

RODOLFO MIÃO

**IMPLEMENTAÇÃO DE SISTEMAS
ERP (“Enterprise Resource Planning”) SAP R/3® E
SUAS TECNOLOGIAS *MIDDLEWARE***

Monografia apresentada à Escola
Politécnica da Universidade de São
Paulo para conclusão do Curso de
MBA em Tecnologia de Software.

Área de concentração:
Engenharia de Software

Orientador:
Prof. Dr. Carlos Hulot

São Paulo
2007

DEDALUS - Acervo - EPEL



31500020085

FICHA CATALOGRÁFICA

M2007CK

Mião, Rodolfo

Implementação de sistemas ERP ("Enterprise Resource Planning") SAP R/3® e suas Tecnologias "middleware" / R. Mião - São Paulo, 2007.

81p.

Monografia (MBA em Tecnologia de Software) – Escola Politécnica da Universidade de São Paulo. Programa de Educação Continuada em Engenharia.

1.Sistemas ERP 2.Middleware 3.Desenvolvimento de Software I.Universidade de São Paulo. Escola Politécnica. Programa de Educação Continuada em Engenharia II.t.

1825602

OK

À EDS do Brasil por disponibilizar
seus ambientes ERP's para estudos.
A meu pai Carlos e meu irmão Maurício pela ajuda.
À Simone Suguihara pelo apoio e
à Priscila Tavoraro pelas revisões.

Agradecimentos

Ao professor Carlos Hulot pelas rápidas respostas, atenção, dedicação e orientação.

À Profa. Dra. Selma Shin Shimizu Melnikoff pela motivação e especial atenção.

Aos amigos Arthur Utiyama e Christian Reis Meneguín pelas incontáveis horas que trabalhamos em grupo.

RESUMO

O objetivo deste trabalho é abordar o tema ERP (“**Enterprise Resource Planning**” ou Planejamento de Recursos do Empreendimento) que corresponde a um Sistema de Informação Integrado, dividido por vários departamentos que se comunicam e atualizam a mesma base de dados. Será utilizado o ERP de Mercado SAP R/3, demonstrando de forma sucinta os principais fatores críticos de sucesso, os fatores de insucesso, suas implicações e inter-relações a fim de oferecer uma visão mais aprofundada de um dos fatores, julgados como importante: as formas de *Middleware* do ERP e suas tecnologias como: RPC, DCOM, RMI, CORBA, RFC, EDI, **Web Services**, SOA, ESA, XML, XML para EDI e Arquivos textos (TXT). Este fator será explorado em maior profundidade em função da variedade de tecnologias que possibilitam a integração do ERP com sistemas legados, cuja seleção pode impactar no resultado de um projeto.

ABSTRACT

The objective of this work is to approach subject ERP (Enterprise Resource Planning) which corresponds to an Integrated Information System, divided for some departments that communicates and updates the same database.

It will be used a commercial ERP, SAP R/3, showing in a very succinct format the main critical factors of success, the failure factors, implications and inter-relations in order to offer a deepened vision of one factors, judged as important: the Middleware ERP's forms and tecnology like: RPC, DCOM, RMI, CORBA, RFC, EDI, Web Services, SOA, ESA, XML, XML for EDI and Text files (TXT). This factor will be explored in depth as function of the variety technologies that make possible the ERP integration with legacies systems, whose the choice can impact in the result of a project.

SUMÁRIO

1 INTRODUÇÃO	13
1.1 OBJETIVO	13
1.2 MOTIVAÇÃO	13
1.3 JUSTIFICATIVA	14
1.4 ESTRUTURA DO TRABALHO	14
2 IMPLEMENTAÇÃO DE SISTEMAS ERP	15
2.1 O SISTEMA ERP - ENTERPRISE RESOURCE PLANNING	15
2.2 A ARQUITETURA DE SISTEMAS ERP SAP R/3	19
2.3 SISTEMA ERP COMO DIFERENCIAL COMPETITIVO	20
2.4 A ESCOLHA E IMPLEMENTAÇÃO DO ERP	21
2.5 PROBLEMAS ENFRENTADOS AO IMPLEMENTAR ERP	23
2.6 FATORES CRÍTICOS DE SUCESSO	29
2.7 FASES DA IMPLEMENTAÇÃO	33
2.8 RECOMENDAÇÕES AO IMPLEMENTAR ERP	38
3 TECNOLOGIAS DISPONÍVEIS PARA MIDDLEWARE COM SISTEMAS ERP ...	42
3.1 INTRODUÇÃO	42
3.2 HISTÓRICO	43
3.3 TEXT FILES (TXT);	45
3.4 REMOTE PROCEDURE CALLS (RPC)	46
3.5 DISTRIBUTED COM (DCOM)	46
3.5.1 Visão Geral do COM	47
3.6 REMOTE METHOD INVOCATION (RMI)	51
3.7 COMMON OBJECT REQUEST BROKER ARCHITECTURE (CORBA)	53
3.8 REMOTE FUNCTION CALL (RFC)	56
3.9 EDI (ELECTRONIC DATA INTERCHANGE)	57
3.10 XML	59
3.11 XML PARA EDI	61
3.12 WEB SERVICES	62
3.13 SERVICE ORIENTED ARCHITECTURE (SOA)	63
3.14 ENTERPRISE SERVICES ARCHITECTURE (ESA)	65
4 COMPARATIVOS ENTRE AS INTERFACES	67
4.1 INTRODUÇÃO	67
4.2 CONSIDERAÇÕES AO COMPARAR	67
4.3 IMPORTÂNCIA DESTA COMPARAÇÃO	67
4.4 COMPARATIVO	68
4.4 PRÓS E CONTRAS DE CADA SOLUÇÃO	69
5 CONCLUSÃO	71
5.1 TRABALHOS FUTUROS	72
LISTA DE REFERÊNCIAS	73
ANEXO A	79
ERP ainda é desafio em algumas empresas de grande porte	79
Vale do Rio Doce estreia ERP de US\$ 55 milhões	83

LISTA DE FIGURAS

Figura 1 – Módulos do ERP SAP R/3	17
Figura 2 – Hierarquia de fila única	18
Figura 3 – Hierarquia de duas filas	18
Figura 4 – Hierarquia de três filas	18
Figura 5 – Estrutura típica de funcionamento de um sistema ERP	19
Figura 6 – Escopo funcional do ERP	20
Figura 7 – Modelo do ciclo de vida dos sistemas ERP.....	21
Figura 8 – ERP e suas principais fases.....	34
Figura 9 – Middleware.....	37
Figura 10 – Áreas Problemáticas em sistemas de Informação ERP	38
Figura 11 – Uso de sistemas ERP.....	41
Figura 12 – Text files.	44
Figura 13 – Remote Procedure Calls (RPC)	45
Figura 14 – Empacotamentos de Objetos do DCOM.....	48
Figura 15 – Arquitetura DCOM	49
Figura 16 – Arquitetura RMI	51
Figura 17 – Modelo de IDL	54
Figura 18 – IDL unificado.	55
Figura 19 – RFC como meio de acesso	56
Figura 20 – EDI.	57
Figura 21 – EDI Baseado na WEB com XML	60
Figura 22 – Web Services	61
Figura 23 – SOA	64

LISTA DE TABELAS

Tabela I – Fatores organizacionais.....	24
Tabela II – Benefícios e problemas dos sistemas ERP.....	27
Tabela III – “Doenças” mais frequentes.	30
Tabela IV – Principais mudanças tecnológicas identificadas nas organizações.....	30
Tabela V – Principais mudanças estruturais identificadas nas organizações	31
Tabela VI – Principais mudanças comportamentais identificadas nas organizações.....	32
Tabela VII – “Quantidade” de esforço para a implementação do middleware.....	67
Tabela VIII. – “Quantidade” de esforço para a implementação do middleware - cont.....	67

LISTA DE ABREVIATURAS E SIGLAS

BD - Banco de dados
COM - Component Object Model
CORBA – Common Object Request Broker Architecture
CSV - comma-separated values
DAT - Data
DCOM - Distributed Component Object Model.
EDI - Eletronic Data Interchange.
EDS – Electronic Data System.
ERP – Enterprise Resource Planning
ESA - Enterprise Services Architecture
FTP – File Transfer protocol
GUI – Graphical User Interface
HTML - HyperText Markup Language
HTTP – HyperText Transfer Protocol
HTTPS – HyperText Transfer Protocol Secure
IBM – International Business Machines Corporation
IDL - Interface Definition Language
J2EE – Java 2 Platform, Enterprise Edition
MRP – Manufacturing Resource Planning
NCP - Network Control Protocol
POS - Persistent Object Service
PPP - Point-to-Point protocol
RFC - Remote Function Call
RMI - Remote Method Invocation
RPC - Remote Procedure Calls.
SAML – Security Assertion Markup Language
SAP (Sistema) - Systeme, Anwendungen, Produkte in der Datenverarbeitung
SAP (Empresa) - Systemanalyse und Programmentwicklung
SCM – Supply Chain Management
SOA – Service-Oriented architecture
SOAP – Simple Object Access Protocol

TCP/IP - Transfer Control Protocol/Internet Protocol

TXT – Text

UDDI – Universal Description Discovery and Integration

URL – Uniform Resource Locator

WSDL – Web Services Description Language

XML – Extensible Markup Language

1 INTRODUÇÃO

1.1 OBJETIVO

O ERP é um Sistema de Informação Integrado, dividido por vários departamentos que se comunicam e atualizam a mesma base de dados conforme Corrêa H.L. (1999). O objetivo dele é focar a empresa como um todo e permitir que as informações geradas por um departamento sejam compartilhadas com os outros.

Esta monografia discorre sobre as implementações de sistemas ERP – “**Enterprise Resource Planning**” ou Planejamento de Recursos do Empreendimento, utilizando principalmente o ERP de Mercado SAP R/3. A sua importância deve-se aos pontos críticos de sucesso da implementação de sistemas ERP e aponta alguns fatores que podem levar ao insucesso nas implementações, mostrando que há muitos benefícios na adoção de um sistema ERP que, se bem implementado, tornar-se-á fundamental para a gestão das empresas modernas.

Serão ainda abordados neste trabalho as principais razões ou fatores pelos quais as empresas não alcançam os resultados a princípio esperados e apresentará um conjunto de boas práticas de planejamento e estruturação de projetos para auxiliar a minimização dos problemas enfrentados na sua implantação.

Em suma, serão tratados aqui os principais fatores críticos de sucesso de projetos ERP dentre, os quais receberá destaque o **Middleware** do ERP, mostrando inicialmente os principais mecanismos existentes para esta integração e indicando quais as melhores alternativas dentro do contexto tecnológico atual.

1.2 MOTIVAÇÃO

Muitos fornecedores de ERP trabalham com tecnologias diversas. O ERP SAP R/3 possui diversas formas de **Middleware**, ou seja, interfaces que permitem a interação entre diferentes aplicações de softwares, geralmente também sobre diferentes plataformas de hardware e infraestrutura para troca de dados. Desta forma a escolha correta da

tecnologia é de extrema importância, pois, conhecendo-se antecipadamente as diversas características de cada uma, permitirá a correta decisão.

1.3 JUSTIFICATIVA

O sistema ERP é uma ferramenta que pode trazer inúmeros benefícios para a empresa, porém deve ser implementado e utilizado de maneira correta para que possam ser eficazes inclusive a integração com outros sistemas da empresa atual ou que possam vir a surgir. O fracasso em uma implementação de grande porte, como a de um ERP, envolve dispendioso custo financeiro e esforço por parte da equipe de implementação. Para que não haja problemas posteriores, deve-se haver um bom planejamento antes de iniciar o processo de implementação de ERP.

