**UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE CIÊNCIAS MATEMÁTICAS DE SÃO CARLOS**

**Utilização de Sistemas OLTP no
Gerenciamento de Ambientes de
Manufatura Integrada
voltados à Produção Discreta**

Wellington José Brigante

Orientador : Prof. Dr. Edson dos Santos Moreira

**Dissertação apresentada ao Instituto de Ciências Matemáticas
de São Carlos, da Universidade de São Paulo, como parte dos
requisitos para obtenção do título de Mestre na área de
Ciências de Computação e Matemática Computacional.**

**São Carlos - SP
1995**

*“Um ideal custa uma vida,
mas vale uma eternidade”*

DEDICATÓRIA

*Dedico este trabalho
em memória de meus avós,
aos meus pais, às minhas irmãs,
e em especial, à minha esposa Silvana.*

AGRADECIMENTOS

Expresso os agradecimentos ao meu orientador Prof. Dr. Edson dos Santos Moreira, pelo mérito de sua orientação, dedicação, compreensão e confiança.

Ao Projeto CIM, do Departamento de Engenharia Mecânica, da Escola de Engenharia de São Carlos, da Universidade de São Paulo, na pessoa de seu coordenador, Prof. Dr. Henrique Rozenfeld, pela oportunidade, condições proporcionadas, e sugestões apresentadas.

À Unidade de Negócios Software, da Itautec Informática S.A., por disponibilizar suas ferramentas de desenvolvimento e autorizar a utilização do núcleo do seu monitor OLTP.

Ao Instituto de Ciências Matemáticas de São Carlos, da Universidade de São Paulo, pela oportunidade em seu programa de mestrado, possibilitando o desenvolvimento deste trabalho.

À minha esposa Silvana, pela motivação, incentivo e apoio oferecidos durante todos os momentos deste programa de mestrado.

À todas as pessoas que de alguma forma contribuíram para a realização deste trabalho.

ÍNDICE

Lista de Figuras

Lista de Tabelas

Lista de Telas

Lista de Abreviaturas

Resumo

Abstract

1. Introdução	1
2. Integração da Manufatura	3
2.1. Produção, Sistemas Produtivos e Empresas	4
2.2. Processo de Produção Discreta	7
2.3. Automação da Manufatura	9
2.4. Manufatura Integrada por Computador - CIM	11
3. Sistemas Computacionais Distribuídos	15
3.1. Sistemas Operacionais	15
3.2. Sistemas Distribuídos	17
3.2.1. Sistemas Cliente-Servidor	19
3.2.2. Processamento Transacional 'On-Line' - OLTP	22
3.2.3. Monitores OLTP	25
3.3. Modelo Hierárquico de Interconexão - RM-OSI	26
3.4. Arquitetura IEEE 802	31
3.5. Redes Locais de Computadores	32
3.5.1. Topologias	33
3.5.2. Suportes de Transmissão	35
3.5.3. Mecanismos de Controle de Acesso ao Meio	36
3.5.4. Padrões de Comunicação	40
3.5.5. Redes Industriais	42
3.6. Protocolos de Comunicação	42
3.6.1. Protocolo IPX/SPX	44
3.6.2. Protocolo TCP/IP	46
3.6.3. Protocolo MMS	48

4. Especificação do Sistema OLTP Proposto.....	49
4.1. Estrutura Funcional do Monitor OLTP	52
4.2. Estrutura Funcional da Aplicação Cliente.....	57
5. Apresentação do Sistema OLTP.....	63
5.1. Unidade Remota de Tomada de Decisões.....	69
5.2. Unidade Remota de Projeto - CAD.....	74
5.3. Unidade Remota de Processo - CAPP.....	76
5.4. Unidade Remota de Planejamento - PCP.....	78
5.5. Unidade Remota de Produção - CAM.....	80
6. Conclusões e Sugestões para Trabalhos Futuros.....	85
7. Bibliografia	88
Apêndice A. Ambiente Novell NetWare	92
A.1. Ambiente 'Multithreading'	93
A.2. Desenvolvimento de NLMs.....	94
A.3. Desenvolvimento de Aplicações Cliente-Servidor.....	95
A.4. Características do NetWare.....	96
Apêndice B. Descrição dos Arquivos	100
Apêndice C. Roteiro de Apresentação.....	105

LISTA DE FIGURAS

Fig. 2.01 - Arquitetura de um ambiente de manufatura integrada.....	4
Fig. 2.02 - Processo de produção contínuo	5
Fig. 2.03 - Processo de produção discreto	5
Fig. 2.04 - Processo de produção por projeto	6
Fig. 2.05 - Pirâmide da produtividade	12
Fig. 3.01 - Modelo minicomputador.....	17
Fig. 3.02 - Modelo estação	18
Fig. 3.03 - Modelo banco de processadores	18
Fig. 3.04 - Sistema cliente-servidor.....	19
Fig. 3.05 - Comunicação assíncrona.....	20
Fig. 3.06 - Comunicação síncrona.....	20
Fig. 3.07 - Mecanismo RPC.....	20
Fig. 3.08 - Modelo de acesso remoto a dados	21
Fig. 3.09 - Modelo servidor de banco de dados.....	22
Fig. 3.10 - Modelo servidor de aplicações.....	22
Fig. 3.11 - Arquitetura de um monitor OLTP.....	26
Fig. 3.12 - Arquitetura RM-OSI	27
Fig. 3.13 - Camadas da arquitetura RM-OSI.....	28
Fig. 3.14 - Arquitetura IEEE 802.....	31
Fig. 3.15 - Topologia em malha irregular	33
Fig. 3.16 - Topologia em estrela	34
Fig. 3.17 - Topologia em árvore	34
Fig. 3.18 - Topologia em anel.....	34
Fig. 3.19 - Topologia em barramento.....	35
Fig. 3.20 - Técnicas de multiplexação	37
Fig. 3.21 - Ethernet	40
Fig. 3.22 - Token Bus.....	40
Fig. 3.23 - Token Ring	41
Fig. 3.24 - FDDI.....	41
Fig. 3.25 - ARCnet	41
Fig. 3.26 - Protocolo de comunicação.....	43
Fig. 3.27 - Protocolo IPX/SPX.....	44
Fig. 3.28 - Formato de um pacote IPX.....	45
Fig. 3.29 - Formato de um pacote SPX.....	46
Fig. 3.30 - Protocolo TCP/IP.....	46
Fig. 3.31 - Formato de um pacote IP	47
Fig. 3.32 - Formato de um pacote TCP.....	47
Fig. 3.33 - Formato de um pacote UDP	47
Fig. 3.34 - Padronizações MMS	48
Fig. 4.01 - Estrutura funcional do sistema OLTP	51
Fig. 4.02 - Estrutura funcional do monitor OLTP.....	54
Fig. 4.03 - Estrutura funcional da aplicação cliente	58
Fig. A.01 - Arquitetura modular Novell NetWare.....	92

LISTA DE TABELAS

<i>Tabela 4.01 - Monitores OLTP existentes.....</i>	<i>49</i>
<i>Tabela 4.02 - Descrição das transações utilizadas</i>	<i>62</i>
<i>Tabela 5.01 - Descrição dos parâmetros de ativação do monitor OLTP</i>	<i>65</i>
<i>Tabela 5.02 - Descrição dos parâmetros de ativação do API IPX/SPX cliente.....</i>	<i>65</i>
<i>Tabela 5.03 - Descrição dos parâmetros de ativação da interf. cli. de acesso</i>	<i>65</i>
<i>Tabela 5.04 - Sessões clientes definidas</i>	<i>67</i>
<i>Tabela 5.05 - Pedidos definidos</i>	<i>71</i>
<i>Tabela 6.01 - 16 usuários / 1 transação por segundo (1 monitor OLTP).....</i>	<i>86</i>
<i>Tabela 6.02 - 64 usuários / 4 transações por segundo (2 monitores OLTP).....</i>	<i>86</i>
<i>Tabela 6.03 - 128 usuários / 10 transações por segundo (2 monitores OLTP)</i>	<i>86</i>
<i>Tabela B.01 - Arquivos utilizados.....</i>	<i>100</i>
<i>Tabela B.02 - Estrutura das tabelas de aplicativos.....</i>	<i>100</i>
<i>Tabela B.03 - Estrutura da tabela de comandos numéricos.....</i>	<i>100</i>
<i>Tabela B.04 - Estrutura da tabela de desenhos</i>	<i>100</i>
<i>Tabela B.05 - Estrutura da tabela de estado das sessões clientes</i>	<i>100</i>
<i>Tabela B.06 - Estrutura da tabela de falhas das unidades produtivas.....</i>	<i>101</i>
<i>Tabela B.07 - Estrutura da tabela de famílias</i>	<i>101</i>
<i>Tabela B.08 - Estrutura da tabela de folhas de processo.....</i>	<i>101</i>
<i>Tabela B.09 - Estrutura da tabela de monitores OLTP</i>	<i>101</i>
<i>Tabela B.10 - Estrutura da tabela de ordens de fabricação.....</i>	<i>101</i>
<i>Tabela B.11 - Estrutura da tabela de pedidos</i>	<i>102</i>
<i>Tabela B.12 - Estrutura da tabela de produtos</i>	<i>102</i>
<i>Tabela B.13 - Estrutura da tabela de sessões clientes</i>	<i>103</i>
<i>Tabela B.14 - Estrutura da tabela de unidades produtivas.....</i>	<i>103</i>
<i>Tabela B.15 - Estrutura da tabela de unidades remotas.....</i>	<i>103</i>
<i>Tabela B.16 - Relacionamento entre os arquivos.....</i>	<i>104</i>

LISTA DE TELAS

Tela 5.01 - Gerenciador de tabelas.....	63
Tela 5.02 - Tabela de transações e aplicações.....	63
Tela 5.03 - Tabela de arquivos.....	64
Tela 5.04 - Tabela de áreas permanentes.....	64
Tela 5.05 - Tabela de localidades remotas.....	64
Tela 5.06 - Ativação do monitor OLTP.....	64
Tela 5.07 - Ativação da aplicação cliente	66
Tela 5.08 - Definições.....	66
Tela 5.09 - Tabela de monitores OLTP	66
Tela 5.10 - Tabela de unidades remotas.....	66
Tela 5.11 - Tabela de sessões clientes	67
Tela 5.12 - Estabelecimento das conexões.....	67
Tela 5.13 - Conexão das sessões clientes.....	67
Tela 5.14 - Sessão cliente conectada	67
Tela 5.15 - Gerenciamento das sessões clientes.....	68
Tela 5.16 - Desconexão das sessões clientes.....	68
Tela 5.17 - Sessão cliente desconectada	68
Tela 5.18 - Liberação das sessões clientes.....	68
Tela 5.19 - Desativação do monitor OLTP.....	68
Tela 5.20 - Uni. rem. de tomada de decisões	69
Tela 5.21 - Definições / Decisões.....	69
Tela 5.22 - Tabela de famílias	70
Tela 5.23 - Tabela de produtos	70
Tela 5.24 - Tabela de unidades produtivas.....	70
Tela 5.25 - Tabela de falhas das uni. produtivas	70
Tela 5.26 - Tabela de pedidos	71
Tela 5.27 - Monitoração / Decisões	71
Tela 5.28 - Estados das sessões clientes.....	71
Tela 5.29 - Estados dos pedidos concluídos	71
Tela 5.30 - Estados dos pedidos pendentes	72
Tela 5.31 - Estados das unidades produtivas	72
Tela 5.32 - Liberação dos pedidos / Decisões.....	72
Tela 5.33 - Escolhas na lib. dos ped. / Decisões.....	72
Tela 5.34 - Pedido liberado	73
Tela 5.35 - Envio de mensagens.....	73
Tela 5.36 - Elaboração da mensagem.....	73
Tela 5.37 - Mensagem enviada.....	73
Tela 5.38 - Recepção de mensagens.....	74
Tela 5.39 - Desconexão / Decisões	74
Tela 5.40 - Uni. rem. de projeto - CAD.....	74
Tela 5.41 - Definições / Projeto	74
Tela 5.42 - Tabela de aplicativos / Projeto.....	75
Tela 5.43 - Elaboração / Projeto	75
Tela 5.44 - Escolhas na elaboração do desenho	75
Tela 5.45 - Aplicativo para elab. do desenho	75

<i>Tela 5.46 - Liberação dos pedidos / Projeto</i>	<i>76</i>
<i>Tela 5.47 - Desconexão / Projeto</i>	<i>76</i>
<i>Tela 5.48 - Uni. rem. de processo - CAPP.....</i>	<i>76</i>
<i>Tela 5.49 - Definições / Processo</i>	<i>76</i>
<i>Tela 5.50 - Elaboração / Processo</i>	<i>77</i>
<i>Tela 5.51 - Liberação dos pedidos / Processo.....</i>	<i>77</i>
<i>Tela 5.52 - Desconexão / Processo</i>	<i>77</i>
<i>Tela 5.53 - Uni. rem. de planejamento - PCP.....</i>	<i>78</i>
<i>Tela 5.54 - Definições / Planejamento</i>	<i>78</i>
<i>Tela 5.55 - Monitoração / Planejamento.....</i>	<i>78</i>
<i>Tela 5.56 - Elaboração / Planejamento</i>	<i>78</i>
<i>Tela 5.57 - Controle da produção</i>	<i>79</i>
<i>Tela 5.58 - Alteração da situação da produção.....</i>	<i>79</i>
<i>Tela 5.59 - Situação da produção alterada.....</i>	<i>79</i>
<i>Tela 5.60 - Planejamento da produção</i>	<i>79</i>
<i>Tela 5.61 - Pedido incluído na produção</i>	<i>80</i>
<i>Tela 5.62 - Liberação dos ped. / Planejamento</i>	<i>80</i>
<i>Tela 5.63 - Desconexão / Planejamento.....</i>	<i>80</i>
<i>Tela 5.64 - Uni. rem. de produção - CAM.....</i>	<i>81</i>
<i>Tela 5.65 - Monitoração / Produção</i>	<i>81</i>
<i>Tela 5.66 - Processo de produção</i>	<i>81</i>
<i>Tela 5.67 - Escolhas no processo de produção.....</i>	<i>81</i>
<i>Tela 5.68 - Início da simulação da produção</i>	<i>82</i>
<i>Tela 5.69 - Falha na simulação</i>	<i>82</i>
<i>Tela 5.70 - Interrupção da simulação</i>	<i>82</i>
<i>Tela 5.71 - Simulação interrompida.....</i>	<i>82</i>
<i>Tela 5.72 - Simulação liberada</i>	<i>82</i>
<i>Tela 5.73 - Prosseguimento da simulação.....</i>	<i>82</i>
<i>Tela 5.74 - Conclusão da simulação</i>	<i>83</i>
<i>Tela 5.75 - Conclusão do proc. de produção</i>	<i>83</i>
<i>Tela 5.76 - Liberação dos pedidos / Produção.....</i>	<i>83</i>
<i>Tela 5.77 - Escolhas na lib. dos ped. / Produção.....</i>	<i>83</i>
<i>Tela 5.78 - Pedido liberado e concluído</i>	<i>84</i>
<i>Tela 5.79 - Desconexão / Produção</i>	<i>84</i>
<i>Tela B.01 - Relacionamento entre os arquivos.....</i>	<i>104</i>

LISTA DE ABREVIATURAS

ACID : Atomicidade, consistência, isolamento e durabilidade.

ANSI : American National Standards Institute.

API : Application Program Interface / Interface para programa de aplicação.

ARCnet : Attached Resource Computer Network.

ATDM : Asynchronous Time Division Multiplexing / Multiplexação assíncrona por divisão no tempo

AGV : Automated Guided Vehicle / Veículo guiado automaticamente.

CAD : Computer Aided Design / Projeto assistido por computador.

CAE : Computer Aided Engineering / Engenharia assistida por computador.

CAM : Computer Aided Manufacturing / Manufatura assistida por computador.

CAPP : Computer Aided Process Planning / Planejamento do processo assistido por computador.

CIM : Computer Integrated Manufacturing / Manufatura integrada por computador.

CNC : Computer Numerical Control / Controle numérico por computador.

CLP : Controlador lógico programável.

CSMA/CD : Carrier-Sense Multiple Access with Collision Detection / Acesso múltiplo com detecção de portadora e detecção de colisão.

DARPA : Defense Advanced Research Projects Agency.

DNC : Direct Numerical Control / Controle numérico direto.

ECB : Event Control Block.

FDDI : Fiber Distributed Data Interface.

FDM : Frequency Division Multiplexing / Multiplexação por divisão em frequência.

FIFO : First In First Out.

FTP : File Transfer Protocol.

IEEE : Institute of Electrical and Eletronic Engineers.

IMP : Interface Message Processor.

IP : Internet Protocol.

IPC : Inter Process Communication / Comunicação entre processos.

IPX : Internetwork Packet Exchange.

ISA : Instruments Society of America.

ISO : International Standards Organization.

JIT : Just-in-Time.

LAN : Local Area Network / Rede local de computadores.

LLC : Logical Link Control / Sub-camada de controle de enlace lógico.

MAC : Medium Access Control / Sub-camada de controle de acesso ao meio.

MAP : Manufacturing Automation Protocol.

MMS : Manufacturing Message Specification.

MRP : Material Requirements Planning / Planejamento das necessidades de materiais.

NFS : Network File System.

NLM : NetWare Loadable Module.

OLTP : On-Line Transaction Processing / Processamento transacional 'on-line'.

OSI : Open Systems Interconnection.

PCP : Planejamento e controle da produção.

QMS : Queue Management System / Sistema de gerenciamento de filas.

RFC : Remote File Sharing.

RIM : Resource Interface Module.

RM-OSI : Reference Model for Open Systems Interconnection.

RPC : Remote Procedure Call / Chamada remota a procedimentos.

SAP : Service Access Point / Ponto de acesso de serviços.

SC16 : Subcomitê 16 da ISO.

SMTP : Simple Mail Transfer Protocol.

SQL : Structured Query Language.

SPX : Sequenced Packet Exchange.

STDM : Synchronous Time Division Multiplexing / Multiplexação síncrona por divisão no tempo.

TCP : Transmission Control Protocol

TDM : Time Division Multiplexing / Multiplexação por divisão no tempo.

TLI : Transport Layer Interface / Interface a nível de transporte.

TOP : Technical Office Protocol.

UDP : User Datagram Protocol.

VMD : Virtual Manufacturing Device / Dispositivo virtual de manufatura.

WAN : Wide Area Network / Rede de computadores de longa distância.

RESUMO

Este projeto propõe a utilização de sistemas OLTP (Processamento Transacional 'On-Line') no gerenciamento do fluxo de informações entre os segmentos de um ambiente de manufatura integrada voltado à produção discreta. Para tanto, são analisados os aspectos relacionados com o desenvolvimento de aplicações cliente-servidor para sistemas transacionais utilizando-se monitores OLTP. Como plataforma base para o sistema escolheu-se o ambiente Novell NetWare, por este oferecer uma rede de processamento distribuído 'multithreading' aberta e flexível. O núcleo do sistema (monitor OLTP) reside, juntamente com as aplicações servidoras, no servidor Novell, sendo acessado por todas as estações clientes da rede através de uma arquitetura cliente-servidor. Será mostrado que a utilização de sistemas OLTP no gerenciamento de ambientes de manufatura integrada voltados à produção discreta, disponibiliza mecanismos para alta performance e disponibilidade do sistema em situações críticas, mantendo a segurança e integridade dos dados, através do compartilhamento e utilização eficiente dos recursos computacionais oferecidos.

ABSTRACT

The idea behind this project is the use of OLTP (On-Line Transactional Processing) systems in the management of information flow between the segments of the integrated manufacturing environment designed for the discrete production. The features related to the development of client-server applications for transactional systems using OLTP monitors are reviewed. The Novell's NetWare environment has been selected as the base platform, since it provides a "multithreading" distributed network which allows for an open and flexible environment. The system kernel (OLTP monitor) resides, along with the server applications, in the Novell's server, being that accessed by all clients stations through a client-server architecture. It is shown that the use of OLTP systems in the management of integrated manufacturing environments designed for the discrete production makes possible the implementation of mechanisms to allow for high performance and availability in critical situations, keeping security and integrity of data through the sharing and the efficient use of the computing resources available.

1. INTRODUÇÃO

A modernização industrial dos ambientes de manufatura, para alcançar a produtividade, a competitividade, a qualidade, a redução de custos, para assimilar as constantes variações nas características dos produtos, a utilização racional da energia e matéria-prima, entre outros, caracteriza-se pela introdução do computador em praticamente todos os seus segmentos, desde o nível de projeto ao nível de produção.

Esta introdução do computador resulta no surgimento de um conjunto de atividades adicionais, cada qual, com o objetivo de buscar a máxima eficiência do seu segmento no ambiente de manufatura. Tais atividades correspondem à implantação de processos automatizados, entre eles :

- *Projeto Assistido por Computador - CAD;*
- *Engenharia Assistida por Computador - CAE;*
- *Planejamento do Processo Assistido por Computador - CAPP;*
- *Manufatura Assistida por Computador - CAM.*

A simples implantação destes processos automatizados favorece o surgimento das chamadas "ilhas isoladas de automação", não destacando, por si só, um dos principais componentes do custo operacional de um ambiente de manufatura, o gerenciamento do fluxo de informações entre os segmentos.

A consideração de todos os segmentos de um ambiente de manufatura, integrados como um conjunto harmonioso, é designada por *Manufatura Integrada por Computador - CIM*, que é vista como uma tecnologia capaz de viabilizar a cooperação e integração entre todos os segmentos, garantindo, a pronta disponibilidade de informações precisas a qualquer segmento que as tenha solicitado.

Como condição primeira e necessária à implantação de um ambiente de manufatura integrada, tem-se uma série de medidas organizacionais que visam a uma completa reestruturação do ambiente, envolvendo os sistemas produtivos, as atividades de negócios e as pessoas, a fim de possibilitar a integração destes diferentes elementos.

Satisfeita a condição anterior, uma próxima condição para a integração total do ambiente de manufatura se refere à maneira pela qual as informações serão trocadas. A utilização de redes locais de computadores, como ferramentas de integração, vem de encontro a esta condição. Uma rede local, deve prover serviços que assegurem, além da interconexão e compartilhamento de recursos, um desempenho compatível e uma confiável transmissão e recepção das informações, de tal forma que os procedimentos necessários para a obtenção de tais objetivos sejam transparentes ao ambiente.

No caso de ambientes de manufatura integrada voltados à produção discreta, a troca de informações ocorre de maneira intensa e variável devido ao grande número de operações simultâneas efetuadas pelos seus segmentos, ocasionando um aumento do tráfego das informações na rede local e, conseqüentemente, uma queda de performance em todo o sistema, não garantindo a pronta disponibilidade das informações.

A utilização de sistemas para *Processamento Transacional 'On-Line' - OLTP* tem o objetivo de garantir a performance de todo o sistema juntamente com a integridade e a pronta disponibilidade das informações pois, baseados em sistemas distribuídos cliente-servidor, fornecem suporte a um grande número de operações simultâneas, permitindo compartilhamento de recursos, dados e aplicações, além de alta disponibilidade para situações críticas.

Logo, diante destes aspectos, o presente projeto se propõe a levantar os principais aspectos relacionados aos ambientes de manufatura e ao desenvolvimento de aplicações em sistemas distribuídos cliente-servidor, propondo a *utilização de um sistema OLTP no gerenciamento do fluxo de informações entre os segmentos de um ambiente de manufatura integrada voltado à produção discreta*.

Esta dissertação está organizada em sete capítulos, onde o primeiro corresponde a esta breve introdução, o último apresenta a bibliografia utilizada e os demais apresentam os conceitos envolvidos juntamente com os elementos integrantes do sistema proposto. Estes demais capítulos são descritos abaixo.

O segundo capítulo apresenta uma visão geral sobre integração da manufatura, considerando as principais características de um sistema produtivo, do processo de produção discreta, do processo de automação da manufatura e da manufatura integrada por computador.

O terceiro capítulo envolve os fundamentos para sistema computacionais distribuídos, abordando sistemas operacionais, sistemas distribuídos cliente-servidor, processamento transacional 'on-line', monitores OLTP, modelo RM-OSI, arquitetura IEEE 802, redes locais de computadores e protocolos de comunicação.

O quarto capítulo descreve a especificação do sistema OLTP proposto, expondo os motivos que levaram à escolha do ambiente Novell NetWare como plataforma base para o sistema, juntamente com as estruturas funcionais do monitor OLTP, das aplicações servidoras e da aplicação cliente.

O quinto capítulo apresenta o sistema OLTP implementado, demonstrando a viabilidade e as vantagens do sistema, através do acompanhamento de um processo de elaboração, execução e conclusão de um determinado pedido em um ambiente de manufatura integrada voltado à produção discreta.

O sexto capítulo apresenta as conclusões sobre o sistema OLTP proposto e sugestões para trabalhos futuros.

2. INTEGRAÇÃO DA MANUFATURA

As atividades produtivas da humanidade evoluíram lentamente até a metade do século XVIII quando se iniciou na Inglaterra a época industrial. Estas idéias propagaram-se em tempos diferentes ao restante da Europa Ocidental e à América. O trabalho braçal, preponderante até então, foi sendo substituído pelo trabalho das máquinas (mecanização), uma vez que fontes de energia mais poderosas tornaram-se disponíveis para a realização dos processos de transformação da matéria-prima.

Depois das máquinas a vapor, foram os motores a combustão interna e a energia elétrica, que deram origem a novos desenvolvimentos da manufatura. A mecanização oriunda desta época teve como primeira consequência o aumento da produtividade. Apesar dos progressos na produtividade, a parcela da população em atividades de manufatura começou a crescer somente a partir do século XIX com a consequente diminuição da parcela dedicada à agricultura, ocasionando uma profunda modificação nos parâmetros sociais e econômicos.

Verificou-se o surgimento de novas classes e a nítida separação dos fatores de produção capital e trabalho. O controle de uma empresa de manufatura deixou de ser puramente um mecanismo relacionado com os fatores internos, passando a considerar também as forças políticas e econômicas. Desenvolveu-se assim, o conceito de sucesso financeiro, as técnicas de contabilidade, o mercado de capitais e as teorias econômicas e sócio-políticas.

Um crescimento acentuado da produtividade foi alcançado através das primeiras tecnologias de automação (tecnologia destinada à substituição ou auxílio do esforço mental). Aos primeiros reguladores mecânicos desenvolvidos por James Watt, em 1788, seguiram-se a instrumentação e os reguladores pneumáticos e hidráulicos, e após as guerras mundiais, os eletrônicos e os com tecnologia analógica.

A evolução nas últimas três décadas foi rápida com a introdução da tecnologia digital. Embora a construção de máquinas simples e a mecanização se iniciassem em 1770, somente no princípio deste século surgiram os mecanismos automáticos fixos e as linhas de montagem para a produção em massa. Em seguida foram desenvolvidas máquinas-ferramenta com controle automático simples, para executarem sequências fixas de operações. A introdução do controle ou comando numérico no final da década de 50, abriu uma nova etapa na automação.

O progresso tecnológico, até a metade deste século, foi decorrente da concentração tecnológica nos campos da energia e dos materiais. As novas tecnologias de informação vieram provocar, pouco a pouco, uma nova revolução a partir de 1950. A introdução do controle numérico em 1952 abriu uma nova era na automação, uma vez que se baseava na utilização de computadores digitais. Os computadores digitais representavam na época uma nova tecnologia, constituindo-se hoje no núcleo da tecnologia de informática.

A extensão lógica do controle numérico correspondeu ao controle numérico computadorizado para máquinas-ferramenta e máquinas operatrizes, onde o computador digital passou a ser incluído como parte integrante do sistema de controle. Robôs industriais foram desenvolvidos simultaneamente aos sistemas de controle numérico computadorizados. Porém, não tiveram um papel preponderante nos ambientes de manufatura até o final dos anos 70.

No controle de processos foi introduzida a instrumentação eletrônica, o que possibilitou a instalação de salas de controle a grandes distâncias dos processos. Com o avanço das técnicas de transmissão de dados passou-se a utilizar, no lugar de uma única sala de controle central, diversas salas de controle distribuídas geograficamente, interligadas entre si e conectadas a uma sala central. Surgiram, então, as idéias básicas de controle hierárquico, evoluindo para o que pode-se chamar de controle distribuído.

A idéia de utilizar-se computadores digitais no controle de processos foi uma decorrência imediata. Sistemas digitais distribuídos vêm sendo construídos, a comunicação com o homem (operadores, supervisores, etc.) vem sendo aprimorada e sistemas de controle distribuído com grande número de computadores vêm sendo interconectados por redes de computadores. A automação penetrou definitivamente em todos os setores produtivos de um ambiente de manufatura, representando um fator decisivo no aumento da produtividade, da qualidade dos produtos e da flexibilidade de produção (fig. 2.01).

Fig. 2.01 - Arquitetura de um ambiente de manufatura integrada

2.1. PRODUÇÃO, SISTEMAS PRODUTIVOS E EMPRESAS

Produção corresponde a um processo, através do qual, bens ou serviços são gerados por meio da transformação de recursos. Já os sistemas produtivos, ou empresas, combinam fatores como materiais, trabalho, energia e recursos de capital, através de um processo organizado, com o objetivo de produzir algum produto ou serviço. Assim, as empresas são identificadas como empresas de manufatura (indústrias), empresas financeiras (bancos), empresas de negócios (escritórios), etc. [Rese91a].

Segundo COURTOIS [Cour91a], uma empresa é caracterizada pela sua organização e particularidade de seus produtos. Contudo, apesar das peculiaridades referentes a cada uma, são classificadas em função de alguns critérios, tais como, organização do fluxo de produção e relação com o cliente.

Na classificação das empresas segundo a organização do fluxo de produção, distinguem-se três fluxos básicos de produção : produção contínua ('flow shop'), produção discreta ('job shop') e produção por projeto.

A produção contínua (fig. 2.02) trata de quantidades significativas de um produto ou de uma família de produtos. Sua implantação é realizada em linhas de produção, tornando linear o fluxo do produto. É caracterizada por :

- Muita eficácia;
- Pouca flexibilidade (linhas rígidas de produção);
- Sistemas de gestão relativamente simples;
- Produção equilibrada em cada uma das máquinas;
- Automatização intensa dos processos de produção e dos sistemas de manutenção.

Fig. 2.02 - Processo de produção contínuo

A produção discreta (fig. 2.03) trata de quantidades relativamente pequenas de numerosos e variados produtos, a partir de um conjunto diversificado de máquinas. Sua implantação é realizada pelo reagrupamento das máquinas em função do processo tecnológico que executam, direcionando o fluxo do produto em função do encadeamento dos processos a serem realizados. É caracterizada por :

- Pouca eficácia;
- Muita flexibilidade (oficinas de produção flexíveis);
- Sistemas de gestão mais complexos;
- Equilíbrio difícil das tarefas, gerando elevados níveis de estoque e custos de fabricação;
- Máquinas capazes de realizar um grande número de operações, não sendo exclusivas a um só produto.

Fig. 2.03 - Processo de produção discreto

A produção por projeto (fig. 2.04) trata de um único produto, geralmente de grande porte e complexo, resultando em um processo de produção único e não renovável. Sua implantação é realizada pelo encadeamento de todas as operações que conduzem ao termo do projeto. É considerada uma variante da produção discreta.

Fig. 2.04 - Processo de produção por projeto

Após estas análises, conclui-se que os processos contínuos são bem mais indicados que os discretos, por serem muito mais simples de se gerir. Então, como utilizar processos discretos, uma vez que são de gestão complexa? Uma alternativa seria reestruturá-los, de tal modo que operassem como processos contínuos, possuindo assim, uma gestão mais simples.

