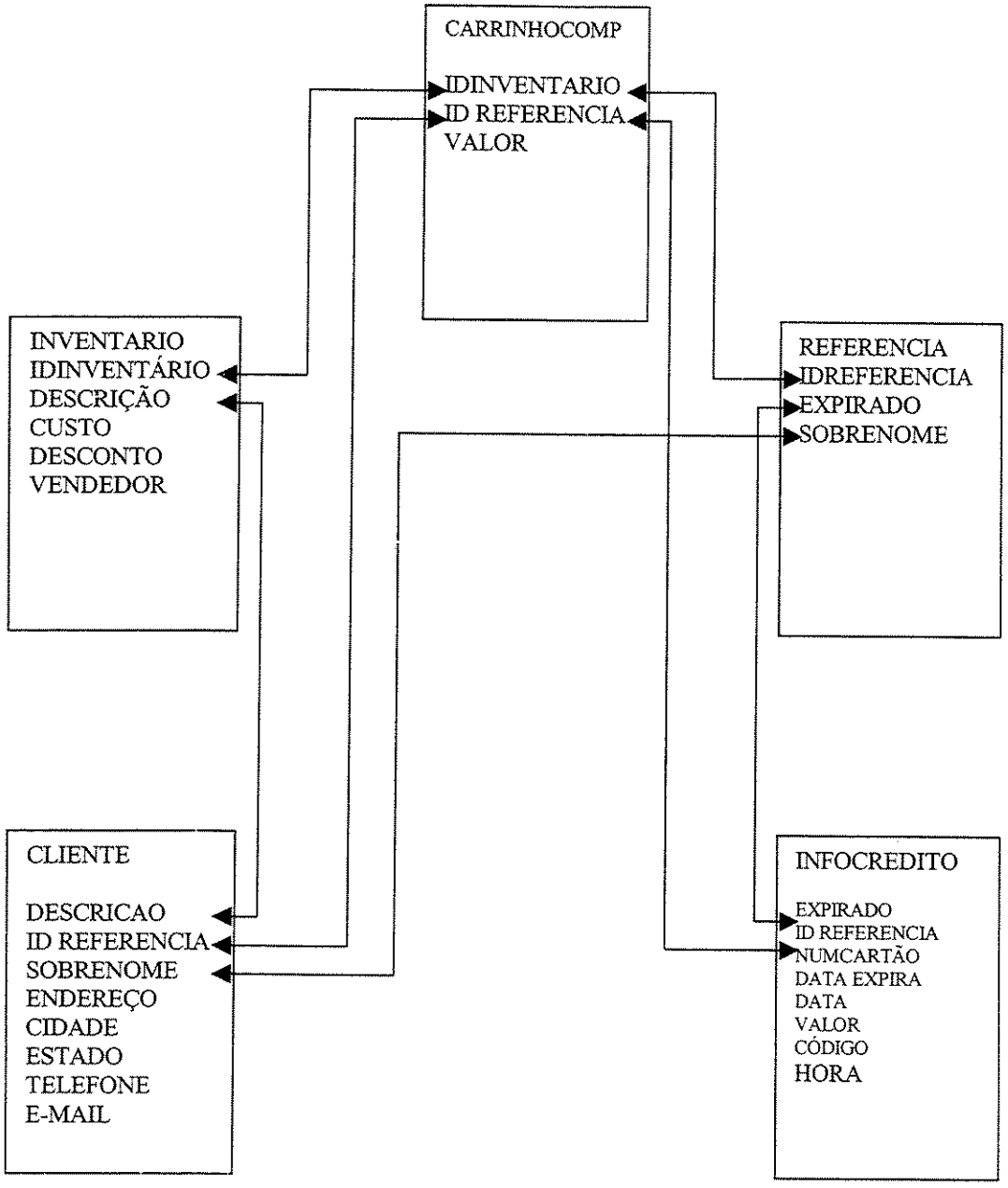


Figura 32: *Diagrama de relacionamento das entidades*

Um banco de dados relacional, neste caso o Oracle foi utilizado para criar as tabelas. Pode-se utilizar outros tipos de bancos de dados relacionais como por exemplo, o MySql (*freeware*) rodando com PHP e o Apache em um servidor Linux. O *software* Erwin como o Oracle

designer pode ser utilizado para criar diagramas de relacionamentos de entidades mais complexos e para criar estruturas de tabelas.

Pode-se estabelecer permissões de acesso nas tabelas. É possível, também, se conectar ao banco de dados via ODBC (Open *Database Connectivity*) ou JDBC (Java *Database Connectivity*) com um *login* e senha por razões de segurança.

VIII.3 Interface ODBC e Banco de dados

Após criar o banco de dados, é necessário compartilhar usando o ODBC ou o JDBC. O ODBC permite acessar as informações no banco de dados como descrito na figura33: Acesso ao banco de dados usando ODBC ou JDBC.