F. Heinemann (2004) afirma que “Os softwares de ERP trabalham muito com integração devido às necessidades dos negócios”.

1.4 ESTRUTURA DO TRABALHO

O Capítulo 1 apresenta a Introdução, o Objetivo, a Justificativa e a Estrutura deste trabalho.

O Capítulo 2 expõe um histórico do surgimento do ERP, suas características, arquitetura, principais problemas na implementação, os fatores críticos de sucesso, as fases de implementação e as recomendações ao implementar o ERP.

O Capítulo 3 identifica as principais tecnologias de **Middleware** disponíveis atualmente para o sistema ERP SAP R/3, com um breve histórico sobre o surgimento das redes de computadores.

O capítulo 4 faz um comparativo entre as principais interfaces existentes no mercado abordando os prós e contras delas.

O Capítulo 5 expõe a conclusão deste trabalho, bem como, oferece sugestões para trabalhos futuros.

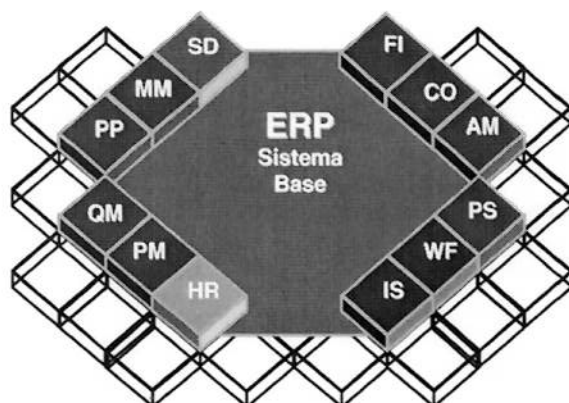
2 IMPLEMENTAÇÃO DE SISTEMAS ERP

2.1 O Sistema ERP - Enterprise Resource Planning

Durante a década de 1960 surgiu uma nova técnica de Planejamento de Pedidos de Material, a qual foi chamada de MRP – “**Manufacturing Resource Planning**” ou Planejamento de Recursos de Manufatura – e tinha por objetivo ajudar a produzir e comprar apenas o necessário e no momento certo conforme o livro Planejamento, Programação e Controle da Produção – MRP II / ERP (1999). Essa técnica permitia com base na decisão de produção dos produtos finais e suas quantidades determinar em que momento e o quê comprar, p.e. semi-acabados, componentes e matérias-primas. Esse controle terminava por reduzir a quantidade de produtos em estoques, diminuir o tempo de produção e distribuição e ainda aumentar a eficiência do processo.

Com o passar do tempo o MRP provou ser uma excelente técnica para gestão de inventários, mas não tratava as outras diferentes áreas das empresas. Isso provocou na década de 80 a evolução do sistema para o MRP II, que se diferencia do MRP na questão da decisão, pois além das decisões de o que comprar, quanto e quando, auxilia a tomada de decisões referentes a como produzir, ou seja, com quais recursos. Conseqüentemente, deixava de atender apenas as necessidades de informação referentes ao inventário de materiais para atender também às necessidades de informação para tomada de decisão gerencial.

Embora houvesse este considerável diferencial, para o funcionamento adequado do MRP II, era necessário informações de outros sistemas da organização. Com a necessidade dessa integração, surgiu o ERP – “**Enterprise Resource Planning**” ou, Planejamento de Recursos do Empreendimento. A figura 1 ilustra os módulos atuais do ERP SAP R/3.



Módulos tratados pelo ERP:

SD (**Sales and Distribution**) – Vendas e Distribuição

MM – **Material Management** – Gerenciamento de Materiais

PP – **Production Planning** – Planejamento da Produção

QM – **Quality Management** – Gerenciamento de Qualidade

PM – **Plant Maintenance** – Manutenção da Planta

HR – **Human Resources** – Recursos Humanos

FI – **Financial Accounting** – Contabilidade Financeira

CO – **Controlling** – Controle

AM – **Fixed Assets Management** – Gerenciamento de ativos fixos

PS – **Project System** – Sistema de projetos

WF – **Workflow** – Fluxo de Trabalho

IS – **Industry Solutions** – Soluções industriais

Figura 1. Módulos de um ERP (Schneider, Neureither (2005))

O ERP não atua somente no planejamento, mas também controla e fornece suporte a todos os processos operacionais, produtivos, administrativos e comerciais da empresa, facilitando o tráfego de informações dentro da organização, integrando todos os departamentos da empresa, tanto do departamento financeiro, como o departamento de Marketing ou de Recursos Humanos. Isto acontece pelo fato do ERP ser um Sistema de Informação Integrado, dividido por vários departamentos que comunicam e atualizam a mesma base de dados.

O sistema ERP SAP R/3 (**Systeme, Anwendungen, Produkte in der Datenverarbeitung** ou Sistema, Aplicações e Produtos em Processamento de Dados) surgiu em primeiro de Abril de 1972 quando cinco funcionários da IBM fundaram a divisão SAP como “**Systemanalyse und Programmentwicklung**” (Sistemas Análises e Desenvolvimento de Programas) em Mannheim na Alemanha e atualmente mantém a matriz em Walldorf, também na Alemanha, conforme livro **Web Programming with SAP Web Application Server** (2004). Essa divisão foi criada para desenvolver e vender softwares com padrão empresarial ao qual integraria todos os processos de negócios.

Fatos relevantes sobre a SAP:

- Líder global em prover softwares de soluções de negócios na arquitetura cliente/servidor.
- Quarto maior fornecedor de software independente no mundo.
- Disponível em 14 idiomas.

- 34% dos clientes da SAP no mundo possuem faturamento na casa dos US\$ 200 milhões.

O sistema ERP SAP R/3, conforme Schneider, Neureither (2005), é baseado na arquitetura cliente-servidor de três camadas nas quais as mesmas podem estar dispostas em um mesmo computador chamado de hierarquia de uma fila (**“one tier hierarchy”**), ilustrado na figura 2.

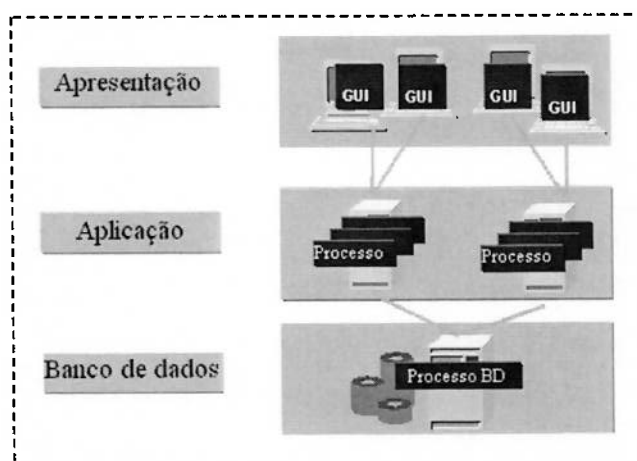


Figura 2 – Hierarquia única fila - Único computador

Dispostas em dois ou mais computadores onde a camada de apresentação ao usuário se comunica com as camadas de aplicação e de banco de dados instanciadas de forma agrupada ao qual é chamado de hierarquia de duas filas (**“two tier hierarchy”**) ilustrado na figura 3.

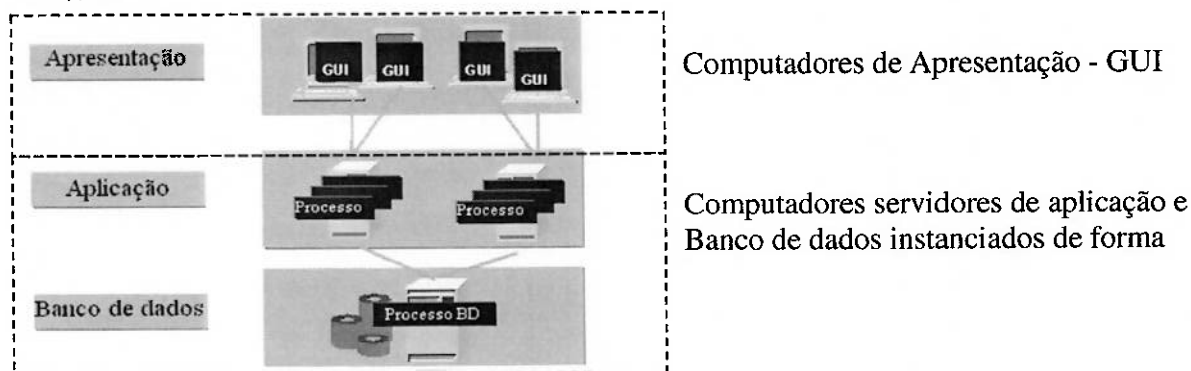


Figura 3 – Hierarquia de duas filas.

Dispostas em três ou mais computadores onde a camada de apresentação ao usuário se comunica com as camadas de aplicação e estas se comunicam com a camada de banco de

dados instanciadas de forma separada ao qual é chamado de hierarquia de três filas (**“three tier hierarchy”**), ilustrado na figura 4.

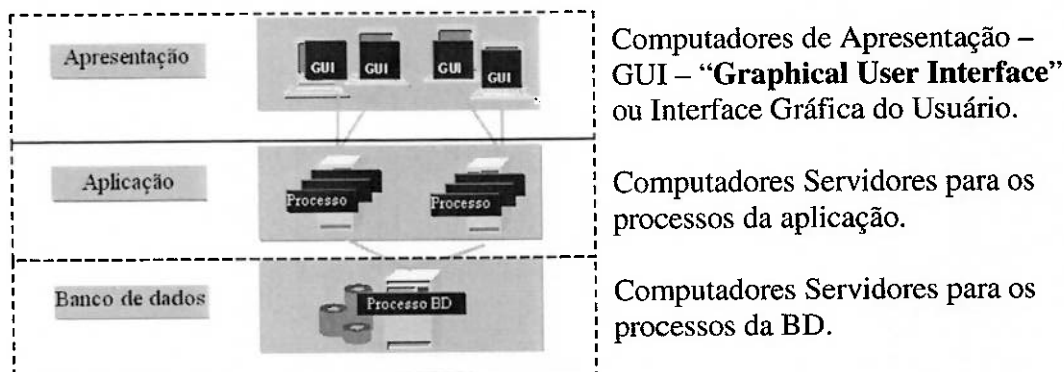


Figura 4 – Hierarquia de três filas.

Na figura 5 é representada a arquitetura repositória de sistema ERP segundo Davenport, Thomas (1998), as quais são construídas utilizando-se a arquitetura cliente-servidor.