A *Tecnologia de Grupo* corresponde a uma filosofia que pode servir a este objetivo, através da organização e agrupamento dos produtos e máquinas em conjuntos com características técnicas e funcionais semelhantes entre si, chamados *famílias*. Por exemplo, em uma oficina mecânica de produção discreta, todas as máquinas que possuem características semelhantes são organizadas e agrupadas em famílias, gerando células de manufatura. Cada célula de manufatura tem a função de fabricar produtos sempre da mesma família, à qual, suas máquinas também pertencem, criando-se assim uma organização contínua entre as células de manufatura.

A vantagem desse reagrupamento consiste no ganho de eficácia, diminuindo-se os prazos e custos de fabricação. Já a desvantagem desse reagrupamento consiste na perda de flexibilidade das máquinas introduzidas nas células de manufatura. Cada solução deve implicar sempre na observação de dois itens conflitantes:

- Grande flexibilidade com organização e gestão mais complexa;
- Flexibilidade mais baixa com organização e gestão relativamente simples.

Na classificação das empresas segundo a relação com o cliente, distinguem-se dois tipos de produção:

- *Produção para estoque*: Os produtos acabados são armazenados em estoques. Precisa-se produzir antecipadamente para estoque, quando o tempo de fabricação é superior ao prazo da entrega, ou uma grande quantidade produzida ocasionar diminuição de custos;
- *Produção por encomenda*: Conduz a uma diminuição do estoque, logo, diminuição dos custos financeiros.

Toda empresa tem o interesse de produzir apenas o que já está vendido, mas para isso, é preciso que o seu prazo de produção seja inferior a um prazo aceitável de venda. Se a empresa pode ser capaz de tratar e produzir o produto dentro de um prazo aceitável, então deverá produzir por encomenda, caso contrário, deverá produzir para estoque [Cour91a].

Exemplos de atuação das empresas, mediante o fluxo e tipo de produção :

- *Fluxo contínuo e produção para estoque* : refinarias;
- *Fluxo contínuo e produção por encomenda* : linhas de montagem automotiva;
- *Fluxo discreto e produção para estoque* : componentes eletrônicos;
- *Fluxo discreto e produção por encomenda* : equipamentos industriais;
- *Fluxo por projeto e produção para estoque* : apartamentos;
- *Fluxo por projeto e produção por encomenda* : navios.

2.2. PROCESSO DE PRODUÇÃO DISCRETA

As empresas de manufatura voltadas à produção discreta necessitam de informações que definam seus produtos através de características que descrevam suas formas, quantidades, materiais e especificações [Rese91a]. Estas informações são descritas através das listas de materiais, dos desenhos, das especificações, do planejamento do processo, do planejamento e controle da produção, dos cronogramas de fabricação, dos tempos de operação, de montagem e improdutivos, das informações sobre materiais, dos planos de produção, das ordens de fabricação, da programação, da emissão de ordens, dos sistemas de controle, e do replanejamento e reprogramação.

As listas de materiais, além de identificar o produto, relacionam todas as suas partes e componentes, identificando-os por nome e número, e mostrando também a quantidade necessária destas partes e componentes para se montar uma unidade completa do produto.

Os desenhos estão entre os meios mais importantes de se introduzir informações relativas ao produto para os planejamentos de processo e controles de produção, mostrando o projeto do produto em forma gráfica e descrevendo suas medidas físicas, tolerâncias, formas e contornos.

As especificações são informações relativas ao produto, separadas do desenho, pertinentes ao trabalho de fabricação, como instruções de como realizar determinadas operações, testes necessários, procedimentos de inspeções e outras exigências que não podem ser transmitidas por um desenho.

Todas as partes fabricadas do produto devem passar por operações que mudam a forma inicial de sua matéria-prima. Para tanto, deve ser elaborado o planejamento do processo produtivo. A primeira fase sempre analisa o trabalho a ser feito e identifica as mudanças que devem ser efetuadas no estado do material. O próximo passo é dividir o trabalho a ser feito em diversas operações e decidir qual máquina será utilizada em cada operação. Finalmente, deve-se decidir a sequência ideal na qual as operações devem ser realizadas, com o devido estudo do ferramental a ser utilizado. Todo este trabalho é registrado em folhas de processo, que mostram todas as operações a serem feitas, as máquinas envolvidas, além de outras informações a respeito do ferramental.

O planejamento e controle da produção deve estar informado da indisponibilidade de máquinas e equipamentos provocada por defeitos, programas de manutenção, desativações, bem como da entrada em operação de novos recursos.

A montagem de produtos relativamente complexos só deve ter início, em princípio, assim que todas as suas partes e componentes estiverem prontos. Consequentemente, esses itens terão datas de início diferentes. Certas partes podem formar subconjuntos antes da montagem final, de maneira que todos os subconjuntos precisariam estar prontos antes de iniciar-se a montagem final. Os cronogramas de fabricação são diagramas que ilustram a programação das partes do produto, mostrando as datas de início e término das diversas fases.

Os tempos de operação são aqueles necessários para a execução de cada operação em um produto. Tempo de montagem é o tempo consumido para a colocação e retirada da máquina, das ferramentas, dispositivos e materiais. Tempos improdutivos são aqueles que são perdidos por outras causas que não as de montagem.

É de fundamental importância o acesso do planejamento e controle da produção às informações sobre materiais que definam a identificação, o estado, a política de estoque e a procedência dos materiais, além do conhecimento dos índices de eliminação por defeito dos componentes dos produtos fabricados.

Os planos de produção são programas emitidos periodicamente, aprovados pela alta administração da empresa de manufatura, a fim de autorizar a produção. Constituem-se em uma informação fundamental que autoriza a operação de todas as instalações produtivas durante um determinado período, especificando quantidades de produtos a serem montados em períodos compatíveis com a capacidade, além de incluir a fabricação de partes do produto e subconjuntos, se houver alguma forma de solicitação. Os fatores importantes na elaboração dos planos de produção são :

- Plano de vendas;
- Capacidade produtiva;
- Disponibilidade de matéria-prima no mercado;
- Recursos financeiros da empresa de manufatura.

Uma vez recebido, o pedido do cliente é adicionado ao plano de produção. A seguir, ordens de fabricação, devem explicitar os tipos e quantidades dos componentes do produto a fabricar e materiais que devem ser utilizados. Além disso, as empresas de manufatura geralmente possuem componentes prontos e matérias-primas disponíveis em estoque. Diante disso e da impossibilidade de se considerar que o estoque estará em um nível que possa atender aleatoriamente qualquer requisição, é necessário fazer um estudo para determinar a quantidade necessária de cada componente que entra na composição de cada produto, que compõem o plano de produção. Este processo converte o plano de produção em ordens de fabricação e ordens de compra com datas definidas de término e recebimento, respectivamente. Compreende algumas tarefas :

- Cálculo da quantidade líquida (política de estoque que visa abastecimento garantido);
- Aprazamento das ordens (data em que as ordens deverão estar concluídas);
- Estudo da carga (avaliação detalhada da carga de trabalho das máquinas);
- Emissão das ordens de compra de materiais;
- Emissão das ordens de fabricação.

A programação é definida como a determinação de datas nas quais iniciar e/ou completar cada operação que compõem um procedimento, e como a determinação de quando e onde cada operação necessária para a fabricação deve ser executada.

A função da elaboração do plano de produção está relacionada com o planejamento do que a empresa de manufatura deve fazer em termos de produto e tempo. Este tempo mais amplo é dividido na etapa da emissão das ordens, que representa o processo final nessa análise do plano de produção e está relacionada com o planejamento das operações em cada máquina, sendo responsável pela decisão de iniciar a produção ao nível de chão-de-fábrica, segundo um determinado roteiro e na ocasião programada.

Os sistemas de controle testam a execução do plano e programa de produção, através das informações coletadas pelo sistema de emissão, exercendo uma função de acompanhamento a nível de :

- Controle do plano de produção;
- Acompanhamento de ordens;
- Levantamento de faltas;
- Acompanhamento do plano diário;
- Controle do desempenho dos departamentos.

Causas diversas provocam o desvio da produção em relação à programação. Além de problemas que frequentemente ocorrem com operadores e máquinas causando transtornos à produção, modificações no projeto do produto, alterações nas quantidades solicitadas pelo cliente, operações necessárias e não planejadas, rejeições pelo controle de qualidade, etc., são responsáveis por consideráveis trabalhos de replanejamento e reprogramação. Replanejamento e reprogramação são tarefas de grande monta, devido à grande quantidade de informações que devem ser manuseadas em um curto intervalo de tempo, e do grau de dificuldade muito maior que o de elaboração dos planos iniciais.

2.3. AUTOMAÇÃO DA MANUFATURA

Segundo GERELLE [*Gere_a*], a manufatura de bens é definida como um sistema que integra as diferentes informações de um processo de produção que definem seus produtos e os seus diferentes estágios, necessitando para isso dados de entrada definidos para se obter resultados esperados. Já um sistema de manufatura pode ser entendido como composição dos quatro elementos básicos : engenharia, chão-de-fábrica ('shop floor'), suporte e negócios.

A engenharia é o elemento responsável pela criação e desenvolvimento dos produtos a serem fabricados, assim como o desenvolvimento dos meios de produção necessários (processos de fabricação, ferramental, equipamentos, etc.).

O chão-de-fábrica corresponde às instalações de manufatura (máquinas, equipamentos, etc.) responsáveis por fabricar os produtos nos prazos e quantidades determinados. Os recursos disponíveis, além das máquinas e equipamentos, correspondem a mão-de-obra direta (operadores) e indireta (suporte diretamente relacionado à produção).

O suporte é responsável por manter o desempenho e características, tanto de qualidade, quanto dos equipamentos. Pode ser subdividido em suporte à qualidade (provê meios para manter a qualidade dos produtos controlada e estável) e suporte à operação (provê meios para manter os equipamentos e instalações do chão-de-fábrica em condições operacionais adequadas, através de um processo de manutenção, tanto dos equipamentos, quanto das instalações).

Os negócios devem ser a interface do sistema de manufatura com o mundo exterior, tanto do mercado consumidor (clientes), como do mercado supridor (fornecedores). Compreende as atividades de marketing, suprimentos, e planejamento e controle da produção.

Nos ambientes modernos de manufatura, os sistema de manufatura são bastante complexos, envolvendo uma série de atividades dependentes e independentes entre si. A sua complexidade será tanto maior quanto mais complexos forem os produtos a serem fabricados e a estrutura de produção vinculada a esses produtos. Assim, algumas características principais determinam a sua menor ou maior complexidade :

- Complexidade do produto a ser fabricado (folhas de processo com menor ou maior número de operações);
- Diversificação de produtos a serem fabricados (muitas folhas de processo envolvem um maior número de máquinas, ferramentas e controles);
- Variação das quantidades dos produtos fabricados (situações de variação na carga das máquinas provocam sobrecarga e ociosidade, gerando como consequência, maior dificuldade de planejamento).

A grande complexidade de um sistema de manufatura resulta numa grande complexidade de coordenação, exigindo um grande número de pessoas envolvidas, bem como, uma enorme burocracia do registro manual de informações geradas em cada elemento do ambiente de manufatura, o que implica na necessidade do processo de automação da manufatura, envolvendo as áreas de coordenação, projeto, produção (chão-de-fábrica) e fabricação [Mill86a].

A área de coordenação compreende quatro grandes grupos de atividades : suprimentos, produção, vendas e finanças. Essas quatro sub-áreas tem em comum o fato de trabalharem basicamente com informações, sendo envolvidas com geração, arquivamento, utilização, disponibilidade e transmissão de dados. A automação está relacionada com a utilização de computadores interconectados por redes, possibilitando a transmissão de dados entre vários tipos de equipamentos.

Na área de projeto, a difusão da automação ocorreu em paralelo ao desenvolvimento da tecnologia computacional. A utilização do computador como auxílio do projeto forneceu uma técnica poderosa no trato com o trabalho intelectual. A utilização dessa técnica permitiu uma sensível economia no custo do desenho envolvido em um projeto. Na etapa de projeto preliminar, a produtividade de um sistema automatizado é um pouco maior que a dos sistemas convencionais, porém nas modificações de desenho, os ganhos de produtividade são enormes.

Na área de produção, os sistemas automáticos de produção são classificados em dois tipos principais :

- *Sistemas de automação fixa, ou rígida* : São aqueles, nos quais, a sequência das operações da folha de processo é definida pela configuração dos equipamentos. Os exemplos típicos correspondem às linhas de montagem automáticas;
- *Sistemas de automação flexíveis, ou programáveis* : São aqueles, nos quais, os equipamentos são projetados para serem flexíveis. Os exemplos típicos correspondem aos Controladores Lógicos Programáveis - CLPs, às máquinas-ferramenta comandadas por Controle Numérico por Computador - CNC e às máquinas de Controle Numérico Direto - DNC.

Na área de fabricação, a automação tem origem no desenvolvimento dos dispositivos de controle onde através da informática, é promovida a conversão da lógica da tecnologia eletrônica digital em informações, associada à comunicação entre os equipamentos. A automação nessa área está relacionada com a utilização dos equipamentos de controle numérico. Essa base técnica (informática associada à eletrônica) permite, através de linguagens de programação, a conversão das informações do desenho em uma descrição matemática numérica. As aplicações de automação são classificadas por :

- Utilização intensiva de máquinas-ferramenta a controle numérico;
- Utilização de centros de usinagem e controle numérico, juntamente com linhas de transferência dedicadas;
- Sistemas flexíveis de manufatura.

No momento atual, a automação da manufatura concentra-se na integração dos processos automatizados isolados através de um sistema completo. De uma certa forma, este desenvolvimento pode ser visto como uma mudança da automação da manufatura para a automação completa do ambiente de manufatura, designada por *Manufatura Integrada por Computador - CIM*.

2.4. MANUFATURA INTEGRADA POR COMPUTADOR - CIM

Torna-se muito difícil uma definição clara e precisa do conceito de Manufatura Integrada por Computador - CIM, pois abrange todos os setores de um ambiente de manufatura, desde o desenvolvimento até o fornecimento do produto. Cada empresa de manufatura possui uma estrutura organizacional e funcional, assim sendo, cada uma precisa definir sua própria estratégia de integração [Clar88a].

O conceito de Manufatura Integrada por Computador difere da teoria da Divisão do Trabalho proposta por Taylor, onde o ser humano tem uma capacidade limitada de trabalhar informações, pois, ao, ser destinado um pequeno número de funções específicas, o trabalhador poderá realiza-las mais rapidamente e com menos erros. O surgimento do computador, associado a utilização de bancos de dados e sistemas distribuídos, ajuda o trabalhador a executar um número maior de tarefas mais diversificadas.

No início da década de 80, o conceito de Manufatura Integrada por Computador era um conceito meramente tecnológico. Naquele período, a manufatura estava sendo impulsionada por uma série de outros conceitos que representavam novas tecnologias. Essas novas tecnologias surgiram na forma de sistemas computacionais, equipamentos computadorizados ou softwares, gerando processos automatizados que, na maioria das vezes, resultaram em ilhas isoladas de automação.

Durante os últimos anos, o conceito de Manufatura Integrada por Computador foi reformulado. O computador deixou de ser a figura principal, passando a assumir o papel de recurso para facilitar a integração.

A representação estrutural da Manufatura Integrada por Computador pode ser feita através da figura da *Pirâmide da Produtividade* (fig. 2.05), com os seus cinco níveis de atividades existentes [Cost89a]. Cada nível representa um conjunto diferente de tarefas, cada uma delas utilizando uma forma diferente de automação, de tal modo que os cinco níveis formam uma hierarquia de operações produtivas, na base, até o nível de administração global, no topo.

Fig. 2.05 - Pirâmide da produtividade

O primeiro nível (processos) representa as máquinas, os equipamentos, os sensores e atuadores envolvidos nos processos produtivos, organizados como unidades produtivas.

O segundo nível (controle de processos) faz o controle direto e supervisão das unidades produtivas do primeiro nível. Além das funções de controle e supervisão, o sistema de automação gera e armazena uma grande quantidade de dados, podendo também, receber a supervisão de níveis mais altos para determinar variações na produção ou em operações específicas dos processos. Neste nível são formadas as interfaces homem-máquina.

O terceiro nível (engenharia) tem a função de supervisionar e coordenar as atividades de todas as unidades situadas nos níveis inferiores. Informações sobre máquinas, produção ou qualidade que estão disponíveis em cada computador de supervisão do segundo nível podem ser transmitidas para este nível, para utilização pela engenharia de processos e/ou produto. Neste nível, redes de comunicação de dados convencionais são utilizadas em conjunto com as interfaces com o operador, a fim de configurar um sistema de controle distribuído, trabalhando 'on-line'. Neste nível também são gerados e distribuídos os softwares de comando numérico para as máquinas CNCs do primeiro nível.

O quarto nível (planejamento operacional) é organizacional e está ligado às gerências de manufatura encarregadas de avaliar o desempenho da produção, examinando os dados específicos de cada máquina (ou processo), as informações sobre produção e quantidades produzidas, sendo que as gerências definem então as ações a serem tomadas.

O quinto nível (alta administração) avalia o desempenho da empresa de manufatura, como um todo, analisando as informações relatadas pelos outros níveis. Gera relatórios e orienta a tomada de decisões estratégicas, administrativas e financeiras. O processo de comunicação neste nível difere dos níveis inferiores, pois, a operacionalidade não precisa ser feita 'on-line'. Redes locais de computadores, através de canais de comunicação de alto nível, têm sido largamente utilizadas entre os sistemas de nível hierárquico mais elevado.

A implantação da Manufatura Integrada por Computador envolve dois elementos básicos : elementos técnicos e elementos organizacionais [Fros89a].

Os elementos técnicos são classificados em cinco categorias principais :

- *Projeto* : Compreende o Projeto Assistido por Computador - CAD, a Engenharia Assistida por Computador - CAE e o Planejamento do Processo Assistido por Computador - CAPP;
- *Manufatura* : Compreende a Manufatura Assistida por Computador - CAM, que se refere aos equipamentos e máquinas utilizados para a produção de componentes e produtos;
- *Planejamento e controle da produção* : Compreende as tecnologias de planejamento dos recursos disponíveis (sistemas de Planejamento das Necessidades de Materiais - MRP), distribuição desses recursos e as metodologias para produção Just-in-Time - JIT;
- *Tecnologia de movimentação de materiais* : Compreende o desenvolvimento e aplicação de tecnologias ligadas à estocagem (Magazines), distribuição e movimentação de materiais (Veículos Guiados Automaticamente - AGVs, Robôs, etc.);
- *Tecnologia de integração* : Compreende os sistemas computacionais de comunicação para congregar e agrupar informações de áreas isoladas da empresa de manufatura. Os sistemas computacionais incluem hardware, software e redes de computadores especializadas (LAN, WAN, etc.).

Os elementos organizacionais são classificados em três categorias principais :

- *Projeto da organização* : Implica na alteração em relação à unidade de trabalho totalmente integrada e auto-suficiente, e no redimensionamento de níveis hierárquicos;
- *Reprojeto de funções* : A aplicação de mão-de-obra em empresas de manufatura integrada, deverá ter menor especialização e habilidade manual, e maior amplitude intelectual, além de maior generalismo no trato de problemas;
- *Alteração da função de supervisão* : Envolve controle do estado da mão-de-obra, inspeção de qualidade e produção, e interfaces entre as áreas do ambiente de manufatura.

Toda empresa de manufatura deve se precaver e preparar métodos para evitar problemas e dificuldades na implantação da Manufatura Integrada por Computador, de forma a minimizar os prejuízos ou até evitar a rejeição total de toda a estratégia definida. Os problemas e dificuldades mais encontrados na implantação e utilização da Manufatura Integrada por Computador, são :

- Dificuldades na integração de computadores e equipamentos de diferentes fornecedores;
- Solução de problemas organizacionais para a realização do pensamento de integração;
- Resistência humana, principalmente a nível de direção, devido ao medo de perda da competência;
- Falta de experiência e qualificação na área de Manufatura Integrada por Computador, inclusive dos fornecedores de software;

- Falta de um banco de dados lógico único;
- Problemas de interface.

A realidade da Manufatura Integrada por Computador exige uma definição clara das metas da empresa de manufatura, já que são exigidos grandes volumes de investimento em capital e disponibilidades para mudanças abrangentes, assim como uma metodologia de implantação por etapas, devido à complexidade das atividades a serem desenvolvidas. É também de importância a utilização de novas filosofias gerenciais como a Tecnologia de Grupo, a Reengenharia de Processos e o Just-in-Time, pois, só com elas é possível um aproveitamento total das potencialidades oferecidas pela Manufatura Integrada por Computador [Roze89a].

Atualmente, Manufatura Integrada por Computador não é apenas um projeto de automação industrial, é uma nova maneira de se administrar a manufatura, através de estratégias de projetos para instalações de manufatura e de filosofias de organização, onde, um dos grandes desafios, é a implantação de sistemas computacionais, com o objetivo de tornarem as empresas de manufatura mais *flexíveis, integráveis e previsíveis*.

3. SISTEMAS COMPUTACIONAIS DISTRIBUÍDOS

Quando os primeiros computadores começaram a ser utilizados, tiveram sua utilização restrita a órgãos do governo e grandes empresas. Eram utilizados apenas em aplicações onde a rapidez do processamento da informação era tão vital que justificava o alto investimento da sua aquisição.

Os avanços na tecnologia de novos sistemas operacionais e da microeletrônica levaram a uma contínua redução do custo dos equipamentos, que culminaram com o surgimento dos mini e microcomputadores de custo mais acessível, estando ao alcance de pequenas empresas e até mesmo de cidadãos comuns.

Como consequência dessa evolução, a forma como a atividade de processamento de dados estava organizada nas empresas mudou. Nas grandes empresas, além dos computadores de grande porte, passou a ser comum a existência de mini e microcomputadores instalados junto ao usuário final. Nesse contexto, a comunicação cresce de importância por ser o elemento que permite a essas máquinas trabalharem de forma corporativa.

Na comunicação entre computadores sob uma rede a longa distância, são normalmente utilizadas as redes públicas de telefonia como meio de interconexão. Essa técnica tem a grande vantagem de utilizar uma infra-estrutura já instalada, cujas linhas chegam a qualquer ponto do mundo. No entanto, por ser um sistema projetado para transmitir voz, impõe limites na taxa máxima de transmissão de dados entre os computadores. As redes locais, ao contrário, por serem o meio de interconexão de máquinas localizadas geograficamente próximas, não precisam utilizar a rede telefônica, podendo ser projetadas com altas taxas de transmissão, baixo custo e especificamente para transmissão de dados ou transmissão integrada de dados, voz e imagem.

3.1. SISTEMAS OPERACIONAIS

Um sistema operacional caracteriza-se como um sistema de software responsável pelo gerenciamento dos recursos de um computador, capaz de fornecer interfaces eficientes e amigáveis ao usuário [Tane92a].

A interface entre o sistema operacional e os programas do usuário é definida por um conjunto de serviços que o sistema operacional oferece, conhecidos como *chamadas de sistema*. As chamadas de sistema criam, excluem e utilizam vários objetos gerenciados pelo sistema operacional. Um desses objetos, o processo, se refere a um conceito chave em todo sistema operacional.

Um *processo* é basicamente um programa em execução. Consiste do programa executável, dos dados e pilhas utilizados pelo programa, do contador de programas, do ponteiro para a pilha e outros registradores, e de todas outras informações necessárias para a execução do programa.

Os processos de um sistema operacional são independentes entre si, encontrando-se em algum dos estados :

- *Em execução* : Utilizando a capacidade do processador no determinado instante;
- *Em espera* : Aguardando temporariamente a execução, até que um outro processo libere a capacidade de processamento;
- *Bloqueado* : Aguardando a execução, até que algum evento do sistema o desbloqueie e o coloque em estado de espera.

Segundo MILENKOVIC [Mile87a], os processos frequentemente se comunicam com outros processos, necessitando, para isso, de mecanismos para comunicação entre processos, *InterProcess Communication - IPC*. São eles :

- *Seções críticas* : Garante que um processo acesse um recurso e seja executado com exclusividade, evitando assim, condições de disputa;
- *Exclusão mútua com 'Busy Waiting'* : Quando um processo estiver acessando uma seção crítica, nenhum outro poderá alocar a seção, restando ao outro processo apenas esperar por um aviso;
- *'Sleep and Wakeup'* : Quando um processo desejar acessar uma seção crítica, este verifica se a seção está livre para ser alocada. Se não estiver, então irá esperar por um aviso até que a seção esteja livre;
- *Semáforos* : Correspondem a variáveis que armazenam o número de avisos que um processo pode receber para sair do estado de espera. Utilizado também para sincronismo entre processos. Neste caso, em um determinado instante, um processo em execução sinaliza outro processo e entra em espera; o outro processo sinalizado entra e permanece em execução, até sinalizar o primeiro processo e entrar novamente em espera, até que algum outro processo o sinalize;
- *Contadores de eventos* : Correspondem a variáveis acumulativas de um processo. Um processo em execução incrementa o contador de eventos de outro processo; o outro processo, quando estiver em execução, incrementa o contador de eventos de um outro processo e decrementa o seu contador, permanecendo em execução enquanto o seu contador não for zerado;
- *Monitores* : Correspondem a conjuntos de procedimentos que podem ser acessados por processos, fornecendo meios de exclusão mútua, evitando assim, situações de conflito;
- *Passagem de mensagens* : Corresponde a duas primitivas de comunicação (envio e recepção). Em um determinado instante, um processo em execução envia uma mensagem para outro processo e entra em recepção; o outro processo recebe a mensagem e entra em execução, até enviar uma mensagem para o primeiro processo e entrar novamente em recepção, até que algum outro processo envie uma mensagem.

Quando vários processos estão "logicamente" em execução, o sistema operacional deve decidir qual deles estará realmente em execução. Para realizar essa decisão, o sistema operacional dispõe de técnicas de escalonamento, sendo o responsável pelo chaveamento do estado em que cada processo deve se encontrar. O processo corrente realmente em execução é chamado de tarefa, ou 'task'.

A estratégia de alocação, pela qual, processos que estão em execução podem ser suspensos pelo sistema operacional, ou podem ser interrompidos por outro processo é chamada de *escalonamento preemptivo*. Já quando só o processo é responsável pelo seu tempo de execução, sendo executado até terminar ou até repassar o controle para outro processo, independentemente da ação do sistema operacional, a estratégia de alocação é chamada de *escalonamento não preemptivo*.

3.2. SISTEMAS DISTRIBUÍDOS

Um sistema distribuído caracteriza-se como um sistema capaz de administrar e gerenciar várias unidades de processamento, sem mecanismos de compartilhamento de memória e independentes entre si, de modo que a utilização destas unidades seja totalmente transparente e imperceptível ao usuário, que deve reconhecer todo o sistema como um único "processador virtual". É importante salientar que é o software, não o hardware, o fator determinante para um sistema ser distribuído ou não [Tane85a].

Em relação aos sistemas mais tradicionais, sistemas distribuídos possuem as vantagens :

- *Relação custo/desempenho* : Utilização de vários microprocessadores ao invés de minicomputadores e computadores de grande porte;
- *Software relativamente simples* : Cada processador possui uma função dedicada;
- *Tolerância a falhas* : Componentes do sistema podem ser desativados sem afetar os demais.

Porém, sistemas distribuídos apresentam como problema, o alto grau de dificuldade na tarefa de escalonar otimamente os processos entre os processadores, quando estes estão executando outros processos no mesmo instante.

Segundo MULLENDER [Mull89a], os modelos de sistemas distribuídos são agrupados em três modelos : modelo minicomputador, modelo estação e modelo banco de processadores. Estes modelos podem, inclusive, serem combinados entre si, gerando novas associações.

No modelo minicomputador (fig. 3.01), o sistema consiste de alguns minicomputadores, cada um com vários usuários. Cada usuário acessa uma máquina específica, podendo também acessar remotamente outras máquinas. Este modelo é visto como uma peculiaridade dos sistemas centralizados 'time-sharing'.

Fig. 3.01 - Modelo minicomputador

Neste modelo, todos os serviços são executados em uma máquina servidora, proporcionando maior segurança e pequeno tráfego de dados pela rede de computadores. Por outro lado, praticamente nenhum processamento é realizado no terminal do usuário. A máquina servidora executa todos os serviços, inclusive aqueles que não envolvem dados centralizados e que poderiam ser executados localmente, resultando em uma sobrecarga da máquina servidora.

No modelo estação (fig. 3.02), cada usuário possui um microcomputador, usualmente equipado com um processador potente, memória volátil, disco rígido e vídeo de alta resolução. O trabalho é todo realizado no microcomputador, porém o sistema começa a operar de forma distribuída quando suporta um único ou vários sistemas de arquivos, geralmente interconectados por uma rede local, onde os dados centralizados podem ser acessados sem a especificação da localização do seu sistema de arquivos.

Fig. 3.02 - Modelo estação

Neste modelo, a capacidade de processamento da estação cliente é amplamente aproveitada para efetuar a interface com o usuário. Entretanto, quando são precisos dados centralizados, a estação cliente os requisita à máquina servidora que contém o sistema de arquivos envolvido, que por sua vez, os envia pela rede local à estação cliente. Todo o processamento é realizado na estação cliente e praticamente nenhum na máquina servidora. Como o volume de dados pode ser muito grande, este modelo pode acarretar um enorme tráfego de dados pela rede local, proporcionando queda de performance em todo o sistema.

A primeira geração deste modelo permitiu o compartilhamento de disco do sistema de arquivos da máquina servidora. A segunda geração permitiu o compartilhamento de arquivos. Já a terceira geração, é caracterizada pelo sistema cliente-servidor, o qual permite que várias estações da rede local compartilhem também a capacidade de processamento da máquina servidora para a execução de seus serviços.

O modelo banco de processadores (fig. 3.03) corresponde a uma evolução do modelo estação, onde um ou mais processadores são alocados temporariamente para um determinado serviço do usuário. Quando o serviço é completado, o processador é liberado e retorna ao banco para próximas requisições.

Fig. 3.03 - Modelo banco de processadores

3.2.1. SISTEMAS CLIENTE-SERVIDOR

Um sistema cliente-servidor consiste em uma interconexão entre máquinas, cuja interação resulta em sinergia, utilizando também a capacidade da estação cliente de apresentar e manipular dados através de interfaces gráficas avançadas voltadas para o usuário final [Data91a]. Apresenta as características necessárias para a implementação dos modelos de sistemas distribuídos anteriores :

- Maior segurança dos dados centralizados;
- Pequeno tráfego de dados pela rede de computadores;
- Aproveitamento da capacidade de processamento da estação cliente;
- Possibilidade de execução de serviços em uma máquina servidora.

Dentro do contexto de sistemas cliente-servidor, as aplicações podem ser distribuídas em aplicações clientes e servidoras, isto é, aplicações de processamento intensivo que distribuem as tarefas entre as diversas máquinas interconectadas entre si, de acordo com as necessidades do sistema. A *aplicação cliente* provê o 'front end' da aplicação, interagindo com o usuário, montando requisições e solicitando serviços. Já a *aplicação servidora* provê o 'back end' da aplicação, oferecendo e executando os serviços requisitados, e enviando a resposta à respectiva aplicação cliente (fig. 3.04).