O ODBC tem diferentes chamadas de funções baseadas em SQL. Algumas rotinas que foram utilizadas são listadas na tabela seguinte:

Nome da função	**Descrição**
SQLConnect()	Conecta na base de dados especificado
SQLAllocConnect()	Aloca uma conexão ao dispositivo de ODBC
SQLExecute	Executa o procedimento especificado pelo SQLAllocStmt()
SQLAllocStmt()	Aloca um *handle* para o procedimento do SQL especificado a ser enviado ao DB

Tabela 7: *Funções de chamadas ODBC*

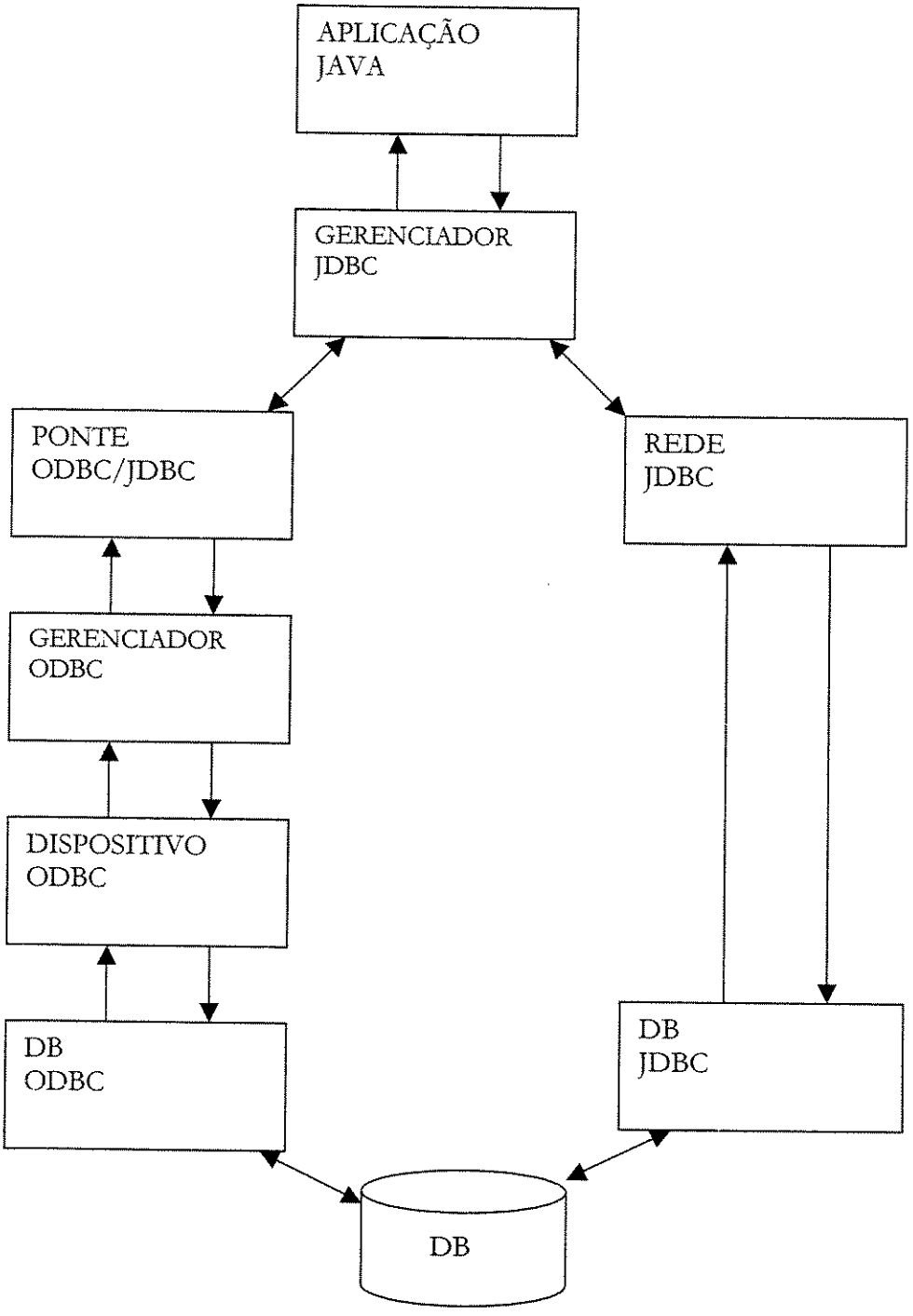


Figura 33: *Acesso ao banco de dados usando* ODBC *ou* JDBC

VIII.4 Algoritmo de Luhn mod 10

O algoritmo de Luhn (LUHN) é usado para gerar o dígito de verificação do Cartão e comparar com o número de cartão fornecido pelo usuário.

Com a implementação do algoritmo de *Luhn*, pode-se processar em tempo real com segurança o cartão de crédito digitado pelo *cardholder.*

O número do cartão de crédito do *cardholder* é primeiramente verificado sem gerar tráfego na rede. Se o número estiver de acordo com os parâmetros da instituição detentora da *brand* conforme a tabela: Informação sobre o cartão de crédito. Todos os números de cartões providos pelos *brands* listados nesta tabela serão considerados na pré-autorização. Depois de encerrar a pré-autorização, inicia-se a fase de autorização conforme descritos nos capítulos 1 e 2. Se o cartão for autorizado, então a compra é permitida. Depois disso inicia-se o processo de *Clearing & Settlement* definido periodicamente, entre as diferentes instituições financeiras envolvidas na transação.