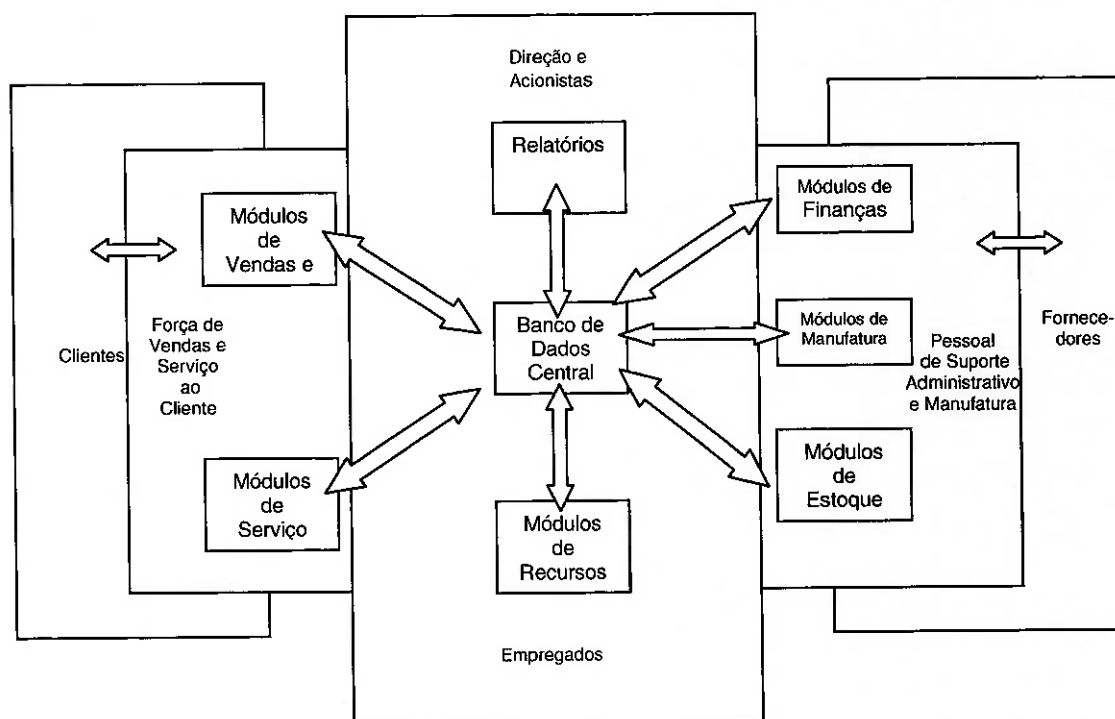


Figura 5. Estrutura típica de funcionamento de um sistema ERP segundo Davenport, Thomas (1998).

O auge dos sistemas ERP ocorreu em torno de 1998-1999 por causa do **bug** do milênio, quando diversas empresas optaram por adotá-lo como nova plataforma tecnológica, uma vez que seus sistemas legados precisariam ser adaptados segundo Cliffe, S.(1999).

Algumas das características fundamentais dos sistemas ERP são:

- a) Flexibilidade: o faz adaptar-se às mudanças no ambiente externo.
- b) Modularidade: permite utilizar um módulo independente do outro.
- c) Adaptabilidade às necessidades funcionais da empresa.
- d) Integridade das informações obtidas: todas as transações realizadas pela empresa devem ser registradas de forma centralizadas para que as consultas extraídas do sistema possam refletir ao máximo possível suas realidades operacionais, fornecendo acesso às informações em tempo real.

O sistema ERP apresenta vantagens adicionais sobre o MRPII e “(...) talvez seja esta a principal motivação de grande número de empresas que optam por adotá-lo: a integração entre as várias áreas e setores funcionais da organização, todas compartilhando uma mesma base de dados única e não redundante.” conforme Corrêa, H.L.(1999) .

2.2 A Arquitetura de Sistemas ERP SAP R/3

Davenport, Thomas (1998) divide os ERP em quatro blocos: financeiro, recursos humanos, operações e logística e vendas e marketing. Os módulos do bloco financeiro seriam divididos em contabilidade, contas a pagar, contas a receber e fluxo de caixa. Os módulos do bloco de recursos humanos seriam a folha de pagamento, gerenciamento de recursos humanos e controle de despesas de viagem. Os módulos de operações e logística seriam o gerenciamento de estoques, o MRP e o faturamento. Os módulos de vendas e marketing seriam processamento de pedidos e gerenciamento e planejamento de vendas, como ilustrado na figura 6.

ESCOPO FUNCIONAL DO ERP SAP R/3				
Financeiro	Contabilidade	Contas a pagar	Contas a receber	Fluxo de Caixa
Recursos Humanos	Folha de pagamento	Gerenciamento de RH		Controle de despesas
Operações e Logísticas	Gerenciamento de estoque	MRP		Faturamento
Vendas e Marketing	Processamento de Pedidos		Planejamento de vendas	

Figura 6 – Escopo funcional do ERP SAP R/3

2.3 Sistema ERP como diferencial competitivo

Empresas investem em sistemas que custam milhões de dólares (veja anexo A) buscando um diferencial, para depois da implementação, descobrirem que não funcionam, ou pelo menos, não atendem às suas necessidades. Um exemplo é o caso da implementação de um Sistema ERP em uma empresa do ramo de seguros atuante no Brasil que foi considerada mal-sucedida com a decisão da retirada do sistema ERP, após a identificação de uma série de problemas de diversas ordens técnicas, operacionais entre outras, segundo Souza, Cesar (2003).

O momento da aquisição de um sistema ERP é normalmente complexo para todos em uma empresa, pois essa decisão só é tomada quando a empresa percebe que os sistemas atuais não atendem as necessidades da organização, e também porque poderá envolver ajustes em procedimentos internos.

No entanto, o sistema ERP é uma solução vantajosa com um diferencial competitivo e, se a implementação for bem conduzida, pode trazer inúmeros benefícios para a empresa, pois integram a informação dentro da organização. Todavia, apesar de considerarem estas transformações necessárias e importantes, muitas organizações nem sempre se preparam adequadamente para enfrentar esse projeto, conduzindo a implementação de maneira não planejada.

A adoção de um sistema ERP exige que a empresa se reorganize tendo o foco no processo do negócio como um todo e não mais dividido por departamentos, uma vez que o sistema é integrado e as informações que são geradas por um departamento são compartilhadas por outros. Isso facilita o controle, pois estando todas as informações armazenadas no sistema, pode-se verificar o desempenho das várias áreas da empresa, além de envolver considerável análise dos processos, treinamento dos colaboradores, investimentos em informática (equipamentos) e reformulação nos métodos de trabalho e na cultura da empresa.

2.4 A escolha e implementação do ERP

As diversas etapas pelas quais passam um projeto de desenvolvimento e utilização de sistemas de informação é representada pelo ciclo de vida de sistemas de acordo com Souza; Zwicker (2000).

Abaixo (figura 7) segue a representação de um modelo específico do ciclo de vida de sistemas ERP que consideram as etapas de decisão e seleção, implementação e utilização. Vale ressaltar que esta representação não segue o rigor formal de engenharia de software.

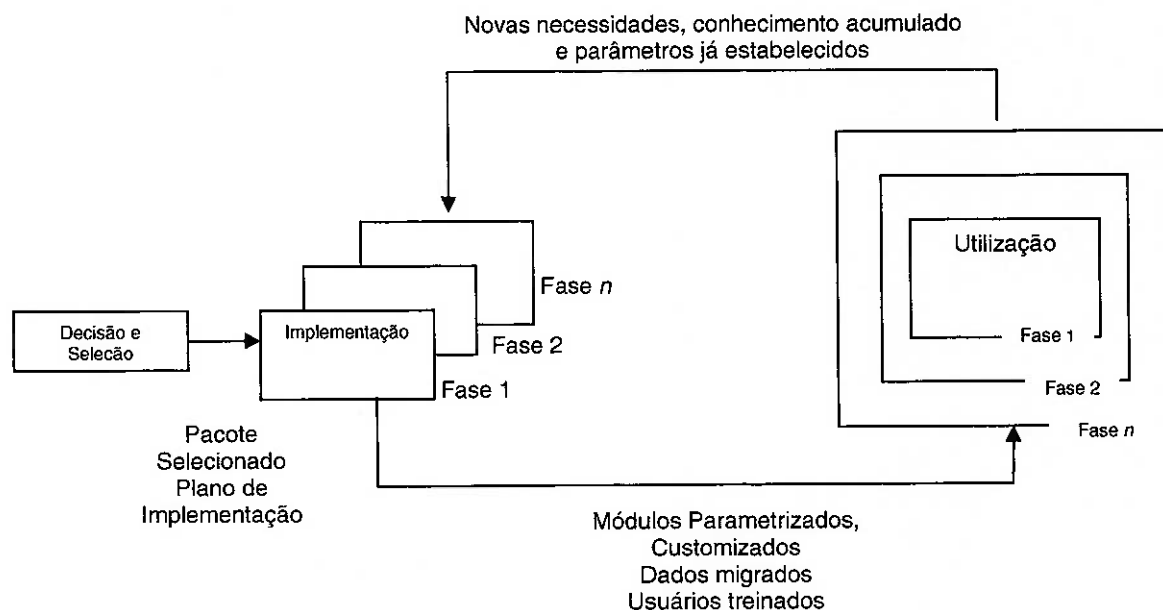


Figura 7 – Modelo do ciclo de vida dos sistemas ERP, apresentado por Souza; Zwicker (2000)

Na etapa de decisão e seleção a empresa decide utilizar um sistema ERP e procura um fornecedor. Para escolher o sistema a ser adquirido, a empresa deve estudar vários aspectos, tais como o custo de manutenção, licenciamentos futuros e atualização de software, correção de erros, capacidade de adequação às mudanças (do negócio e da legislação), assim como a própria longevidade da aplicação adquirida devem ser considerados. Também deve ser feita uma análise do ROI (**return of investment**) e de uma forma mais ampla a implementação de ERP em outras organizações do mesmo setor. A implementação é a segunda etapa do ciclo de vida de sistemas ERP, embora o termo seja utilizado para representar o ciclo de vida completo. Essa etapa pode ser definida como o processo pelo qual os módulos do sistema são colocados em funcionamento em uma empresa e envolve a adaptação dos processos de negócio ao sistema, a parametrização e modificação do sistema, a configuração de hardware e software, o treinamento de usuários e a disponibilidade de suporte.

Na etapa de utilização o sistema passa a fazer parte do dia-a-dia das operações. Essa etapa realimenta a etapa de implementação, pois durante o processo de utilização surgem novas possibilidades e necessidades que podem ser resolvidas por novos módulos, pela parametrização ou pela modificação dos módulos atuais.

A implementação de um sistema ERP é um investimento que consome muitos recursos, entre eles, tempo e dinheiro conforme matéria **Living with ERP** apresentada na revista CIO Magazine (1998). Se não houver um rigoroso controle de custos, diversos problemas podem surgir em consequência disto, levando algo que deveria oferecer uma solução a se tornar um problema. Muitas vezes ocorre problemas organizacionais durante a implementação e a utilização de sistemas ERP, sendo relativamente comum as empresas relatarem suas traumáticas experiências durante e após estes processos conforme NAH, F.F.H et al (2001).

2.5 Problemas enfrentados ao implementar ERP

A forte expansão dos ERPs na indústria de software, trouxe também vários casos de fracassos em relação às implementações. Nestas implementações mal sucedidas as empresas perderam não só o capital e tempo investidos, mas também prejudicaram o andamento dos processos de negócios.

Muitos casos ocorreram devido aos problemas relacionados à organização dos processos empresariais ou às perspectivas inadequadas levantadas pela implementação do ERP. Os principais problemas enfrentados pela empresa referiram-se às tentativas de se integrar à funções e atividades que na organização sempre foram tratadas em separado.

Uma outra situação ocorre porque a empresa busca uma padronização dos procedimentos e das normas a serem seguidas pela empresa. “Esta busca por uma cultura mais disciplinada no que diz respeito a informações, processos e sistemas, conduz a conflitos pessoais e setoriais na organização e a conseqüente fortificação de cada setor ao procurarem evidenciar suas especificidades em detrimento de uma integração maior. (...) Este tipo de situação é facilmente encontrada em empresas do setor de tecnologia da informação, em ambientes de trabalhadores do conhecimento, liberais e empreendedores.” conforme Davenport, Thomas (1998).

Davenport, Thomas (1998) ressalta também que a estratégia financeira sofre pelo menos dois impactos significativos em sua elaboração, decorrente da implementação de sistemas ERP. O primeiro é quando ocorrem os altos custos ocasionados pela decisão de investimento em uma determinada tecnologia como os sistemas ERP. O segundo ocorre quando os resultados decorrentes do bom ou mau uso destes sistemas geram provisões positivas ou negativas para o fluxo financeiro sendo necessários reposicionamento monetário de caixa.