Fig. 3.04 - Sistema cliente-servidor

Um sistema cliente-servidor caracteriza-se por quatro níveis de aplicações :

- *Primeiro nível* : Compartilhamento de arquivos e periféricos;
- *Segundo nível* : Aplicações nas estações clientes acessando bancos de dados distribuídos;
- *Terceiro nível* : Aplicações nas máquinas servidoras disponibilizando serviços para troca de mensagens;
- *Quarto nível* : Aplicações distribuídas, clientes e servidoras.

Um fator que determina o tipo de comunicação cliente-servidor depende de como as aplicações cliente e servidor gerenciarão as requisições e serviços. Há dois tipos de comunicação cliente-servidor : assíncrona e síncrona.

Na comunicação síncrona (fig. 3.06), o cliente requisita um serviço, suspendendo o processamento até que o servidor envie a resposta.

Fig. 3.06 - Comunicação síncrona

Vários mecanismos de comunicação podem ser utilizados nos sistemas cliente-servidor, no entanto o de maior destaque é o *Remote Procedure Call - RPC* [Mull89a]. O RPC é constituído por um mecanismo simples, transparente e rápido. Sua transparência ideal ocorre quando os procedimentos remotos não podem ser distinguidos dos procedimentos locais.

É importante notar que o RPC exige sincronismo (uma aplicação cliente é bloqueada assim que fizer o pedido). Neste momento, uma mensagem é construída contendo os argumentos necessários para a sua transferência à uma máquina remota, onde o procedimento será executado. Quando o procedimento for finalizado, os resultados serão devolvidos sob a forma de uma mensagem à máquina requisitante, com o cliente finalmente reassumindo a execução.

A operação do RPC reflete o sistema cliente-servidor síncrono. O cliente faz as requisições através de um procedimento de interface ('stub' do cliente), que tem como função isolar a visão do cliente dos procedimentos utilizados para conversão, empacotamento e transmissão dos dados para o servidor. Uma vez feita a requisição, o 'stub' do cliente aguarda a resposta do servidor. No servidor, a requisição é recebida, identificada e tratada por um procedimento de interface ('stub' do servidor), com função análoga ao 'stub' do cliente (fig. 3.07).

Fig. 3.07 - Mecanismo RPC

Como as chamadas de procedimentos locais, as chamadas de procedimentos remotos estão sujeitas a perda de mensagens, falha no servidor, falha no cliente e falha no meio de comunicação. Em caso de problemas, podem ocorrer efeitos colaterais, tais como, duplicação de mensagens e bloqueio indefinido do cliente ou servidor ('deadlock'). Desse modo, há necessidade de mecanismos tanto do lado do cliente quanto do servidor capazes de resolver esses impasses. Uma solução para o problema de bloqueio é a utilização de 'timeouts' [Tane85a].

As aplicações distribuídas são tipicamente compostas de três componentes lógicos, distintos entre si :

- Componentes para interface com vídeo, teclado e outros dispositivos, incluindo também elementos utilizados na validação de dados;
- Componentes para acesso a dados por recuperação e atualização das informações armazenadas;
- Componentes responsáveis pela implementação da lógica em situações críticas, onde deva existir alta disponibilidade para os acessos que necessitem de respostas rápidas e precisas.

Existem três abordagens para a manipulação dos dados e dos componentes da aplicação em um sistema cliente-servidor : modelo acesso remoto a dados, modelo servidor de banco de dados e modelo servidor de aplicações [Stan93b].

No modelo de acesso remoto a dados (fig. 3.08), a aplicação e a apresentação dos dados são executados em cada estação cliente. A estação cliente processa os dados e gerencia o teclado e os dispositivos de entrada. As aplicações também executam a lógica da solução. Os dados são armazenados e atualizados através de chamadas padrões (por exemplo, SQL), enviadas pela rede de computadores ao servidor que contém o banco de dados. Operacionalmente, as características que definem esse modelo são a grande quantidade de mensagens e dados trocados entre o servidor e a estação cliente, além da replicação das aplicações entre as estações clientes.

Fig. 3.08 - Modelo de acesso remoto a dados

No modelo servidor de banco de dados (fig. 3.09), a apresentação dos dados é executada em cada estação cliente. A estação cliente é limitada a validar os dados, além de gerenciar o teclado e os dispositivos de entrada. As aplicações são executadas no servidor, executando a lógica da solução, e gerenciando e acessando o banco de dados. O banco de dados também assume a liderança e a responsabilidade pelos dados neste modelo.

Fig. 3.09 - Modelo servidor de banco de dados

No modelo servidor de aplicações (fig. 3.10), a estação cliente usualmente processa os dados e gerencia o teclado e os dispositivos de entrada. A aplicação lógica, neste caso, é definida na estação cliente e processada remotamente em um conjunto de servidores de aplicação. O servidor de aplicação contém os componentes da aplicação e fornece mecanismos para comunicação entre as aplicações no mesmo ou em outros servidores. As aplicações geralmente suportam acesso a banco de dados por chamadas padrões (por exemplo, SQL). A integração das aplicações servidoras é realizada pela ativação de uma outra aplicação capaz de gerenciar todas as demais aplicações juntamente com o ambiente e os recursos envolvidos. Esta aplicação corresponde a um gerenciador de sistemas transacionais, ou simplesmente, monitor OLTP (Processamento Transacional 'On-Line'). É importante ressaltar que servidores de aplicações dedicados podem residir na mesma plataforma física, como o banco de dados, ou em outras máquinas. Eles são logicamente distintos no fato das aplicações servidoras não estarem sendo executadas ou gerenciadas pela máquina com o servidor de banco de dados, mas sendo controladas pelo monitor OLTP.

Fig. 3.10 - Modelo servidor de aplicações

3.2.2. PROCESSAMENTO TRANSACIONAL 'ON-LINE' - OLTP

No passado, o Processamento Transacional 'On-Line' - OLTP, estava limitado somente a ambientes proprietários de máquinas de grande porte ('mainframe'), especificamente voltados ao mercado financeiro. Atualmente, devido à necessidade da integridade de informação, sistemas para processamento transacional estão sendo solicitados em quase todas as áreas da computação.

No processamento transacional, uma sequência de operações discretas executadas como uma única unidade de trabalho, é denominada de *transação*. Cada transação tem um início e um fim bem definidos. Uma transação é considerada transiente, até que uma decisão seja tomada para completar seu trabalho com sucesso ('*commit*'), ou então até que o trabalho seja cancelado ('*rollback*'). O trabalho de uma transação pode ser cancelado a qualquer momento antes que um '*commit*' ocorra. Uma vez processado um '*commit*', o trabalho efetuado é considerado permanente [Bers92a].

Uma transação possui quatro propriedades básicas (*ACID*) :

- *Atomicidade* : O trabalho executado por uma transação, ou é efetivado integralmente ou então, não é realizado;
- *Consistência* : Uma transação ao ser efetivada, deve deixar os recursos afetados em um estado novo e válido. Caso a transação seja cancelada, os recursos afetados deverão ser retornados ao estado anterior em que se encontravam antes do início da execução da transação;
- *Isolamento* : Quando uma transação está em execução, o seu processamento deve ser mantido separado e isolado de outras transações também em execução. Como uma transação em execução depende de determinados dados, estes dados devem ser bloqueados, a fim de evitar que outras transações em execução tentem acessá-los;
- *Durabilidade* : Os resultados e efeitos gerados por uma transação efetivada, não devem ser afetados devido a falhas de software ou hardware do sistema.

Um sistema para processamento transacional 'on-line', ou sistema OLTP, representa um modo de processamento de aplicações, onde mensagens são recebidas de fontes transacionais, como terminais ou outras aplicações, e são processadas 'on-line', compartilhando recursos comuns, como bancos de dados, sistemas de comunicação de dados, impressoras, etc.. Todo o processo fundamenta-se em mecanismos de comunicação padrão RPC [Clay92a].

Esses sistemas caracterizam-se pelo suporte a um grande número de operações simultâneas e alta disponibilidade para aplicações que necessitem de respostas rápidas e precisas, distinguindo-se do processamento em 'batch' (lotes) ou do processamento em 'time-sharing' (compartilhamento de tempo) pela ênfase em :

- Atomicidade das operações;
- Respostas rápidas;
- Ação baseada em eventos.

As propriedades ACID definem ferramentas para sistemas OLTP. Como esses sistemas simplesmente requisitam serviços, aguardando respostas, as aplicações transacionais são comparadas como um conjunto de eventos distribuídos vindos de várias estações [Stan93a]. Esse conjunto de transações subdivide-se em :

- *Transações 'flat'* : Transações sem nenhuma sub-transação. Todo o trabalho da transação ou é executado com um '*commit*' ou é interrompido com um '*rollback*';
- *Transações pseudo-conversacionais* : Transações projetadas com uma série de transações relacionadas que podem ser chamadas a cada '*commit*', passando o mesmo contexto da informação em áreas compartilhadas;
- *Transações conversacionais* : Transações que suportam com o escopo de uma única transação, múltiplas interações entre os clientes (requisitantes) em uma aplicação específica;
- *Transações distribuídas* : Transações que envolvem mais que um gerenciador de transações, podendo acessar um ou mais gerenciadores de recursos.

Para o processamento de várias transações, um sistema OLTP necessita determinar quais aplicações estarão habilitadas a executar imediatamente determinadas transações, armazenando as demais transações que não estiverem sendo executadas, para que possam depois entrar em execução. Para tanto necessita tomar decisões de escalonamento entre as tarefas [Stan93a]. Esses critérios de escalonamento subdividem-se em :

- *Prioridade de escalonamento baseada em transações* : Corresponde a prioridade de escalonamento baseada na requisição da transação;
- *Prioridade de escalonamento baseada no conteúdo dos dados* : Corresponde a prioridade de escalonamento baseada no conteúdo dos dados. Admite escalonamento preemptivo entre requisições enfileiradas;
- *Escalonamento multi-nó* : O escalonador gerencia as requisições de vários nós, associando-os a vários servidores de aplicações, de acordo com uma carga balanceada entre eles e algoritmos de escalonamento envolvendo prioridades;
- *Replicação automática do servidor* : Para otimizar a carga e o tempo de resposta, o sistema OLTP pode replicar automaticamente os servidores de aplicações;
- *Replicação administrativa do servidor* : Replicação dos servidores de aplicações via interface do administrador;
- *Roteamento baseado no conteúdo dos dados* : Requisições de transações podem ser roteadas para servidores de aplicações específicos, baseado no conteúdo dos dados.

Formas de integridade são atributos das transações que contribuem para a garantia das propriedades ACID em situações de execução normal e de falhas de sistema [Stan93a]. Essas formas de integridade subdividem-se em :

- *'Backout' para interromper transações incompletas* : Corresponde a habilidade para interromper uma transação ativa, que poderá novamente ser inicializada por uma transação de aplicação;
- *Proteção 'firewall'* : Corresponde a habilidade para proteger transações de aplicações da interferência de outras funções do sistema, garantindo o isolamento das transações;
- *'Two-phase commit'* : Protocolo de sincronismo para a função de 'commit' entre transações distribuídas;
- *Migração coordenada* : Corresponde a habilidade para o gerenciador de transações assumir o controle do 'two-phase commit' para transações distribuídas;
- *'Commit/rollback' heurístico* : Evento de 'commit' ou 'rollback' unilateralmente pelo gerenciador de recursos para o resultado das transações locais após a primeira fase do 'two-phase commit'.

Sistemas de reserva de passagens aéreas podem ser encarados como exemplo de sistemas OLTP. Estes sistemas precisam permitir que milhares de usuários acessem concorrentemente, 'on-line', uma base de dados comum. Outro exemplo muito conhecido, é encontrado na automação bancária, onde muitos usuários, em diferentes localidades, efetuam transações financeiras (depósitos, retiradas, consultas, etc.) a uma base de dados comum.

Para que uma aplicação servidora execute transações provenientes de várias fontes e localidades diferentes, é necessário o isolamento das diferenças destas várias fontes de transações.

Para garantir este isolamento e facilitar o desenvolvimento e a integração de aplicações servidoras que respeitem as quatro propriedades básicas (ACID) de uma transação, existem os gerenciadores de sistemas transacionais, ou simplesmente, monitores OLTP, que são capazes de gerenciar todas as aplicações servidoras juntamente com o ambiente e os recursos envolvidos, representando um conjunto de interfaces e procedimentos que gerenciam a alocação de recursos do sistema, provendo mecanismos para escalonamento de transações, manipulação das filas de mensagens, segurança, tolerância a falhas e recuperação do estado anterior no caso de transações canceladas [Open92a].

3.2.3. MONITORES OLTP

Monitores OLTP são gerenciadores de sistemas transacionais que suportam um grande número de operações simultâneas, muitas vezes, compartilhando dados, aplicações, ambientes e recursos envolvidos. Os usuários que necessitam dos resultados destas operações, ou transações, não devem se preocupar com a localização dos recursos que estão sendo utilizados, pois isto lhe fica de forma transparente pelo sistema [Open92a]. Seus objetivos básicos são :

- *Processamento 'on-line'* : O usuário está normalmente conectado ao sistema através de uma rede de computadores. As transações são ativadas para fazerem consultas e alterações a dados que precisam estar atualizados e disponíveis imediatamente;
- *Alta disponibilidade* : Muitas corporações requerem que seus sistemas fiquem disponíveis 24 horas por dia, 7 dias por semana. O monitor OLTP utiliza uma combinação dos recursos de hardware e software para se recuperar de eventuais falhas e minimizar o impacto de imprevistos;
- *Respostas rápidas* : Os usuários normalmente necessitam de acesso imediato às informações atualizadas. O monitor OLTP tem mecanismos de prioridade para que transações com menor importância possam ser executadas sem afetar a performance do sistema;
- *Baixo custo por transação* : O custo de processamento de uma transação deve ser uma pequena fração do custo total do negócio. Cada transação deve ser organizada e processada de maneira eficiente, explorando os recursos que o hardware e o sistema oferecem;
- *Acesso aos recursos compartilhados com integridade* : Nos sistemas OLTP, várias transações podem estar acessando os mesmos dados ao mesmo tempo. É fundamental que cada transação consiga ser executada sem interferir no processamento de qualquer outra transação. Os dados alterados precisam estar imediatamente disponíveis para as outras transações.

Um monitor OLTP (fig. 3.11) é composto pelos gerenciadores :

- *Gerenciador de aplicações* : É um software que interage com as aplicações clientes, possibilitando a interface entre estas e as aplicações servidoras;
- *Gerenciador de recursos* : É um software que disponibiliza todos os recursos compartilhados do sistema;
- *Gerenciador de transações* : É um software que interage com as aplicações servidoras, fornecendo às aplicações clientes os recursos do sistema necessários para o desenvolvimento de um sistema OLTP de forma transparente.

Fig. 3.11 - Arquitetura de um monitor OLTP

É o gerenciador de transações, o responsável pelas funções de :

- *Segurança* : Não permite que algum usuário possa executar transações se não estiver devidamente autorizado (acesso ao sistema). Não permite também nenhum tipo de acesso a dados, aos quais não foram fornecidos permissões (acesso a dados);
- *Integridade* : As atualizações são feitas tornando os dados corretos, sem a possibilidade de dois usuários alterarem o mesmo dado ao mesmo tempo, e sempre que transações não sejam completadas apropriadamente, o estado anterior é restaurado;
- *Compartilhamento de recursos* : Libera todos os recursos utilizados pelas transações imediatamente após a execução, permitindo que os recursos compartilhados fiquem alocados o menor tempo possível;
- *Interconectividade* : Permite a interconexão entre redes de computadores de vários padrões e topologias diferentes;
- *Economia* : A utilização de máquinas menores aumenta a capacidade de processamento, aproveitando os recursos do sistema de maneira mais eficiente, com aplicações menores e de fácil manutenção;
- *Facilidade de desenvolvimento* : As aplicações utilizam uma interface padrão para acessar todas as facilidades do sistema. Logo, tornam-se portáteis entre plataformas distintas e distantes das particularidades de cada sistema operacional.

3.3. MODELO HIERÁRQUICO DE INTERCONEXÃO - RM-OSI

As redes de computadores surgiram para viabilizar o compartilhamento eficiente dos recursos computacionais (hardware, software e dados) dos usuários. Em geral, esses recursos são sistemas heterogêneos, onde equipamentos de diferentes fabricantes possuem características diferentes, utilizando e executando softwares com características específicas e distintas para as aplicações dos usuários, podendo inclusive, manipular e produzir dados com formatos incompatíveis. Até mesmo quando equipamentos idênticos de um só fabricante são empregados em aplicações distintas, estes podem apresentar características heterogêneas.

Essa heterogeneidade de sistemas beneficia o usuário, por não estar restrito a um único tipo de sistema para as suas várias aplicações, podendo assim selecionar um sistema que melhor se adapte às suas necessidades. Porém, tal heterogeneidade dificulta consideravelmente a interconexão de equipamentos de fabricantes diferentes.

Convém observar que já há algum tempo, os grandes fabricantes desenvolveram soluções para interconexão de seus próprios equipamentos, referindo-se a tais soluções (particulares) como arquiteturas de rede, isto é, conjuntos de convenções para interconexão de seus equipamentos.

A interconexão entre as redes de computadores, por sua vez, contribui para dificultar o problema ainda mais : redes diferentes podem ter serviços de transmissão diferentes que requerem interfaces diferentes. É necessária, pois, uma maneira pela qual o problema de heterogeneidade não torne a interconexão de sistemas distintos inviável.

Diante da necessidade de padrões para a interconexão de sistemas heterogêneos, a International Standards Organization - ISO criou um subcomitê (SC16) para analisar o problema, com o objetivo de definir padrões para a interconexão de sistemas abertos (Open Systems Interconnection - OSI).

O SC16 sugeriu uma arquitetura estratificada em camadas para que se desenvolvesse, a partir daí, um padrão para o modelo de arquiteturas que servisse de suporte para o desenvolvimento de protocolos padronizados. Assim, o desenvolvimento do modelo de arquiteturas foi concluído e recebeu o nome de Reference Model for Open Systems Interconnection - RM-OSI (fig. 3.12).

Fig. 3.12 - Arquitetura RM-OSI

A arquitetura RM-OSI foi desenvolvida a partir de três elementos básicos :

- As aplicações existentes no ambiente RM-OSI;
- As conexões que ligam as aplicações e que lhes permitem trocar informações;
- Os sistemas.

No ambiente RM-OSI, um *sistema* é um conjunto de computadores com software associado, periféricos, terminais, operadores humanos, meios para transferência de informações, etc., capaz de processar informações. A transferência de informações (comunicação) entre sistemas é feita por meios físicos. É importante notar que o RM-OSI não se refere apenas aos aspectos de comunicação entre sistemas, mas também como estes sistemas interagem para executar uma tarefa distribuída de um processo de aplicação. Um *processo de aplicação*, no conceito do RM-OSI, é um elemento pertencente a um sistema aberto que processa informações para uma aplicação particular [Gloz86a].

O termo aberto não se aplica a nenhuma implementação, tecnologia ou interconexão particular de sistemas, mas sim à adoção de padrões, isto é, qualquer sistema adotando os padrões do SC16 pode interagir com qualquer outro sistema que obedeça aos mesmos padrões. O termo RM-OSI refere-se, pois, aos padrões para a troca de informações entre terminais, computadores, pessoas, redes de computadores, etc., que estejam abertos aos demais, com o propósito de trocar informações através da utilização dos padrões aplicáveis. Um sistema que obedece aos padrões RM-OSI na sua comunicação com outros sistemas, pode ser visto como um sistema aberto [Tane88a].

Na representação da arquitetura RM-OSI, onde as camadas sucessivas são representadas numa sequência vertical, com os meios físicos para a interconexão embaixo, cada sistema individual é logicamente decomposto em *subsistemas*, cada um correspondendo a interseção do sistema com uma camada, ou seja, cada camada é logicamente composta de subsistemas de mesmo nível de todos os subsistemas interconectados.

Cada subsistema é composto por uma ou várias *entidades*, assim, cada camada é composta de entidades, onde cada entidade pertence a um dos sistemas. Entidades em uma mesma camada são chamadas de entidades pares.

O princípio de projetar uma rede de computadores como um conjunto hierárquico de camadas, com cada camada baseada na sua camada inferior, reduz o problema global da rede ao projeto de cada uma das camadas, simplificando consideravelmente o trabalho de desenvolvimento e de manutenção. O projeto de uma camada é restrito ao contexto dessa camada e supõem que os problemas fora deste contexto já estejam devidamente resolvidos. Não interessa para uma determinada camada como as demais camadas implementam o fornecimento de seus serviços; só importa o que a camada oferece como serviço, isto é, as demais camadas são verdadeiras caixas-pretas (fig. 3.13).

Fig. 3.13 - Camadas da arquitetura RM-OSI

Com a camada física, a arquitetura RM-OSI permite a flexibilidade da utilização de vários meios físicos para a interconexão com procedimentos de controle diferentes. Esta camada define as características mecânicas, elétricas, funcionais e de procedimentos para ativar, manter e desativar conexões físicas para a transmissão de 'bits' entre as entidades da camada de enlace de dados.

Uma conexão física é identificada pela camada de enlace de dados através de identificadores de pontos de terminação da conexão física. Quando a conexão tem dois pontos de terminação, tem-se uma conexão física *ponto-a-ponto* ('store-and-forward'). Por outro lado, quando a conexão física é caracterizada por vários pontos de terminação, tem-se uma conexão física *multiponto* ('broadcast').

A transmissão das sequências de 'bits' em uma conexão física pode ser *'half-duplex'* (dados trafegando em direções diferentes, mas não simultaneamente) ou *'full-duplex'* (dados trafegando simultaneamente em direções diferentes). Além disso, a transmissão pode ser *síncrona* (cadeia fixa para a transmissão sequenciada dos 'bits') ou *assíncrona* (não exige a fixação prévia de um padrão de tempo entre o transmissor e o receptor, através da transmissão de caracteres tipo 'start-stop'). Porém, a transmissão assíncrona é muito lenta e ineficiente a nível de 'bit'. O sinal transmitido pode ser amplificado através de *repetidores*.

Alguns meios físicos de comunicação requerem a utilização de técnicas específicas para que se possa transmitir dados entre sistemas, mesmo quando sob condições de taxas de erro relativamente altas. Essas técnicas são utilizadas em procedimentos de controle de enlace de dados. As funções desta camada são : detectar, e possivelmente corrigir erros na camada física, além de fornecer à camada de rede a capacidade de pedir estabelecimento de circuitos de dados na camada física.

O propósito desta camada é prover os meios funcionais e procedimentos para ativar, manter e desativar uma ou mais conexões de enlace de dados entre as entidades da camada de rede. A interconexão entre dois segmentos com o propósito de isolar o tráfego na rede pode ser feita através de *pontes*, que captam a localização dos sistemas pelo endereço de origem, podendo assim filtrar as mensagens que devem ou não passar para outro segmento.

Na arquitetura de sistemas abertos, alguns sistemas serão destinatários de dados, enquanto outros funcionarão apenas como nós intermediários que repassam dados a outros sistemas. Logo, a camada de rede deve fornecer uma trajetória de conexão entre um par de entidades da camada de transporte, possivelmente passando por nós intermediários. Protocolos voltados para a operação da rede propriamente dita, tais como algoritmos de roteamento e de controle de congestionamento, serão agrupados nesta camada.

A camada de rede provê os meios para estabelecer, manter e terminar conexões de rede entre sistemas contendo entidades de aplicação de comunicação. Provê também os meios funcionais e de procedimento para a troca de informações, através de conexões de rede, entre duas entidades da camada de transporte. O roteamento entre segmentos pode ser realizado através de *roteadores*.

A camada de transporte foi definida para aliviar as entidades de camadas superiores da tarefa do transporte de dados entre si, controlando o transporte do sistema-fonte ao sistema-destino, isto é "fim-a-fim", fornecendo um serviço de transferência transparente de dados entre entidades da camada de sessão. O termo transparente refere-se ao fato de que as entidades de sessão (usuários de transporte) não necessitam conhecer os detalhes pelos quais é alcançada uma transferência de dados confiável e econômica.

Os usuários de transporte são identificados para a camada de transporte apenas pelos seus endereços de transporte. O serviço de transferência de dados é fornecido às entidades endereçáveis sem considerar sua localização.

Para que a transferência de dados seja econômica, a camada de transporte otimiza a utilização dos recursos de comunicação disponíveis. Esta otimização é feita respeitando as restrições de nível de desempenho, da demanda global e simultânea de todos os usuários de transporte, e do limite geral de recursos disponíveis.

O serviço da camada de rede fornece conexões de rede de uma entidade de transporte a outra. Essas conexões podem ser estabelecidas em uma única rede de comunicação (sistemas estarem localizados na mesma rede) ou através de um número de redes em série. A camada de rede se encarrega de todos os detalhes de chaveamento, roteamento e passagem de mensagens de uma rede para outra. Logo, a complexidade das funções na camada de transporte, que são responsáveis pela qualidade do serviço oferecido, depende da qualidade do serviço de rede disponível. Se a conexão oferecida pela camada de rede for confiável e econômica, as funções necessárias na camada de transporte serão proporcionalmente reduzidas. A interconexão de redes de diferentes arquiteturas pode ser realizada através de um 'gateway'.

O propósito da camada de sessão é organizar e sincronizar o diálogo, gerenciando a troca de dados na comunicação entre entidades da camada de apresentação. Para isso, fornece serviços para o estabelecimento de uma conexão de sessão entre duas entidades de apresentação, através da utilização de uma conexão de transporte.

Os serviços da camada de sessão são classificados em duas categorias :

- *Serviços de administração de sessão* : Unem e desunem duas entidades para um relacionamento. Por exemplo : 'login' e 'logout';
- *Serviços de diálogo de sessão* : Controlam a troca de dados, delimitando e sincronizando operações de transferência de dados entre duas entidades de apresentação. Por exemplo : troca de dados em 'half-duplex', 'full-duplex', etc..

Os serviços oferecidos pela camada de sessão são os primeiros a se preocuparem com as aplicações propriamente ditas, além dos serviços de mera comunicação.

A camada de apresentação fornece os serviços que podem ser selecionados pela camada de aplicação para a interpretação da sintaxe dos dados trocados. Esses serviços gerenciam a entrada, troca, apresentação e controle de dados estruturados. A camada de apresentação resolve problemas de diferença de sintaxe entre sistemas abertos na comunicação. É através dos serviços da camada de apresentação que as aplicações no ambiente RM-OSI podem se comunicar sem custos excessivos oriundos de variações de interfaces, de transformações ou de modificações das próprias aplicações.

Todas as outras camadas existem para dar suporte à camada de aplicação (camada mais alta). Os serviços desta camada são utilizados pelos próprios usuários no ambiente RM-OSI. O propósito da camada de aplicação é servir de interface para os usuários na comunicação do RM-OSI. Cada usuário é representado para os demais pela sua entidade de aplicação correspondente.

Apesar da alternativa de padronização proposta pelo RM-OSI, implementações que sigam este modelo são raras. Justifica-se a não utilização do modelo, considerando-se entre outros, aspectos como adequação e eficiência (muito da eficiência é perdida com a sobrecarga imposta pelo modelo, proveniente de suas sete camadas) [Come88a].

3.4. ARQUITETURA IEEE 802

A arquitetura IEEE 802 (fig. 3.14) resulta de um projeto do Institute of Electrical and Eletronic Engineers - IEEE, no sentido de estabelecer uma arquitetura padrão em camadas, nos moldes da arquitetura RM-OSI, só que desta vez, orientada para o desenvolvimento de redes locais de computadores, com os objetivos:

- Corresponder ao máximo possível com o RM-OSI;
- Permitir a interconexão eficiente de equipamentos de custo moderado;
- Permitir a implantação desta arquitetura a um custo moderado.

Fig. 3.14 - Arquitetura IEEE 802

Segundo GIOZZA [Gioz86b], a arquitetura IEEE 802 pode ser vista, em princípio, como uma adaptação particular das duas camadas inferiores da arquitetura RM-OSI. As características da tecnologia associada às redes locais, motivaram uma redefinição de serviços e funções das duas camadas, em três camadas :

A **camada física** se encarrega de prover os serviços básicos de transmissão e recepção de 'bits' através de conexões físicas, de maneira semelhante à camada de mesmo nome na arquitetura RM-OSI.

A **sub-camada de controle de acesso ao meio - MAC** especifica os mecanismos que permitem gerenciar a comunicação a nível de enlace de dados. Particularmente, a sub-camada MAC fornece os serviços que permitem disciplinar o compartilhamento de um meio de transmissão comum aos sistemas usuários da rede.

A **sub-camada de controle de enlace lógico - LLC** se encarrega de prover às camadas superiores os serviços que permitem uma comunicação confiável de sequência de 'bits' entre os sistemas usuários da rede. Sua especificação prevê a existência de dois tipos de serviços básicos oferecidos à camada superior. São eles :

- Um serviço que permite que as unidades de informação sejam trocadas sem o estabelecimento prévio de uma conexão a nível de enlace de dados, não havendo controle para recuperação de erros ou anomalias, e nem controle da cadência de transferência das unidades de dados;
- Um segundo serviço que permite o estabelecimento de uma conexão a nível de enlace de dados, antes da troca de dados propriamente dita, de modo a incorporar as funções de recuperação de erros, de sequenciamento e de controle de fluxo.

3.5. REDES LOCAIS DE COMPUTADORES

Segundo TANENBAUM [Tane88a], as redes locais de computadores são consideradas como soluções ao problema de interconexão de vários computadores autônomos. Seus principais objetivos são :

- Compartilhar recursos computacionais geograficamente dispersos;
- Permitir alta confiabilidade, através da existência de recursos alternativos;
- Reduzir gastos, visto que computadores de porte menor apresentam vantagens na relação custo/desempenho;
- Ser um meio de comunicação entre as pessoas.

As redes locais têm se desenvolvido com a finalidade de permitir a comunicação de dados localmente e em áreas geograficamente dispersas, com requisitos de desempenho e confiabilidade.

Os computadores componentes de uma rede local são chamados de *estações*, sendo conectadas por uma *sub-rede de comunicação*, cuja tarefa é transferir mensagens de uma estação a outra. Uma sub-rede é vista através de dois componentes distintos : os nós de comunicação e as linhas de transmissão.

Os *nós de comunicação* são chamados Interface Message Processor - IMPs, sendo utilizados para conectar duas ou mais linhas de transmissão, recebendo e transmitindo dados através da *linha de transmissão* correta. Nas sub-redes ponto-a-ponto, a informação é enviada de um nó de comunicação para outro, passando por vários nós no caminho, sendo que cada nó intermediário recebe a informação, a armazena temporariamente e a retransmite por um canal de saída. Já as sub-redes multiponto utilizam um único canal de comunicação compartilhado por todos os nós da sub-rede.

Segundo TAROUÇO [Taro86a], a principal característica das redes locais é a relativa flexibilidade na conceituação do grau de localidade da rede. No entanto, costuma-se atribuir às redes locais uma série de características técnicas gerais consideradas como típicas, de modo a distingui-las principalmente das redes de computadores de longa distância. São elas :

- Dimensões moderadas;
- Capacidade de transmissão de informações;
- Taxa de erros de transmissão;
- Atrasos de transmissão;
- Conectividade.

3.5.1. TOPOLOGIAS

A topologia de uma rede corresponde à estrutura de interconexão física das várias estações que a compõem. Essa estrutura de interconexão pode refletir tanto a localização geográfica das estações como o fluxo de informação gerado entre elas. No caso das redes locais, a interconexão das estações é feita através dos nós de comunicação.