INFORMAÇÃO SOBRE CARTÃO DE CRÉDITO		
TIPO DE CARTÃO	PREFIXO	COMPRIMENTO
MasterCard	51 a 55	16
Visa	4	13 ou 16
American Express (AMEX)	34 ou 37	15
Diner's club/Carte Blanche	300 a 305 ou 36 ou 38	14
Discover	6011	16
JCB	3	16
JCB	2131 ou 1800	15

Tabela 8: *Prefixo e comprimento de cartões de crédito*

VIII.5 Código do programa do algoritmo de Luhn

```
//Arquivo: Cartao.cpp

    // headers

    #include "Cartao.hpp"

//*********************************************************************//
    #include <sys/types.h>

    #include <stdio.h>

    #include <ctype.h>

//*********************************************************************//

// função:  CCartao::DoCartao()
//   objetivo: chama a função e processa qualquer pedido de um  cliente
// retorna:   nada
//
void CCartao::DoCartao()


{

cout << "<CENTER>Verifica o número... <BR>" << endl;
if ( VerficaCartao() )
{
cout << "OK...<BR>Conexão ao banco para aprovação...<BR>" << endl;

if ( SeguraCartao() )
                                        MostraAprovado();
else
                                        MostraRejeitado();
}
else
MostraCartaoRuim();
cout << "O Cartão existe<BR>";
} // CCartao::DoCartao...



//
```

```
// função: CCartao::MostraAprovado()
// objetivo: exibe mensagem aprovada
// retorna: nada
//
    void

    CCartao::MostraAprovado()

{
cout << "<CENTER>Seu pedido foi aprovado."

"A remessa está sendo encaminhada pelo correio. Obrigado por ter feito negócio

conosco.<BR><BR><BR>";

} // CCartao::MostraAprovado...


//
// função: CCartao::MostraCartaoRuim()
// objetivo: mostra a mensagem Cartao não aprovado
// retorna: nada
//
    void

    CCartao::MostraCartaoRuim()

{
cout << "<CENTER> Este número de Cartão não consta. Digite

novamente o seu número de Cartão."<BR><BR><BR>";

} // CCartao::MostraCartaoRuim...

//
// função: CCartao::MostraRejeitado()
// objetivo: mostra a mensagem Cartao rejeitado.
// retorna: nada
//
    void

    CCartao::MostraRejeitado()

{
cout << "<CENTER>Desculpe, mas seu pedido foi rejeitado."

"chame nosso central de assistência ou"
```

```
"digite novamente um numero de Cartão válido.<BR><BR><BR>";

} // CCartao::DispalyRejeitado...


//
//  função:  CCartao::VerficaCartao()
//   objetivo: Verifica o comprimento e os dígitos do Cartao
//  retorna:  verdade se for válida, falso no caso contrário
//
bool CCartao::VerificaCartao()

{
bool   resultado = false;
switch ( m_numeroCartao.comprimento() )
{
case 13:
        // deve ser VISA

        if ( m_acceptVisa )

resultado = VerificaVisa();
break;
case 14:
        // Diners Club

        if ( m_acceptDiners )

resultado = VerificaDiners();
break;
case 15:
        // American Express, JCB

        if ( m_acceptJCB )

resultado = VerificaJCB();
if ( !resultado && m_acceptAmex )
resultado = VerificaAmex();
break;
case 16:
// MasterCard, Visa, Discover, JCB
if ( m_acceptMasterCard )
resultado = VerificaMasterCard();
if ( !resultado &&  m_acceptVisa )
resultado = VerificaVisa();
if ( !resultado && m_acceptDiscover )
resultado = VeirficaDiscover();
```

```
else if ( !resultado && m_acceptJCB )
resultado == VerificaJCB();
break;
default:


} // switch...
return resultado;
} // CCartao::VerificaCartao...


//
//  função:  CCartao::VerificaVisa()
//  objetivo:   Verifique o Cartao de Crédito de tipo Visa  CCartao::VerificaVisa()
{
bool resultado = false;
short numero = m_numeroCartao.c_str()[0] - '0'; if ( numero == 4 )
resultado = DoVerificaSoma();
return resultado;


} // CCartao::VerificaVisa...


//
//  função:  CCartao::VerificaMasterCard()
//  objetivo: verifica o Cartao como do tipo MasterCard
//  retorna:  verdade se for válido e falso caso contrário
//
bool CCartao::VerificaMasterCard()


{
bool resultado = false;
char  valor[2];
        strncpy(valor, m_numeroCartao.c_str(),sizeof(valor));

        short numero = atoi( valor );

        if ( numero >=51 && numero <=55 )

resultado = DoVerificaSoma();
return resultado;
} // CCartao::VerificaMasterCard...


//
//  função:  CCartao::Amex()
```

```
//   objetivo: verifica o Cartao de crédito do tipo
// American Express.
//   retorna:  verdade se for válido, falso caso contrário
//
bool CCartao::VerificaAmex()


{
bool resultado = false;
char  valor[2];
        strncpy(valor, m_numeroCartao.c_str(),sizeof(valor));

        short numero = atoi( valor );

        if ( numero ==34 || numero ==37 )

resultado = DoVerificaSoma();
    return resultado;

    } // CCartao::VerificaAmex...


//
//   função:  CCartao::VerificaDiners()
//   objetivo: verifica o Cartao de crédito do tipo
// Diner's Club ou Carte Blanche.
//   retorna:  verdade se for válido, falso caso contrário
//
bool CCartao::VerificaDiners()


{
bool resultado = false;
char  valor[2];
        strncpy(valor, m_numeroCartao.c_str(),sizeof(valor));

        short numero = atoi( valor );

        if ( numero ==36 || numero ==38 )

resultado = DoVerificaSoma();
else
{
char temp[3];
strncpy( temp, m_numeroCartao.c_str(),sizeof(temp) );


numero = atoi( temp );
if ( numero >=300 && numero <= 305 )
resultado = DoVerificaSoma();
}
```

```
    return resultado;

    } // CCartao::VerificaDiners...


//
// função:  CCartao::VerificaDiscover()
// objetivo: Verifica  Cartao de crédito do tipo
//Discover.
// retorna:  verdade se for verdade, falso caso contrário
//
bool CCartao::VerificaDiscover()


{
bool resultado = false;
char  valor[4];
        strncpy(valor, m_numeroCartao.c_str(),sizeof(valor));

        short numero = atoi( valor );

        if ( numero == 6011 )

resultado = DoVerificaSoma();
    return resultado;

    } // CCartao::VerificaDiscover...


//
// função:  CCartao::JCB()
// objetivo: Verifica o Cartao de crédito do tipo
//
// retorna:  verdade caso verdade, falso caso contrário
//
bool CCartao::VerificaJCB()


{
bool resultado = false;
char  valor[4];
        strncpy(valor, m_numeroCartao.c_str(),sizeof(valor));
        short numero = atoi( valor );
        if ( m_numeroCartao.comprimento() == 15 &&
(numero ==2131 || numero ==1800) )
resultado = DoVerificaSoma();
else if ( m_numeroCartao.comprimento() == 16 )
{
char temp = m_numeroCartao.c_str()[0];
numero = m_numeroCartao.c_str()[0] - '0';
if ( numero == 3 )
resultado = DoVerificaSoma();
```

```
}

    return resultado;

    } // CCartao::VerificaJCB...


//
//  função:  CCartao::DoVerificaSoma()
//  objetivo:   Verifica os dígitos do Cartão usando o algoritmo Mod 10 de
//              Luhn
//  retorna:   verdade se os dígitos são válidos, caso contrário falso
//
bool


CCarrinho::DoVerificaSoma()


{
  bool   resultado = false;
  vector<short>   digitArray;
  short          modelo;

for (short índice = 0; índice < m_numeroCartao.comprimento(); índice ++)


  {
// converte para inteiro
        modelo = m_numeroCartao.c_str()[índice] - '0';
//  o array toma conta de tudo
        digitArray.pback( modelo );
  }

// agora totaliza os dígitos reversos e os dobra
  //   parte do algoritmo do Luhn
  vector<short>::reverso_iterator   iter;
        bool passo = true;

        short soma = 0;

for (iter = digitArray.rbegin(); iter != digitArray.rend(); iter++)


  {
if ( passo )
        // totaliza os dígitos
soma += *iter;
        else
    {
        short valor = *iter * 2;
```

```
        if ( valor >= 10 )

        {
          soma += 1;

          valor -= 10;

        }
soma += valor;
    }
passo = !passo;                    // pula cada numero


  }

return ( soma %10 == 0 );
}  // CCartao::DoVerificaSoma
```