Segundo autores Smith H. Jeff et al (2001) e Keil; Robey (2001), qualquer que seja o campo de projeto, atrasos, falhas e dúvidas muitas vezes só são comunicados à administração sênior quando já é tarde demais.

Diversos estudos indicam que 70 por cento de todos os projetos de reengenharia de processos de negócios fracassam em proporcionar os benefícios prometidos. Do mesmo

modo, uma alta porcentagem dos projetos de planejamento de recursos não é implementada totalmente e não atinge seus objetivos mesmo após três anos de trabalho segundo Cliffe, S.(1999).

Nem todos os aspectos do processo de implementação podem ser facilmente controlados ou planejados (Alter; Ginzberg(1978)). Usuários que lideram atividades de projeto costumam usar sua posição para favorecer interesses privados e conquistar poder, ao invés de promover objetivos organizacionais (Franz; Robey(1984)). Ou seja, nem sempre os usuários se envolvem nos projetos de sistemas de modo produtivo.

Enquanto alguns recebem bem o novo sistema porque percebem que a mudança introduzida será benéfica para si, outros podem considerá-la prejudicial aos seus interesses e resistir a ela (Joshi, Kailash(1991)).

Por conseguinte, a estratégia de implementação deve não apenas incentivar a participação e o envolvimento do usuário, como também abordar a questão da contra-implementação (Keen, Peter(1981)). Contra-implementação é uma estratégia deliberada para frustrar a implementação de um sistema de informação ou de uma inovação na organização.

Os principais fatores que podem ocasionar o insucesso da implementação do sistema ERP são:

- **Funcionalidade** – a empresa deve verificar se o sistema se adapta à maioria das práticas, políticas e regras da organização.
- **Resistência Organizacional** - os empregados podem acreditar que a mudança irá prejudicar o processo e comprometer seus empregos, uma vez que a implantação do ERP traz uma melhor distribuição da informação. Muitas vezes, inclusive, serão necessárias menos pessoas para desempenhar a atividade agora automatizada, criando o receio de possíveis demissões. Por se tratar de projeto complexo, as implementações de sistemas ERP necessitam de uma equipe de projeto dedicada, o que leva as organizações a enfrentarem um outro problema: o da resistência de líderes funcionais em disponibilizar seus recursos mais valiosos ao projeto, pois isso compromete suas capacidades em executar as atividades rotineiras da organização. Na tabela I são listados alguns fatores importantes na estrutura organizacional.

Fatores organizacionais no planejamento e implementação de sistemas
Participação e envolvimento do usuário
Projeto de Cargos
Monitoração de padrões e de desempenho
Ergonomia (incluindo equipamento, interface do usuário e ambiente de trabalho).
Procedimentos para resolução de queixas dos funcionários
Saúde e segurança
Cumprimento de regulamentações governamentais

Tabela 1 – Fatores organizacionais (Franz; Robey(1984).

- **Tecnologia** - desenvolver a tecnologia apropriada e integrada ao ERP requer cuidado com a integração com outros sistemas (**Middleware**), uma vez que, em alguns casos, pode-se chegar a uma situação de existir tantos sistemas paralelos, que o ambiente volte a apresentar os mesmos problemas que o ERP deveria solucionar, como inconsistência entre os dados, não obtenção das informações em tempo real com necessidade de atualizações não automáticas, etc. Deve-se levar em conta também que os sistemas ERP evoluem e assim podem passar a incorporar novas funções e informações que na versão anterior não eram necessárias, criando a possibilidade de que, a cada atualização de versão do sistema ERP a empresa tenha que analisar e em alguns casos reescrever, as aplicações de integração entre os sistemas. Será abordado mais adiante algumas das principais tecnologias de integração com o ERP, baseados em um ERP de mercado SAP R/3 e suas tecnologias envolvidas.

- **Funcionários despreparados** - Um programa de treinamento intensivo é necessário para que as pessoas entendam como utilizar o novo sistema, o que passará a ser esperado delas e como suas atitudes afetarão a organização como um todo. Sem treinamento adequado, os empregados da organização não estarão aptos a utilizar o novo sistema.

Outro problema existente durante a implementação é que, à medida que vai se aproximando o tempo determinado para o fim do projeto, a pressão aumenta e há o risco

de profissionais abandonarem o processo ao meio. Para evitar isso é preciso haver um plano de contingências.

Existem alguns pontos que são igualmente importantes e que influenciam na implementação do ERP, devendo ser considerados ao adotar o sistema. Merecem destaque:

- **Alta rotatividade** - a alta rotatividade de pessoas envolvidas na implementação dificulta algumas etapas do andamento do projeto, chegando a ser necessário em um determinado ponto de uma reciclagem completa dos usuários na utilização da ferramenta e a contratação de consultores externos para oferecer o treinamento.
- **Sobrecarga de funções por parte dos usuários** - em algumas áreas os usuários ficam sobrecarregados com o novo sistema, pois antes mesmo de completar sua implementação já ocorre corte de pessoal. Isto dificulta os testes, já que as pessoas que permanecem ficam encarregadas de fazer o seu trabalho normal e o das pessoas cortadas.
- **Falta de capacitação dos funcionários** - alguns funcionários da empresa podem apresentar limitações e não conseguir se enquadrar na nova realidade trazida com o sistema ERP. Com as atualizações de versões, também é gerado um grande problema devido à necessidade de readaptação dos funcionários às novas rotinas do sistema, bem como toda a readaptação dos módulos específicos que foram desenvolvidos para atender às necessidades de adaptações do sistema.
- **Vínculo com a empresa fornecedora do ERP** - cria-se um vínculo de dependência com a empresa fornecedora do ERP. A Empresa fica obrigada a custear o contrato de manutenção e taxas de atualização do software.
- **Necessidade constante de manutenção** - com a dinâmica dos negócios, a Empresa continua precisando de novos desenvolvimentos no sistema ERP, uma vez que nem todos os recursos necessários são contemplados pelo sistema original. Isso gera a necessidade de profissionais caros, qualificados tanto nos negócios, como na tecnologia utilizada pelo ERP. Com as atualizações de versões no próprio produto gera-se também um grande problema devido à necessidade de novos treinamentos dos funcionários às novas rotinas do sistema, bem como a de revisão de todos os módulos específicos criados

pela própria empresa e que foram desenvolvidos para atender às necessidades de adaptações do sistema.

De acordo com Corrêa, H.L.(1999) ao tomar a decisão da utilização de sistemas ERP, as empresas esperam obter diversos benefícios. Entre os benefícios apontados pelas empresas fornecedoras estão: a integração, o incremento das possibilidades de controle sobre os processos da empresa, a atualização tecnológica, a redução de custos de informática e o acesso a informações de qualidade em tempo real para a tomada de decisões sobre toda a cadeia produtiva. Entretanto há também problemas a considerar. A tabela II apresenta uma síntese que relaciona dificuldades e benefícios às características desses sistemas que são:

- **São pacotes comerciais:** Para o sistema ser um pacote comercial ele deve ser fornecido e suportado por uma empresa através de meios comerciais ao qual já se encontram disponíveis no mercado empresarial.
- **Usam modelos de processos:** A empresa fornecedora utiliza padrões, certificados ou não, para o processo de implementação e manutenção do sistema.
- **São sistemas integrados:** O sistema trabalha de forma integrada para suas funcionalidades e características.
- **Usam banco de dados corporativos:** O sistema trabalha sobre um único banco de dados corporativo.
- **Possuem grande abrangência funcional:** O sistema possui diversos módulos funcionais nativos atendendo aos diversos departamentos da empresa.

Características	Benefícios	Problemas
São pacotes comerciais	<ul style="list-style-type: none"> - redução de custos de informática; - foco na atividade principal da empresa; - redução do número de aplicações independentes; - atualização tecnológica permanente por conta do fornecedor. 	<ul style="list-style-type: none"> - dependência do fornecedor; - empresa não detém o conhecimento sobre o pacote.

Características	Benefícios	Problemas
Usam modelos de processos	<ul style="list-style-type: none"> - difunde conhecimento sobre “best practices”; - facilita a reengenharia de processos; - impõe padrões. 	<ul style="list-style-type: none"> - necessidade de adequação do pacote à empresa; - necessidade de alterar processos empresariais; - alimenta a resistência à mudança.
São sistemas integrados	<ul style="list-style-type: none"> - redução do re-trabalho e inconsistências; - redução da mão-de-obra relacionada a processos de integração de dados; - maior controle sobre a operação da empresa; - eliminação de interfaces entre sistemas isolados; - melhoria na qualidade da informação; - contribuição para gestão integrada; - otimização global dos processos da empresa. 	<ul style="list-style-type: none"> - mudança cultural da visão departamental para a de processos; - maior complexidade de gestão da implementação; - maior dificuldade na atualização do sistema, pois exige acordo entre vários departamentos; - um módulo não disponível pode interromper o funcionamento dos demais; - alimenta a resistência à mudança.
Usam banco de dados corporativos	<ul style="list-style-type: none"> - padronização de informações e conceitos; - eliminação de discrepâncias entre informações de diferentes departamentos; - melhoria na qualidade da informação; - acesso a informações para toda a empresa. 	<ul style="list-style-type: none"> - mudança cultural da visão de “dono da informação” para a de “responsável pela informação”; - mudança cultural para uma visão de disseminação de informações dos departamentos por toda a empresa; - alimenta resistência à mudança.
Possuem grande abrangência funcional	<ul style="list-style-type: none"> - eliminação da manutenção de múltiplos sistemas; - padronização de procedimentos; - redução de custos de treinamento; - interação com um único fornecedor. 	<ul style="list-style-type: none"> - dependência de um único fornecedor; - se o sistema falhar, toda a empresa pode parar.

Tabela II – Benefícios e problemas dos sistemas ERP - Corrêa, H.L.(1999) .

2.6 Fatores Críticos de Sucesso

Para a implementação de ERP dar certo deve-se considerar alguns fatores críticos de sucesso:

- **Envolvimento da alta direção** – Segundo Bancroft et al (1998) para se obter o apoio da alta administração é necessário, antes de tudo, justificar o custo / benefício da implantação do sistema ERP, estar ciente do investimento e conseqüentemente comprometido a se envolver com todos os processos. Os benefícios incluem redução de trabalho redundante e melhoria na base de dados para estimativas.
- **Equipe de projeto com dedicação em tempo integral e capacitado em ERP** – Um dos problemas enfrentados no início das implantações de sistemas ERP é a escassez de pessoal especializado. A baixa disponibilidade pode fazer com que o cronograma da implantação se estenda muito além do prazo. Bancroft et al (1998) sugere alguns passos para esse planejamento, entre os quais estão a definição do líder do projeto, a formação do comitê executivo, a definição do plano geral de implementação e a estruturação das equipes do projeto.
- **Compromisso de que a organização aceitará a mudança** – Bingi; Sharma; Godla(1999) afirmam que “a implementação de sistemas causam mudanças maciças nas organizações e devem ser cuidadosamente gerenciadas para que os benefícios possam ser obtidos”. Veja na tabela III os problemas mais freqüentes que ocorrem em uma implementação de ERP.

O comprometimento e a motivação dos recursos humanos são cruciais neste processo de mudança. Além da necessidade de saber utilizar a tecnologia, a apreensão do novo modelo de trabalho e a cooperação com outras equipes são fatores que devem ser trabalhados. O sucesso na condução e assimilação dessas mudanças depende da cultura da organização. Organizações, em que a alta administração não possui a confiança dos empregados, em que a tomada de decisões é centralizada e em que também a moral dos empregados é baixa, tendem a ser resistentes às mudanças encarando qualquer nova implantação como algo passageiro e que não merece dedicação ou esforço pessoal. Desta

forma, possuir um plano de comunicação, envolvimento e comprometimento de todos na organização é de extrema relevância.