Existem várias maneiras de se configurar a interconexão dos nós de comunicação em um ambiente localizado. Cada uma dessas maneiras apresenta características próprias, com diferentes implicações quanto ao desenvolvimento, operação e manutenção da rede. Essas implicações podem ser fatores determinantes da relação custo/desempenho global da rede. Assim, por exemplo, é fundamental a simplicidade do nó, a fim de viabilizar a interconexão de hardware usuário relativamente barato. É também importante, por exemplo, manter um certo grau de confiabilidade para a rede, em face das falhas individuais dos nós. O alcance ou capacidade de extensão da rede é uma outra implicação associada à escolha de uma determinada topologia.

A topologia não só define como os vários nós de comunicação estão interconectados, como também afeta o desempenho do fluxo de dados, os custos de implementação e a confiabilidade. Comumente, uma rede de grandes proporções compreende várias topologias interligadas.

Segundo TANENBAUM [Tane88a], as topologias existentes que melhor se adaptam às redes locais são : malha irregular, estrela, árvore, anel e barramento.

A malha irregular (fig. 3.15) constitui a estrutura topológica mais geral possível. Cada nó de comunicação conecta-se diretamente a um número variável de outros nós, dando origem a uma estrutura irregular de interconexão.

Fig. 3.15 - Topologia em malha irregular

Esta topologia é normalmente associada às redes de computadores a longa distância. Apresenta como vantagem o fato do arranjo de interconexões poder ser realizado em função do tráfego de informações, permitindo uma otimização nos meios de transmissão, relativamente caros. Por outro lado, a variedade de caminhos (rotas) resultante, introduz a necessidade de decisão de encaminhamento (roteamento) das mensagens em cada nó da rede, implicando em uma considerável capacidade de processamento e de armazenamento das mensagens em cada nó.

A topologia em estrela (fig. 3.16) é uma estrutura de interconexão bastante familiar aos sistemas computacionais tradicionais. Nessa estrutura, todas as decisões de roteamento são concentradas em um único nó de comunicação chamado nó central, simplificando muito a implementação de cada um dos outros nós. Cada nó secundário é ligado fisicamente ao nó central.

Fig. 3.16 - Topologia em estrela

Esta topologia é uma estrutura bem adaptada a situações onde o fluxo de comunicação também é centralizado, isto é, quando um certo número de nós secundários se comunicam com um nó primário central. Porém, quando o fluxo de comunicação é mais geral, a dependência de um nó centralizado pode ser uma desvantagem, pois além tornar a rede mais vulnerável, aumenta consideravelmente a complexidade do nó central relacionado com o número de nós interconectados.

A topologia em árvore (fig. 3.17) é uma expansão da topologia em estrela, pois, cada nó intermediário se encarrega da transferência de mensagens até os nós folhas. Nessa estrutura, é muito simples a inclusão de novos ramos e folhas.

Fig. 3.17 - Topologia em árvore

Nesta topologia, o roteamento é simples, mas as falhas em nós intermediários podem levar ao isolamento de múltiplos nós, e as falhas nos enlaces físicos podem levar à criação de redes não comunicantes.

A topologia em anel (fig. 3.18) é uma das estruturas de interconexão para redes que permite eliminar a necessidade de um nó de comunicação central, pois o fluxo de comunicação entre os nós é descentralizado.

Fig. 3.18 - Topologia em anel

Nesta topologia, não há necessidade de decisões de roteamento. As mensagens geradas são transmitidas unidirecionalmente de nó em nó até atingir o nó destinatário. A única decisão necessária em cada nó é a capacidade de reconhecer seu próprio endereço nas mensagens circulando no anel e copiar as que lhe são destinadas.

O meio físico de transmissão é comum a todos os nós, sendo constituído por vários segmentos de transmissão ponto-a-ponto entre pares de nós adjacentes.

Esta topologia caracteriza-se pela participação dos nós intermediários, funcionando como repetidores no processo de transmissão de uma mensagem entre dois nós não adjacentes, o que permite um maior alcance da rede. Porém, o fato de cada nó participar do processo de transmissão, coloca a confiabilidade da rede dependente da confiabilidade individual dos elementos repetidores distribuídos pelos nós.

A topologia em barramento (fig. 3.19) é uma estrutura de interconexão bastante adaptada às redes locais. Nesta topologia, também não existe a necessidade de decisões de roteamento. As mensagens são trocadas pelos nós de comunicação sem a participação dos nós intermediários. A única decisão necessária em cada um dos nós é a identificação das mensagens que lhe são destinadas.

Fig. 3.19 - Topologia em barramento

Nesta topologia, não existe armazenamento intermediário das mensagens ao nível da rede. O meio físico de transmissão é composto por um único segmento de transmissão multiponto, compartilhado pelos diversos nós interconectados.

Os nós intermediários têm geralmente um comportamento passivo em face ao processo de transmissão das mensagens no meio físico. Essa característica constitui uma importante vantagem da topologia em barramento quanto à confiabilidade da rede. Porém, essa mesma característica tende a limitar o alcance da rede.

3.5.2. SUPORTES DE TRANSMISSÃO

Segundo TANENBAUM [Tane88a], existem basicamente quatro tipos de suporte de transmissão utilizados nos sistemas de transmissão em redes locais: par trançado, cabo coaxial, fibra ótica e espaço livre.

O par trançado é um suporte de transmissão composto por dois fios metálicos (geralmente de cobre) enrolados em espiral. O trançado dos fios tende a manter constantes as propriedades elétricas ao longo do suporte, garantindo um melhor desempenho na transmissão em relação a um par de fios em paralelo. É considerado um suporte de transmissão com razoável imunidade a ruídos, além de ser de fácil manuseio devido a suas pequenas dimensões e grande maleabilidade.

O cabo coaxial é um suporte de transmissão formado basicamente por um condutor cilíndrico (geralmente de cobre) dentro de um tubo metálico concêntrico (também geralmente de cobre), que serve ao mesmo tempo como condutor de retorno e como blindagem eletrostática. Os dois condutores são separados por um material dielétrico, podendo ser o ar seco ou algum material plástico. Duas fitas de aço podem ser aplicadas em espiral sobre o condutor externo a fim de fornecer, além da proteção mecânica, uma eficiente blindagem magnética (cabos triaxiais), aumentando a imunidade às diferenças de potencial entre os aterramentos nas extremidades do cabo.

A transmissão por cabo coaxial é também considerada uma tecnologia tradicional, sendo portanto bastante disponível. Em particular, os países que operam sistemas de televisão por cabo (CATV) apresentam um grande desenvolvimento dessa tecnologia. Os cabos coaxiais são classificados conforme a tecnologia de transmissão utilizada : *banda básica* ('baseband') (50 ohms) e *banda larga* ('broadband') (75 ohms).

Os sistemas de transmissão banda larga, que utilizam procedimentos de modulação e multiplexação no domínio de frequência, tais como sistemas CATV, são particularmente interessantes no caso de integração de serviços em redes locais, como por exemplo, serviços de transmissão de dados e imagem.

A fibra ótica é um suporte de transmissão para luz infravermelho caracterizado como guia de onda luminosa. Uma fibra é constituída basicamente de uma "casca" envolvendo um núcleo composto de um material semicondutor com índice de refração ligeiramente superior ao da "casca". A fibra é confeccionada basicamente de sílica, onde o encapsulamento plástico externo provê uma proteção mecânica à mesma. Apresenta total imunidade às interferências e ruídos eletromagnéticos, constituindo um recurso confiável, além de fornecer completa isolamento elétrica das estações, permitindo grandes diferenças de potencial entre seus aterramentos, sendo também muito cômoda em função do seu pequeno tamanho e peso. A maior vantagem da fibra ótica é, sem dúvida, a alta capacidade de transmissão alcançada em decorrência da alta frequência da luz.

O espaço livre é um recurso de transmissão que corresponde a um meio natural para a propagação de sinais eletromagnéticos de uma maneira geral. Do ponto de vista da conectividade que interessa particularmente às redes locais, o espaço livre é o melhor suporte de transmissão existente. Além de prover naturalmente uma interconexão completa, permite uma grande flexibilidade na localização das estações. Essas qualidades interessam principalmente aos sistemas de comunicação onde a mobilidade das estações é um fator essencial. Apesar das vantagens apresentadas, a utilização do espaço livre como suporte de transmissão em redes locais, esbarra em uma série de inconvenientes. Os principais são o custo dos equipamentos de radiodifusão e a dependência de regulamentação pública quanto à utilização do espectro de frequências.

3.5.3. MECANISMOS DE CONTROLE DE ACESSO AO MEIO

As topologias básicas eliminam a necessidade de roteamento intermediário das mensagens, permitindo a difusão destas entre os nós de comunicação. Porém, as mesmas topologias introduzem a necessidade de se disciplinar a utilização do recurso de transmissão comum pelos vários nós interconectados.

Uma rede de comunicação, estruturada em anel ou em barramento, necessita de um mecanismo que controle o acesso dos nós ao meio de transmissão compartilhado, determinando, por exemplo, qual nó está habilitado a transmitir em um determinado instante de tempo. Os mecanismos de controle de acesso em redes locais podem ser definidos através das técnicas de multiplexação utilizadas (fig. 3.20).

Fig. 3.20 - Técnicas de multiplexação

Segundo GIOZZA [Gioz86a], a classificação das técnicas de multiplexação é feita observando-se os fatores :

- Variável de separação dos sinais (tempo ou frequência);
- Modo da temporização no caso da variável tempo ser utilizada (síncrona ou assíncrona);
- Localização das funções de gerência e controle do compartilhamento (centralizado ou distribuído).

A multiplexação por divisão em frequência - **FDM** caracteriza-se pela separação de sinais elétricos em diferentes faixas de frequência (canais banda larga) do meio físico de transmissão. Esta técnica permite que dois ou mais nós transmitam simultaneamente, sem conflitos, em um mesmo suporte físico de transmissão

A multiplexação por divisão no tempo - **TDM** caracteriza-se pela divisão do tempo de utilização do suporte físico de transmissão compartilhado pelos diversos nós. Esta técnica adapta-se perfeitamente ao caso de multiplexação de sinais elétricos digitais, sendo realizada segundo os dois modos básicos de temporização : síncrono e assíncrono. No caso *síncrono*, a alocação do tempo de utilização é feita em intervalos regulares (periódicos). Já no caso *assíncrono*, o intervalo de tempo alocado é de duração e periodicidade variável.

Na multiplexação síncrona por divisão no tempo - **STDM**, os intervalos de tempo alocados aos nós são determinados a partir de um sinal padrão de sincronismo presente periodicamente no meio físico de transmissão. O tempo entre dois padrões de sincronismo sucessivos é geralmente dividido em intervalos regulares, alocados fixamente aos nós, garantindo a todos um acesso regular ao meio de transmissão comum.

Sua utilização, por ser relativamente simples, é indicada quando se tratar de suportar aplicações por um tráfego regular contínuo, que necessitem de uma garantia e regularidade no acesso ao meio de transmissão compartilhado. Porém, torna-se muito ineficiente no suporte de aplicações caracterizadas por um tráfego irregular e intermitente.

Na multiplexação assíncrona por divisão no tempo - **ATDM**, o tempo de utilização do meio físico de transmissão é dividido dinamicamente entre os diversos nós, segundo o tráfego de mensagens gerado entre eles.

Sua utilização permite um compartilhamento mais eficiente da capacidade de transmissão no caso de tráfego tipicamente irregular, sendo adotada como base das principais técnicas de controle de acesso ao meio.

Esta técnica, por caracterizar-se pela dinâmica de alocação dos intervalos de transmissão, necessita de uma estratégia de gerenciamento e compartilhamento do meio de transmissão.

No caso de mecanismos onde as funções de gerenciamento e controle de acesso ao meio de transmissão são implantadas em um único nó, diz-se que o mecanismo de controle de acesso é do tipo *centralizado*. Porém, quando as funções de gerenciamento e controle de acesso ao meio de transmissão são distribuídas entre todos os nós ativos, tem-se caracterizado um mecanismo de controle de acesso do tipo *distribuído*.

A centralização do mecanismo de controle de acesso coloca um problema de vulnerabilidade da rede em relação a um nó central. Além disso, a existência de um nó centralizando o gerenciamento do compartilhamento do meio de transmissão, resulta, em geral, em uma baixa eficiência na utilização da capacidade de transmissão, consumida por um tráfego de mensagens de controle entre o nó central e os demais nós da rede. Por outro lado, o controle de acesso centralizado permite a implantação mais facilmente de algoritmos de prioridade, de maneira a aumentar a disponibilidade de acesso a determinados nós.

A principal técnica de controle de acesso centralizado, o '*pooling*' é muito utilizada nos sistemas de interconexão de terminais com um computador central. O nó controlador central (geralmente associado a um computador), envia aos nós secundários (associados às estações), um a um, uma mensagem de controle, convidando-os a transmitir. A ordem dos convites é estabelecida por uma lista armazenada no nó central. Os nós secundários ao serem consultados, possuindo alguma mensagem a transmitir, realizam a transmissão. Caso contrário, o nó central é notificado e passa, então, a consultar o próximo nó da lista de consulta sequencial.

Essa técnica de controle de acesso requer um constante intercâmbio de mensagens de controle entre o controlador central e os nós secundários, comprometendo de uma maneira geral, o desempenho da rede. Para ser eficiente, o controle de acesso centralizado do tipo '*pooling*' necessita que :

- O tempo de propagação de ida-e-volta seja pequeno;
- A sobrecarga de mensagens de consulta seja pequena;
- O número de estações não seja grande.

O mecanismo de controle de acesso distribuído permite, em geral, uma maior eficiência na utilização da capacidade de transmissão. A distribuição reduz consideravelmente o tráfego de mensagens de controle, característico do mecanismo centralizado, adequando-se perfeitamente à natureza aleatória e intermitente, da demanda de acesso ao meio de transmissão compartilhado.

A pioneira das técnicas de controle de acesso distribuído, o '*token passing*', permite o compartilhamento sob demanda, da banda passante de um meio de transmissão em anel. Essa técnica é baseada na existência de uma entidade única regulando a inserção de mensagens no anel, isto é, o acesso dos nós ao meio de transmissão compartilhado. Essa entidade única, denominada '*token*', pode ser entendida inicialmente como sendo uma mensagem de controle que, uma vez inicializada, circula entre todos os nós, portando o direito de transmissão no meio compartilhado.

Uma estação qualquer que deseja inserir uma mensagem no meio, deve primeiro acionar o seu nó correspondente para adquirir o 'token'. A fase de aquisição do 'token' corresponde à sua identificação e à sua consequente retirada de circulação do anel. Uma vez adquirido o 'token', o nó tem o direito exclusivo de transmissão. A possibilidade de conflito na aquisição do 'token' por parte dos vários nós, é eliminada implicitamente pela estrutura cíclica unidirecional da topologia em anel. Ao completar a transmissão de sua mensagem, o nó com posse temporária do 'token' deve remetê-lo em circulação, enviando-o ao próximo nó. A próxima estação do anel, caso tenha alguma mensagem para inserir, procede da mesma maneira descrita anteriormente. Os nós correspondentes às estações inativas, isto é, aquelas sem mensagem para transmitir, funcionam como repetidores intermediários, permitindo a circulação do 'token' entre os nós associados às estações ativas.

O procedimento de inserção de mensagem em um anel com acesso controlado por 'token' é realizado pelo procedimento :

- A estação ativa aguarda a passagem do 'token';
- A estação de posse do 'token' transmite a mensagem;
- Ao fim da transmissão, o 'token' é restituído à circulação no anel.

Esta técnica ganhou muita popularidade em aplicações de tempo real, uma vez que o seu comportamento determinístico permite o cálculo preliminar dos tempos máximos de transmissão, mesmo nos casos de congestionamento do meio físico.

A técnica de controle de acesso distribuído, *Acesso Múltiplo com Detecção de Portadora e Detecção de Colisão - CSMA/CD*, constitui a principal técnica de controle de acesso associada às redes configuradas em barramento, e à tecnologia de redes locais de uma maneira geral. Esta técnica resulta da evolução de uma classe mais geral de técnicas de controle de acesso para redes denominadas Técnicas de Contenção ou Acesso Aleatório, que caracterizam-se pelo fato das estações e, particularmente dos nós, poderem transmitir (difundir suas mensagens no meio de transmissão) quando quiserem. Entretanto, como as estações interconectadas pela rede geram de modo independente (aleatório) suas mensagens no tempo, pode acontecer que dois ou mais nós queiram ter acesso ao meio de transmissão comum no mesmo instante. Quando isso ocorre, tem-se uma situação de colisão no acesso ao meio compartilhado. O princípio de contenção, neste caso, é baseado na capacidade do nó detectar as situações de colisão. Uma vez detectada uma colisão no acesso ao meio, cada um dos nós conflitantes, aguarda um novo instante para retransmitir sua mensagem. Esse atraso obedece, em geral, a uma distribuição aleatória a fim de minimizar a probabilidade de uma nova colisão ocorrer no momento das retransmissões.

Esta técnica é muito popular em ambientes orientados a transações, onde o tráfego na rede é intermitente e os atrasos no acesso ao meio, provenientes de possíveis colisões, podem ser tolerados, situação contrária aos ambientes de tempo real.

3.5.4. PADRÕES DE COMUNICAÇÃO

O padrão IEEE 802.3 - Ethernet (fig. 3.21), desenvolvido pela Xerox Corporation, refere-se a comunicação em redes do tipo CSMA/CD, com topologia em barramento.

Fig. 3.21 - Ethernet

A nível físico, consideram-se as topologias de barramento :

- "10 Base 2" e "10 Base 5", para comunicação em banda básica a 10 Mbps, utilizando cabo coaxial (50 ohms) e suportando distância máxima de 200 a 500 metros;
- "10 Broad 36", para comunicação em banda larga a 10 Mbps, utilizando cabo coaxial (75 ohms) e suportando distância máxima de 3.600 metros;
- Par trançado, para comunicação a 1 Mbps.

O padrão IEEE 802.4 - Token Bus (fig. 3.22), desenvolvido pelo Institute of Electrical and Electronic Engineers - IEEE, refere-se a comunicação em redes do tipo 'token passing', com topologia em barramento.

Fig. 3.22 - Token Bus

Consiste na interconexão lógica das estações na forma de anel, embora a estrutura de conexão física siga a topologia em barramento. Ou seja, à cada estação é assinalada uma posição numa ordem sequencial, com o último membro da sequência seguida pelo primeiro. Cada estação conhece a identidade da estação sucessora e predecessora. Veio a ter grande influência no meio industrial.

A nível físico, consideram-se dois tipos de transmissão :

- Sistema de transmissão para comunicação em banda larga a 10 Mbps, utilizando cabo coaxial (75 ohms) e suportando distância máxima de 3.750 metros (com até 1024 estações);
- Sistema de transmissão para comunicação em banda portadora ('carrierband' - sistema em banda básica mais imune a ruídos, devido a sua codificação ser baseada em onda senoidal - recomendada para aplicações de chão-de-fábrica, pois os custos são mais baixos que o sistema em banda larga) a 5 Mbps e suportando distância máxima de 1.000 metros (com até 32 estações).

O padrão *IEEE 802.5 - Token Ring* (fig. 3.23), desenvolvido pela International Business Machines Corporation - IBM, refere-se a comunicação em redes do tipo 'token passing', com topologia em anel.

Fig. 3.23 - Token Ring

Como na Token Bus, a Token Ring tem a vantagem de possuir esquema de prioridades, o que garante sua adequação à aplicações de tempo real. A comunicação é realizada em banda básica a 4 Mbps, utilizando par trançado blindado (150 ohms).

O projeto *Fiber Distributed Data Interface - FDDI* (fig. 3.24), desenvolvido pelo American National Standards Institute - ANSI, visava a padronização de uma interface de alto desempenho entre processadores e memórias de computadores, separados entre si. Logo a seguir foi adaptado para servir de 'backbone' em redes de alto desempenho na transmissão pura de dados. Originou depois, uma variante que inclui a transmissão de voz e imagem.

Fig. 3.24 - FDDI

Refere-se a comunicação em redes com topologia em anel duplo bidirecional, um anel transmitindo no sentido horário e o outro no sentido anti-horário.

A comunicação é realizada a 100 Mbps e suportando distância máxima de 200 Km (com até 1000 estações).

O padrão *Attached Resource Computer Network - ARCnet* (fig. 3.25), desenvolvido pela Datapoint Corporation, vem assumindo importância crescente no mercado de automação industrial.

Fig. 3.25 - ARCnet

O projeto ARCnet está baseado em uma topologia em árvore, onde os nós estão interconectados em módulos designados Resource Interface Module - RIM, com os diversos RIMs interconectados em estrela.

Refere-se a comunicação em redes do tipo 'token passing', onde o 'token' é passado de RIM para RIM. A comunicação é realizada em banda básica a 2,5 Mbps, utilizando cabo coaxial (RG 62/U de 93 ohms) e suportando distância máxima 600 metros (com até 255 estações) e espalhamento de 6,5 Km : diversos segmentos podem ser interligados por repetidores infravermelho a distâncias de 2 a 3 Km.

3.5.5. REDES INDUSTRIAIS

A presença do projeto MAP/TOP foi decisiva no campo da automação industrial.

O projeto *Manufacturing Automation Protocol* - MAP, foi impulsionado pela General Motors - GM, visando essencialmente a comunicação chão-de-fábrica dos dispositivos programáveis como CLPs, CNCs, etc..

O projeto MAP enquadra-se no RM-OSI, com a definição de protocolos em cada uma das sete camadas. Está baseado nas redes padrão Token Bus, com topologia em barramento, com os protocolos de aplicação indo na direção de aplicações de tempo real e de células de manufatura flexíveis.

O projeto *Technical Office Protocol* - TOP nasceu na mesma época que o MAP, como esforço da Boeing Co. para a área técnica e de engenharia. Também enquadra-se no RM-OSI. Além disso, existe compatibilidade total do TOP com MAP nas camadas 2, 3, 4, 5, e 6 do RM-OSI, o que facilita a integração das duas redes, inclusive permitindo a utilização do mesmo meio físico para a transmissão de dados das redes MAP e TOP, em canais distintos. Está baseado nas redes padrão Ethernet, com topologia em barramento.

O projeto *Field-Bus* (barramento de campo), em desenvolvimento pela Instruments Society of America - ISA, é destinado à comunicação no chão-de-fábrica de equipamentos mais simples e especializados, como sensores de sinais analógicos, transformadores de sinal, unidades de sinais binários, contadores de impulsos, controladores, reguladores, atuadores, CLPs, etc.. Estes equipamentos são hoje interligados aos equipamentos de monitoramento e de supervisão de nível superior, através de enlaces ponto-a-ponto, utilizando as interfaces série RS-232-C.

O projeto Field-Bus deverá fornecer soluções, ainda suportáveis nos custos, para transmissão digital dos sinais, através de topologias em barramento.

3.6. PROTOCOLOS DE COMUNICAÇÃO

Qualquer que seja o tipo de comunicação entre sistemas computacionais, é necessário a observação de um conjunto de regras que discipline a troca de informações. Este conjunto de regras recebe o nome de protocolo de comunicação (fig. 3.26). Logo, qualquer processo de comunicação, independentemente dos sistemas envolvidos, pressupõe a existência de algum tipo de protocolo [Tane88a].

Fig. 3.26 - Protocolo de comunicação

Na terminologia da ISO, o elemento ativo que executa as ações determinadas para a camada (N) é chamado *entidade da camada (N)*. As entidades (N) representam a capacidade de processamento distribuído da camada (N) no desempenho de suas funções.

Exceto para a camada mais alta, cada camada (N) fornece *serviços (N)* às entidades ($N + 1$), isto é, entidades (N) adicionam valores, utilizando as funções (N) implementadas na camada (N), aos serviços ($N - 1$) que recebem da camada ($N - 1$), e oferecem esses serviços, agora mais valiosos, serviços (N), às entidades ($N + 1$). As entidades (N) se comunicam via serviços ($N - 1$), da camada imediatamente inferior. A comunicação das entidades (N) é regulamentada pelos *protocolos (N)*.

Os serviços (N) são oferecidos às entidades ($N + 1$) nos *Pontos de Acesso de Serviços - SAP*, que representam os lugares onde as entidades devem buscar os serviços da entidade de nível inferior. Assim, uma entidade (N) oferece um serviço (N) no SAP (N), e uma entidade ($N + 1$) procura o mesmo serviço (N) no SAP (N). Um *SAP (N)* é servido e utilizado apenas por uma entidade (N) e uma entidade ($N + 1$), respectivamente. Porém, uma entidade (N) pode servir vários SAPs (N) e uma entidade ($N + 1$) pode utilizar vários SAPs (N).

A comunicação entre entidades ($N + 1$) é feita através de uma associação chamada *conexão (N)*, estabelecida na camada (N) utilizando um protocolo (N). A camada (N) oferece conexões (N) como parte dos serviços (N).

Os serviços que uma camada deve oferecer à camada superior podem ser de dois tipos : orientados à conexão, ou não orientados à conexão.

Um serviço orientado à conexão (circuito virtual) é similar a um sistema telefônico, onde o serviço primeiro estabelece uma conexão, utiliza a conexão e finaliza a conexão. É um serviço confiável, pois utiliza 'acknowledgement'. Já um serviço não orientado à conexão (datagrama) é similar a um sistema postal, onde o serviço envia uma mensagem a um endereço destino, independente da verificação deste. Não é um serviço confiável, pois não utiliza 'acknowledgement'.

A comunicação entre as camadas pode ser '*simplex*' (dados trafegando em apenas uma direção), '*half-duplex*' (dados trafegando em direções diferentes, mas não simultaneamente) ou '*full-duplex*' (dados trafegando simultaneamente em direções diferentes).

Através da alocação das entidades relacionadas com a execução de um determinado serviço, são fornecidos '*sockets*' [Tane92a], possuindo as características:

- Podem ser criados e destruídos dinamicamente, e cada '*socket*' criado tem um endereço de rede associado;
- Cada '*socket*' é associado a uma entidade, suportando um tipo particular de serviço;
- Os '*sockets*' executam serviços confiáveis e orientados à conexão, ou seja, antes da execução de um serviço, é necessário o estabelecimento das conexões entre os '*sockets*' referentes às entidades envolvidas.

3.6.1. PROTOCOLO IPX/SPX

No protocolo IPX/SPX (fig. 3.27), desenvolvido pela Xerox Corporation, todos os nós de comunicação em uma rede, possuem dois endereços: endereço do grupo e endereço do nó. O endereço do grupo é realmente conhecido como endereço de rede ('network'), que descreve o meio físico no qual o nó está localizado. Exceto pela topologia em estrela (na qual, o meio físico indica o servidor da rede), é possível que vários servidores da rede possam existir com o mesmo endereço de rede. Além do mais, o endereço de rede é definido apenas em software, e o endereço do nó é definido em hardware ou 'firmware' [Macc90a].

Fig. 3.27 - Protocolo IPX/SPX

O protocolo *Internetwork Packet Exchange - IPX* (fig. 3.28) suporta somente datagramas para a comunicação na rede. É um protocolo ponto-a-ponto, livre de conexões, endereçando e direcionando mensagens de dados através da rede, correspondendo à camada de rede do RM-OSI.

	Tamanho (Bytes)
'Checksum'	2
Tamanho	2
Controle de Transp.	1
Tipo de Pacote	1
Rede Destino	4
Nó Destino	6
'Socket' Destino	2
Rede Origem	4
Nó Origem	6
'Socket' Origem	2
Dados	0-546

Fig. 3.28 - Formato de um pacote IPX

O IPX lê o endereço associado ao retorno da informação, direcionando esse endereço à área devida no sistema operacional da estação ou do servidor da rede. O IPX tem algumas desvantagens:

- Não garante o envio correto da informação;
- Não garante que a informação será devidamente sequencializada entre as transmissões;
- Não suprime a recepção de mensagens duplicadas.

O endereço destino de um pacote IPX corresponde a um endereço formado pelo endereço da rede, endereço do nó e 'socket'. O endereço da rede identifica cada segmento de uma rede com vários servidores. O endereço do nó identifica unicamente cada adaptador de rede. O 'socket' representa a aplicação destino; a aplicação pode abrir e fechar 'sockets' da mesma maneira que abrir e fechar arquivos.

Para realizar comunicações IPX, cada nó deve preparar duas estruturas de dados especiais, um Event Control Block - ECB e um cabeçalho IPX. O ECB é utilizado para se obter informações sobre as mensagens IPX e SPX: seus formatos e localização dos seus componentes na memória. O cabeçalho IPX é fisicamente acoplado à mensagem antes de ser enviada ao seu destino, sendo utilizado para determinar o comprimento da mensagem, endereço do fonte, interface de comunicação e rotina a ser chamada após um envio ou recebimento.

O protocolo *Sequenced Packet Exchange - SPX* (fig. 3.29), é orientado à conexão, formando entre dois nós, uma ligação ou "tubo", pelo qual os nós se comunicam. O SPX é mais lento que o IPX, porém suprime as deficiências do IPX, fornecendo entrega garantida da informação, mensagens sequencializadas e supressão de mensagens duplicadas. Onde o IPX é rápido, o SPX é robusto, monitorando as transmissões na rede para garantir o sucesso da entrega.

	Tamanho (Bytes)
'Checksum'	2
Tamanho	2
Controle de Transp.	1
Tipo de Pacote	1
Rede Destino	4
Nó Destino	6
'Socket' Destino	2
Rede Origem	4
Nó Origem	6
'Socket' Origem	2
Controle de Conex.	1
Tipo de 'Datastr.'	1
Id. da Conex. Orig.	2
Id. da Conex. Dest.	2
Número Seq.	2
Número 'Ack'.	2
Número Alloc.	2
Dados	0-534

Fig. 3.29 - Formato de um pacote SPX

O SPX também requer e envia reconhecimentos a um parceiro de comunicação, garantindo o sucesso da transmissão. Se um pedido não fornece uma resposta em um tempo específico, o SPX retransmite o pedido. Após um número razoável de transmissões com falha, o SPX assume que a conexão está perdida e informa o usuário.

3.6.2. PROTOCOLO TCP/IP

O protocolo TCP/IP (fig. 3.30) corresponde a um conjunto de protocolos desenvolvido pela Defense Advanced Research Projects Agency - DARPA, para ser o protocolo base da rede Internet americana. Uma grande vantagem desse protocolo é a sua capacidade de acomodar uma grande variedade de redes de comunicação, fornecendo ainda serviços a vários tipos diferentes de máquinas clientes conectadas a redes heterogêneas.

Fig. 3.30 - Protocolo TCP/IP

No protocolo TCP/IP, as camadas *TCP* e *IP* correspondem às camadas de *rede* e de *transporte*, respectivamente, do RM-OSI. As camadas física e de enlace de dados são independentes da rede de comunicação utilizada, podendo assim serem utilizadas sobre qualquer padrão IEEE 802 [Come88a].

O protocolo *Internet Protocol - IP* (fig. 3.31) define a unidade básica de transferência e o formato exato de todos os dados que passam pela rede. Em adição à especificação formal dos dados, o IP inclui um conjunto de regras que especificam como as operações devem ser processadas e como os erros devem ser tratados. Este protocolo define um mecanismo não confiável e não orientado à conexão para a transmissão dos dados.

	Tamanho (bits)
Versão	4
Compr. do 'Header'	4
Tipo de Serviço	8
Compr. Total	16
Identificação	16
Flags	3
'Offset' de Fragm.	13
Tempo de Vida	8
Protocolo	8
'Checksum'	16
Endereço Orig.	32
Endereço Dest.	32
Opções + 'Padding'	32
Dados	0-576 (octetos)

Fig. 3.31 - Formato de um pacote IP

O protocolo *Transmission Control Protocol - TCP* (fig. 3.32) fornece um serviço confiável, 'full-duplex', do qual dependem muitas aplicações. O TCP permite a uma aplicação em uma máquina enviar uma sequência de dados a uma outra aplicação em outra máquina. Este protocolo é orientado à conexão no sentido de que, antes da transmissão dos dados ser iniciada, as partes envolvidas na comunicação devem necessariamente estabelecer uma conexão. O software responsável pela implementação do TCP usualmente reside no sistema operacional e utiliza o protocolo IP para transmitir a informação pela rede.