Referências Bibliográficas

[1] M. Ellis and B. Stroustrup, The Annotated C++ Reference Manual, Addison-Wesley, Massachusetts, 1990.

[2] D. Kapur, D. R. Musser, and A. A. Stepanov, "Tecton, A Language for Manipulating Generic Objects," Proc. of Workshop on Program Specification, Aarhus, Denmark, August 1981, Lecture Notes in Computer Science, Springer-Verlag, vol. 134, 1982.

[3] D. Kapur, D. R. Musser, and A. A. Stepanov, "Operators and Algebraic Structures," Proc. of the Conference on Functional Programming Languages and Computer Architecture, Portsmouth, New Hampshire, October 1981.

[4] A. Kershenbaum, D. R. Musser, and A. A. Stepanov, "Higher Order Imperative Programming," Technical Report 88-10, Rensselaer Polytechnic Institute, April 1988.

[5] A. Koenig, "Associative arrays in C++," Proc. USENIX Conference, San Francisco, CA, June 1988.

[6] S. Gonzalez, "eLibrary," MasterCard, 1997.

[7] D. R. Musser and A. A. Stepanov, "A Library of Generic Algorithms in Ada," Proc. of 1987 ACM SIGAda International Conference, Boston, December, 1987.

[8] D. R. Musser and A. A. Stepanov, "Generic Programming," invited paper, in P. Gianni, Ed., ISSAC '88 Symbolic and Algebraic Computation Proceedings, Lecture Notes in Computer Science, Springer-Verlag, vol. 358, 1989.

[9] Carlos F. Franco Jr., "e-Business, Tecnologia de Informação e Negócios na Internet," Editora Atlas, 2001.

[10] D. R. Musser and A. A. Stepanov, "Algorithm-Oriented Generic Libraries," Software Practice and Experience, vol. 24(7), July 1994.

[11] B. Stroustrup, The Design and Evolution of C++, Addison-Wesley, Massachusetts, 1994.