- **Planejamento adequado, visão e objetivos claros** – o que é esperado do novo sistema deve estar bem definido, uma vez que a percepção dos resultados esperados pela utilização/ exploração do software são inferiores às expectativas e exigem prazos e custos superiores aos inicialmente estimados – o que é uma das principais causas de cancelamento dos projetos de implementação de ERP. Segundo Slater, Derek(1999), comentando a respeito da necessidade de um processo formal e extenso para a seleção do fornecedor de sistemas ERP, afirma que: “empresas compram sistemas que custam milhões de dólares para depois descobrir que não funcionam – ou pelo menos não funcionam bem – para um dos seus principais processos de negócios”. Segundo o autor, “uma das razões para isso é que os sistemas ERP estão de tal maneira em alta e a imprensa e consultores insistem tanto em suas possibilidades, que muitas empresas embarcam nessa solução sem fazer o estudo necessário”.
- **Middleware** – Heinemann F.(2004) afirma que “Os softwares de ERP trabalham muito com integração devido às necessidades dos negócios”. Muitos fornecedores de ERP trabalharam com diversas tecnologias. Deve ser muito bem definido como o ERP irá se comunicar com os outros sistemas da organização e quando isso deverá ocorrer.
- **Pós-ERP** - em um grande número de projetos há uma diminuição de eficácia e de capacidade de resposta logo após a implementação. A razão mais comum é o fato da equipe ainda não estar familiarizada com a nova tecnologia, o que pode provocar situações de maior ou menor embaraço. Já a descoberta de que o sistema não atende a uma determinada atividade não planejada na fase de aquisição do ERP é uma das principais razões de insatisfação das empresas que adquiriram e já implementaram.

	PROBLEMAS MAIS FREQUENTES	REQUER
ERP	<ul style="list-style-type: none"> ■ Excesso de "modificações" dificultando a atualização de versão. ■ Erros de configuração. ■ Sistemas incompletos. ■ Hardware inadequado. 	<ul style="list-style-type: none"> ■ Boa configuração, sistemas atualizados e completos, outras ações de caráter puramente técnico.
Nível I Interface usuário-sistema	<ul style="list-style-type: none"> ■ Deterioração decorrente da falta de sistemática para avaliar e treinar os usuários (manutenção do conhecimento). 	<ul style="list-style-type: none"> ■ Programa para manter os usuários capacitados (software + habilidades complementares). ■ Responsável por desenvolver e manter este nível.
Nível II Interface organização- usuário	<ul style="list-style-type: none"> ■ Ausência de padrão de interface. Os processos não são percebidos, não estão estruturados e são realizados de forma deficiente e esporádica. 	<ul style="list-style-type: none"> ■ Processos estruturados de gestão, alimentação e manutenção do ERP. ■ Habilidades complementares (outras mais). ■ Responsáveis por desenvolver e manter este nível.
Nível III Ambiente integrado e orientado para processos	<ul style="list-style-type: none"> ■ Suposição de que a instalação de um ERP é o único requisito necessário. ■ Desconhecimento de como desenvolver o ambiente necessário para operar com sucesso uma tecnologia integrada. 	<ul style="list-style-type: none"> ■ Ciclos de gestão de processos. ■ Times multidisciplinares de processos. ■ Papéis e responsabilidades de gestão de processos. ■ Realinhamento de elementos estruturais. ■ Valores e crenças organizacionais coerentes com uma filosofia de integração e orientação para processos.

Tabela III – Problemas mais frequentes Bingi; Sharma; Godla(1999)

Serão apresentadas nas tabelas IV, V e VI a seguir, as principais mudanças organizacionais pós-ERP. As mudanças estão divididas entre tecnológicas, estruturais e comportamentais. Contudo, cabe destacar que esta classificação não é estanque, sendo somente para facilitar a comunicação dos resultados.

Variáveis	Mudanças Tecnológicas identificadas
Mudanças na tecnologia de informação e na qualidade da informação	<ul style="list-style-type: none"> • Atualização de hardware e software • Aumento do número de Computadores Pessoais • Unificação das informações • Diminuição dos relatórios impressos • Dificuldade na obtenção de relatórios gerenciais modificados

Mudanças nas técnicas de gestão e processos de trabalho	<ul style="list-style-type: none"> • Incorporação de novas técnicas de gestão (best practices) • Redesenho de processos e sua racionalização • Melhoria no monitoramento dos processos • Maior integração dos processos • Identificação e resolução de problemas nos processos é mais rápida • Melhor sincronização das dimensões física e contábil
Mudanças nos produtos e na eficácia organizacional	<ul style="list-style-type: none"> • Aumento do ritmo de trabalho • Melhoria da imagem organizacional perante o mercado • Tempo maior empregado nas atividades-fins de cada setor, nas atividades de análise de dados e nas atividades gerenciais.
Mudanças na qualificação técnica das pessoas	<ul style="list-style-type: none"> • Necessidade de maior preparo e qualificação técnica das pessoas (2º grau completo, conhecimentos básicos de informática e idiomas etc.).

Tabela IV - Principais mudanças tecnológicas identificadas nas organizações. Souza, Cesar(2003)

Variáveis	Mudanças estruturais identificadas
Mudanças quanto aos mecanismos de coordenação	<ul style="list-style-type: none"> • Sistema auxilia a comunicação inter e intra-unidades (maior rapidez) • Diminuição da quantidade de consultas diretas e trocas de informações verbais
Mudanças nas partes básicas da organização	<ul style="list-style-type: none"> • Eliminação de um nível hierárquico (de natureza tática) • Demissão de pessoas que não tiveram condições de se adaptar à nova tecnologia • Acúmulo de funções por parte de alguns cargos

Mudanças quanto aos parâmetros de desenho das organizações	<ul style="list-style-type: none"> • Aumento do nível de controle sobre o trabalho • Tendência a um aumento da autonomia para a realização de tarefas e decisões rotineiras pelo maior acesso às informações • Aumento no nível de formalização das organizações • Maior padronização dos processos de trabalho
--	---

Tabela V - Principais mudanças estruturais identificadas nas organizações - Souza, Cesar(2003)

Variáveis	Mudanças comportamentais identificadas
Mudanças na cultura organizacional	<ul style="list-style-type: none"> • Aumento da responsabilidade na realização de atividades • Preocupação com a veracidade e precisão dos dados • Maior necessidade de as pessoas pensarem na empresa toda, de se voltarem aos objetivos organizacionais. • Aumento da visão sobre clientes externos da organização
Mudanças quanto ao grau de motivação dos funcionários	<ul style="list-style-type: none"> • Maior conscientização sobre o impacto causado pelo trabalho de cada indivíduo sobre todos os processos • Maior compreensão dos objetivos do trabalho
Mudanças nas habilidades e capacidades requeridas das pessoas	<ul style="list-style-type: none"> • Necessidade de maior disciplina na realização do trabalho • Necessidade de explorar o sistema exige preparo para pesquisa e análise • Valorização da capacidade de trabalhar em grupo • Necessidade de pessoas mais comprometidas e mais ágeis

Tabela VI Principais mudanças comportamentais identificadas nas organizações. Souza, Cesar(2003)

2.7 Fases da Implementação

Não é somente a implantação de um ERP, entretanto, que irá fazer a empresa levar vantagem sobre as outras, uma vez que a empresa não pode ser considerada eficaz apenas

por seus aspectos tecnológicos, ou ainda, mesmo que uma organização implante o mesmo ERP utilizado por suas concorrentes, não necessariamente irá desenvolver a mesma capacidade competitiva delas.

Assim como acontece na maioria das atividades, o planejamento no processo de implementação de um sistema ERP é muito importante, pois as organizações só obtêm o máximo de vantagem ao entenderem que os ERPs são sistemas de suporte aos processos e que, quanto mais aderentes forem os processos ao sistema, maior será o aproveitamento e conseqüentemente levará a melhorias desses processos.

Durante o planejamento, se forem discutidas e estabelecidas alternativas para eventuais contratempos, economiza-se tempo e dinheiro.

Em muitos casos, o fator competitivo de uma organização dependerá da maneira como a informação que é gerenciada por esses sistemas será administrada, pois a qualidade da informação a ser gerada pelo sistema dependerá diretamente da qualidade com que foi definida na etapa de implementação.

Assim, pode-se considerar que, para se tornar competitiva no mercado, a empresa, além de utilizar um sistema ERP, necessita ter a informação de como fazer bom uso dos dados armazenados, devendo se planejar muito bem para conduzir a implantação e com maior eficiência do que os seus concorrentes.

Além da etapa de planejamento é necessário ainda que a implementação seja monitorada com atualização freqüente do cronograma estabelecido, identificando eventuais necessidades de ajustes e revisão dos recursos alocados para permitir cumprimento dos objetivos estabelecidos.

É importante também analisar os processos internos da empresa, identificando melhorias e propondo novas soluções, envolvendo a configuração dos parâmetros do sistema para que ele contemple o novo processo. Essa identificação exige um grande conhecimento do sistema e também um grande conhecimento das características do negócio da empresa onde será implantado o ERP. A complexidade da parametrização depende tanto do sistema (podendo chegar ao nível de se definir quais dados aparecem na tela do computador ou não), quanto da complexidade do processo em si.

Caso a empresa possua características particulares, que não sejam contempladas pelo sistema ERP, o sistema pode ser modificado com soluções específicas para adaptar o sistema aos negócios da empresa.

Após todas essas fases deve ser feita a validação do sistema, o que envolve análises críticas da implantação, confrontando-se o que foi planejado com o que foi executado e verificando se todos os objetivos foram alcançados.

Consequências do mau planejamento da implementação:

- Estouro de custos
- Perda de prazos
- Deficiências técnicas que prejudicam o desempenho
- Fracasso na obtenção dos benefícios operados

O processo de planejamento se compõe de um conjunto de fases, das quais pode-se citar as seguintes como sendo as principais:



Figura 8. ERP e suas principais fases - Laudon (2005).

- **Levantamento das necessidades de informação** - Deve ser detalhado para permitir a identificação de todos os requisitos relevantes a serem atendidos pelo sistema, tanto operacionais como gerenciais, de maneira a evitar que o sistema seja modificado imediatamente após ou mesmo durante sua implementação. Neste levantamento deve-se também abordar a sua disponibilidade aos usuários.

Na prática, um grande número de projetos é entregue com alguma funcionalidade faltando (prometidas, então, para versões posteriores) segundo Keil; Mann; Rai(2000).

- **Definição das informações a serem extraídas do sistema** - Preparar os relatórios e consultas que serão colocados à disposição dos usuários e prever o tempo necessário para treinamento, visando uso eficiente do sistema. Submeter as definições aos usuários para aprovação para assegurar o entendimento adequado de suas necessidades.
- **Entendimento da abrangência dos vários módulos do sistema, alternativas para seu uso, necessidades de eventuais ajustes e tempo requerido** - Sistemas ERP permitem normalmente caminhos distintos para atender a uma determinada necessidade: evitar ajustes e modificações, em casos em que o sistema já oferece a solução desejada, depende de conhecimento detalhado do sistema e análise cuidadosa das alternativas disponíveis.
- **Identificação das simplificações e eliminações permitidas pelo sistema** – Um sistema integrado permite simplificação e eliminação de tarefas normalmente executadas em sistemas manuais, muitas vezes repetitivos, ou com baixa integração. Identificar as principais oportunidades de simplificação faz parte da etapa de planejamento da implantação.
- **Definição da seqüência de implantação, resultados esperados e prazos envolvidos** – Deve-se preparar um cronograma que estabeleça as atividades previstas, quais processos chaves e permita a identificação de pontos críticos e dos resultados bem definidos que marquem o progresso do processo de implementação. É válido ressaltar a importância desta etapa uma vez que é sabido que 30 a 40 por cento de todos os projetos de software fogem do controle; ultrapassam em muito a programação e as projeções de orçamento originais, além de não funcionarem como o especificado Keil, Mann e Rai(2000) .