	Tamanho (bits)
Porta Orig.	16
Porta Dest.	16
Número Seq.	32
Número 'Ack'.	32
Compr. do 'Header'	4
Reservado	6
Bits de Código	6
Janela	16
'Checksum'	16
Ponteiro Urgente	16
Opções + 'Padding'	32
Dados	0-536 (octetos)

Fig. 3.32 - Formato de um pacote TCP

O protocolo *User Datagram Protocol - UDP* (fig. 3.33) fornece um serviço de liberação de informações, não confiável e não orientado à conexão. As mensagens UDP podem ser perdidas, duplicadas ou chegarem fora de ordem, além disso as mensagens podem chegar mais rápido do que o receptor pode processá-los. O UDP utiliza o protocolo IP para transportar as mensagens pelas máquinas.

	Tamanho (bits)
Porta Orig.	16
Porta Dest.	16
Compr. Total	16
'Checksum'	16
Dados	0-536 (octetos)

Fig. 3.33 - Formato de um pacote UDP

Apesar do protocolo TCP/IP apresentar um número menor de camadas (quando comparado ao RM-OSI), ainda existe uma sobrecarga, proveniente dessas camadas. Contudo, o protocolo TCP/IP constitui-se no protocolo mais utilizado atualmente, vindo a estabelecer-se como padrão de fato para redes em ambientes UNIX, em particular envolvendo sistemas distribuídos à base de estações de trabalho. O TCP/IP oferece 'sockets' como interfaces lógicas entre as entidades (correspondentes aos SAPs do RM-OSI), associadas a uma série de comandos. Em cima destas interfaces, são oferecidos diversos protocolos como o Virtual Terminal Protocol - Telnet, o File Transfer Protocol - FTP, o Simple Mail Transfer Protocol - SMTP, já existentes, e outros mais recentes como o Network File System - NFS, o Remote File Sharing - RFS e o sistema X-windows.

3.6.3. PROTOCOLO MMS

O *Manufacturing Message Specification - MMS* é um protocolo de aplicação projetado para suportar a transmissão de mensagens entre equipamentos programáveis dos ambientes de manufatura e de controle de processos. Os equipamentos em questão são, em geral, de pequeno porte e altamente especializados. Por exemplo, em uma célula de manufatura, um computador de supervisão da célula, controla diversos equipamentos interligados por uma rede. O computador da célula terá que se comunicar com os outros computadores da fábrica, utilizando protocolos sofisticados. Através do MMS, permite-se que os equipamentos da célula se comuniquem entre si, de forma simples. Um outro exemplo poderia ser citado para um sistema de instrumentação digital (SDCD), em que um computador de supervisão está interligado a várias estações locais que fazem a aquisição e transmissão de dados em relação a sensores, atuadores e instrumentos digitais no processo [Pime90a].

O MMS é um protocolo genérico de transmissão de mensagens, aplicável a uma vasta gama de sistemas de chão-de-fábrica. Contudo não são nele descritos os aspectos de aplicações, particulares para cada tipo de dispositivo. Hoje em dia, estão em elaboração diversos desses padrões, em particular para dispositivos de comando numérico, controladores programáveis, controladores de robôs e sistemas de controle de processos.

Os serviços MMS são definidos a partir do conceito de Dispositivo Virtual de Manufatura - VMD, pela determinação de algumas características gerais e outras particulares de cada dispositivo de manufatura. Tendo em vista, as padronizações MMS (fig. 3.34) são definidas em dois grandes grupos : o primeiro constitui o núcleo do MMS e o segundo corresponde às padronizações parceiras, que acrescentam ao primeiro grupo definições e significados relevantes a alguns grupos de dispositivos de manufatura [Carv94a].

Fig. 3.34 - Padronizações MMS

4. ESPECIFICAÇÃO DO SISTEMA OLTP PROPOSTO

Nos ambientes de manufatura integrada voltados à produção discreta, a troca de informações ocorre de maneira intensa e variável devido ao grande número de operações simultâneas efetuadas pelos seus segmentos.

Logo, a utilização de sistemas OLTP no gerenciamento do fluxo de informações entre os segmentos de um ambiente de manufatura integrada voltado à produção discreta, tem o objetivo de disponibilizar mecanismos para alta performance e disponibilidade do sistema em situações críticas, mantendo a segurança e integridade dos dados, através do compartilhamento e utilização eficiente dos recursos computacionais oferecidos.

Segundo STANDISH [Stan93a], em levantamento feito sobre o mercado dos Estados Unidos da América, em 1993, os monitores OLTP existentes (relacionados por suas plataformas base) estão descritos na *tabela 4.01*, adotando-se a seguinte convenção :

- *A* : Monitores já disponíveis atuando na plataforma;
- *P* : Monitores planejados para atuarem na plataforma;
- *N* : Monitores não planejados para atuarem na plataforma.

Plataforma	CICS	ACMS	TUXEDO	ENCINA	TOP END	UniKix	VIS/TP	MTS	tp+	TIP/ix
UNIX	A	P	A	A	A	A	A	A	A	A
Windows NT	P	P	P	P	P	P	N	P	N	P
NetWare	N	N	P	P	P	N	N	N	N	N
OS/2	A	N	N	P	N	N	N	A	N	N

Tabela 4.01 - Monitores OLTP existentes

Atualmente, o monitor OLTP *TUXEDO*, comercializado pela Novell Inc., já se encontra disponível atuando na plataforma Novell NetWare.

O sistema OLTP proposto abrange as informações que definem os produtos nos processos de produção discreta, através da descrição dos desenhos, do planejamento do processo, do planejamento e controle da produção, dos cronogramas de fabricação, das ordens de fabricação, da emissão das ordens, dos sistemas de controle, e do replanejamento e reprogramação. Com isso, fundamenta-se em quatro, dos cinco existentes, níveis da pirâmide da produtividade :

- Segundo nível (controle de processos);
- Terceiro nível (engenharia);
- Quarto nível (planejamento operacional);
- Quinto nível (alta administração).

Por basear-se no gerenciamento do fluxo de informações, não estando fundamentado no primeiro nível (processos) da pirâmide da produtividade, o sistema OLTP proposto *não necessita de mecanismos para respostas em tempo real*. Assim, agregando a sua *grande disponibilidade* existente atualmente no mercado, porém, com um *número muito reduzido de monitores OLTP* atuando em sua plataforma, e juntamente com a existência de ferramentas para o desenvolvimento de aplicações cliente-servidor, oferecendo uma rede de processamento distribuído 'multithreading' aberta e flexível, escolheu-se o ambiente Novell NetWare como plataforma base para o sistema, utilizando como interface de comunicação o protocolo IPX/SPX no padrão IEEE 802.3 - Ethernet.

Diante deste contexto e características, o sistema OLTP proposto será muito eficiente em ambientes onde não se encontrem outros aplicativos saturando a rede, e onde o acréscimo de transações disparadas pelas estações clientes conectadas, possa prosseguir até não implicar em queda de performance do sistema.

O sistema OLTP proposto baseia-se no modelo servidor de aplicações, sob uma arquitetura cliente-servidor, envolvendo monitores OLTP como gerenciadores de sistemas transacionais, integrando aplicações clientes e servidoras, juntamente com o ambiente e os recursos envolvidos.

O desenvolvimento de sistemas OLTP no ambiente Novell NetWare caracteriza-se por :

- O monitor OLTP, juntamente com as APIs servidoras de comunicação e as aplicações servidoras, residem no servidor como NetWare Loadable Modules - NLMs, com o monitor OLTP gerenciando o ambiente, os recursos e as aplicações. Todo o sincronismo envolvendo trocas de mensagens e processos do servidor, é efetuado através de semáforos e filas de mensagens;
- As aplicações clientes, juntamente com as APIs clientes de comunicação, residem nas estações clientes, com as estações clientes estabelecendo sessões com o monitor OLTP;
- As aplicações clientes procuram o monitor OLTP, requisitam um serviço e aguardam a resposta. O monitor OLTP passa os serviços às aplicações servidoras, aguarda as respostas e as envia às aplicações clientes. A comunicação cliente-servidor ocorre como um par requisição/resposta, com as aplicações cliente enviando a requisição.

As aplicações clientes podem ativar transações em monitores OLTP localizados em diferentes servidores, sendo a localização destes transparente para o cliente. Uma estação cliente estabelece uma sessão com um determinado monitor OLTP antes de acessar um serviço. Uma vez estabelecida a sessão, fica identificado o requisitante do serviço, o que permitirá o controle de autorização de acesso ao serviço feito pelo monitor OLTP. Dentro desta sessão, será estabelecido um diálogo entre a aplicação cliente e as diversas aplicações servidoras gerenciadas pelo monitor OLTP (*fig. 4.01*).

Fig. 4.01 - Estrutura funcional do sistema OLTP

O sistema operacional cliente é responsável pelo gerenciamento dos recursos da estação cliente, envolvendo o ambiente de execução da aplicação cliente, juntamente com os mecanismos de comunicação oferecidos. Escolheu-se o *MS-DOS* versão 3.x (ou superior) com o ambiente gráfico *WINDOWS* versão 3.1 e os 'drivers' de comunicação padrão IPX/SPX para o envio e recepção de mensagens à rede oferecidos pelo ambiente Novell NetWare (*IPX.COM* e *NETX.EXE*).

O API IPX/SPX cliente é responsável pela interface de comunicação entre os 'drivers' IPX/SPX e a interface cliente de acesso. Foi utilizado o aplicativo *GRCAX34.EXE*, desenvolvido pela Itaotec Informática S.A..

A interface cliente de acesso é responsável pela comunicação entre o API IPX/SPX cliente e a aplicação cliente, permitindo que a aplicação cliente se comunique com o monitor OLTP através de um mecanismo padrão RPC. Esta interface oferece conectividade entre as aplicações cliente-servidor de forma transparente em relação ao meio físico. Foi utilizado o aplicativo *GRUCP34.EXE*, desenvolvido pela Itaotec Informática S.A..

A aplicação cliente é responsável pela interface com o usuário, montando requisições e solicitando serviços através de uma arquitetura cliente-servidor. Esta aplicação está voltada para o gerenciamento do fluxo de informações entre os segmentos de um ambiente de manufatura integrada voltado à produção discreta. Foi desenvolvida pelo autor deste projeto, utilizando-se o ambiente Microsoft Access em Windows, com a inclusão de bibliotecas de ligação dinâmica (DLLs) desenvolvidas em C++, resultando no aplicativo *CIM_APL.MDB*.

O sistema operacional servidor é responsável pelo gerenciamento dos recursos da rede local envolvida, envolvendo o ambiente servidor de execução do monitor OLTP e das aplicações servidoras, juntamente com os mecanismos de comunicação oferecidos. Escolheu-se o *NetWare* versão 3.11 com o protocolo de comunicação IPX/SPX para o envio e recepção de mensagens à rede.

O API IPX/SPX servidor é responsável pela interface de comunicação entre o protocolo IPX/SPX e o monitor OLTP. Foi utilizado o módulo *ITREDE.NLM*, desenvolvido pela Itaotec Informática S.A..

O monitor OLTP é responsável pelo processamento transacional do sistema OLTP e pelo gerenciamento das aplicações servidoras juntamente com o ambiente, os dados e os recursos envolvidos, oferecendo conectividade entre as aplicações cliente-servidor de forma transparente em relação ao meio físico. Foi desenvolvido conjuntamente pelo autor deste projeto e pela equipe de desenvolvimento de software da Itaútec Informática S.A. (quando o autor era membro da equipe), utilizando-se o compilador WATCOM C 32 bits, resultando no módulo *OLTP.NLM*.

As aplicações servidoras são responsáveis pela execução, propriamente dita, do "processo lógico" das aplicações servidoras. Estas aplicações estão voltadas para a ativação das funções e recursos disponibilizados pelo monitor OLTP. Foram desenvolvidas pelo autor deste projeto, utilizando-se o compilador WATCOM C 32 bits, resultando nos módulos :

- *CIM_T01.NLM* : Tem como função atualizar em uma área de memória compartilhada no servidor, o estado momentâneo enviado por cada estação cliente conectada;
- *CIM_T02.NLM* : Tem como função obter da área de memória compartilhada no servidor os últimos estados das estações clientes conectadas;
- *CIM_T03.NLM* : Tem como função enviar mensagens a qualquer estação cliente conectada. As mensagens podem ser endereçadas a uma determinada estação, ou a todas as estações (mensagens em 'broadcasting'), sendo enviadas de maneira instantânea, ou então de forma escalonada, em um horário fixo ou após algum intervalo de tempo;
- *CIM_T04.NLM* : Tem como função realizar toda a interface necessária com o método de acesso a arquivos utilizado pelo monitor OLTP, disponibilizando operações de inclusão, alteração, consulta e exclusão de dados.

4.1. ESTRUTURA FUNCIONAL DO MONITOR OLTP

Devido à sua filosofia transacional, o monitor OLTP oferece às aplicações uma total transparência com relação ao meio de processamento. As aplicações servidoras, ao receberem o controle, recebem também os dados de uma forma padronizada, da mesma forma também que os enviam ao monitor OLTP, que deve retorná-los como resposta à transação requisitada por uma aplicação cliente. É o próprio monitor OLTP que se encarrega de efetuar o controle do protocolo de comunicação para receber e enviar as mensagens que trafegam pela rede, além de tornar transparente para as aplicações, detalhes referentes ao gerenciamento de memória e outras características que dependam do sistema operacional utilizado.

Todo o funcionamento do monitor OLTP está baseado em tabelas que contêm as informações necessárias para o correto roteamento das transações às aplicações servidoras encarregadas de executá-las. Estas tabelas estão relacionadas com as aplicações servidoras e devem estar disponíveis no momento da ativação do monitor OLTP. O processo de geração das tabelas é automático e interativo, através da utilização do aplicativo *GRUTB40.EXE*, desenvolvido pela Itaútec Informática S.A e customizado pelo autor deste projeto.

A principal tabela é a tabela de transações que contém informações sobre todas as transações que poderão ser ativadas durante a execução do monitor OLTP. Por exemplo :

- A que aplicação a transação pertence;
- Que aplicação servidora trata a transação;
- Que tipo ou tipos de estações podem ativar a transação.

As demais tabelas são :

- *Tabela de arquivos* : contém a estrutura de todos os arquivos que serão abertos pelo monitor OLTP e gerenciados pelo módulo de acesso a dados;
- *Tabela de áreas permanentes* : contém a relação de todas as áreas de memória compartilhadas no servidor reservadas pelo monitor OLTP;
- *Tabela de localidades remotas* : contém a identificação dos monitores OLTP, aos quais as estações poderão se conectar, juntamente com a definição do formato da mensagem de dados que cada tipo de estação pode enviar, possibilitando ou não, a formatação dos dados.

Após ativado, o monitor OLTP disponibiliza funções de sistema como recursos que podem ser utilizados pelas aplicações servidoras para executar tarefas específicas que ampliem as funcionalidades do sistema. As funções disponibilizadas são :

- *Ssend* : Permite que a aplicação servidora envie mensagens a qualquer estação que esteja conectada ao monitor OLTP. Utilizada pela aplicação servidora *CIM_T03.NLM*;
- *Strans* : Permite que a aplicação servidora ative uma transação em um horário fixo ou após algum intervalo de tempo. Utilizada pela aplicação servidora *CIM_T03.NLM*;
- *Smem* : Retorna à aplicação servidora o endereço de uma área de memória compartilhada no servidor. Utilizada pelas aplicações servidoras *CIM_T01.NLM* e *CIM_T02.NLM*;
- *Slink* : Permite que a aplicação servidora ative outra aplicação servidora que também esteja sob o monitor OLTP. Após a execução da outra aplicação servidora, o controle volta à aplicação servidora chamadora. Utilizada pela aplicação servidora *CIM_T04.NLM* (uma consulta pode ser executada como várias leituras sequenciais).

Com essas características, o monitor OLTP possui uma arquitetura modular, proporcionando assim :

- Flexibilidade de configuração;
- Eficiência na execução;
- Melhor aproveitamento de memória.

Com isso, a estrutura funcional do monitor OLTP (*fig. 4.02*) foi definida em função da diversidade de configurações nas quais pode atuar. Para que o sistema fosse o mais genérico possível, o monitor OLTP foi desenvolvido em módulos, cada um com uma função específica, dependendo da configuração utilizada.

Fig. 4.02 - Estrutura funcional do monitor OLTP

O módulo inicializador é responsável pelas fases de inicialização e finalização do monitor OLTP. Na fase de inicialização, faz a análise dos parâmetros de ativação, executa a carga das tabelas e demais módulos necessários para sua execução, abre os arquivos utilizados, aloca os processos e recursos envolvidos, e inicializa as áreas de controle. Na fase de finalização, descarrega as tabelas e módulos carregados, fecha os arquivos abertos, e desaloca todos os processos e recursos. É executado como um processo no contexto do módulo *OLTP.NLM*, ganhando o controle assim que o monitor OLTP é ativado e, logo após as inicializações, aciona os módulos de processamento 'on-line'.

Os módulos de processamento 'on-line' são responsáveis pelo ambiente de execução das aplicações servidoras e pelo tratamento das mensagens provenientes de qualquer estação conectada ao monitor OLTP, funcionando como escalonadores de funções. Aguardam as mensagens do módulo de recepção do API, as tratam, as enviam para o módulo gerenciador de aplicações e aguardam as respostas para enviá-las ao módulo de envio para o API. Sendo mensagens de transações disparadas por tempo, as enviam para o módulo gerenciador de aplicações e aguardam as respostas para enviá-las ao módulo de processamento 'batch'. São executados como processos no contexto do módulo *OLTP.NLM*, onde esse número de processos deve ser configurado de acordo com o número de estações envolvidas, envolvendo geralmente um processo para cada estação, se não implicar em queda de performance do sistema.

O módulo de processamento 'batch' é responsável pelo armazenamento das mensagens de transações disparadas por tempo. Aguarda as mensagens do módulo gerenciador de aplicações, as armazena até que o tempo fornecido se esgote, as envia para os módulos de processamento 'on-line' e aguarda as respostas. É executado como um processo no contexto do módulo *OLTP.NLM*.

O módulo gerenciador de aplicações é responsável pela formatação das mensagens de entrada e saída, pela ativação das aplicações servidoras, e pelo gerenciamento do "processo lógico" da transação, marcando o início da transação ('begin transaction') e o fim da transação ('end transaction'). Aguarda as mensagens dos módulos de processamento 'on-line', as envia para as aplicações servidoras associadas com o código da transação e aguarda as respostas para ativar o módulo de acesso a dados, a fim de efetivar ou cancelar as operações efetuadas, para só depois, enviar as respostas aos módulos de processamento 'on-line'. Sendo mensagens de transações disparadas por tempo, as envia para o módulo de processamento 'batch' e envia as respostas para os módulos de processamento 'on-line'. É executado como uma rotina no contexto do processo do módulo inicializador.

O módulo de acesso a dados é responsável pelo acesso aos arquivos abertos pelo monitor OLTP, garantindo uma maior integração, eficiência e segurança na execução dos serviços, além de assegurar transparência quanto aos métodos de acesso e organização empregados. É executado como uma rotina no contexto do processo do módulo inicializador.

O módulo de envio para o API é responsável pelo envio das mensagens do monitor OLTP para o API IPX/SPX servidor. Aguarda as mensagens dos módulos de processamento 'on-line' e as envia para o módulo de envio para o 'driver' IPX/SPX. É executado como um processo no contexto do módulo *OLTP.NLM*.

O módulo de recepção do API é responsável pela recepção das mensagens do API IPX/SPX servidor para o monitor OLTP. Aguarda as mensagens do módulo de recepção do 'driver' IPX/SPX e as envia para os módulos de processamento 'on-line'. É executado como um processo no contexto do módulo *OLTP.NLM*.

O módulo de envio para o 'driver' IPX/SPX é responsável pelo envio das mensagens do API IPX/SPX servidor para o 'driver' IPX/SPX. Aguarda as mensagens do módulo de envio para o API e as envia para o 'driver' IPX/SPX. É executado como um processo no contexto do módulo *ITREDE.NLM*.

O módulo de recepção do 'driver' IPX/SPX é responsável pela recepção das mensagens do 'driver' IPX/SPX para o API IPX/SPX servidor. Aguarda as mensagens do 'driver' IPX/SPX e as envia para o módulo de recepção do API. É executado como um processo no contexto do módulo *ITREDE.NLM*.

As aplicações clientes basicamente tem a função de realizar a interface entre o usuário e as aplicações servidoras, através da montagem de requisições, da solicitação de serviços e do recebimento das respostas para o tratamento e validação dos dados. O processo de comunicação entre as aplicações clientes e o monitor OLTP é efetuado pela interface cliente de acesso, através de um mecanismo padrão RPC com quatro funções básicas :

- *Início* : Permite à aplicação cliente estabelecer uma sessão com o monitor OLTP;
- *Envia* : Permite à aplicação cliente enviar mensagens ao monitor OLTP;
- *Recebe* : Permite à aplicação cliente receber mensagens do monitor OLTP;
- *Fim* : Permite à aplicação cliente finalizar a sessão com o monitor OLTP.

Para a função início, a aplicação cliente fornece o nome da sua sessão cliente juntamente com o nome do monitor OLTP envolvido, recebendo depois uma 'handle' correspondendo à identificação da conexão aberta com o monitor OLTP. A partir daí, para as outras funções de comunicação, a aplicação cliente especifica somente a 'handle' de conexão com o monitor OLTP e os 'buffers' de dados.

O envio de mensagens das aplicações clientes ao monitor OLTP ocorre de forma padronizada, sendo também padronizada a forma como as aplicações clientes recebem os dados que foram devolvidos como resposta à transação requisitada. As mensagens enviadas pelas aplicações clientes são compostas normalmente pelo código da transação e dados, e as aplicações clientes, ao enviarem as mensagens, desconhecem qual aplicação servidora é responsável por executar a transação.

O monitor OLTP, ao receber as mensagens, verifica qual aplicação servidora é responsável por executar a transação e passa o controle à aplicação servidora fornecendo sempre quatro parâmetros :

- *Área de entrada* : Ponteiro para o 'buffer' que contém os dados da mensagem recebida;
- *Área de saída* : Ponteiro para o 'buffer' onde a aplicação servidora deve colocar a mensagem que será enviada de volta como resposta à aplicação cliente;
- *Área de comunicação* : Ponteiro para uma área que contém campos de informação de controle passadas do monitor OLTP para a aplicação servidora e vice-versa;
- *Área exclusiva* : Ponteiro para uma área auxiliar que pode ser utilizada pela aplicação servidora para armazenar dados quaisquer.

Os campos mais utilizados da área de comunicação são :

- *CodTrans* : Campo preenchido pelo monitor OLTP, que contém o código da transação que foi ativada. A aplicação servidora que trata mais de uma transação deve consultar esse campo para saber que transação foi ativada;
- *TamMens* : Campo preenchido pelo monitor OLTP que contém o tamanho da mensagem que está na área de entrada. A aplicação servidora deve preencher este campo com o tamanho da mensagem que será enviada como resposta;
- *SitTrans* : Campo preenchido pela aplicação servidora que informa para o monitor OLTP se a execução da transação ocorreu normalmente ou não;
- *SitCom* : Campo onde a aplicação servidora indica se vai continuar ou finalizar a comunicação com a estação.

O desenvolvimento de aplicações servidoras sob o monitor OLTP não é complicado. Basicamente, a aplicação servidora executa o procedimento :

- Recebe uma mensagem da área de entrada;
- Processa a mensagem;
- Coloca a mensagem de resposta na área de saída;
- Preenche corretamente os campos da área de comunicação;
- Ao retornar, o monitor OLTP envia os dados que estiverem na área de saída novamente para a estação que ativou a transação.

Logo, a utilização de monitores OLTP apresenta às aplicações cliente-servidor importantes funcionalidades, como :

- *Comunicação transparente entre aplicações clientes e servidoras* : A comunicação é feita através de um conjunto de funções que permitem às aplicações clientes se comunicarem com as aplicações servidoras. Esta interface oferece conectividade entre as aplicações clientes e servidoras de forma transparente em relação ao meio físico. As aplicações podem ser executadas em ambientes distintos sem a necessidade de sofrerem alterações;
- *Recuperação dinâmica de transações* : Em qualquer ponto do processamento de uma transação é possível desfazer todas as atualizações efetuadas nos arquivos até então. Isto é feito pelo módulo de acesso a dados através de uma solicitação da aplicação servidora que trata a transação, ou de maneira automática, em caso de qualquer erro durante a execução da mesma. Num ambiente de rede onde é comum vários usuários compartilharem os mesmos arquivos, esta facilidade é fundamental para assegurar a integridade dos dados e facilitar o desempenho das aplicações;
- *Recuperação automática* : Na ativação do monitor OLTP, o módulo de acesso a dados verifica se há necessidade de efetuar recuperação nos arquivos das aplicações. A recuperação será necessária caso tenha ocorrido algum problema grave durante a execução de um ciclo de transação, como falta de energia elétrica, ou mesmo um erro no sistema. Isto é sentido pelo monitor OLTP no instante de sua ativação e a recuperação é automaticamente requisitada;
- *Formatação de dados* : As mensagens que trafegam entre as estações clientes e as aplicações servidoras passam por um processo chamado de formatação, cujo objetivo é tornar transparente à aplicação servidora, o tipo de estação que solicitou a execução da transação. Além disto, a formatação provê mecanismos de edição dos campos, de forma a padronizar o tratamento dos dados feito pelas aplicações. Este tratamento consiste, por exemplo, em ajustar à esquerda ou à direita os dados no campo e/ou preencher campos com um caracter de preenchimento;
- *Ativação de transações por tempo* : Em determinados ambientes, certos procedimentos devem ser ativados com hora marcada, independentemente da iniciativa de seus usuários. O monitor OLTP possibilita a ativação automática de transações em um horário fixo ou após algum intervalo de tempo;
- *Ligação dinâmica entre aplicações servidoras* : Uma aplicação servidora que está sendo executada sob o monitor OLTP pode chamar uma outra aplicação servidora e passar parâmetros sem que as mesmas estejam ligadas. As aplicações servidoras podem estar escritas, inclusive em linguagens diferentes;
- *Roteamento de mensagens entre aplicações clientes e servidoras* : Uma aplicação servidora que está sendo executada sob o monitor OLTP pode endereçar mensagens a qualquer aplicação cliente ou estação que esteja conectada ao monitor OLTP, ou mesmo ativar transações em outro monitor OLTP. A aplicação servidora pode enviar uma mensagem para todos os clientes conectados (mensagem em 'broadcasting'), a um grupo específico de clientes, ou a um determinado cliente.

4.2. ESTRUTURA FUNCIONAL DA APLICAÇÃO CLIENTE

A aplicação cliente, representada pelo aplicativo *CIM_APL.MDB*, tem como objetivo gerenciar o fluxo de informações entre os segmentos de um ambiente de manufatura integrada voltado à produção discreta, envolvendo um contexto cliente-servidor, através da interação com o usuário do sistema para receber, consistir e tratar os dados enviados pelo monitor OLTP.

Com isso, a estrutura funcional da aplicação cliente (fig. 4.03) foi definida pela separação dos segmentos de um ambiente de manufatura em conjuntos chamados de unidades remotas, juntamente com o agrupamento das máquinas envolvidas em unidades produtivas (filosofia de tecnologia de grupo), todos interligados por uma rede local e gerenciados por um monitor OLTP.

Fig. 4.03 - Estrutura funcional da aplicação cliente

A unidade remota de tomada de decisões é responsável pelo gerenciamento de todo o processo de manufatura e pela criação dos pedidos de produção. Gerencia as tabelas de famílias, produtos, pedidos, unidades produtivas e falhas das unidades produtivas. Monitora os estados das sessões clientes, dos pedidos e das unidades produtivas. Após a criação dos pedidos, deve libera-los para a unidade remota de projeto - CAD.

A unidade remota de projeto - CAD é responsável pela elaboração dos desenhos relacionados com os pedidos fornecidos pela unidade remota de tomada de decisões. Gerencia a tabela dos aplicativos utilizados para a elaboração dos desenhos. Elabora os desenhos através da ativação de um determinado aplicativo cadastrado. Após a elaboração dos desenhos, deve liberar os pedidos relacionados para a unidade remota de processo - CAPP.

A unidade remota de processo - CAPP é responsável pela elaboração das folhas de processo e programas de comando numérico relacionados com os pedidos fornecidos pela unidade remota de projeto - CAD. Gerencia as tabelas dos aplicativos utilizados para a elaboração das folhas de processo e dos programas de comando numérico. Elabora as folhas de processo e os programas de comando numérico através da ativação dos seus respectivos aplicativos cadastrados. Após a elaboração das folhas de processo e programas de comando numérico, deve liberar os pedidos relacionados para a unidade remota de planejamento - PCP.

A unidade remota de planejamento - PCP é responsável pela elaboração das ordens de fabricação e pelo planejamento e controle da produção, relacionados com os pedidos fornecidos pela unidade remota de processo - CAPP. Gerencia a tabela dos aplicativos utilizados para a elaboração das ordens de fabricação. Monitora os estados das unidades produtivas. Elabora as ordens de fabricação através da ativação de um determinado aplicativo cadastrado, além de elaborar também o planejamento e o controle da produção. Após a elaboração das ordens de fabricação e do planejamento da produção, deve liberar os pedidos relacionados para a unidade remota de produção - CAM.

A unidade remota de produção - CAM é responsável pela interface direta com as unidades produtivas encarregadas de executar as ordens de fabricação, relacionadas com os pedidos fornecidos pela unidade remota de planejamento - PCP. Monitora os estados das unidades produtivas. Elabora o processo de produção dos pedidos através da ativação das unidades produtivas associadas. Após a elaboração do processo de produção, deve liberar os pedidos relacionados para a unidade remota de tomada de decisões.

O processo de comunicação entre a aplicação cliente e o monitor OLTP é iniciado pelo estabelecimento de uma conexão envolvendo uma sessão cliente e um monitor OLTP. Logo, torna-se necessário, preliminarmente à utilização da aplicação cliente, a definição das sessões clientes, através do gerenciamento das tabelas de :

- *Monitores OLTP* : Contém os códigos e nomes dos monitores OLTP;
- *Unidades remotas* : Contém os códigos e nomes das unidades remotas;
- *Sessões clientes* : Contém os códigos e nomes das sessões clientes. Cada sessão cliente é associada a um monitor OLTP, com a função de uma unidade remota.

Após a definição das sessões clientes, pode-se ativar o processo de conexão de qualquer sessão cliente com o monitor OLTP, passando então esta sessão a executar a função da sua unidade remota associada. Feitas as conexões, as sessões clientes conectadas passam a operar determinadas ações, sempre de maneira semelhante :

- Sob o ponto de vista funcional do processo de manufatura, as sessões clientes aguardam os pedidos e, ao recebê-los, coloca-os como pendentes, até efetuarem as ações relacionadas com esses pedidos, liberando-os depois para as próximas unidades remotas;
- Durante o período de atividade, as sessões clientes enviam mensagens ao monitor OLTP, informando qual ação está realizando ou em qual estado se encontra;
- Durante o período de atividade, as sessões clientes podem enviar mensagens a qualquer sessão cliente conectada. As mensagens são endereçadas a uma determinada sessão, ou a todas de maneira geral (mensagens em 'broadcasting'), sendo enviadas de maneira instantânea, ou então, em um horário fixo ou após algum intervalo de tempo;
- Após o período de atividade, as sessões clientes se desconectam do monitor OLTP.