[12] J. E. Shopiro, "Strings and Lists for C++," AT&T Bell Labs Internal Technical Memorandum, July 1985.

[13] A. A. Stepanov and M. Lee, "The Standard Template Library," Technical Report HPL-94-34, Hewlett-Packard Laboratories, April 1994.

[14] ATM Fórum. Private Network-Network Interface Specification, Version 1.0. Março, 1996.

[15] Laubach, M. RFC 1577: *Classical IP and ARP over ATM.* Janeiro, 1994.

[16] ATM Forum. LAN Emulation over ATM, Version 1.0 Janeiro, 1995.

[17] Mc Gregor, D.N. " A Recommended Error Control Architecture for ATM Networks with Wireless Links, " IEEE Journal on Selected Areas in Communications, Vol. 15, No. 1, January 1997

[18] The Frame Relay Forum. Frame *Relay*/ATM PVC *Network Interworking Implementation Agreement, FRF.* Dezembro, 1994.

[19] The Frame Relay Fórum. Frame Relay/ATM PVC Service Networking Implementation Agreement, FRF. 8.Abril, 1995.

[20] IST, "Information Switching Technology ", May 2000.

[21] ATM Forum. Data Exchange Interface Specification, Version 1.0. Agosto, 1993.

[22] ATM Forum. Frame based User-Network Interface Specifications, Version 1.0. Setembro, 1995.

[23] KENDALL, K.E., KENDALL, J.E. System Analysis and Design. 4th Edition. Upper Saddle River/NJ. Prentice Hall, 1998. p. 903.

[24] KLING, R. Defining The Bounderies Of Computing Across Complex Organizations. Critical Issues in Information Systems Research. New York, Edited by R. J. Boland Jr. and R. A. Hirschheim , John Wiley & Sons Ltd. pp. 307-361, 1987.

[25] LAUDON, K. C., LAUDON, J. P. Management Information System: New Approaches to Organization & Technology. 5th Ed. Upper Saddle River/NJ: Prentice Hall, 1998. p. 693.

[26] OXNER, W & CHARLAB, S. O Seu Futuro Eletrônico: A Revolução da Informação. Gráfica e Editora JB S.A. Rio De Janeiro/RJ. Brasil. 1995.

[27] STAR, R.M. Princípios de Sistemas de Informação: Uma Abordagem Gerencial. 2ª edição. Rio de Janeiro/RJ: LTC–Livros Técnicos e Científicos Ltda, 1998. p.451.

[28] D. Raychaudhuri and N. D. Wilson, ATM-Based transport architecture for multi-services wireless personal communication networks," IEEE Journal on Selected Areas in Communications, vol. 12, pp. 1401{1414, October 1994.

[29] R. O. LaMaire, A. Krishna, P. Bhagwat, and J. Panian, Wireless LANs and mobile networking: Standards and future directions," IEEE Communications Magazine, vol. 34, pp. 86{94, August 1996.

[30] Y. J. Kim, J. K. Kim, and Y. H. Lee, \A new medium access control scheme for wireless ATM networks," in Proceedings VTC'97, (Phoenix, AZ), May 1997.

[31] O. Kubbar and H. T. Mouftah, Multiple access control protocols for wireless ATM: Problems definition and design objectives," IEEE Communications Magazine, vol. 35, pp. 93{99, November 1997.

[32] E. Ayanoglu, K. Y. Eng, and M. J. Karol, Wireless ATM: Limits, challenges, and proposals," IEEE Personal Communications, vol. 3, pp. 18{34, August 1996.

[33] D. Raychaudhuri, L. J. French, R. J. Siracusa, S. K. Biswas, R. Yuan, P. Narasimhan, and C. A. Johnston, WATMnet: A prototype wireless ATM system for multimedia personal

communication," IEEE Journal on Selected Areas in Communications, vol. 15, pp. 83{95, January 1997.

[34] N. Passas, S. Paskalis, D. Vali, and L. Merakos, Quality-of-service-oriented medium access control for wireless ATM networks," IEEE Communications Magazine, vol. 35, pp. 42{50, November 1997.

[35] K.-C. Chen, Medium access control of wireless LANs for mobile computing," IEEE Network, vol. 8, pp. 50{63, September 1994.

[36] D. Sbirk, J. M. Karlsson, L. Falk, and C. Lind, An overview of proposed MAC algorithms for wireless ATM," in The Thirteenth Nordic Teletraffic Seminar, (NTNU, Trondheim, Norway), 1996. available at http://www.tts.lth.se/Personal/daniels/papers/nts13.abstract.html.

[37] S. G. Fischer, T. A. Wysocki, and H.-J. Zepernick, MAC protocol for a CDMA based wireless ATM LAN," in Proceedings ICC'97, (Montr_eal, Canada), June 1997.

[38] R. Pichna and Q. Wang, A MAC protocol for the integrated wireless access network," in Proceedings PIMRC'95, vol. 1, (New-York, NY), pp. 248{252, September 1995.

[39] A. S. Mahmoud, D. D. Falconer, and S. A. Mahmoud, A multiple access scheme for wireless access to a broadband ATM LAN based on polling and sectored antennas," IEEE Journal on Selected Areas in Communications, vol. 14, pp. 596-608, May 1996.