- **Estabelecimento de responsabilidades** – Delegar responsabilidades aos usuários envolvidos no atendimento dos analistas encarregados dos aspectos técnicos da implementação e no processo para aprovação dos resultados.
- **Treinamento:** consiste no treinamento dos usuários sobre as funcionalidades do sistema pertinentes às suas atividades operacionais e gerenciais. A participação dos usuários responsáveis dos processos envolvidos é importante na questão do treinamento como um dispositivo de qualificação, desenvolvimento e adequação dos usuários ao novo sistema.
- **Seleção do *Middleware* do ERP:** *Middleware* de acordo com o livro *Web Programming with SAP Web Application Server*(2004) é o software que interliga duas ou mais aplicações permitindo que se comuniquem e transmitam dados uma para a outra (Veja a Figura 9).



Figura 9 – *Middleware* - É um software que funciona como uma camada de tradução entre duas aplicações separadas para que possam trabalhar em conjunto.

O **Middleware** pode compor-se de software personalizado escrito pela própria organização ou de um pacote de software. No item 3.0, serão abordados as técnicas de **Middleware** com maiores detalhes.

Deve-se efetuar um levantamento de quais sistemas atuais da organização devem-se comunicar diretamente com o ERP e qual o impacto dessa comunicação no processo atual. Faz-se necessária ainda uma análise de risco, bem como a identificação de potenciais, duplicidade de informações e re-trabalhos que possam vir a ser gerados, caso não optar pelo **Middleware** neste momento.

A empresa necessitará desenvolver mecanismos de tratamento de dados e informações integrados, considerando o ambiente da organização e traçando suas estratégias baseadas nesses recursos. Grande parte das empresas não podem simplesmente livrarem-se de

todos os seus sistemas e criar integração total a partir do zero. Muitas das aplicações de mainframe (legado) que utilizam são essenciais para suas operações diárias e é muito arriscado modificá-las. De qualquer forma, isto não impedem estas empresas de tornarem-se ainda mais úteis se suas informações e sua lógica empresarial puderem ser integradas a outras aplicações, segundo W. Noffsinger et al(1998).

Um modo de integrar diversas aplicações legadas é utilizar um software especial chamado *Middleware* para criar uma interface ou ponte entre dois sistemas diferentes.

2.8 Recomendações ao Implementar ERP

Antes de discutir as recomendações de seleção do *Middleware* como um dos elementos chaves para uma boa implantação de um ERP, é sempre importante lembrar outros aspectos mencionados anteriormente neste trabalho, os quais não estão ligados diretamente à tecnologia, mas cuja importância é essencial para o sucesso da implantação. Assim sendo, não podem deixar de ser mencionados.

Se as empresas não considerarem os custos da mudança organizacional associada a qualquer sistema novo, nem executarem essas mudanças efetivamente, os benefícios trazidos pelos investimentos em tecnologia da informação serão reduzidos, segundo Irani; Love(2000).

Os problemas que causam fracasso de sistema ERP encaixam-se em múltiplas categorias como ilustrados na figura 10.

As principais áreas problemáticas são: projeto, dados, custo e operações de acordo com Laudon (2005).

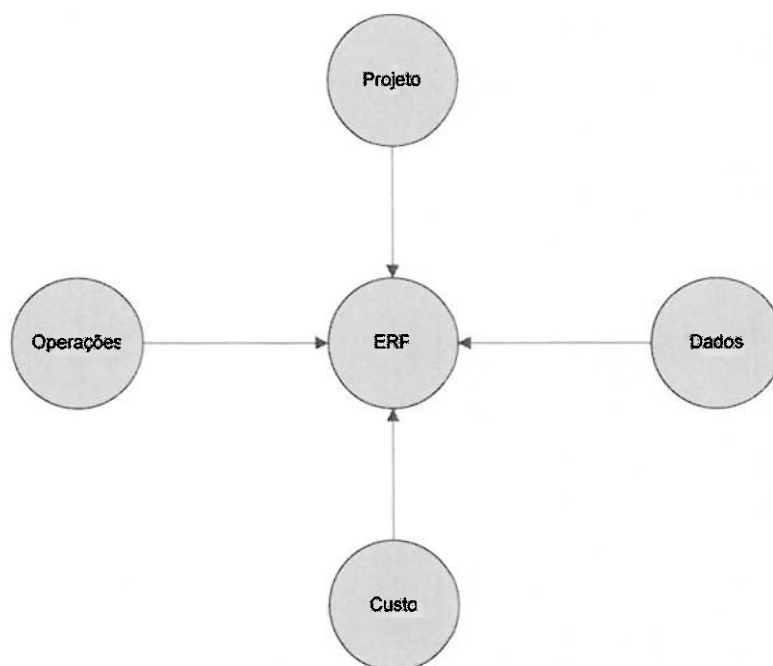


Figura. 10 Áreas Problemáticas em sistemas de Informação ERP. Problemas com projeto, dados, custo ou operações podem ser sintomas do fracasso de um sistema Laudon (2005).

Segundo Cliffe, Sarah(1999) a melhor maneira para se abordar uma implementação de um sistema ERP e encará-lo como um novo negócio, uma vez que ambos compartilham características como tendência a mudar o formato conforme o andamento, altos custos e riscos e envolvimento de diversos times. A autora apresenta as seguintes recomendações para as companhias evitarem custosas e grandes falhas em seus projetos de implementação de ERP:

1. **Investir em estágios** - devido aos altos riscos envolvidos em projetos de ERP, o investimento em estágios permite a cada etapa concluída uma reavaliação do projeto, reestruturando-o ou mesmo abandonando-o. Esta abordagem não é fácil para a equipe responsável pela implementação e as empresas poderão gastar mais recursos financeiros no projeto do que se fosse aplicada a abordagem tradicional. Em contrapartida, reduz-se com isto os riscos de perdas de muitos investimentos, caso ocorra uma falha no projeto.
2. **Compartilhamento de riscos** - a autora argumenta que os riscos podem ser compartilhados por todos os participantes, como se estivessem em um novo negócio. Pelo fato dos projetos de ERP serem muito complexos, muitas vezes os funcionários da

empresa fornecedora de ERP ou da consultoria de implementação precisam tomar importantes decisões que afetam a forma de operacionalizar os negócios da empresa. Estas decisões precisam estar consistentes com o plano estratégico da companhia e com os interesses dos acionistas.

3. Composição da equipe - como em novos negócios deve ser dada importância vital para a composição da equipe que fará a implementação, os membros devem ser selecionados considerando sua reputação, realizações passadas, flexibilidade e habilidades.

A implementação do sistema ERP não termina com o gerenciamento e a finalização de sua implantação. Pelo contrário, após o término do processo, podem surgir novas situações a serem tratadas, como novas ferramentas que se integram ao sistema ou a necessidade de manutenção e atualização do sistema, entre outros. Possíveis campos futuros para a atuação do sistema ERP devem ser pesquisados, bem como outros sistemas que possam surgir e se integrar ao sistema da empresa.

De acordo com Robey, Daniel(2000), a utilização de um sistema ERP apresenta algumas vantagens e desvantagens que são listadas abaixo:

Vantagens:

- Reduz o número de documentos em papel, disponibilizando consulta e introdução on-line de informação.
- A informação é detalhada e vinda de várias áreas da empresa.
- Menor tempo na resposta.
- Maior controle da informação.
- Melhor monitorização do sistema e rápida consulta às bases de dados.
- Providencia uma base de dados única para ser utilizada por todas as aplicações.

Desvantagens:

- Custo de Implementação é muito elevado, fora do alcance das pequenas e médias empresas;

- Durante a implementação do sistema a empresa poderá ficar sujeita a uma queda na produtividade e, como a implementação de um sistema ERP pode levar muito tempo, o resultado da empresa poderá também ficar comprometido.
- Altos gastos com treinamentos.
- Poderá haver demissão de pessoal, pois após a implantação alguns dos processos deixam de ser manuais para serem automatizados.

Tratando-se agora de recomendações para o **Middleware**, tomar a decisão sobre qual tecnologia a ser usada dependerá de diversos fatores, tais como:

- Custos;
- Tempo de projeto;
- Se os sistemas que irão interagir as suportam;
- Se for uma comunicação “real-time” ou pode ser feito sem a interação de um usuário, chamado de “processos em background”;
- Determinação das regras necessárias para que o processo flua sem problemas;
- Estabelecimento das atividades de suporte, em caso de falhas (chamados de contingências).

Todas essas atividades devem ser levadas em consideração no momento da escolha da tecnologia, dependendo do grau de importância da informação trafegada. Se a tecnologia escolhida não for adequada, além de todos os fatores anteriores serem afetados diretamente, o suporte pós-implantação se tornará muito complexo, fazendo com que pequenas melhorias ou falhas encontradas possam levar muito tempo para serem implementadas ou corrigidas, podendo até ser necessário um novo projeto.

3 Tecnologias disponíveis para *Middleware* com Sistemas ERP

3.1 Introdução

As empresas não podem se dar ao luxo de fixarem-se em uma tecnologia. Elas têm que se adaptar para oferecer serviços diferenciados para clientes e parceiros.

Segundo Sampaio, Cleuton(2006), não é mais possível para uma empresa manter uma interface através de p.e. disquetes, muito utilizados na década passada entre as empresas, com seus parceiros, fornecedores e clientes utilizando recursos propensos ao desuso.

Muitos fornecedores de ERP trabalham com diversas tecnologias de **Middleware**. Neste tópico será analisado os principais métodos do mercado, focando principalmente o SAP R/3 (**S**ysteme, **A**nwendungen, **P**rodukte in der **D**atenverarbeitung ou Sistema, Aplicações e Produtos em Processamento de Dados), visto que este é um produto já consolidado e com a maior base instalada atualmente. O gráfico abaixo (figura 11) provê informações de abril de 2005, apresentada pela IPM – Impacta Pesquisa Periódica de Mercado.

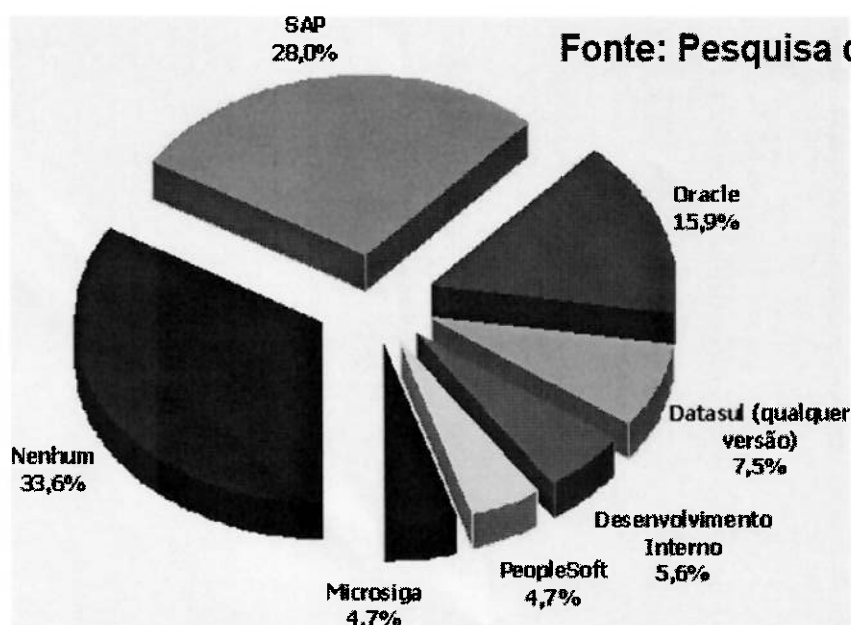


Figura. 11 Uso de sistemas ERP IPM – Impacta Pesquisa Periódica de Mercado(2005).