A ação de *gerenciamento das tabelas* efetuada pela unidade remota de tomada de decisões compreende as tabelas :

- *Famílias* : Contém os códigos e nomes das famílias. Utilizando a filosofia de tecnologia de grupo, uma família é responsável pela relação de um pedido com suas unidades produtivas;
- *Produtos* : Contém os códigos e nomes dos produtos. Os produtos são padronizados, servindo como base para os pedidos;
- *Pedidos* : Contém os códigos e nomes dos pedidos, além das datas de início e de previsão relacionadas. Cada pedido é relacionado a uma família, partindo de um produto padrão. Após a criação de um pedido, armazena-se o seu horário de início;

- *Unidades produtivas* : Contém os códigos e nomes das unidades produtivas. Cada unidade produtiva é relacionada a uma família. Uma unidade produtiva pode estar em quatro estados : ativa, disponível, em manutenção e em falha. Após a criação de uma unidade produtiva, esta se encontra disponível, armazenado-se a data e o horário de disponibilidade;
- *Falhas das unidades produtivas* : Contém os códigos e os nomes das falhas das unidades produtivas. Corresponde a uma relação de possíveis erros que possam ocorrer durante o processo de produção de um pedido por uma unidade produtiva.

A ação de *monitoração dos estados* efetuada pela unidade remota de tomada de decisões compreende os estados :

- *Estados das sessões clientes* : Informa quais sessões clientes estão conectadas e desconectadas. As sessões conectadas informam os monitores OLTP envolvidos, as funções das unidades remotas, os estados ou situações em que se encontram e os horários de conexão. Já as sessões desconectadas informam os monitores OLTP envolvidos, as funções das unidades remotas e os horários de desconexão;
- *Estados dos pedidos* : Informa quais pedidos estão concluídos e pendentes. Os pedidos informam as famílias relacionadas, os produtos padrões, as datas e horários de início e término, as datas previstas, os horários de liberação das unidades remotas envolvidas, as unidades produtivas associadas, e os desenhos, folhas de processo, programas de comando numérico e ordens de fabricação gerados;
- *Estados das unidades produtivas* : Informa quais unidades produtivas estão ativas, disponíveis, em manutenção e em falha. As unidades produtivas informam as famílias relacionadas, o estado em que se encontram, as datas e horários de início e término dos últimos estados em que se encontraram, e as descrições das últimas falhas.

A ação de *gerenciamento das tabelas* efetuada pelas unidades remotas de projeto - CAD, de processos - CAPP e de planejamento - PCP, compreende as tabelas para os *aplicativos utilizados para a elaboração dos desenhos, das folhas de processo, dos programas de comando numérico, e das ordens de fabricação*. Todas as quatro tabelas contém os códigos e nomes dos aplicativos, os comandos de ativação, os diretórios para armazenamento e as extensões para os nomes dos desenhos, folhas de processo, programas de comando numérico e ordens de fabricação.

A ação de *elaboração* efetuada pelas unidades remotas de projeto - CAD, de processos - CAPP e de planejamento - PCP, compreende a elaboração dos *desenhos, das folhas de processo, dos programas de comando numérico, e das ordens de fabricação*. Todas as quatro elaborações englobam os pedidos envolvidos, as famílias relacionadas, os produtos padrões, os aplicativos utilizados, os nomes dos desenhos, folhas de processo, programas de comando numérico e ordens de fabricação gerados, indicativos dizendo se os nomes gerados serão passados ou não, como parâmetros para os aplicativos, e o próprio processo de ativação e execução do aplicativo.

A ação de *monitoração dos estados* das unidades produtivas efetuadas pelas unidades remotas de planejamento - PCP e de produção - CAM, é idêntica a mesma ação efetuada pela unidade remota de decisões.

A ação de *elaboração* efetuada pela unidade remota de planejamento - PCP compreende as elaborações :

- *Planejamento da produção* : Engloba os pedidos envolvidos, as famílias relacionadas, os produtos padrões, os nomes das ordens de fabricação, e os códigos e nomes das unidades produtivas. Cada unidade produtiva é associada a um pedido, mediante a família envolvida;
- *Controle da produção* : Engloba as unidades produtivas, as famílias relacionadas, o estado em que se encontram, as datas e horários de início e término dos últimos estados em que se encontraram, e as descrições das últimas falhas. O estado de cada unidade produtiva pode ser alterado, alterando-se com isso as datas e horários de início e término do estado anterior e do novo estado da unidade produtiva.

A ação de *elaboração do processo de produção* efetuada pela unidade remota de produção - CAM engloba os pedidos envolvidos, as famílias relacionadas, os produtos padrões, os nomes das ordens de fabricação, as unidades produtivas associadas, indicativos dizendo se os pedidos já foram produzidos ou não, e a própria ativação e execução do processo de produção. Como o sistema gerencia fluxo de informações, não estando fundamentado no primeiro nível (processos) da pirâmide da produtividade, o processo de produção das unidades produtivas foi reproduzido através de um *processo de simulação das unidades produtivas*. Este processo engloba as unidades produtivas, as famílias relacionadas e indicativos representando sensores recebendo sinais de falha, situações de pausa, e situações de conclusão antecipada. Existem três sensores e, ao serem pressionados, ter-se-a disponível uma lista de falhas, devendo-se escolher uma entre elas, e decidindo se a unidade produtiva será colocada momentaneamente em estado de falha, bloqueando o processo até que este sensor seja pressionado novamente, ou então se a unidade produtiva será colocada imediatamente em estado de falha, cancelando o processo (neste caso, somente a unidade remota de planejamento - PCP, através da elaboração do *controle da produção*, poderá ativar novamente a unidade produtiva). Caso seja solicitado uma situação de pausa, o processo permanecerá bloqueado, até que a situação de pausa seja solicitada novamente. Caso seja solicitado uma conclusão antecipada, o processo será imediatamente concluído. É importante ressaltar que não havendo falha, e não sendo concluído antecipadamente, o processo estará concluído após *trinta segundos*.

A ação de *envio de mensagens* a quaisquer sessões clientes conectadas efetuada por todas as unidades remotas engloba as sessões clientes envolvidas, indicativos dizendo se as mensagens serão enviadas a uma determinada sessão, ou em 'broadcasting', indicativos dizendo se as mensagens serão enviadas instantaneamente, ou em um horário fixo ou após algum intervalo de tempo, as mensagens específicas, e o próprio processo de envio de mensagens. Caso seja escolhido envio de mensagem a uma determinada sessão, ter-se-a disponível uma lista de sessões conectadas, devendo-se escolher uma entre elas. Caso seja escolhido envio em um horário fixo, este horário deve ser fornecido, e caso seja escolhido envio após algum intervalo de tempo, então este intervalo que deve ser fornecido.

A ação de *liberação dos pedidos* efetuada por todas as unidades remotas engloba os pedidos envolvidos, as famílias relacionadas, os produtos padrões, as próximas unidades remotas a receberem os pedidos e o próprio processo de liberação dos pedidos. Cada pedido liberado implica no armazenamento do seu horário de liberação pela unidade remota envolvida, com este pedido passando a ser pendente somente para a próxima unidade remota. Quando o pedido for liberado pela unidade remota de produção - CAM, além do armazenamento do seu horário de liberação por essa unidade remota, este pedido não passará mais a ser pendente, passando para concluído e armazenando a data e horário de término do mesmo.

Todas as ações efetuadas pelas sessões clientes são executadas por meio de requisições de serviços feitas ao monitor OLTP, através da ativação das transações descritas na *tabela 4.02*.

<i>Transação</i>	<i>Aplicação servidora</i>	<i>Função</i>
<i>Transação T01</i>	<i>CIM T01.NLM</i>	Atualizar no monitor OLTP o estado momentâneo em que a sessão cliente se encontra.
<i>Transação T02</i>	<i>CIM T02.NLM</i>	Obter no monitor OLTP os estados momentâneos das sessões clientes conectadas.
<i>Transação T03</i>	<i>CIM T03.NLM</i>	Enviar mensagens às sessões conectadas, de maneira instantânea, ou então, em um horário fixo ou após algum intervalo de tempo.
<i>Transação T04</i>	<i>CIM T04.NLM</i>	Inclusão de dados nos arquivos gerenciados pelo monitor OLTP.
<i>Transação T05</i>	<i>CIM T04.NLM</i>	Alteração de dados dos arquivos gerenciados pelo monitor OLTP.
<i>Transação T06</i>	<i>CIM T04.NLM</i>	Consulta de dados dos arquivos gerenciados pelo monitor OLTP.
<i>Transação T07</i>	<i>CIM T04.NLM</i>	Exclusão de dados dos arquivos gerenciados pelo monitor OLTP.

Tabela 4.02 - Descrição das transações utilizadas

5. APRESENTAÇÃO DO SISTEMA OLTP

A apresentação do sistema OLTP foi elaborada de maneira a demonstrar a viabilidade e as vantagens do sistema proposto. A abordagem básica da apresentação envolve o acompanhamento do processo de elaboração, execução e conclusão de um determinado pedido em um ambiente de manufatura integrada voltado à produção discreta.

Antes do início do processo, propriamente dito, se faz necessário a geração das tabelas a serem utilizadas pelo monitor OLTP, através da execução do aplicativo *GRUTB40.EXE* (tela 5.01). Foram geradas as quatro tabelas: tabelas de transações e aplicações, tabelas de arquivos, tabelas de áreas permanentes, e tabelas de localidades remotas.

A *tabela de transações e aplicações* gerada, *MTBTCIM* (tela 5.02) descreve as transações disponíveis pelo monitor OLTP, associando cada transação a uma aplicação servidora, além de especificar os tipos de terminais habilitados a ativar as transações (no caso, as estações clientes correspondem ao tipo *TERMINAL*), e definir as características relativas a cada transação (por exemplo, se a transação pode ser ativada por tempo).

Tela 5.01 - Gerenciador de tabelas

Tela 5.02 - Tabela de transações e aplicações

A *tabela de arquivos* gerada, *MTDDCIM* (tela 5.03) descreve a estrutura dos arquivos a serem abertos e gerenciados pelo monitor OLTP.

A *tabela de áreas permanentes* gerada, *GRTCCIM* (tela 5.04) descreve o formato das áreas de memória compartilhadas no servidor reservadas pelo monitor OLTP.

Tela 5.03 - Tabela de arquivos

Tela 5.04 - Tabela de áreas permanentes

A *tabela de localidades remotas* gerada, MTAGCIM (tela 5.05) descreve a identificação dos monitores OLTP, aos quais as estações clientes poderão se conectar, juntamente com a definição do formato da mensagem de dados que cada tipo de estação pode enviar, possibilitando ou não, a formatação dos dados enviados.

Após a geração das tabelas, para iniciar o processo no servidor, deve-se ativar o *monitor OLTP* (tela 5.06), através da carga do módulo OLTP.NLM no console do servidor NetWare, segundo o comando 'load sys:cim\.nlm\oltp mtbtcim mtddcim grtccim mttagcim id=OLTP0001 rede=(mstat=n) path=cim\.nlm;cim\tab'. Os parâmetros utilizados são descritos na *tabela 5.01*.

Tela 5.05 - Tabela de localidades remotas

Tela 5.06 - Ativação do monitor OLTP

Parâmetro	Função
MTBTCIM	Carregar a tabela de transações e aplicações MTBTCIM, ativando automaticamente as aplicações servidoras nela definidas, através da carga dos módulos CIM T01.NLM, CIM T02.NLM, CIM T03.NLM e CIM T04.NLM.
MTDDCIM	Carregar a tabela de arquivos MTDDCIM.
GRTCCIM	Carregar a tabela de áreas permanentes GRTCCIM.
MTAGCIM	Carregar a tabela de localidades remotas MTAGCIM.
ID	Identificar este monitor OLTP como OLTP0001.
REDE	Ativar automaticamente o API IPX/SPX servidor, através da carga do módulo ITREDE.NLM. O argumento mstat=n indica para o API não gerar mensagens de status para o monitor OLTP. Uma mensagem de status é gerada quando alguma sessão conectada permanece sem enviar mensagens durante um certo período de tempo.
PATH	Definir o caminho para procura das tabelas e módulos a serem carregados.

Tabela 5.01 - Descrição dos parâmetros de ativação do monitor OLTP

Ativado o monitor OLTP deve-se preparar cada estação cliente para o início do processo, ativando-se em cada uma :

- O API IPX/SPX cliente, através da carga do aplicativo GRCA34.EXE na estação cliente, segundo o comando 'grca34.exe a nnom=10 nses=10 mstat=n'. Os parâmetros utilizados são descritos na tabela 5.02;
- A interface cliente de acesso, através da carga do aplicativo GRUCP34.EXE na estação cliente, segundo o comando 'grucp34.exe a ses=10 trec=10'. Os parâmetros utilizados são descritos na tabela 5.03.

Parâmetro	Função
A	Carrega o aplicativo como módulo residente na estação.
NNOM	Indica o número máximo de identificações que a estação pode assumir, sendo criado um nome para cada sessão conectada.
NSES	Indica o número máximo de sessões que a estação pode ter conectada ao mesmo tempo.
MSTAT	Indica para o API não gerar mensagens de status para o API IPX/SPX servidor. Uma mensagem de status é gerada quando a sessão conectada permanece sem enviar mensagens durante um certo período de tempo.

Tabela 5.02 - Descrição dos parâmetros de ativação do API IPX/SPX cliente

Parâmetro	Função
A	Carrega o aplicativo como módulo residente na estação.
SES	Indica o número máximo de sessões que podem estar conectadas ao mesmo tempo pela aplicação cliente.
TREC	Indica o tempo máximo, em segundos, que a interface cliente de acesso pode aguardar quando chamada pela aplicação cliente para receber uma mensagem.

Tabela 5.03 - Descrição dos parâmetros de ativação da interf. cli. de acesso

Já preparada a estação cliente deve-se iniciar o processo, ativando-se a aplicação cliente (tela 5.07), através da carga do aplicativo CIM_APL.MDB, segundo o comando 'c:\cim\msarn200.exe c:\cim\cim_apl.mdb /ini c:\windows\cim.ini', onde msarn200.exe corresponde ao 'runtime' do Microsoft Access em Windows e o parâmetro /ini indica a utilização do arquivo de inicialização cim.ini.

Preliminarmente ao estabelecimento de uma conexão envolvendo uma sessão cliente e o monitor OLTP, torna-se necessário a definição dos monitores OLTP, unidades remotas e sessões clientes a serem utilizados (tela 5.08).

Tela 5.07 - Ativação da aplicação cliente

Tela 5.08 - Definições

A tabela de monitores OLTP (tela 5.09) define quais os monitores OLTP estão habilitados a receberem conexões de sessões clientes. Foram definidos os monitores OLTP0001 e OLTP0002.

A tabela de unidades remotas (tela 5.10) define quais os segmentos de um ambiente de manufatura estão disponíveis para utilização e gerenciamento. As unidades remotas foram padronizadas em:

- Unidade remota de tomada de decisões;
- Unidade remota de projeto - CAD;
- Unidade remota de processo - CAPP;
- Unidade remota de planejamento - PCP;
- Unidade remota de produção - CAM.

Tela 5.09 - Tabela de monitores OLTP

Tela 5.10 - Tabela de unidades remotas

A tabela de sessões clientes (tela 5.11) define quais sessões estão habilitadas para conexão a um determinado monitor OLTP, exercendo a função de uma determinada unidade remota. Cada sessão cliente está associada a somente um monitor OLTP, com a função de somente uma unidade remota. Toda sessão cliente recém criada, inicialmente se encontra desconectada. As sessões clientes definidas são descritas na tabela 5.04.

Após as definições preliminares, pode-se iniciar o estabelecimento das conexões das sessões clientes com o monitor OLTP (tela 5.12).

Tela 5.11 - Tabela de sessões clientes

Tela 5.12 - Estabelecimento das conexões

Sessão cliente	Monitor OLTP	Unidade remota
DEC00001	OLTP0001	Unidade remota de tomada de decisões
DEC00002	OLTP0002	Unidade remota de tomada de decisões
CAD00001	OLTP0001	Unidade remota de projeto - CAD
CAD00002	OLTP0002	Unidade remota de projeto - CAD
CAPP0001	OLTP0001	Unidade remota de processo - CAPP
CAPP0002	OLTP0002	Unidade remota de processo - CAPP
PCP00001	OLTP0001	Unidade remota de planejamento - PCP
PCP00002	OLTP0002	Unidade remota de planejamento - PCP
CAM00001	OLTP0001	Unidade remota de produção - CAM
CAM00002	OLTP0002	Unidade remota de produção - CAM

Tabela 5.04 - Sessões clientes definidas

Com o estabelecimento das conexões, inicia-se efetivamente o processo de execução da aplicação cliente. Basicamente, este processo consiste na conexão das sessões clientes (telas 5.13 e 5.14), no gerenciamento pelo monitor OLTP das sessões clientes conectadas (tela 5.15), no desempenho das funções de cada sessão cliente durante o seu período de conexão, na desconexão das sessões clientes (telas 5.16 e 5.17) acarretando a liberação pelo monitor OLTP das sessões clientes (tela 5.18), e na desativação do monitor OLTP (tela 5.19). Somente uma sessão cliente pode estar conectada por estação, e após uma conexão, a sessão cliente torna-se indisponível para outras conexões, permanecendo até a desconexão, onde torna-se disponível novamente.

Tela 5.13 - Conexão das sessões clientes

Tela 5.14 - Sessão cliente conectada

5. Apresentação do Sistema OLTP

Tela 5.15 - Gerenciamento das sessões clientes

Tela 5.16 - Desconexão das sessões clientes

Tela 5.17 - Sessão cliente desconectada

Tela 5.18 - Liberação das sessões clientes

Tela 5.19 - Desativação do monitor OLTP

Esta apresentação retrata uma situação do sistema OLTP em um ambiente de manufatura integrada voltado à produção discreta, onde :

- O monitor *OLTP0001* já está ativo;
- As sessões clientes *DEC00001*, *CAD00001*, *CAPP00001*, *PCP00001* e *CAM00001* já estão conectadas;

- O fluxo de execução de um pedido origina-se na unidade remota de tomada de decisões, passando depois pelas unidades remotas de projeto, de processo, de planejamento e de produção, nesta ordem. O pedido é considerado concluído após ser liberado pela unidade remota de produção;
- Existe um pedido *PED_01* já concluído, ou seja, já liberado por todas as unidades remotas;
- Existe um pedido *PED_02* pendente na sessão cliente *CAM00001*;
- Existe um pedido *PED_03* pendente na sessão cliente *PCP00001*;
- Existe um pedido *PED_04* pendente na sessão cliente *CAPP00001*;
- Existe um pedido *PED_05* pendente na sessão cliente *CAD00001*;
- A unidade remota de tomada de decisões cria um pedido *PED_06*, liberando-o a seguir. A partir daí, todo o fluxo de desenvolvimento dos pedidos é acompanhado por entre as unidades remotas, dando sempre como prioridade de execução os pedidos pendentes anteriormente.

5.1. UNIDADE REMOTA DE TOMADA DE DECISÕES

A unidade remota de tomada de decisões (*tela 5.20*) gerencia todo o processo de manufatura e de criação dos pedidos de produção, sendo responsável pela definição das famílias, produtos, pedidos, unidades produtivas e falhas das unidades produtivas (*tela 5.21*).

Tela 5.20 - Uni. rem. de tomada de decisões

Tela 5.21 - Definições / Decisões

A tabela de famílias (*tela 5.22*) define as famílias responsáveis pela relação de pedidos com suas unidades produtivas. Cada pedido será produzido somente nas unidades produtivas que pertençam a sua família. Foram definidas as famílias *FAM_0001* e *FAM_0002*.

A tabela de produtos (*tela 5.23*) define os produtos padrões, que servem como base para os pedidos. Foram definidos os produtos *PRODUTO_01* e *PRODUTO_02*.

Código	Nome da família
1	FAM_0001
2	FAM_0002
3	

Tela 5.22 - Tabela de famílias

Código	Nome do produto
1	PRODUTO 01
2	PRODUTO 02
3	

Tela 5.23 - Tabela de produtos

A tabela de unidades produtivas (tela 5.24) define as máquinas envolvidas. Cada unidade produtiva será associada a um pedido, através da sua família. Foram definidas as unidades produtivas *UNIDADE 01 - FAMÍLIA 01* associada à família *FAM_0001*, *UNIDADE 02 - FAMÍLIA 01* associada à família *FAM_0001*, *UNIDADE 03 - FAMÍLIA 02* associada à família *FAM_0002* e *UNIDADE 04 - FAMÍLIA 02* associada à família *FAM_0002*. Toda unidade produtiva recém criada, inicialmente se encontra disponível.

A tabela de falhas das unidades produtivas (tela 5.25) define os possíveis erros que possam ocorrer durante o processo de produção de um pedido por uma unidade produtiva. Foram definidos os erros *FALHA FORÇADA (CONTROLE)*, *FALHA_02*, *FALHA_03*, *FALHA_04*, *FALHA_05*, *FALHA_06*, *FALHA_07*, *FALHA_08*, *FALHA_09* e *FALHA_10*.

Código	Nome da unidade produtiva	Família
1	UNIDADE 01 - FAMÍLIA 01	FAM_0001
2	UNIDADE 02 - FAMÍLIA 01	FAM_0001
3	UNIDADE 03 - FAMÍLIA 02	FAM_0002
4	UNIDADE 04 - FAMÍLIA 02	FAM_0002

Tela 5.24 - Tabela de unidades produtivas

Código	Nome da falha da unidade produtiva
1	FALHA FORÇADA (CONTROLE)
2	FALHA_02
3	FALHA_03
4	FALHA_04
5	FALHA_05
6	FALHA_06
7	FALHA_07
8	FALHA_08
9	FALHA_09
10	FALHA_10

Tela 5.25 - Tabela de falhas das uni. produtivas

A tabela de pedidos (tela 5.26) define os pedidos a serem executados, tomando-se como base os produtos já definidos. Cada pedido será associado a uma unidade produtiva através da sua família. Os pedidos definidos são descritos na *tabela 5.05*. Nesse instante, é criado um novo pedido, *PED_06* pertencente à família *FAM_0001*, tendo como base o produto *PRODUTO_01*. Todo pedido recém criado, inicialmente se encontra pendente, até ser liberado.

Com os elementos da unidade remota de tomada de decisões já definidos e com o pedido *PED_06* já criado, pode-se iniciar o processo de monitoração dos estados das sessões clientes, dos pedidos e das unidades produtivas (tela 5.27).

Tela 5.26 - Tabela de pedidos

Tela 5.27 - Monitoração / Decisões

Pedido	Família	Produto
PED 01	FAM 0001	PRODUTO 01
PED 02	FAM 0001	PRODUTO 02
PED 03	FAM 0002	PRODUTO 01
PED 04	FAM 0002	PRODUTO 02
PED 05	FAM 0001	PRODUTO 01

Tabela 5.05 - Pedidos definidos

A monitoração dos estados das sessões clientes (tela 5.28) informa toda a situação das sessões clientes conectadas e desconectadas, inclusive os estados e situações em que se encontram no momento.

A monitoração dos estados dos pedidos informa toda a situação dos pedidos concluídos e pendentes, inclusive horários de liberação por cada unidade remota envolvida. A monitoração dos estados dos pedidos concluídos (tela 5.29) informa também as unidades produtivas associadas, e os desenhos, folhas de processo, programas de comando numérico e ordens de fabricação gerados.

Tela 5.28 - Estados das sessões clientes

Tela 5.29 - Estados dos pedidos concluídos

Através da monitoração dos estados dos pedidos pendentes (tela 5.30), também é possível saber a cada momento, em que estado um pedido se encontra e qual unidade remota o está tratando, inclusive a quanto tempo.

Tela 5.34 - Pedido liberado

Tela 5.35 - Envio de mensagens

O processo de envio de mensagens para as demais unidades remotas consiste basicamente na elaboração da mensagem (tela 5.36) e na parametrização do mecanismo de envio, indicando as sessões clientes destino e a maneira de envio.

Após a elaboração da mensagem, esta é enviada (tela 5.37) sendo recebida instantaneamente, ou após algum intervalo de tempo (dependendo da maneira escolhida de envio), pela sessão cliente destino (tela 5.38).

Após todas as operações e funções executadas, a unidade remota de tomada de decisões pode desconectar-se do monitor OLTP (tela 5.39).

Tela 5.36 - Elaboração da mensagem

Tela 5.37 - Mensagem enviada

Tela 5.38 - Recepção de mensagens

Tela 5.39 - Desconexão / Decisões

5.2. UNIDADE REMOTA DE PROJETO - CAD

A unidade remota de projeto - CAD (tela 5.40) gerencia todo o processo de elaboração dos desenhos relacionados com os pedidos liberados pela unidade remota de tomada de decisões, sendo responsável pela definição dos aplicativos utilizados para a elaboração dos desenhos (tela 5.41). Tem como pendentes, os pedidos *PED_05* e *PED_06*.

Tela 5.40 - Uni. rem. de projeto - CAD

Tela 5.41 - Definições / Projeto

A tabela de aplicativos para elaboração dos desenhos (tela 5.42) define os possíveis aplicativos a serem utilizados na elaboração dos desenhos, permitindo total flexibilidade para a escolha e execução automática dos aplicativos. Para teste, foi definido o aplicativo *CAD_01* (*PAINTBRUSH*), ativado através do comando '*pbrush.exe*', tendo como base para armazenamento dos desenhos o diretório '*c:\cim*', armazenando os desenhos com extensão '*bmp*'.

Com os aplicativos para elaboração dos desenhos já definidos, pode-se iniciar o processo de elaboração dos desenhos para os pedidos pendentes, no caso o pedido *PED_05* (tela 5.43).

Tela 5.42 - Tabela de aplicativos / Projeto

Tela 5.43 - Elaboração / Projeto

O processo de elaboração dos desenhos envolve a escolha dos pedidos pendentes, no caso o pedido *PED_05*, dos aplicativos para elaboração dos desenhos e dos nomes dos desenhos (tela 5.44), indicando também se o nome dos desenhos serão passados como parâmetro na chamada do aplicativo. Após a escolha, o aplicativo é ativado automaticamente, possibilitando o início efetivo da elaboração do desenho (tela 5.45).

Tela 5.44 - Escolhas na elaboração do desenho

Tela 5.45 - Aplicativo para elab. do desenho

Após a elaboração do desenho do pedido *PED_05*, pode-se iniciar o processo de liberação dos pedidos (tela 5.46), totalmente análogo ao processo de liberação de pedidos efetuado pela unidade remota de tomada de decisões. Após este processo, o pedido *PED_05* é então liberado, deixando de ser pendente para a unidade remota de projeto - CAD, passando como pendente para a unidade remota de processo - CAPP.

Após todas as operações e funções executadas, a unidade remota de projeto - CAD pode desconectar-se do monitor OLTP (tela 5.47).

Tela 5.46 - Liberação dos pedidos / Projeto

Tela 5.47 - Desconexão / Projeto

5.3. UNIDADE REMOTA DE PROCESSO - CAPP

A unidade remota de processo - CAPP (tela 5.48) gerencia todo o processo de elaboração das folhas de processo e programas de comando numérico relacionados com os pedidos liberados pela unidade remota de projeto - CAD, sendo responsável pela definição dos aplicativos utilizados para a elaboração das folhas de processo e programas de comando numérico (tela 5.49). Tem como pendentes, os pedidos PED_04 e PED_05.

Tela 5.48 - Uni. rem. de processo - CAPP

Tela 5.49 - Definições / Processo

O processo de definição dos aplicativos para elaboração das folhas de processo e programas de comando numérico é totalmente análogo ao processo de definição dos aplicativos para elaboração dos desenhos efetuado pela unidade remota de projeto - CAD.

Com os aplicativos para elaboração das folhas de processo e programas de comando numérico já definidos, pode-se iniciar o processo de elaboração das folhas de processo e programas de comando numérico para os pedidos pendentes, no caso o pedido PED_04 (tela 5.50).

O processo de elaboração das folhas de processo e programas de comando numérico é totalmente análogo ao processo de elaboração dos desenhos efetuado pela unidade remota de projeto - CAD.

Após a elaboração das folhas de processo e programas de comando numérico do pedido *PED_04*, pode-se iniciar o processo de liberação dos pedidos (*tela 5.51*), totalmente análogo ao processo de liberação dos pedidos efetuado pela unidade remota de tomada de decisões. Após este processo, o pedido *PED_04* é então liberado, deixando de ser pendente para a unidade remota de processo - CAPP, passando como pendente para a unidade remota de planejamento - PCP.

Tela 5.50 - Elaboração / Processo

Tela 5.51 - Liberação dos pedidos / Processo

Após todas as operações e funções executadas, a unidade remota de processo - CAPP pode desconectar-se do monitor OLTP (*tela 5.52*).

Tela 5.52 - Desconexão / Processo

5.4. UNIDADE REMOTA DE PLANEJAMENTO - PCP

A unidade remota de planejamento - PCP (*tela 5.53*) gerencia todo o processo de elaboração das ordens de fabricação e de planejamento e controle da produção relacionados com os pedidos liberados pela unidade remota de processo - CAPP, sendo responsável pela definição dos aplicativos utilizados para a elaboração das ordens de fabricação (*tela 5.54*). Tem como pendentes, os pedidos *PED_03* e *PED_04*.

Tela 5.53 - Uni. rem. de planejamento - PCP

Tela 5.54 - Definições / Planejamento

O processo de definição dos aplicativos para elaboração das ordens de fabricação é totalmente análogo ao processo de definição dos aplicativos para elaboração dos desenhos efetuado pela unidade remota de projeto - CAD.

O processo de monitoração das unidades produtivas (*tela 5.55*) é totalmente análogo ao processo de monitoração das unidades produtivas efetuado pela unidade remota de tomada de decisões.

Com os aplicativos para elaboração das ordens de fabricação já definidos, pode-se iniciar o processo de elaboração das ordens de fabricação, do planejamento da produção e do controle da produção para os pedidos pendentes, no caso o pedido *PED_03* (*tela 5.56*).

Tela 5.55 - Monitoração / Planejamento

Tela 5.56 - Elaboração / Planejamento

O processo de elaboração das ordens de fabricação é totalmente análogo ao processo de elaboração dos desenhos efetuado pela unidade remota de projeto - CAD.

O processo de elaboração do controle da produção (*tela 5.57*) envolve a situação momentânea do processo de produção, através do estado momentâneo de cada unidade produtiva. Este estado pode ser alterado, implicando na mudança desta unidade produtiva para um novo estado, alterando também a situação do processo de produção (*tela 5.58*). A nova situação da produção (*tela 5.59*) é observada pela alteração das datas e horários de início e término de cada último estado que as unidades produtivas se encontravam.

O processo de elaboração do planejamento da produção envolve a escolha dos pedidos pendentes, no caso o pedido *PED_03*, e das unidades produtivas associadas a mesma família do pedido (*tela 5.60*). Após a escolha, este pedido é incluído automaticamente no processo de produção (*tela 5.61*).