[40] D. J. Goodman, R. A. Valenzuela, K. T. Gayliard, and B. Ramamurthi, Packet reservation multiple access for local wireless communications," IEEE Transactions on Communications, vol. 37, pp. 885{890, August 1989.

[41] J. Dunlop, J. Irvine, D. Robertson, and P. Cosimini, Performance of a statistically multiplexed access mechanism for a TDMA radio interface," IEEE Personal Communications, vol. 2, pp. 56{64, June 1995.

[42] N. Amitay and S. Nanda, Resource auction multiple access (RAMA) for statistical multiplexing of speech in wireless PCS," IEEE Transactions on Vehicular Technology, vol. 43, pp. 584{596, August 1994.

[43] D. Raychaudhuri, Wireless ATM networks: Architecture, system design and prototyping," IEEE Personal Communications, vol. 3, pp. 42{49, August 1996.

[53] A. Papoulis, Probability, Random Variables, and Stochastic Processes. McGraw-Hill, third ed., 1991.

[44] V. Paxson and S. Floyd, \Wide area traffic: The failure of Poisson modeling," IEEE/ACM Transactions on Networking, vol. 3, pp. 226{244, June 1995.

[45] A. Leon-Garcia, Probability and Random Processes for Electrical Engineering. Addison-Wesley, second ed., 1994.

[46] R. Jain, Congestion control and traffic management in ATM networks: Recent advances and a survey," Computer Networks and ISDN Systems, vol. 28, pp. 1723{ 1738, 1996.

[47] Esin Haritaoglu, Wireless Communication Networks over TCP, ATM and Location Update and Paging Policies, " ENEE609A Independent Study Term Report"

[48] J. F. Frigon, "Dynamic reservation TDMA medium access control protocol for wireless ATM networks," Master's thesis, University of British Columbia, August 1998.

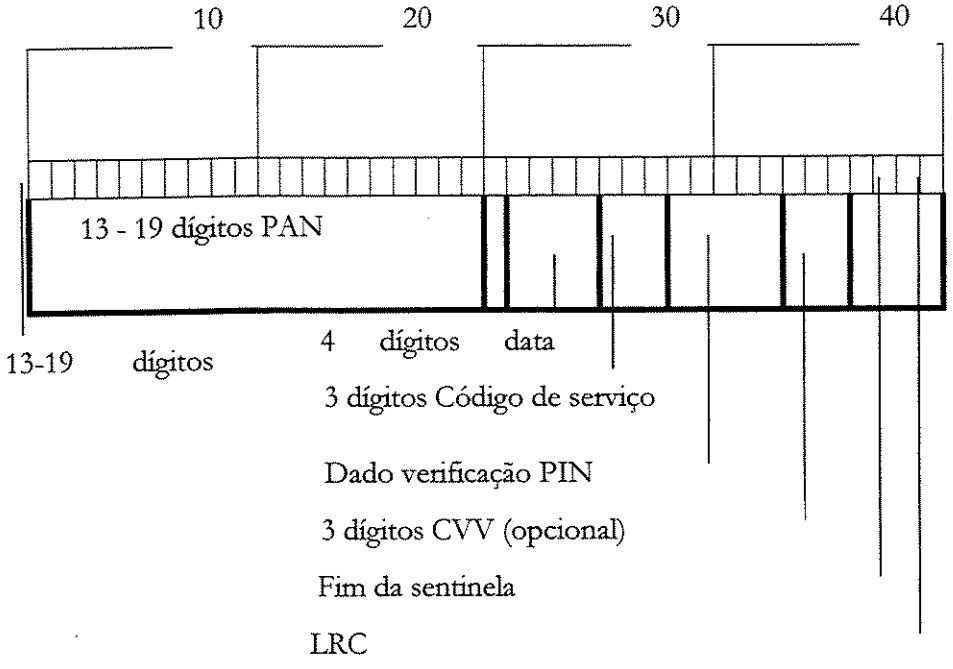
[49] Mamadou M. Ndiaye, " Luhn Algorithm Implementations and Applications," Smart Cards Seminary, 2000. www.expertnet.com.br/e-education/luhn.htm.

[50] Michael Woodford, "Interest and Prices: Monetary Policy in a Cashless Economy, "Chapter 2, pp41-45, Princeton University, January 2002.

Apêndice A

Tecnologia de Cartões

O layout da banda magnética é mostrado na figura seguinte. O comprimento máximo do *trak* é de 40 caracteres. O comprimento varia de acordo com o layout definido pelo issuer de cartões.



Campo	Comprimento	Descrição
1	1	Começo da sentinela
2	1	Código do formato
3	13-19	Número da conta primária
4	E	Separador de Campo
5	2-26	Nome do Cardholder
6	1	Separador de Campo
7	4	Data de Expiração
8	3	Código de Serviço
9	Método dependente	Verificação do PIN
10		Dado Discreto
11		Fim da Sentinela
12		Verificação de redundância

Apêndice B

Glossário de Termos Técnicos

ACQUIRER
Acquirer é uma instituição financeira ou um *switch* que adquire e gera transação.