É importante também saber como tal tecnologia se originou. Para isso, será relatado um pouco da origem das redes de computadores - o elemento base nos mecanismos de **Middleware** atuais.

3.2 Histórico

No início da década de 70, universidades e outras instituições que faziam trabalhos relacionados à defesa tiveram permissão para se conectar à Arpanet (L. Roberts(1988)). Em meados de 1975 existiram aproximadamente 100 sites interligados.

Pesquisadores que trabalhavam na Arpanet estudaram como o crescimento da rede alterou o modo de como as pessoas a usavam. No final dos anos 70 a ARPANET tinha crescido tanto que o seu original protocolo de comutação de pacotes, chamado de **Network Control Protocol** (NCP), tornou-se inadequado. Foi aí, então, que a ARPANET começou a usar um novo protocolo chamado TCP/IP (**Transfer Control Protocol/Internet Protocol**), o qual é amplamente utilizado pela internet e redes locais atuais.

Atualmente há cerca de 400 milhões de computadores permanentemente conectados a internet, além de muitos sistemas portáteis e de computadores pessoais que ficam on-line por apenas alguns momentos. (Informações obtidas no **Network Wizard Internet Domain Survey**,(2006)).

Para se utilizar eficazmente uma rede deve-se satisfazer a um determinado grupo de requisitos. Os principais deles são: segurança e velocidade da transmissão da informação.

Diversas empresas propuseram diferentes soluções de protocolos de comunicações. Os protocolos de comunicações especificam como os computadores interagem e trocam mensagens. Cada protocolo (ex. PPP, FTP, HTTP, IP, TCP...) tem diferentes responsabilidades e regras. Conseqüentemente, desenvolver aplicações usando essas regras, cria-se laços específicos.

Diferentes tecnologias de integração foram criadas e utilizadas para possibilitarem essa integração podendo-se utilizar diferentes protocolos. A maioria delas fazem a troca de dados entre as aplicações de uma forma ou de outra. (Os mais populares são: **Remote Procedure Calls (RPC)**, **Distributed Component Object Model (DCOM)**, **Remote Method Invocation (RMI)**, **Common Object Request Broker Architecture (CORBA)**, **Remote Function Call (RFC)**, **Electronic Data Interchange (EDI)**, **Web Services**, **Service Oriented Architecture (SOA)**, **Enterprise Services Architecture (ESA)**, **XML**, **XML para EDI** e **Text files (TXT)**). Será abordado aqui, em tópicos, cada uma delas.

A dificuldade com a comunicação via Internet ou com qualquer outra rede é que é preciso haver uma estrutura previamente combinada da mensagem. O destinatário tem que conhecer a estrutura da mensagem para ser capaz de recebê-la e tratá-la automaticamente. Um dos problemas com a comunicação eletrônica entre diferentes sistemas computacionais reside nas diferenças existentes entre formatos de arquivos, esquemas relacionais, protocolos de troca de dados, entre outros, que tornam o processo de troca de dados complicado.

No desenvolvimento de padrões entre uma implementação rápida e o nível de colaboração atingido entre os atores existe um sinuoso caminho a ser percorrido. Os grupos de desenvolvimento de padrões geralmente têm visões de alto nível do grau de colaboração e querem gastar um tempo considerável para desenvolver uma solução que sirva a todas as partes. As empresas, por sua vez, querem uma implementação rápida para que sejam transmitidos os dados corretos.

Como tendência do mercado, especialistas de TI identificaram a base de tecnologias WEB, que serve de base para toda a internet. Estas bases consistem nas seguintes tecnologias:

TCP/IP – Protocolo universal, entendido por todos os dispositivos de rede.

HTML – Linguagem universal de textos, usado para mapeamento de informação.

XML – Linguagem universal para trabalhar com qualquer tipo de dados.

São padrões abertos e independentes de fornecedor de tecnologia. Todos esses princípios fazem com que o **Web-Services** ganhe vantagem sobre os demais, tornando-o um sistema independente.

O conceito de **Web-services** nasceu depois de várias tentativas sem sucesso de muitos grupos de analistas, arquitetos e desenvolvedores de todo o mundo para criar mecanismos imediatos de interação entre diferentes sistemas de informação. Baseado nessas necessidades, surgiu a arquitetura de software chamada SOA.

Será tratada agora as tecnologias descritas acima, de uma forma independente, para que o leitor possa entender e, de acordo com as necessidades, conseguir optar entre uma delas e se aprofundar. Não serão discutidos aspectos específicos de uma determinada tecnologia de integração ou **Middleware**. O foco será dado nas decisões de projeto.

3.3 Text files (TXT);

Uma aplicação escreve em um arquivo, o qual outra aplicação lê-lo-á depois (Silva, Muniz(2006)). As aplicações devem ter um acordo quanto a: nome, localização do arquivo, formato do arquivo, tempo em que será escrito e lido e responsável pela exclusão do arquivo.

Os integradores têm a responsabilidade de transformar os arquivos em formatos diferentes. Os arquivos são produzidos a intervalos regulares de acordo com a natureza do negócio. Veja figura 12 como ilustração.

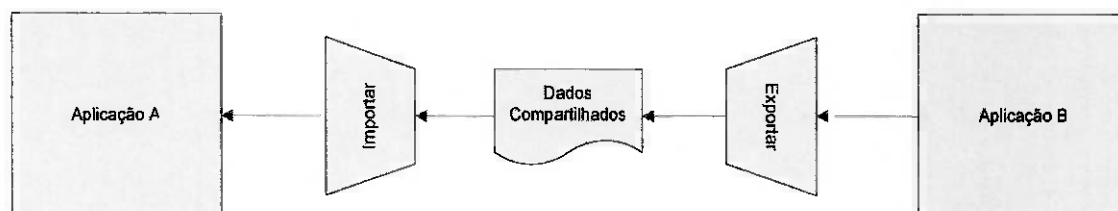


Figura 12 – Cada aplicação produz os arquivos contendo as informações que a outra aplicação deve consumir.

3.4 Remote Procedure Calls (RPC)

Em RPC (Chamada de Procedimento Remoto) (Silva, Muniz(2006)), uma aplicação disponibiliza parte da funcionalidade, a qual é remotamente acessada por outras aplicações, como um procedimento remoto, conforme ilustrado na figura 13. Neste processo tipicamente ocorre comunicação síncrona.

Deve-se desenvolver cada aplicação como um componente com comportamento e dados encapsulados e prover uma interface para permitir que a outra aplicação possa interagir com a aplicação em execução.

RPC são funções de softwares hospedadas em máquinas remotas.

Parâmetros para a função são passados através da rede na sua chamada (processo chamado de “**marshalling**”), em que a função remota é executada e retorna um resultado. O resultado, então, é retornado através da rede para o executor. Tudo isso ocorre de uma forma transparente.

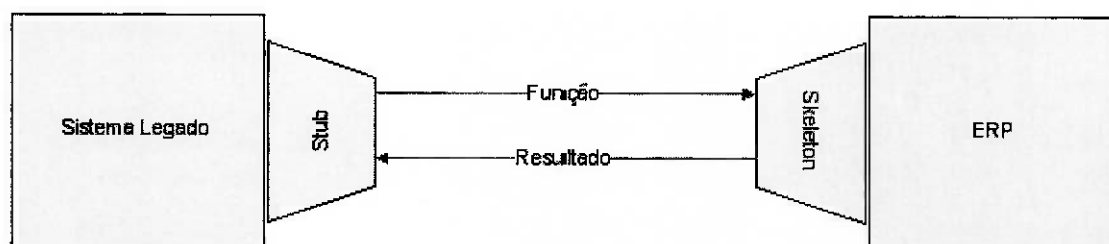


Figura 13 - Desenvolver cada aplicação com um componente com comportamento e dados encapsulados. Prover uma interface para permitir que a outra aplicação possa interagir com a aplicação em execução.

3.5 Distributed COM (DCOM)

O **Distributed Component Object Model** (DCOM) ou Modelo de Objetos de componentes Distribuídos é um modelo de objetos distribuídos proprietário e definido pela empresa Microsoft ® que concorre com outras soluções na arena da computação distribuída, sendo um protocolo que permite que componentes de software se comuniquem através de uma rede.

O DCOM é uma extensão do **Component Object Model (COM)** ou Modelos de objetos de componentes, que é uma estrutura baseada em objetos para desenvolvimento e distribuição de componentes de software.

O material aqui apresentado foi baseado nas obras - **COM Developer's Eagle Eye on the future of COM and DCOM**.(2006); **Inside Visual C++**. (1997); **Inside DCOM**. (1997); **Component Software: Beyond Object-Oriented Programming**. (1997). Referências ou informações técnicas detalhadas podem ser obtidas em **DCOM Architecture**, 1997.

3.5.1 Visão Geral do COM

O COM é uma arquitetura de software orientada a objetos que permite a criação de componentes de software por diferentes fabricantes com uma variedade de linguagens e ferramentas (C, C++, Java, JScript, VBScript, Delphi, PowerBuilder e MicroFocus COBOL). Além de especificação de um padrão para criação de componentes interoperáveis, configuráveis e atualizáveis, o COM também é um conjunto de serviços de suporte. Este modelo corresponde a uma evolução unificada e expansível da tecnologia OLE, integrada à família de sistemas operacionais Windows ® e suporta as seguintes características:

- Um padrão independente de linguagem para um executável cliente Win32 carregar e chamar uma DLL Win32;
- Uma forma genérica de um executável controlar outro, substituindo o **Dynamic Data Exchange (DDE)**;
- Uma versão 32 bits para substituição dos antigos controles VBX, chamados controles ActiveX, que são controles que utilizam a tecnologia COM. Projetados para facilitar sua distribuição em redes de alta latência, provêm sua integração em navegadores Web que podem ser incluídos em aplicações ou páginas Web;
- Uma forma poderosa para aplicativos interagirem com o sistema operacional;
- Um suporte para criação de documentos compostos, configuração de controles, transferência de dados, **scripting** e outras interações entre aplicações;

- Expansão para acomodar novos protocolos como a interface de acesso para banco de dados OLEDB (OLEDB é uma especificação aberta da Microsoft, designada para acessar todos os tipos de dados).

O padrão de interoperabilidade binária do COM permite que os componentes sejam distribuídos sem código fonte e integrados no ambiente dos clientes. O COM é um protocolo que permite a conexão entre um objeto cliente e outro servidor e depois sai de cena. Após a conexão ser realizada, os objetos podem se comunicar através de um mecanismo chamado de interface. Uma interface COM define o comportamento ou as capacidades de um componente de software com um conjunto de métodos e propriedades – é um contrato que garante a consistência semântica do objeto que a suporta, sendo que cada objeto pode suportar diversas interfaces.

As interfaces são definidas com **Microsoft's Interface Definition Language (MIDL)**, uma extensão da linguagem DCE RPC IDL, especificado pela **Open Software Foundation (OSF)**. O compilador MIDL gera os seguintes códigos:

- **Proxy** que corresponde ao lado do cliente da API para os objetos que suportam a interface, sendo responsável pela linearização dos parâmetros (**marshaling**);
- **Stub** para decodificar as requisições recebidas dos clientes e disparar o objeto apropriado no servidor. Estes dois códigos garantem a compatibilidade binária entre as diferentes linguagens utilizadas ou, no caso do DCOM, diferentes sistemas operacionais, máquinas e protocolos de rede.