Após a elaboração das ordens de fabricação e do planejamento da produção para o pedido *PED_03*, pode-se iniciar o processo de liberação dos pedidos (*tela 5.62*), totalmente análogo ao processo de liberação dos pedidos efetuado pela unidade remota de tomada de decisões. Após este processo, o pedido *PED_03* é então liberado, deixando de ser pendente para a unidade remota de planejamento - PCP, passando como pendente para a unidade remota de produção - CAM.

Tela 5.57 - Controle da produção

Tela 5.58 - Alteração da situação da produção

Tela 5.59 - Situação da produção alterada

Tela 5.60 - Planejamento da produção

Tela 5.61 - Pedido incluído na produção

Tela 5.62 - Liberação dos ped. / Planejamento

Após todas as operações e funções executadas, a unidade remota de planejamento - PCP pode desconectar-se do monitor OLTP (tela 5.63).

Tela 5.63 - Desconexão / Planejamento

5.5. UNIDADE REMOTA DE PRODUÇÃO - CAM

A unidade remota de produção - CAM (tela 5.64) gerencia a interface direta com as unidades produtivas encarregadas de executar as ordens de fabricação relacionadas com os pedidos liberados pela unidade remota de planejamento - PCP. Tem como pendentes, os pedidos PED_02 e PED_03.

O processo de monitoração das unidades produtivas (tela 5.65) é totalmente análogo ao processo de monitoração das unidades produtivas efetuado pela unidade remota de tomada de decisões.

Tela 5.64 - Uni. rem. de produção - CAM

Tela 5.65 - Monitoração / Produção

A elaboração do processo de produção (tela 5.66) envolve a escolha dos pedidos pendentes, no caso o pedido *PED_02*, a serem produzidos pelas unidades produtivas associadas no processo de planejamento da produção pela unidade remota de planejamento - PCP (tela 5.67).

Tela 5.66 - Processo de produção

Tela 5.67 - Escolhas no processo de produção

O processo de produção baseia-se na simulação de um processo de produção nas unidades produtivas (tela 5.68). Este processo de simulação engloba indicadores representando sensores recebendo sinais de falha, situações de pausa e situações de conclusão antecipada.

Um sensor de falha ao ser pressionado, interrompe imediatamente o processo de simulação, disponibilizando uma lista de possíveis erros para escolha (tela 5.69). Escolhido o erro, tem-se a opção de colocar a unidade produtiva em estado de falha, ou interromper temporariamente a simulação (tela 5.70). Colocando a unidade produtiva em falha, o processo de produção será cancelado, podendo ser retomado somente após a intervenção da unidade remota de planejamento - PCP, com a elaboração do controle da produção. Interrompendo temporariamente a simulação, esta será interrompida (tela 5.71), sendo liberada somente após o mesmo sensor de falha ser pressionado novamente (tela 5.72), dando prosseguimento normal ao processo de simulação (tela 5.73).

Após a conclusão do processo de simulação (tela 5.74), o pedido submetido a este processo é tido como produzido, concluindo-se assim o processo de produção (tela 5.75).

Tela 5.68 - Início da simulação da produção

Tela 5.69 - Falha na simulação

Tela 5.70 - Interrupção da simulação

Tela 5.71 - Simulação interrompida

Tela 5.72 - Simulação liberada

Tela 5.73 - Prosseguimento da simulação

Tela 5.74 - Conclusão da simulação

Tela 5.75 - Conclusão do proc. de produção

Após a conclusão do processo de produção do pedido *PED_02*, pode-se iniciar o processo de liberação dos pedidos (tela 5.76).

O processo de liberação dos pedidos envolve a escolha dos pedidos pendentes, no caso o pedido *PED_02* e das próximas unidades remotas, em particular a unidade remota de tomada de decisões responsável pela criação deste pedido (tela 5.77).

Tela 5.76 - Liberação dos pedidos / Produção

Tela 5.77 - Escolhas na lib. dos ped. / Produção

Após a escolha do pedido pendente, pedido *PED_02*, e da unidade remota de tomada de decisões, este pedido é então liberado e concluído (tela 5.78), deixando de ser pendente para a unidade remota de produção - CAM, finalizando todo o acompanhamento do fluxo de execução de um pedido.

Após todas as operações e funções executadas, a unidade remota de produção - CAM pode desconectar-se do monitor OLTP (tela 5.79).

Tela 5.78 - Pedido liberado e concluído

Tela 5.79 - Desconexão / Produção

6. CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS

Em um sistema cliente-servidor simples, vários clientes emitem requisições e um servidor responde. Mas, quando a escala de requisições cresce muito, por exemplo, de cinquenta para quinhentas ou mais, a maioria desses sistemas sofrem quedas consideráveis de performance. Assim, das três abordagens analisadas para a manipulação dos dados e dos componentes das aplicações em sistemas cliente-servidor, somente o modelo servidor de aplicações envolvendo monitores OLTP resolve este problema de crescimento, modificando o fluxo de requisição-resposta e oferecendo integralmente os benefícios da filosofia cliente-servidor, ou seja, modelamento e desenvolvimento flexível das aplicações, escolha de componentes de alta tecnologia e utilização eficiente dos recursos computacionais.

Os sistemas OLTP posicionam-se como arquiteturas modernas de computação distribuída e processamento cliente-servidor, provendo mecanismos que oferecem integridade dos dados e um ótimo ambiente de execução para aplicações envolvendo situações críticas, através de uma gama de serviços que garantem :

- Alta performance e balanceamento de carga devido ao processamento concorrente;
- Alta disponibilidade devido a mecanismos de recuperação automática;
- Integridade de dados devido a mecanismos de recuperação dinâmica;
- Segurança devido a mecanismos de gerenciamento integrado de recursos;
- Possibilidade de ativação de transações em um horário fixo ou após algum intervalo de tempo;
- Possibilidade de envio de mensagens entre estações clientes;
- Possibilidade das aplicações compartilharem uma mesma área de memória no servidor;
- Possibilidade da coexistência de bancos de dados heterogêneos;
- Grande capacidade de expansão do sistema, devido à total transparência com relação ao ambiente de processamento;
- Redução de custos computacionais através da utilização eficiente dos recursos.

Esta redução de custos pode ser observada através das *tabelas 6.01, 6.02 e 6.03* elaboradas por STANDISH [Stan93a], em levantamento feito sobre os monitores OLTP existentes no mercado dos Estados Unidos da América, em 1993. Estas tabelas envolvem os custos de um sistema cliente-servidor simples baseado em um servidor de banco de dados, e de sistemas cliente-servidor baseados em servidores de aplicações envolvendo monitores OLTP. O *servidor de aplicações (1)*, corresponde a este servidor com um monitor OLTP. O *servidor de aplicações (2)*, corresponde a outro servidor com outro monitor OLTP, porém fazendo parte juntamente com o servidor de aplicações (1) do mesmo sistema. É importante salientar que à medida que necessita-se de um número maior de usuários, juntamente com um volume maior de transações por segundo, os sistemas OLTP tornam-se cada vez mais rentáveis. Todos os custos envolvidos foram obtidos com base nos produtos existentes no mercado dos Estados Unidos da América, em 1993.

6. Conclusões e Sugestões para Trabalhos Futuros

Arquitetura	Sistema	Banco de dados	Monitor OLTP	Total	Redução
Servidor de banco de dados	US\$ 20.000,00	US\$ 79.000,00	-	US\$ 99.000,00	US\$ 52.500,00 (53 %)
Servidor de aplicações (1)	US\$ 20.000,00	US\$ 19.000,00	US\$ 7.500,00	US\$ 46.500,00

Fonte : The Standish Group International [Stan93b]

Tabela 6.01 - 16 usuários / 1 transação por segundo (1 monitor OLTP)

Arquitetura	Sistema	Banco de dados	Monitor OLTP	Total	Redução
Servidor de banco de dados	US\$ 70.000,00	US\$ 120.000,00	-	US\$ 190.000,00	US\$ 89.000,00 (47 %)
Servidor de aplicações (1)	US\$ 20.000,00	US\$ 48.000,00	US\$ 10.000,00	US\$ 101.000,00	
Servidor de aplicações (2)	US\$ 20.000,00	-	US\$ 3.000,00

Fonte : The Standish Group International [Stan93b]

Tabela 6.02 - 64 usuários / 4 transações por segundo (2 monitores OLTP)

Arquitetura	Sistema	Banco de dados	Monitor OLTP	Total	Redução
Servidor de banco de dados	US\$ 98.000,00	US\$ 200.000,00	-	US\$ 298.000,00	US\$ 159.000,00 (53 %)
Servidor de aplicações (1)	US\$ 41.000,00	US\$ 63.000,00	US\$ 12.000,00	US\$ 139.000,00	
Servidor de aplicações (2)	US\$ 20.000,00	-	US\$ 3.000,00

Fonte : The Standish Group International [Stan93b]

Tabela 6.03 - 128 usuários / 10 transações por segundo (2 monitores OLTP)

Devido à sua filosofia transacional, um sistema OLTP simplifica o acesso de um grande número de operações simultâneas com redução de custos, justamente por fornecer de forma transparente um ambiente concorrente de processamento distribuído, tornando muito eficiente o gerenciamento do fluxo de informações entre os segmentos de um ambiente de manufatura integrada voltado à produção discreta, garantindo performance, integridade e pronta disponibilidade de informações. Isso tudo, criando para o processo envolvido uma única imagem do sistema e contribuindo para um ambiente mais integrável nas empresas de manufatura.

Como sugestões para trabalhos futuros, tem-se as propostas descritas abaixo :

- Considerando-se a proposta do sistema OLTP implementado não abranger o primeiro nível (processos) da pirâmide da produtividade, espera-se que seja possível complementá-lo com esse nível, de modo que o sistema OLTP passe a gerenciar também o fluxo de informações entre os elementos do chão-de-fábrica, envolvendo robôs, sensores, CLPs, CNCs, entre outros, envolvendo assim, situações de tempo real;
- A nível de segurança do sistema, poderia ser implementado um mecanismo de usuários e senhas, a fim de garantir o acesso de determinados usuários às sessões clientes;
- A nível de flexibilidade total do sistema perante o fluxo de execução de um pedido, poderia ser permitido que um pedido tivesse vários desenhos, folhas de processo, programas de comando numérico e ordens de fabricação, além de poder estar também associado a várias unidades produtivas;
- A nível de portabilidade, poderia implementar-se o sistema para ambientes UNIX, utilizando como interface de comunicação o protocolo TCP/IP.

7. BIBLIOGRAFIA

- [Abra84a] ABRAMS, A.; COTTON, I. W. - Computer Networks : A Tutorial / Fourth Edition - *IEEE Computer Society* - 1984.
- [Adai91a] ADAIR, M. B. - Case Study : Implementing CIM with NetWare at Compaq Computer Corporation - *Novell Research* - Mai. 1991 - p. 39-61.
- [Barr89a] BARRA, R.; MENDES, M. D. J. - Padronização de Modelos de Produtos e Redes de Comunicação em MAP/TOP - *Automação & Indústria* - Out. 1989 - p. 17-18.
- [Bers92a] BERSON, A. - *Client/Server Architecture* - McGraw-Hill - 1992.
- [Blac87a] BLACK, U. D. - *Computer Networks* - Prentice-Hall - 1987.
- [Blac89a] BLACK, U. D. - *Data Networks : Concepts, Theory, and Practice* - Prentice-Hall - 1989.
- [Bras88a] BRASIL, SEI - Proposta de Plano Setorial de Automação Industrial - *Rel. SEI 029* - 1988.
- [Bres__a] BRESSAN, G.; RUGGIERO, W. V. - Caracterização de Sistemas Distribuídos - *Boletim Scopus*.
- [Byte95a] BYTE + Computação Cliente/Servidor - *BYTE Especial* - Mai. 1995 - p. 57-82.
- [Carv94a] CARVALHO, T. C. M. D. B. - *Arquiteturas de Redes de Computadores OSI e TCP/IP / BRISA* - Makron Books - 1994.
- [Clar88a] CLARK, K. B.; et al. - *A Perspective on Computer Integrated Manufacturing Tools* - Harvard Business School - 1988.
- [Clay92a] CLAYBROOK, B. - *OLTP Online Transaction Processing Systems* - John Wiley & Sons - 1992.
- [Chon86a] CHONG, P. S. - Developing a Prototype Micro-Computer Network for Implementing a Manufacturing Planning and Control System in Small Manufacturing Companies - *Departament of Industrial Engineering & Management* - 1986 - p. 69-73.
- [Come87a] COMER, E. D. - *Operating System Design, Internetworking with Xinu / Second Edition* - Prentice-Hall - 1987.

- [Come88a] COMER, E. D. - *Internetworking with TCP/IP Principles, Protocols and Architecture* - Prentice-Hall - 1988.
- [Cost89a] COSTA, C. - Aplicação de Redes Locais de CLPs no Chão da Fábrica (níveis 1 e 2) : Um Passo Fundamental Rumo ao CIM - *Automação & Indústria* - Ago. 1989 - p. 14-18.
- [Coul90a] COULOURIS, G. F.; DOLLIMORE, J. - *Distributed Systems : Concepts and Design* - Addison-Wesley Publishing Company - 1990.
- [Cour91a] COURTOIS, A.; et al. - *Gestão da Produção* - Lidel Ed. Técnicas - 1991.
- [Cruz__a] CRUZ, F. W.; CRUZ, J. - Uma Abordagem para a Gerência de Produção em Redes Industriais - *X Simpósio Brasileiro de Redes de Computadores* - p. 121-132.
- [Data91a] DATAMATION - The Best in Client/Server Computing ! - *Datamation* - Out. 1991 - p. 1-24.
- [Derf92a] DERFLER Jr., F. J. - *PC Magazine Guide to Connectivity / Second Edition* - Ziff-Davis Pres - 1992.
- [Fros89a] FROST & S. - *Computer Integrated Manufacturing* - Frost & Sullivan, Inc. - 1989.
- [Gere__a] GERELLE, E. G. R.; STARK, J. - *Integrated Manufacturing : Strategy, Planning and Implementation* - McGraw-Hill Manufacturing and Systems Engineering Series.
- [Gioz86a] GIOZZA, W. F.; et al. - *Redes Locais de Computadores : Tecnologia e Aplicações* - McGraw-Hill - 1986.
- [Gioz86b] GIOZZA, W. F.; et al. - *Redes Locais de Computadores : Protocolos de Alto Nível e Avaliação de Desempenho* - McGraw-Hill - 1986.
- [Gomi86a] GOMIDE, F. A. D. C.; NETTO, M. L. D. A. - Introdução a Automação Industrial Informatizada - *I Escola Brasileiro-Argentina de Informática (EBAI)* - 1986.
- [Gril__a] GRILLO, R. C. M. - Pré-requisitos para a Automação da Pequena e Média Empresa - *Automação & Indústria* - p. 18-19.
- [Hard88a] HARDING, B. - Sharing Design Data Through Networking - *Computer Design* - 15 Out. 1988 - p. 50-63.
- [Jeee86a] IEEE C. S. - XI Conference on Local Computer Networks - *IEEE Computer Society* - 1986.
- [Lee_92a] LEE, R.; OUDERKIRK, C. - Integrating CAD/CAM and Database Technology : The AutoFACT Demo - *Novell Research* - Mar. 1992 - p. 33-43.
- [Macc90a] MACCANN, J. T. - *NetWare Programmer's Guide* - Prentice-Hall - 1990.

- [Maga__a] MAGALHÃES, M.; MENDES, M. D. J. - Redes Locais Industriais e o Projeto de Padronização MAP/TOP - *SBA : Controle & Automação* - Vol. 2 - No. 1 - p. 56-70.
- [Mend86a] MENDES, M. D. J. - Redes Locais de Comunicação em Ambiente Industrial - *VI Seminário de Comando Numérico no Brasil* - UNICAMP - 1986 - p. 27.
- [Mend88a] MENDES, M. D. J. - Redes Locais na Comunicação Fabril - *Automação & Indústria* - Dez. 1988 - p. 14-17.
- [Mile87a] MILENKOVIC, M. - *Operating System / Concepts and Design* - McGraw-Hill - 1987.
- [Mill86a] MILLER, R. K. - *Strategic Planning for Computer Integrated Manufacturing* - SEAI Technical Publications - 1986.
- [Mull89a] MULLENDER, S. - *Distributed Systems* - ACM Press Addison-Wesley Publishing Company - 1989.
- [Nanc90a] NANCE, B. - *Network Programing in C* - QUE Corporation - 1990.
- [Nobr84a] NOBRE Fo.; W. - A Automação Industrial Baseada na Tecnologia CAD/CAM - *Mundo Mecânico* - Set. 1984 - p. 34-36 - Out. 1984 - p. 20-23.
- [Nove91a] NOVELL - NetWare Concepts Reference - *Novell Inc.* - 1991.
- [Nove91b] NOVELL - Developing Client-Server NLMs - *Novell Inc.* - 1991.
- [Oliv90a] OLIVEIRA, F. J. D.; RIVERA, R. R. M. - *Redes Locais nas Empresas* - Ed. Campus - 1990.
- [Open92a] OPEN O. R. - Open Systems OLTP Monitors : Abstract - *Open OLTP Report* - Vol. 3 - No. 2 - 1992 - p. 5-14.
- [Orfa91a] ORFALI, R.; et al. - TxE : An Enabler For Client-Server OLTP - *IBM Personal Systems Developer* - 1991 - p. 25-31.
- [Pime90a] PIMENTEL, J. R. - *Communication Networks for Manufacturing* - Prentice-Hall - 1990.
- [Rese91a] RESENDE, M. O.; SACOMANO, J. B. - Princípios dos Sistemas de Planejamento e Controle da Produção - *USP-EESC 059/91* - Mar. 1991.
- [Ripp93a] RIPPS, D. L. - *Guia de Implementação para Programação em Tempo Real* - Ed. Campus - 1993.
- [Rosa__a] ROSA, P. F.; STIUBIENER, S. - Conceitos de Engenharia de Protocolos Utilizados na Implementação de um Protocolo de Aplicação em Ambiente de Manufatura - *X Simpósio Brasileiro de Redes de Computadores* - p. 51-60.

- [Rose89a] ROSE, C. - Client-Server Computing for the Sophisticated Lan - *Lan Technology* - Dez. 1989 - p. 50-54.
- [Roze89a] ROZENFELD, H.; TAKAHASHI, S. - Gerenciamento da Manufatura Integrada por Computador - *Automação & Indústria* - Out. 1989 - p. 18-21.
- [Rugg83a] RUGGIERO, W. V.; TACHLITSKY, M. - Redes Locais de Computadores para Sistema de Tempo Real - *Boletim Scopus* - Ano 6 - No. 63 - Set. 1983.
- [Rugg83b] RUGGIERO, W. V.; ZACHI, W. - Protocolos para Serviços de Tempo Real em Redes de Computadores - *Boletim Scopus* - Ano 6 - No. 61 - Jul. 1983.
- [Stan93a] STANDISH G. I. - Open OLTP Monitors - *Standish Group International* - 1993 - p. A2-B14.
- [Stan93b] STANDISH G. I. - Client-Server Goes Business Critical : OLTP Monitors Key to Next Generation of Applications - *Standish Group International* - 1993.
- [Tane85a] TANENBAUM, A. S.; RENESSE, R. V. - Distributed Operating Systems - *Computing Surveys* - Vol. 17 - No. 4 - Dez. 1985 - p. 419-469.
- [Tane88a] TANENBAUM, A. S. - *Computer Networks / Second Edition* - Prentice-Hall - 1988.
- [Tane92a] TANENBAUM, A. S. - *Modern Operating Systems* - Prentice-Hall - 1992.
- [Taro86a] TAROUCO, L. M. R. - *Redes de Computadores Locais e de Longa Distância* - McGraw-Hill - 1986.

APÊNDICE A. AMBIENTE NOVELL NETWARE

O ambiente Novell NetWare possui um projeto modular, permitindo facilmente, carregar, descarregar e gerenciar as aplicações e componentes do sistema operacional. Esta arquitetura modular permite que as aplicações sejam ligadas diretamente ao sistema operacional.

Esse ambiente consiste de um núcleo de sistema operacional e de uma coleção de módulos runtimes NetWare Loadable Modules - NLMs, que correspondem a aplicações com a finalidade de estender os serviços centrais do sistema operacional. Justamente como placas de circuitos podem ser instaladas no barramento de hardware para estender a funcionalidade de um computador, pode-se carregar NLMs em um barramento de software (fig. A.01) para estender a funcionalidade do NetWare [Nove91b].

Fig. A.01 - Arquitetura modular Novell NetWare

O NetWare é um sistema operacional não preemptivo, logo, os processos são executados até terminarem ou repassarem o controle para outro processo. Os processos são escalonados de acordo com níveis de prioridade e executados até serem completados. Com isso, múltiplos processos esperando ser executados são escalonados baseadas em outros níveis de prioridades. Frequentemente, se um processo com baixa prioridade estiver sendo executado e um novo processo for escalonado com alta prioridade, o novo processo não poderá interromper o processo corrente.

Devido ao NetWare não restringir a execução de processos, impossibilitando limites de tempo ou controle de interrupção pelo processador, cada processo é responsável pela sua auto-condução. Logo, para obter-se as vantagens do projeto 'multithreading' do NetWare, todo processo deverá ser executado rapidamente e repassar o controle do processador para o próximo processo.

A.1. AMBIENTE 'MULTITHREADING'

O ambiente 'multithreading' do NetWare suporta aplicações distribuídas, sob a forma de vários processos. Tipicamente, um NLM cliente-servidor corresponde a um processo, composto de vários mini-processos, chamados de 'threads' de controle. A primeira 'thread' é reservada para a função principal, e as demais, para o provimento de diferentes serviços. Isto aloca o NLM a servir múltiplos clientes concorrentemente. Adicionalmente, os NLMs sempre estabelecem 'threads' especializadas, que podem ser utilizadas, por exemplo, para acessar comandos do console do servidor.

Uma 'thread' retém o controle do processador até repassar o controle. Nenhuma outra 'thread' pode interferir na 'thread' corrente, mesmo que outra 'thread' tenha um nível de prioridade mais alto. Uma 'thread' pode repassar implicitamente o controle quando chamar alguma função que a suspenda, como por exemplo, uma função que espere uma entrada do teclado. As 'threads' podem também chamar funções que explicitamente repassem o controle para a próxima 'thread' lida.

Utilizando múltiplas 'threads' tem-se muitas vantagens, como :

- *Simplificação do código pela modularização* : Separando as aplicações em 'threads', estas tornam-se mais legíveis, sendo de mais fácil manutenção e atualização. 'Multithreading' permite ao usuário a manipulação do chaveamento lógico dos processos;
- *Ganho de performance* : A execução das aplicações pode ser separada do fluxo de execução principal, reduzindo o tráfego de execução. Por exemplo, uma aplicação de base de dados trabalhará mais eficientemente se a atualização dos registros ocorrer em 'background';
- *Serviços distribuídos igualmente* : Com múltiplas 'threads', nenhuma estação requisitante de um serviço poderá monopolizar a aplicação. Todas as estações poderão requisitar igualmente os serviços;
- *Desenvolvimento de múltiplos contextos através de grupos de 'threads'* : Um grupo de 'threads' consiste de uma ou mais 'threads' definidas pelo programador. 'Threads' do mesmo grupo compartilham muitos recursos similares, como, o diretório de trabalho corrente e a conexão corrente.

'Threads' agrupadas permitem compartilhar três níveis de contexto :

- *Contexto de funções relacionadas* : Inclui os itens que somente uma única 'thread' irá acessar, como, códigos de erro e ponteiros para pilhas;
- *Contexto corrente* : Inclui os itens que afetam todas as 'threads' de um único grupo de 'threads', como, diretório de trabalho corrente, conexão corrente e tela corrente;
- *Contexto de propriedade* : Inclui os itens que afetam todas as 'threads' em um único NLM, como, memória, arquivos abertos, funções definidas e vetores de argumentos passados para a função principal.

Diferente do ambiente DOS, onde todas as aplicações compartilham um único diretório de trabalho corrente para cada 'drive' DOS, o ambiente NetWare aloca cada grupo de 'threads' a terem o seu diretório de trabalho corrente. Não existe a noção de 'drive' em um ambiente de

NLMs. Instantaneamente, o diretório de trabalho corrente diferencia o servidor e volume como partes de um caminho.

Uma 'thread' se encontra sempre em um de três estados : em execução, esperando na fila de execução, ou bloqueada aguardando um evento do sistema. Enquanto uma 'thread' está em execução, ela tem o controle do processador.

Se uma 'thread' estiver na fila de execução, gerenciada pelo sistema operacional, ela esperará o controle do processador. A fila de execução é ordenada por níveis de prioridade. 'Threads' com mais alto nível de prioridade são executadas antes que 'threads' com mais baixo nível de prioridade.

Uma 'thread' bloqueada não pode conhecer-se, devendo ser conhecida por outras 'threads' ou por alguma rotina com eventos do sistema. Uma vez conhecida, a 'thread' vai para a fila de execução atrás de outras 'threads' com mesmo nível de prioridade. Tipicamente, uma 'thread' bloqueada está aguardando por eventos do sistema que venham a ocorrer.

A.2. DESENVOLVIMENTO DE NLMs

Os NLMs podem estender a funcionalidade do NetWare oferecendo muitos serviços, como serviços de impressão e comunicação, serviços de protocolos, serviços de redes locais e 'device drivers', chamando diretamente rotinas internas do sistema operacional, como se os NLMs estivessem no 'firmware' do sistema operacional. Devido aos NLMs chamarem diretamente o sistema operacional, estes não precisam utilizar os serviços de protocolos, como o NetWare Core Protocol - NCP para comunicação com o sistema operacional.

Os NLMs aumentam a flexibilidade do NetWare, podendo :

- Serem carregados e descarregados quando preciso;
- Alocar e desalocar memória quando preciso durante a carga, descarga e tempo de execução;
- Ligar outros NLMs e acessar o sistema operacional.

Os NLMs são agrupados em quatro categorias, cada uma definida por sua função :

- Módulos para serviços e aplicações;
- Módulos LAN 'drivers';
- Módulos 'device drivers';
- Módulos para nomenclatura de sistema de arquivos.

A criação de um NLM é similar ao desenvolvimento de uma outra aplicação para algum sistema operacional. Desenvolvendo NLMs dispõe-se das vantagens do sistema operacional NetWare e do processador de 32 bits.

Geralmente, no desenvolvimento de um NLM, tem-se os passos :

- Planejar como as 'threads' de execução do NLM serão executadas. Todos NLMs devem ter no mínimo uma 'thread' contida em um grupo de 'threads' acomodadas em uma função principal. Pode-se configurar outras 'threads' para acompanhar tarefas diferentes;
- Organizar 'threads' em grupos;

- Escrever o código fonte do NLM como uma aplicação servidora, utilizando as regras para NLMs em ambiente não preemptivo.

Devido aos NLMs serem executados em um ambiente não preemptivo, estes deverão se auto-conduzir seguindo um conjunto de regras. Estas regras definem que um NLM não pode alterar a operação de outros NLMs :

- NLMs não podem tomar o controle do processador por períodos indefinidos de tempo. NLMs devem, ou completar rapidamente a requisição, para repassar o controle (como requisições de I/O), ou repassar explicitamente o controle;
- NLMs deverão referenciar somente a memória que for alocada para si. Se propriamente coordenados, NLMs podem ler memória alocada por outros NLMs. Contudo, um NLM não poderá escrever na memória alocada por outros NLMs, a menos que estes outros NLMs fiquem esperando uma operação. Se um NLM tentar acessar memória que não exista, o servidor interromperá a execução, "derrubando" o sistema;
- Um NLM não poderá chamar uma função que não reconheça os parâmetros requeridos e valores retornados;
- NLMs não podem alterar o estado do processador (por exemplo, chaveamento para modo real). O NetWare é executado somente em modo protegido. Contudo, se um NLM chavear o processador para modo real, o servidor "derrubará" o sistema.

Os NLMs são responsáveis pela liberação dos recursos por ele alocados, como memória, 'sockets', telas, 'devices', entre outros. Um NLM deve controlar o seu processo de término, incluindo mecanismos que manipulem :

- Descarregamento, quando o NLM for descarregado por linha de comando no console do sistema;
- Término normal, quando o NLM terminar;
- Término anormal, quando uma causa de erro inesperado ocasionar um 'shutdown'.

O ambiente não preemptivo requer mais tempo na fase de projeto, porém retornando uma maior rapidez na execução de NLMs com um ganho de performance do sistema.

A.3. DESENVOLVIMENTO DE APLICAÇÕES CLIENTE-SERVIDOR

Geralmente o modelo cliente-servidor do NetWare oferece estas regras :

- A aplicação servidora reside no computador servidor como um NLM;
- A aplicação servidora configura os serviços, inicializa variáveis e aguarda por requisições. O servidor não tenta encontrar os clientes;
- Quando uma aplicação cliente precisa de um serviço, esta procura o servidor e então requisita o serviço;
- A comunicação cliente-servidor ocorre como um par requisição/resposta, com o cliente enviando a requisição.

A aplicação servidora oferece serviços a múltiplos clientes. Similarmente, em uma configuração mais complexa, cada cliente acessa mais que um tipo de aplicação servidora para completar sua tarefa.

A utilização de múltiplos servidores pode ampliar a performance, otimizando a utilização dos recursos da rede. Por exemplo, em um determinado instante, um servidor pode ser habilitado a calcular mais rapidamente que em outros servidores.

Quando projeta-se uma aplicação cliente-servidor, deve-se considerar vários itens, como, divisão de cargas de trabalho e métodos de comunicação. Geralmente, tem-se os passos para o projeto de uma aplicação cliente-servidor :

- Divisão dos módulos da aplicação em tarefas baseadas em clientes e em tarefas baseadas em servidores. As tarefas por estarem próximas do computador, podem aumentar a eficiência das aplicações. Por exemplo, em um determinado instante, as operações de I/O no servidor, estão próximas dos dados com as operações;
- Escalabilidade é um importante fator no projeto de aplicações cliente-servidor. Por exemplo, se o processador for sobrecarregado com operações intensivas, as aplicações conviverão melhor com dez usuários do que mil usuários;
- Observação de um método e maneira de comunicação entre aplicações cliente e servidora. Deve-se considerar a performance característica das interfaces e protocolos. Uma boa regra prática é observar a interface de mais baixo nível possível com o compromisso significativo do nível de esforço necessário para implementá-la.

Para limitar o número de clientes, o NLM, como servidor, deverá determinar o número máximo de clientes a obterem acesso, considerando-se fatores, como :

- *Considerações de memória* : Se as aplicações clientes requisitarem ao servidor uma grande área de memória, deve-se limitar o número de clientes para evitar aplicações executadas fora da região de memória;
- *Considerações de performance* : O servidor será mais eficiente, quanto mais clientes este puder suportar. Por outro lado, se as tarefas dos clientes proporcionarem uma redução na performance do servidor, deve-se limitar o número de clientes para não reduzir a velocidade da rede;
- *Considerações de conexão* : O número de clientes que um NLM puder acessar é limitado pelo número de conexões que o computador servidor puder acessar. Por exemplo, se clientes comunicam-se com o servidor via 'modem', o NLM não poderá acessar mais de quatro clientes ao mesmo tempo.

A.4. CARACTERÍSTICAS DO NETWARE

O NetWare oferece seções críticas que permitem a somente uma, entre um conjunto de 'threads' concorrendo ao mesmo recurso, ter o controle deste recurso. Nesta situação, com exceção da 'thread' corrente, todas as demais são bloqueadas. Porém, a 'thread' deve deixar a seção crítica antes que outra 'thread' possa ser executada.

Uma alternativa para a utilização de seções críticas é a utilização de semáforos locais ou de rede, que limitam o acesso das 'threads' para os recursos e bloqueiam as 'threads' em filas de semáforo.