ATM
Automatic Teller Machine provê acesso automático para o *Cardholder* na sua conta corrente por intermediário de um catão plástico e o PIN de identificação

ATM
Asynchronons Transfer Mode, Modo de Transferência Assíncrona

AUTORIZAÇÃO
Autorização se refere a uma aprovação dada pelo issuer ao acquirer para completar o processamento da transação.

AUTORIZAÇÃO POSITIVA
é uma autorização baseada na informação de conta corrente do *cardholder*.

AUTORIZAÇÃO NEGATIVA
é uma autorização baseada no limite pré - definido pelo issuer ao *cardholder*.

BIN
Bank Identification Number, é um número de 6 a 12 dígitos atribuídos pela rede para apenas identificar os membros issuers. O BIN de um issuer geralmente aparece nos primeiros 12 dígitos do número do cartão do *cardholder*. O BIN informa o acquirer a agência que deve ser contatada para ter as informações sobre o cliente.

DES
Data Encryption Standard, refere-se a um processo de criptografia reversível.

EDC

Eletronic Draft Capture é o processo de transações POS (*Point Of Sale*), que captura transações do *Merchant* e encaminha ao *switch* para efeito de *settlement*.

EFT

Eletronic Fund Transfer é o gerenciamento de informação financeira eletrônica.

EMBOSSING

É o processo de colocar as informações sobre o cliente no cartão de débito ou crédito (PAN, nome, data de vencimento).

HSM

Hardware Security Module usado para criptografar/decriptografar informações codificadas com chaves.

INET

(*International Network for Electronic Transfer*) é um sistema de SETTLEMENT e CLEARING, todos os membros da rede devem usar INET para realizar o SETTLEMENT.

ISSUER

Um estabelecimento que fornece cartão plástico é ISSUER – o número do cartão é chamado de PAN.

SEMÁFOROS

refere a uma função do software que permite a execução de processamento crítico permitindo apenas um processo específico ter acesso a memória ou disco para efeito de leitura e escrita.

PAN

Primary Accont Number identifica apenas a pessoa a quem pertence o cartão. PANs são formados de 13 a 19 dígitos os primeiros 6 a 10 dígitos identificam a instituição financeira e os dígitos restantes a pessoa da instituição.

PIN

Personal Identification Number é um código apenas de 4 a 12 dígitos conhecido apenas pelo *cardholder*. É usado para verificar se a pessoa que utiliza o cartão é autorizada.

POS

Point of Sale, dispositivo que gera transação EFT no momento de compra de serviços ou bens.

Quando um cliente paga com cartão de credito para uma loja, o lojista (*merchant*) envia os dados do cartão do cliente pelo terminal POS.

SET (SECURE ELETRONIC TRANSACTIONS)

A Mastercard, a Visa e outros fornecedores de tecnologia estão cada vez desenvolvendo métodos simples que consumidores e vendedores podem utilizar para conduzir transações na Internet de forma segura e simples como acontece hoje no comércio tradicional.

Esses métodos permitem que clientes realizem compras e paguem com seus cartões de crédito, sem que o vendedor tenha acesso aos dados dos cartões.A entidade financeira (normalmente o issuer do cartão) irá validar os dados do cartão e transferir os recursos diretamente para o merchant.

SETTLEMENT

É definido como a transferência de fundos para completar uma ou mais transações já feitas, sujeitas à contabilidade final. O objetivo é gerar um extrato e distribuir os dados entre os membros, para definir o valor de cada membro nas transações diárias realizadas pelo banco de *clearing* onde o membro tem uma conta com fundos para pagar débitos calculados pelo sistema de SETTLEMENT.

Apêndice C

Termos Técnicos usados no mundo de negócios

Internet

BRICKS-AND-MORTAR (TIJOLOS E CIMENTOS)
Jargão usado para designar negócios que existem no mundo real que é o oposto do mundo puramente virtual da internet. Alguns exemplos de empresas reais são as lojas de varejo e empresas de advocacia, que realizam negócios pelos métodos tradicionais de comunicação e tem presença física.

CLICKS-AND-MORTAR (CLIENTES E CIMENTO)
Jargão usado para descrever uma empresa que integrou sua existência no mundo real ao mundo virtual da Internet, através de e-commerce ou Web Services.

Uma loja online de eletrodomésticos que permite aos clientes agendarem em seu Web site visitas para conserto é um exemplo de empresa clicks-and-mortar.

Outro exemplo é uma rede de livrarias que vende pela Web e permite buscar a mercadoria nos pontos de vendas.

CTI (COMPUTER TELEPHONY INTEGRATION)
Permite que os computadores realizem funções de controle de telefonia, como fazer e receber ligações de voz, fax e dados, bem como identificar a ligação (caller identification). Algumas das funções básicas de aplicações baseadas em CTI são: fazer e receber chamadas, consultas, transferências e conferência; associação de chamadas com dados – provisão de informações de quem liga, a partir de bancos de dados e outras aplicações de forma automática antes da resposta ou transferência da ligação.

CUSTOMER RELATIONSHIP MANAGEMENT (CRM)

CRM é o mesmo que one-to-one marketing.Esse modelo de negócios centrado no cliente também é identificado pelos nomes marketing de relacionamento e marketing em tempo real. A idéia é estabelecer relacionamentos com clientes de forma individual e depois usar as informações coletadas para tratar clientes diferentes de maneira diferente. O intercâmbio entre um cliente e a empresa torna-se mutuamente proveitoso, uma vez que os clientes oferecem informações em retribuição aos serviços personalizados que atendem às suas necessidades individuais.