As funções para os serviços de sincronização, parte da interface para desenvolvimento de NLMs, permite às aplicações coordenarem o acesso aos arquivos da rede e outros recursos. Estes serviços estão divididos em duas categorias : 'locking' e semáforos.

'Locking' permite a uma 'thread' tornar exclusivo o acesso a um serviço relacionado a arquivos, como registros físicos ou registros lógicos. As 'threads' bloqueiam os recursos pela entrada do nome do arquivo ou pela localização do registro em uma tabela, através do seu tamanho. Normalmente, uma 'thread' acessa um grupo de registros e então os bloqueia como se bloqueasse um conjunto. Deve-se tomar a precaução de não ocorrer 'deadlocks', ou seja, uma 'thread' passa o controle a uma outra 'thread' que tenta acessar os recursos bloqueados pela 'thread' anterior. O NetWare utiliza o Btrieve como padrão de método para acesso a dados.

Enquanto o 'locking' permite a somente uma 'thread' acessar recursos relacionados a arquivos, os semáforos limitam a um número configurável, as 'threads' que possam acessar os demais recursos da rede. Pode-se utilizar semáforos para limitar o número de usuários a um recurso particular.

Os semáforos podem ser associados com vários recursos como arquivos, filas de execução, regiões de memória e 'devices'. Existem dois tipos de semáforos :

- *Semáforos de rede* : Aplicados nos recursos disponíveis servidores e estações da rede;
- *Semáforos locais* : Aplicados nos recursos disponíveis para somente um único servidor.

Se um NLM utiliza recursos que podem ser utilizados por outras 'threads' executadas em outras estações ou servidores da rede, deve-se utilizar semáforos de rede. Se semáforos forem utilizados para a comunicação entre 'threads' de um mesmo NLM, pode-se utilizar semáforos locais. Semáforos locais são mais rápidos que semáforos de rede devido a simplicidade e facilidade que um NLM tem para encontra-lo e implementa-lo.

O NetWare associa um único número de conexão para cada cliente a acessar o servidor. Esses números são associados sequencialmente, começando com a conexão número um. Enquanto os clientes mantêm uma conexão aberta com o servidor, o mesmo número de conexão é mantido para cada cliente. Quando um cliente encerra a conexão, o seu número de conexão é liberado para um banco de conexões disponíveis. O servidor pode utilizar outro método para identificação dos clientes, o número do endereçamento de rede de cada cliente.

No NetWare, uma tarefa é uma aplicação em execução na estação ou no servidor. Um número de tarefa é utilizado para identificar cada tarefa. Os números de tarefas são associados sequencialmente, começando com a tarefa número um. A combinação do número de conexão e do número de tarefa estabelece um único par número de conexão/tarefa, que é único para cada tarefa existente. Esse número é utilizado para gerenciar os recursos da rede. Para um NLM poder acessar recursos de outros NLMs, ou de um cliente, este precisará especificar o número de conexão/tarefa quando montar sua requisição.

O número de conexão/tarefa permite :

- *Segurança* : Quando um NLM requisita um serviço, o número de conexão/tarefa verifica qual cliente ou NLM tem o direito de acesso permitido;
- *Contabilidade* : Em um sistema onde as formas de contabilidade estão instaladas, o preço de uma requisição de serviço é repassado ao cliente ou NLM baseado no número de conexão/tarefa;
- *Gerenciamento de recursos* : Quando uma conexão é encerrada, os recursos alocados pelo servidor são liberados através do número de conexão/tarefa.

Quando uma aplicação cliente necessita de um serviço particular, esta primeiro tenta localizar o fornecedor do serviço. O cliente pode utilizar vários métodos para localização do fornecedor do serviço :

- *NetWare 'bindery'* : Todo servidor NetWare contém um 'bindery', formado por uma estrutura de objetos possuindo propriedades associadas contendo valores determinados, que, entre outras coisas, tem a função de listar os serviços avaliados e identificar o fornecedor dos serviços. Os clientes podem procurar este 'bindery' para encontrar serviços;
- *Protocolo de Divulgação de Serviço - SAP* : Os NLMs podem utilizar serviços SAP para divulgar sua presença na rede. Adicionalmente, os clientes podem consultar serviços através de mensagens 'broadcasting' que especifiquem o tipo de serviço procurado, retornando como encontrar todos os servidores da rede que ofereçam este serviço;
- *Serviços de conexão* : O cliente pode utilizar as funções de comunicação da interface para desenvolvimento de NLMs para acompanhar a resolução do endereçamento de serviços pelo protocolo IPX/SPX;
- *Interface a Nível de Transporte - TLI* : O cliente pode utilizar TLI para acompanhar a resolução do endereçamento de serviços pelo protocolo utilizado, como protocolos TCP/IP e IPX/SPX;
- *'sockets'* : O cliente pode utilizar 'sockets' para acompanhar o endereçamento de serviços por TCP/IP.

O NetWare oferece um Sistema de Gerenciamento de Filas - QMS, que se utilizado, traz várias vantagens na forma de enfileiramento fornecido. O servidor é liberado da responsabilidade de enfileirar, alocando requisições de aplicações mais eficientemente.

Quando clientes necessitam de requisições de serviços, o QMS coloca a requisição em uma fila e o servidor processa o serviço. Múltiplos servidores podem acessar uma única fila de requisições. Do mesmo modo, um único servidor pode manipular várias filas. O QMS utiliza a forma de 'bindery' para representar filas e definir suas características. Cada fila é representada no 'bindery' como um objeto.

Aplicações clientes e servidoras manipulam as filas utilizando funções de chamada QMS. Chamadas QMS formam um completo conjunto de ferramentas para trabalhar com os aspectos do 'bindery'. Se uma aplicação precisar de mais controle na definição da estrutura das aplicações, pode-se utilizar chamadas QMS exclusivamente para suprir os serviços.

Geralmente, QMS trabalha melhor para aplicações que :

- Processam grandes unidades de dados e aplicações 'batch', como 'spooling' de impressão ou envio de dados para 'modem';
- Não requerem respostas imediatas, utilizando filas de requisição de aplicações na ordem FIFO, com a prontidão dos serviços variando com o tamanho da fila;
- Necessitam de flexibilidade e controle, pois o QMS pode ordenar e gerenciar aplicações por propriedades, como tamanho e prioridade, além de oferecer um ambiente operacional seguro e independente.

Com relação a alocação de memória, o NetWare oferece como vantagens :

- O limite de alocação de memória para um NLM está relacionado somente com a memória que o servidor ocupa. O NetWare precisa aproximadamente de 2 MB de memória para a base do sistema operacional, enquanto que o remanescente de memória forma um banco de 'buffers' para 'cache'. O servidor disponibiliza estes 'buffers' para as alocações dos NLMs;
- A interface para desenvolvimento de NLMs utiliza o modelo 'flat memory', que opera como se não houvesse segmentos. Todo o código e os dados residem no mesmo segmento. O código desenvolvido torna transparente a aparente segmentação de memória tipicamente associada aos processadores Intel. Devido a remoção do registrador de segmentos do código fonte, o modelo 'flat memory' resulta em aplicações mais rápidas com o endereçamento de memória podendo atingir os 4 GB. O modelo 'flat memory' permite a utilização de estruturas de dados maiores, removendo o obstáculo das limitações de segmentos;
- Quando do término de um NLM, todo o bloco estático de memória alocado por ele é liberado automaticamente. Se não ocorrer a liberação de algum recurso alocado, uma mensagem de aviso será exibida no console.

A utilização de memória compartilhada permite múltiplas 'threads' se comunicarem. Para compartilhar memória entre 'threads' de um mesmo NLM, deve-se utilizar ponteiros globais ou estáticos para um único bloco de memória. Para compartilhar memória entre 'threads' de múltiplos NLMs, deve-se escrever uma função que passe um ponteiro para o endereço de memória, entre os NLMs.

APÊNDICE B. DESCRIÇÃO DOS ARQUIVOS

O sistema OLTP implementado, utiliza os arquivos descritos na *tabela B.01*.

<i>Tabela de aplicativos para comando numérico</i>	<i>Tabela de aplicativos para desenho</i>
<i>Tabela de aplicativos para folha de processo</i>	<i>Tabela de aplicativos para ordem de fabricação</i>
<i>Tabela de comandos numéricos</i>	<i>Tabela de desenhos</i>
<i>Tabela de estado das sessões clientes</i>	<i>Tabela de falhas das unidades produtivas</i>
<i>Tabela de famílias</i>	<i>Tabela de folhas de processo</i>
<i>Tabela de monitores OLTP</i>	<i>Tabela de ordens de fabricação</i>
<i>Tabela de pedidos</i>	<i>Tabela de produtos</i>
<i>Tabela de sessões clientes</i>	<i>Tabela de unidades produtivas</i>
<i>Tabela de unidades remotas</i>	

Tabela B.01 - Arquivos utilizados

As *tabelas de aplicativos para comando numérico, desenho, folha de processo e ordem de fabricação* armazenam os possíveis aplicativos utilizados para elaboração de comandos numéricos, desenhos, folhas de processo, e ordens de fabricação, respectivamente. Possuem a mesma estrutura, descrita na *tabela B.02*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>CódigoAplicativo (*)</i>	Númerico	Código do aplicativo.
<i>NomeAplicativo</i>	Texto	Nome do aplicativo.
<i>ComandoAplicativo</i>	Texto	Comando de ativação do aplicativo.
<i>DirAplicativo</i>	Texto	Diretório do parâmetro do aplicativo.
<i>ExtAplicativo</i>	Texto	Extensão do parâmetro do aplicativo.

Tabela B.02 - Estrutura das tabelas de aplicativos

A *tabela de comandos numéricos* armazena os nomes dos comandos numéricos elaborados pela unidade remota de processo - CAPP. Sua estrutura está descrita na *tabela B.03*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>NomeComNum (*)</i>	Texto	Nome do programa de comando numérico elaborado pela unidade remota de processo - CAPP.

Tabela B.03 - Estrutura da tabela de comandos numéricos

A *tabela de desenhos* armazena os nomes dos desenhos elaborados pela unidade remota de projeto - CAD. Sua estrutura está descrita na *tabela B.04*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>NomeDesenho (*)</i>	Texto	Nome do desenho elaborado pela unidade remota de projeto - CAD.

Tabela B.04 - Estrutura da tabela de desenhos

A *tabela de estado das sessões clientes* armazena os possíveis estados em que se encontram as sessões clientes. Sua estrutura está descrita na *tabela B.05*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>CódEstSesCli (*)</i>	Númerico	Código do estado da sessão cliente.
<i>NomeEstSesCli</i>	Texto	Nome do estado da sessão cliente.

Tabela B.05 - Estrutura da tabela de estado das sessões clientes

A *tabela de falhas das unidades produtivas* armazena as possíveis falhas que as unidades produtivas podem apresentar. Sua estrutura está descrita na *tabela B.06*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>CódFalUniPro (*)</i>	Númerico	Código da falha da unidade produtiva.
<i>NomeFalUniPro</i>	Texto	Nome da falha da unidade produtiva.

Tabela B.06 - Estrutura da tabela de falhas das unidades produtivas

A *tabela de famílias* armazena as famílias que podem relacionar os pedidos com as unidades produtivas. Sua estrutura está descrita na *tabela B.07*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>CódigoFamilia (*)</i>	Númerico	Código da família.
<i>NomeFamilia</i>	Texto	Nome da família.

Tabela B.07 - Estrutura da tabela de famílias

A *tabela de folhas de processo* armazena os nomes das folhas de processo elaboradas pela unidade remota de processo - CAPP. Sua estrutura está descrita na *tabela B.08*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>NomeFolPro (*)</i>	Texto	Nome da folha de processo elaborada pela unidade remota de processo - CAPP.

Tabela B.08 - Estrutura da tabela de folhas de processo

A *tabela de monitores OLTP* armazena os monitores OLTP que podem gerenciar todo o sistema. Sua estrutura está descrita na *tabela B.09*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>CódMonOLTP (*)</i>	Númerico	Código do monitor OLTP.
<i>NomeMonOLTP</i>	Texto	Nome do monitor OLTP.

Tabela B.09 - Estrutura da tabela de monitores OLTP

A *tabela de ordens de fabricação* armazena os nomes das ordens de fabricação elaboradas pela unidade remota de planejamento - PCP. Sua estrutura está descrita na *tabela B.10*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>NomeOrdFab (*)</i>	Texto	Nome da ordem de fabricação elaborada pela unidade remota de planejamento - PCP.

Tabela B.10 - Estrutura da tabela de ordens de fabricação

A *tabela de pedidos* armazena os pedidos a serem executados. Sua estrutura está descrita na *tabela B.11*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>CódigoPedido (*)</i>	Númerico	Código do monitor OLTP.
<i>NomePedido</i>	Texto	Nome do monitor OLTP.
<i>CódigoFamilia</i>	Númerico	Código da família.
<i>CódigoProduto</i>	Númerico	Código do produto.
<i>CódUniPro</i>	Númerico	Código da unidade produtiva.
<i>DataInício</i>	Data	Data do início do pedido.
<i>DataPrevista</i>	Data	Data prevista do pedido.
<i>DataFinal</i>	Data	Data final do pedido.
<i>HoraInício</i>	Data	Hora do início do pedido.
<i>HoraFinal</i>	Data	Hora final do pedido.
<i>FlagConclusão</i>	Lógico	Flag de conclusão do pedido.
<i>FlagDecisão</i>	Lógico	Flag de passagem pela unidade remota de tomada de decisões.
<i>CódigoDecisão</i>	Númerico	Código da unidade remota de tomada de decisões.
<i>NomeDecisão</i>	Texto	Nome da unidade remota de tomada de decisões.
<i>HoraDecisão</i>	Data	Hora de liberação da unidade remota de tomada de decisões.
<i>FlagProjeto</i>	Lógico	Flag de passagem pela unidade remota de projeto - CAD.
<i>CódigoProjeto</i>	Númerico	Código da unidade remota de projeto - CAD.
<i>NomeProjeto</i>	Texto	Nome da unidade remota de projeto - CAD.
<i>HoraProjeto</i>	Data	Hora de liberação da unidade remota de projeto - CAD.
<i>FlagProcesso</i>	Lógico	Flag de passagem pela unidade remota de processo - CAPP.
<i>CódigoProcesso</i>	Númerico	Código da unidade remota de processo - CAPP.
<i>NomeProcesso</i>	Texto	Nome da unidade remota de processo - CAPP.
<i>HoraProcesso</i>	Data	Hora de liberação da unidade remota de processo - CAPP.
<i>FlagPlanejamento</i>	Lógico	Flag de passagem pela unidade remota de planejamento - PCP.
<i>CódigoPlanejamento</i>	Númerico	Código da unidade remota de planejamento - PCP.
<i>NomePlanejamento</i>	Texto	Nome da unidade remota de planejamento - PCP.
<i>HoraPlanejamento</i>	Data	Hora de liberação da unidade remota de planejamento - PCP.
<i>FlagProdução</i>	Lógico	Flag de passagem pela unidade remota de produção - CAM.
<i>CódigoProdução</i>	Númerico	Código da unidade remota de produção - CAM.
<i>NomeProdução</i>	Texto	Nome da unidade remota de produção - CAM.
<i>HoraProdução</i>	Data	Hora de liberação da unidade remota de produção - CAM.
<i>CódAplDes</i>	Númerico	Código do aplicativo para elaboração do desenho.
<i>NomeDes</i>	Texto	Nome do desenho elaborado na unidade remota de projeto - CAD.
<i>CódAplFolPro</i>	Númerico	Código do aplicativo para elaboração da folha de processo.
<i>NomeFolPro</i>	Texto	Nome da folha de processo elaborada pela unidade remota de processo - CAPP.
<i>CódAplComNum</i>	Númerico	Código do aplicativo para elaboração do programa de comando numérico.
<i>NomeComNum</i>	Texto	Nome do programa de comando numérico elaborado pela unidade remota de processo - CAPP.
<i>CódAplOrdFab</i>	Númerico	Código do aplicativo para elaboração da ordem de fabricação.
<i>NomeAplOrdFab</i>	Texto	Nome da ordem de fabricação elaborada pela unidade remota de planejamento - PCP.
<i>FlagProProdução</i>	Lógico	Flag de elaboração do processo de produção da unidade remota de produção - CAM.

Tabela B.11 - Estrutura da tabela de pedidos

A *tabela de produtos* armazena os produtos padrões que podem servir de base para os pedidos. Sua estrutura está descrita na *tabela B.12*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>CódigoProduto (*)</i>	Númerico	Código do produto.
<i>NomeProduto</i>	Texto	Nome do produto.

Tabela B.12 - Estrutura da tabela de produtos

A *tabela de sessões clientes* armazena as sessões clientes que podem se conectar ao monitor OLTP. Sua estrutura está descrita na *tabela B.13*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>CódSesCli (*)</i>	Númérico	Código da sessão cliente.
<i>NomeSesCli</i>	Texto	Nome da sessão cliente.
<i>CódMonOLTP</i>	Númérico	Código do monitor OLTP.
<i>CódUniRem</i>	Númérico	Código da unidade remota.
<i>CódEstSesCli</i>	Númérico	Código do estado da sessão cliente.
<i>FlagConexão</i>	Lógico	Flag de conexão com o monitor OLTP.
<i>HoraInicioConexão</i>	Data	Horário do início da conexão.
<i>HoraFimConexão</i>	Data	Horário do fim da conexão.

Tabela B.13 - Estrutura da tabela de sessões clientes

A *tabela de unidades produtivas* armazena as unidades produtivas que podem executar o processo de produção dos pedidos relacionados com suas famílias. Sua estrutura está descrita na *tabela B.14*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>CódUniPro (*)</i>	Númérico	Código da unidade produtiva.
<i>NomeUniPro</i>	Texto	Nome da unidade produtiva.
<i>CódigoFamilia</i>	Númérico	Código da família.
<i>CódFalUniPro</i>	Númérico	Código da falha da unidade produtiva.
<i>FlagAtivação</i>	Lógico	Flag de ativação.
<i>DataInicioAtivação</i>	Data	Data do início da ativação.
<i>DataFinalAtivação</i>	Data	Data final da ativação.
<i>HoraInicioAtivação</i>	Data	Hora do início da ativação.
<i>HoraFinalAtivação</i>	Data	Hora final da ativação.
<i>FlagDispon</i>	Lógico	Flag de disponibilidade.
<i>DataInicioDispon</i>	Data	Data do início da disponibilidade.
<i>DataFinalDispon</i>	Data	Data final da disponibilidade.
<i>HoraInicioDispon</i>	Data	Hora do início da disponibilidade.
<i>HoraFinalDispon</i>	Data	Hora final da disponibilidade.
<i>FlagManut</i>	Lógico	Flag de manutenção.
<i>DataInicioManut</i>	Data	Data do início da manutenção.
<i>DataFinalManut</i>	Data	Data final da manutenção.
<i>HoraInicioManut</i>	Data	Hora do início da manutenção.
<i>HoraFinalManut</i>	Data	Hora final da manutenção.
<i>FlagFalha</i>	Lógico	Flag de falha.
<i>DataInicioFalha</i>	Data	Data do início da falha.
<i>DataFinalFalha</i>	Data	Data final da falha.
<i>HoraInicioFalha</i>	Data	Hora do início da falha.
<i>HoraFinalFalha</i>	Data	Hora final da falha.

Tabela B.14 - Estrutura da tabela de unidades produtivas

A *tabela de unidades remotas* armazena as unidades remotas que representam os segmentos de um ambiente de manufatura. Sua estrutura está descrita na *tabela B.15*.

<i>Campo</i>	<i>Tipo</i>	<i>Descrição</i>
<i>CódUniRem (*)</i>	Númérico	Código da unidade remota.
<i>NomeUniRem</i>	Texto	Nome da unidade remota.

Tabela B.15 - Estrutura da tabela de unidades remotas

Os campos dos arquivos terminados com (*), representam os campos com função de chave primária nos mesmos arquivos.

O relacionamento entre os arquivos (*tela B.01*) está descrito na *tabela B.16*.

Tela B.01 - Relacionamento entre os arquivos

Arquivo pai	Relacionamento	Arquivo filho
Tabela de monitores OLIP	1 : N	Tabela de sessões clientes
Tabela de unidades remotas	1 : N	Tabela de sessões clientes
Tabela de estado das sessões clientes	1 : N	Tabela de sessões clientes
Tabela de aplicativos para comando numérico	1 : N	Tabela de pedidos
Tabela de aplicativos para desenho	1 : N	Tabela de pedidos
Tabela de aplicativos para folha de processo	1 : N	Tabela de pedidos
Tabela de aplicativos para ordem de fabricação	1 : N	Tabela de pedidos
Tabela de famílias	1 : N	Tabela de pedidos
Tabela de produtos	1 : N	Tabela de pedidos
Tabela de unidades produtivas	1 : N	Tabela de pedidos
Tabela de famílias	1 : N	Tabela de unidades produtivas
Tabela de falhas das unidades produtivas	1 : N	Tabela de unidades produtivas

Tabela B.16 - Relacionamento entre os arquivos

APÊNDICE C. ROTEIRO DE APRESENTAÇÃO

UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE CIÊNCIAS MATEMÁTICAS DE SÃO CARLOS

Unidade de Sistema GLTP no
Gerenciamento de Ambientes de
Manufatura Integrada
com o Sistema Produtivo Distribuído

Coordenador: Prof. Dr. Edson dos Santos Moraes

Desenvolvido por: Instituto de Matemática e Física, Universidade de São Paulo, como parte dos
recursos para obtenção de título de Mestre na área de
Ciência de Computação e Matemática Computacional

São Carlos - SP
1985

Slide 01

Topicos

1. Introdução
2. Integração de Manufaturas
 - 2.1. Produção, Engenharia Produtiva e Engenharia
 - 2.2. Produção da Produção Industrial
 - 2.3. Análises de Manufatura
 - 2.4. Manufatura Integrada por Computador - CIM
3. Sistemas Computacionais Distribuídos
 - 3.1. Sistemas Operacionais
 - 3.2. Sistemas Distribuídos

Slide 02

Topicos

- 3.3.1. Sistemas Cliente-Servidor
- 3.3.2. Processamento Transacional Online - OLTP
- 3.3.3. Manufatura GLTP
- 3.4. Modelos Hierárquicos de Interconexão - RM-QSI
- 3.5. Arquitetura IEEE P15
- 3.6. Redes Locais de Computadores
 - 3.6.1. Topologias
 - 3.6.2. Operações de Transmissão
 - 3.6.3. Algoritmos de Controle de Acesso ao Meio
 - 3.6.4. Protocolos de Comunicação
 - 3.6.5. Redes Industriais
- 3.7. Protocolos de Comunicação

Slide 03

Topicos

- 3.6.1. Protocolo XCPX
- 3.6.2. Protocolo TCP/IP
- 3.6.3. Protocolo MMS
4. Especificação do Sistema GLTP Proposto
 - 4.1. Estrutura Funcional do Modelo GLTP
 - 4.2. Estrutura Funcional da Aplicação Cliente
5. Apresentação do Sistema GLTP
6. Conclusões e Sugestões para Trabalhos Futuros

Slide 04

1

Introdução

Slide 05

Introdução

- Modernização industrial das empresas de manufatura
integração de computadores em praticamente todos os segmentos
- Implantação de processos automatizados
 - ✓ Projeto Assistido por Computador - CAD
 - ✓ Engenharia Assistida por Computador - CAE
 - ✓ Planejamento Assistido por Computador - CAPP
 - ✓ Manufatura Integrada por Computador - CIM
- A simples implantação destes processos e domínios técnicos
o surgimento de "ilhas isoladas de automação", não destacando a
gerenciamento do fluxo de informações entre os segmentos
- A consideração de todos os segmentos de um ambiente
manufatureiro, integrados como um conjunto harmonioso e
desenvolvido por manufatura integrada por computador - CIM

Slide 06

Introdução

- Condições básicas e principais características da produção
- Sistemas de produção: a indústria e a produção em massa
- Condições básicas de produção: a indústria e a produção em massa
- A produção em massa: a indústria e a produção em massa
- A produção em massa: a indústria e a produção em massa
- A produção em massa: a indústria e a produção em massa
- A produção em massa: a indústria e a produção em massa
- A produção em massa: a indústria e a produção em massa

Slide 07

Integração da Manufatura

Slide 08

Integração da Manufatura

- A integração da produção: a indústria e a produção em massa

Slide 09

Produção, Sistemas Produtivos e Empresas

- Produção: Processos, através dos quais, bens e serviços são produzidos
- Sistemas produtivos: a indústria e a produção em massa
- Empresas: a indústria e a produção em massa
- Classificação segundo o tipo de produção
- Produção contínua (flow shop)
- Produção descontínua (job shop)
- Produção por projeto

Slide 10

Produção, Sistemas Produtivos e Empresas

- Produção contínua

Slide 11

Produção, Sistemas Produtivos e Empresas

- Produção descontínua

Slide 12

Produção, Sistemas Produtivos e Empresas

- Produção por projeto

Slide 13

Produção, Sistemas Produtivos e Empresas

- Como utilizar processos discretos, uma vez que não são de padrão contínuo?
- **Integração de dados:** Integração e compartilhamento das informações e materiais em nível global, com o objetivo de melhorar a eficiência e a produtividade, reduzindo custos e aumentando a qualidade.
- **Variação:** Grande variedade, diminuindo os custos e aumentando a flexibilidade.
- **Convergência:** Perda da hierarquia das informações, tornando as informações mais acessíveis.
- **Classificação:** a respeito da relação com o cliente.
- **Produtos para todos:**
 - Por serem por encomenda.

Slide 14

Processo de Produção Discreta

- Informações que definem o produto das empresas de manufatura voltadas à produção discreta:
 - Lista de materiais
 - Desenhos
 - Especificações
 - Planejamento de processos
 - Planos de controle de qualidade
 - Programas de fabricação
 - Tomada de decisões de tecnologia e materiais
 - Informações sobre os materiais
 - Planos de produção
 - Ordem de fabricação
 - Programação
 - Emissão das ordens
 - Sistemas de controle
 - Planejamento e reprogramação

Slide 15

Automação da Manufatura

- **Manufatura de ponta:** Sistema que integra as diferentes informações de processos de produção que permitem a produção de peças diferentes e de alta produtividade para todos os dados de entrada, definindo assim os resultados esperados.
- **Sistema de manufatura:** pode ser entendido como o controle dos elementos básicos:
 - Equipamentos
 - Mão-de-obra (operários)
 - Energia
 - Materiais
- **Processo de automação da manufatura envolve as áreas:**
 - Coordenação
 - Projeto
 - Produção (atuação da automação final do processo)
 - Fabricação

Slide 16

Manufatura Integrada por Computador - CIM

- Cada empresa de manufatura possui uma estrutura organizacional e funcional, assim sendo, cada uma precisa definir sua própria estratégia de integração.
- O computador passa de ser a figura principal, tornando-se apenas o parceiro de trabalho para facilitar a integração.

Slide 17

Manufatura Integrada por Computador - CIM

- **Representação estrutural: Exemplo da produção de uma peça.**

Slide 18

Manufatura Integrada por Computador - CIM

- A importância da manufatura integrada por computador em um sistema de produção e planejamento.
- Os elementos básicos de um sistema de manufatura integrada:
 - ✓ Planejamento
 - ✓ Materiais
 - ✓ Planejamento e controle de produção
 - ✓ Controle da qualidade de materiais
 - ✓ Controle de estoque
- Os elementos básicos de um sistema de manufatura integrada:
 - ✓ Planejamento
 - ✓ Materiais
 - ✓ Planejamento e controle de produção
 - ✓ Controle da qualidade de materiais
 - ✓ Controle de estoque

Slide 19

Manufatura Integrada por Computador - CIM

- Problemas e oportunidades mais encontrados na implementação:
 - ✓ Otimizar as atividades de planejamento diferentes
 - ✓ Planejamento de materiais
 - ✓ Planejamento de produção
 - ✓ Planejamento de estoque
 - ✓ Planejamento de qualidade
- A importância da manufatura integrada por computador em um sistema de produção e planejamento.

Slide 20

3

Sistemas Computacionais Distribuídos

Slide 21

Sistemas Computacionais Distribuídos

- Redes locais de computadores não são apenas locais e tem limitações, devido ao problema de controle de acesso ao sistema, falta de segurança e de confiabilidade para transmissão de dados ou transmissão de dados de dados, voz e imagem.

Slide 22

Sistemas Operacionais

- Sistema operacional: Sistema de software responsável pelo gerenciamento das recursos de um computador, capaz de fornecer interfaces eficientes e amigáveis ao usuário.
- Processos: Programa em execução, juntamente com tudo o que ele necessita de execução normal.
- Os processos estão sempre em algum dos estados:
 - ✓ Pronto para executar
 - ✓ Em execução
 - ✓ Terminado
- O processo corrente sempre possui o chamado de controle de fluxo.
- Estratégias de distribuição de processos:
 - ✓ Escalonamento preemptivo
 - ✓ Escalonamento não preemptivo

Slide 23

Sistemas Distribuídos

- Sistema distribuído: Sistema capaz de administrar e gerenciar várias unidades de processamento, sem necessidade de compartilhamento de memória e independentemente entre si, de modo que a utilização destas unidades seja totalmente transparente e insensível ao usuário.
- Vantagens:
 - ✓ Baixa custo / desempenho
 - ✓ Alta capacidade de expansão
 - ✓ Alta confiabilidade
- Modelos de sistemas distribuídos:
 - ✓ Modelo interconectado
 - ✓ Modelo cliente-servidor
 - ✓ Modelo baseado em mensagens

Slide 24

Sistemas Distribuídos

- Modelo de arquitetura

Slide 25

Sistemas Distribuídos

- Abordagem

- Características

- Compartilhamento de dados e recursos
- Comunicação entre processos
- Sistema distribuído

Slide 26

Sistemas Distribuídos

- Modelo de arquitetura

Slide 27

Sistemas Cliente-Servidor

- Sistema de cliente-servidor

Slide 28

Sistemas Cliente-Servidor

- Níveis de abstração em sistemas cliente-servidor

- Camada de aplicação
- Camada de transporte
- Camada de rede
- Camada de enlace
- Camada física

- Comunicação entre cliente e servidor

Camada	Protocolo	Função
Aplicação	HTTP, FTP, SMTP	Transferência de dados de alto nível
Transporte	TCP, UDP	Transferência de dados de médio nível
Rede	IP	Transferência de dados de baixo nível
Enlace	IEEE 802.3, IEEE 802.11	Transferência de dados de nível de enlace
Física	10BASE-T, 100BASE-T, 1000BASE-T	Transferência de dados de nível físico

Slide 29

Sistemas Cliente-Servidor

- Modelo de arquitetura

- Abordagem para a manipulação dos dados e dos componentes de aplicação em um sistema cliente-servidor

- Modelo de acesso remoto e cliente
- Modelo de acesso remoto e servidor
- Modelo de acesso remoto e rede

Slide 30

Slide 31

Slide 32

Slide 33

Slide 34

Slide 35

Slide 36

Slide 37

Slide 38

Slide 39

Slide 40

Slide 41

Slide 42

Slide 43

Slide 44

Slide 45

Slide 46

Slide 47

Slide 48

Slide 49

Slide 50

Slide 51

Slide 52

Slide 53

Slide 54

Slide 55

Slide 56

Slide 57

Slide 58

Slide 59

Slide 60

Slide 61

Slide 62

Slide 63

Slide 64

Slide 65

Slide 66

Slide 67

Slide 68

Slide 69

Slide 70