DATA MINING

Analisar informações em um banco de dados usando ferramentas que procuram tendências ou anomalias sem o conhecimento do significado dos dados. O data mining é fundamental em estratégias de CRM, especialmente no comércio eletrônico (e-commerce).

DESVIO DE VALOR

O grau em que o valor de uma base de cliente está concentrado em uma pequena porcentagem dos clientes. Um desvio de valor acentuado seria o caso em que uma minúscula porcentagem de clientes é responsável pela maior parte do valor da carteira de clientes .Um desvio de valor leve seria o caso em que o valor dos clientes é distribuído mais uniformemente ao longo de toda a carteira de clientes

E-BUSINESS (NEGÓCIO ELETRÔNICO)

Termo que é mais freqüente aplicado aos negócios resultantes do uso da tecnologia digital e da Internet como principal meio de comunicação e interação.

E-COMMERCE (COMÉRCIO ELETRÔNICO)

e-commerce refere-se a usar a Internet, comunicações digitais e aplicativos de Tecnologia da Informação (IT) para possibilitar o processo de compra ou venda.

Alguns especialistas definem e-commerce como todas as etapas que ocorrem em qualquer ciclo de negócios usando a tecnologia acima descrita. Outros, como compras feitas por

consumidores e empresas pela Internet. Uma outra definição engloba as transações de suporte a IT, como a venda de código de computador por programadores que ocorre on-line.

EDI (ELETRONIC DATA INTERCHANGE)

É a transmissão de dados de negócios entre empresas, de computador, em formato eletrônico. Para os puristas, o EDI é composto somente de dados de negócio (sem mensagens em formato livre ou verbais) com formato padronizado, padrão este aprovado por organizações nacionais ou internacionais.

ERP (ENTERPRISE RESOURCE PLANNING)

ERP é o termo que descreve uma série de atividades de gestão empresarial suportadas por aplicações de IT. Estas são compostas de muitos módulos, incluindo planejamento de produto, compras, estoque, relacionamento com fornecedores, atendimento ao cliente e acompanhamento de pedidos. Em seu uso corrente, o termo ERP engloba também módulo das áreas financeiras e de recursos humanos . Normalmente um ERP é integrado a um banco de dados, e a implantação de um sistema de ERP envolve uma profunda análise do negócio da empresa, treinamento de funcionários e modificações ou criação de procedimentos.

LATÊNCIA ZERO

Termo de computação que descrevem sistema de informação no qual pouco ou nenhum tempo decorre entre a atualização de um registro de informação e sua disponibilidade em qualquer lugar do sistema.Veja também Tempo Real.

RETORNO DE INVESTIMENTO (ROI- RETURN ON INVESTMENT)

Termo que descreve o cálculo do retorno financeiro em uma política ou iniciativa de negócio que implica algum custo. O ROI pode ser medido em termos de um período para a recuperação do investimento, como uma porcentagem de retorno em uma despesa de caixa, ou como valor presente líquido descontado dos fluxos de caixa livres de um investimento. Há muitas maneiras diferentes de calcular o ROI.

SISTEMA LEGADO

Um sistema de computadores ou programa aplicativo mais antigo ou desatualizado que continua a ser usado devido ao custo exorbitante de substituí-lo ou de elaborá-lo novamente.

Quase sempre, tais sistemas oferecem pouca competitividade e compatibilidade com seus equivalentes modernos.Os sistemas legados freqüentemente são grandes, monolíticos e difíceis de modificar e sucatear um sistema legado, em geral, exige também a reengenharia dos processos de negócios de uma empresa.

TCP/IP (TRANSMISSION CONTROL PROTOCOL/INTERNET PROTOCOL)

Um conjunto de protocolos que permite o compartilhamento de aplicações entre computadores heterogêneos (pessoais, servidores e estações de trabalho) em uma rede de comunicação. Devido a padronização dos produtos em todas as camadas (incluindo os que fornecem emulação de terminal e transferência de arquivos) diversos dispositivos rodando TCP/IP podem se comunicar e cooperar em uma mesma rede física.

TECNOLOGIA DA INFORMAÇÃO (IT - INFORMATION TECHNOLOGY)

É o termo que engloba toda a tecnologia utilizada para criar, armazenar, trocar e usar informação em seus diversos formatos (dados corporativos, áudios, imagens, vídeo, apresentações multimídia e outros meios, incluindo os que não foram criados ainda). É um termo conveniente para incluir a tecnologia de computadores e telecomunicações na mesma palavra. Essa convergência está conduzindo a "revolução da informação".

TEMPO REAL

Refere-se ao nível máximo de prontidão relativa à transmissão, processamento e/ou uso de informações. Uma empresa que coleta e usa dados dos clientes em tempo real pode gerenciar os relacionamentos com clientes individuais com muito mais eficiência.

STICKY APPLICATION

Uma parte de um Web site desenvolvida para interagir com os clientes, que exige deles colaboração e que fica mais "inteligente" com o passar do tempo de relacionamento, de modo a atender às necessidades individuais do cliente. O aplicativo torna-se "adesivo" à medida que o cliente torna-se parte dele.