Para eliminar qualquer ambigüidade e colisão de nomes, cada interface recebe uma **Globally Unique Identifier (GUID)**. Esta identificação é denominada **Interface Identifier (IID)** e é substituída a cada nova versão da interface. Um objeto pode suportar novas interfaces ou novas versões da mesma, cada uma com sua própria IID. Desta forma, clientes antigos podem invocar métodos em paralelo com os novos. Além da identificação IID, toda a interface também possui um nome legível, mas neste caso não existe garantia de unicidade.

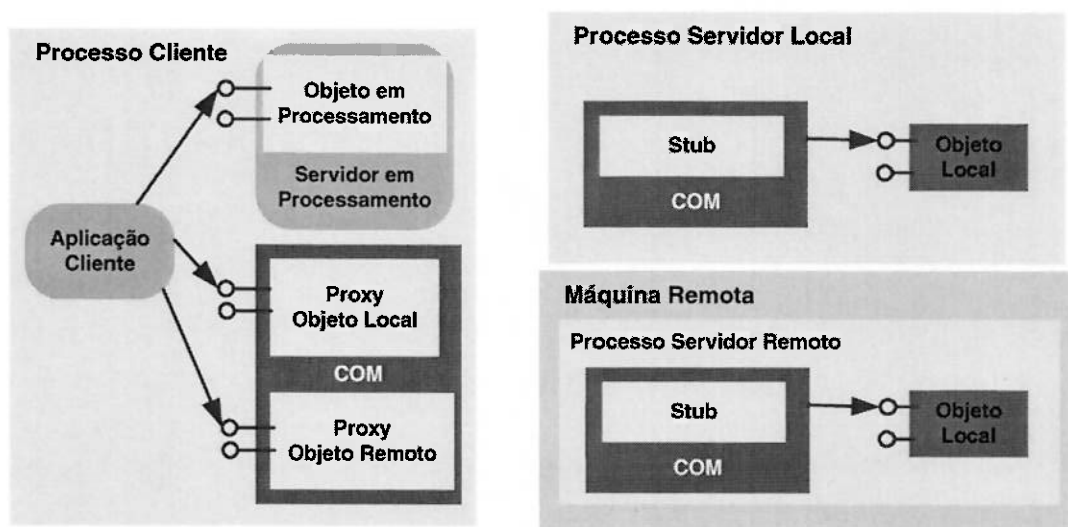


Fig 14 – Empacotamentos de Objetos do DCOM - Roy,; Ewald(1997).

Uma classe COM é o código fonte que implementa uma ou mais interfaces em qualquer uma das linguagens suportadas.

Uma ou mais classes podem ser empacotadas num servidor no formato de DLL, a qual é carregada pelo processo cliente (**in-process server**), como por exemplo, um controle ActiveX. As classes também podem ser empacotadas num executável separado (**out-process server**) que pode executar na mesma máquina que o cliente ou numa máquina remota acessada através do DCOM. Na figura 14 é apresentado um cliente COM invocando métodos nas diferentes formas. Esta transparência de empacotamento é uma característica fundamental do COM.

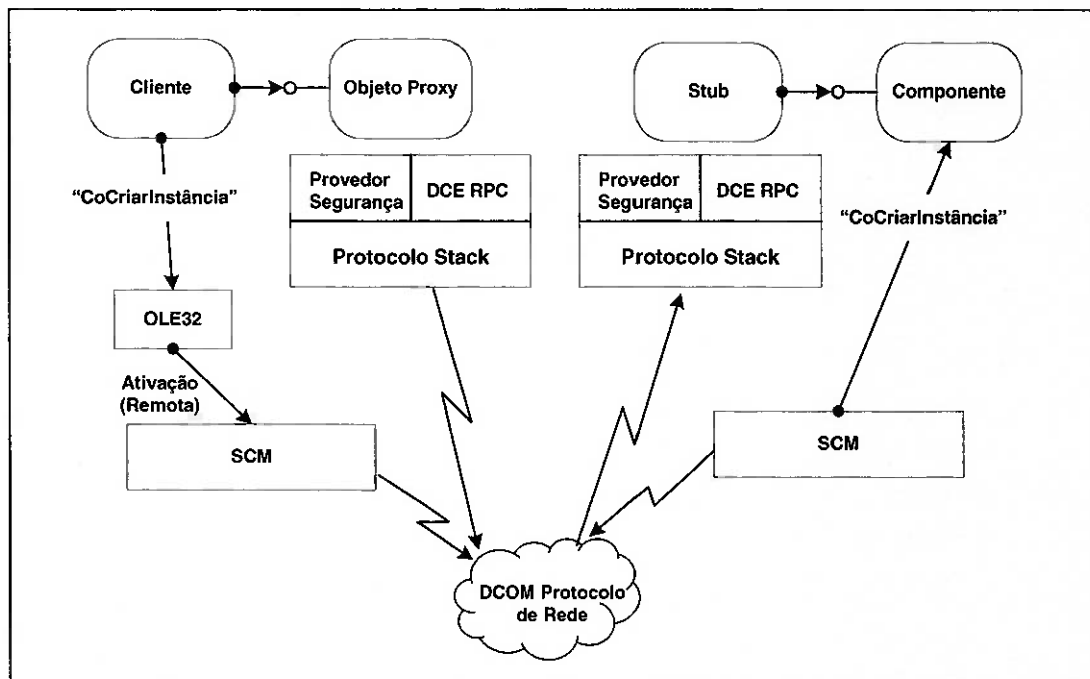


Figura 15 – Arquitetura DCOM – Horstmann; Kirtland(1997).

Para alcançar o máximo desempenho os servidores são tipicamente desenvolvidos como aplicações **multithreaded**. Veja a ilustração arquitetônica na figura 15.

Com o DCOM cada invocação pode ser tratada por uma **thread** separada e um único objeto pode tratar diversas chamadas de forma concorrente.

O custo disto é o aumento de complexidade da aplicação com a utilização de primitivas de sincronização das **threads** no acesso a recursos compartilhados no processo. O DCOM provê diferentes níveis de segurança de forma transparente ao suportar **Access Control List (ACL)** para os componentes COM. Se o cliente não possui direitos de acesso para acessar ou disparar um componente, ocorre falha na requisição sem qualquer envolvimento do código da classe COM.

Um problema comum em aplicações distribuídas é a necessidade de servidores detectarem se seus clientes ainda estão vivos. O DCOM implementa um componente chamado **Object Exporter** o qual verifica referências a objetos que foram trocados com outras máquinas e mantém a consistência da contagem de referências em caso de perda da conexão. Isto é realizado de forma otimizada através de uma única mensagem **keepalive** por máquina, independente da quantidade de clientes ou servidores acessados.

3.6 Remote Method Invocation (RMI)

RMI (**Remote Method Invocation** ou Invocação de métodos Remotos) (Sun Microsystems®) é uma das abordagens da tecnologia Java para prover as funcionalidades de uma plataforma de objetos distribuídos. Esse sistema de objetos distribuídos faz parte do núcleo básico de Java desde a versão JDK 1.1, com sua API especificada através do pacote `java.rmi` e seus subpacotes.

Através da utilização da arquitetura RMI, é possível que um objeto ativo em uma máquina virtual Java possa interagir com objetos de outras máquinas virtuais Java, independentemente da localização dessas máquinas virtuais.

A arquitetura RMI oferece a transparência de localização através da organização de três camadas entre os objetos cliente e servidor:

1. A camada de **stub/skeleton** oferece as interfaces que os objetos da aplicação usam para interagir entre si;
2. A camada de **referência remota** é o **Middleware** entre a camada de **stub/skeleton** e o protocolo de transporte. É nesta camada que são criadas e gerenciadas as referências remotas aos objetos;
3. A camada do **protocolo de transporte** oferece o protocolo de dados binários que envia as solicitações aos objetos remotos pela rede.

A figura 16 ilustra a organização dessas três camadas em uma aplicação RMI:

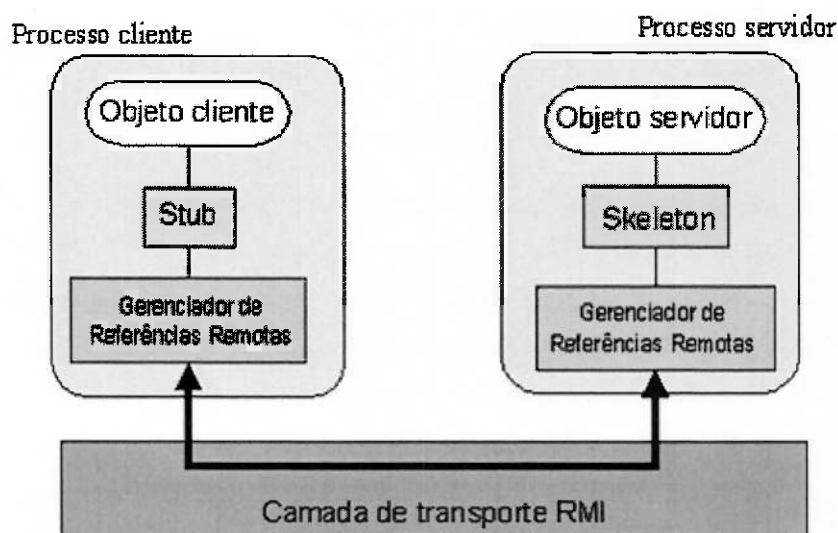


Figura 16. Arquitetura RMI

No desenvolvimento de uma aplicação cliente-servidor usando Java RMI, como para qualquer plataforma de objetos distribuídos, é essencial que seja definida a interface de serviços a ser oferecida pelo objeto servidor.

Os serviços especificados pela interface RMI deverão ser implementados através de uma classe Java. Nessa implementação dos serviços é preciso indicar que objetos dessa classe possam ser acessados remotamente.

A implementação do serviço se dá através da definição de uma classe que implementa a interface especificada. No entanto, além de implementar a interface especificada, é necessário incluir as funcionalidades para que o objeto dessa classe possa ser acessado remotamente como um servidor.

A implementação da interface remota se dá da mesma forma que para qualquer classe implementando uma interface Java, ou seja, a classe fornece implementação para cada um dos métodos especificados na interface.

Com a interface estabelecida e o serviço implementado, é possível criar as aplicações cliente e servidor RMI.

A execução da aplicação cliente-servidor em RMI requer, além da execução da aplicação cliente e da execução da aplicação servidor, a execução do serviço de registro de RMI. Além também do princípio básico de execução de aplicações RMI, a arquitetura RMI oferece facilidades para operação com código disponibilizado de forma distribuída e ativação dinâmica, dentre tantos outros serviços distribuídos.

É interessante observar como padrões de programação distribuída, como a fábrica de objetos remotos e o padrão de **callback**, são trabalhados em aplicações RMI.

No procedimento básico para desenvolver uma aplicação distribuída em RMI, cada servidor remoto criado é cadastrado no serviço de registro RMI. Esse tipo de estratégia, no entanto, pode ser ineficiente quando o número de objetos remotos for grande ou mesmo não previsível antes da execução.

A estratégia para lidar com esse tipo de situação é usar o conceito de fábrica de objetos remotos.

3.7 Common Object Request Broker Architecture (CORBA)

CORBA é a arquitetura padrão criada pelo **Object Management Group (OMG)** para estabelecer e simplificar a troca de dados entre sistemas distribuídos heterogêneos. A sigla CORBA vem de **Common Object Request Broker Architecture**. O material aqui apresentado foi obtido nas obras **What is Corba** (1999); **Instant Corba** (1997); **Overview of CORBA**(2000) .

Em face à diversidade de hardware e software que encontrada atualmente, CORBA atua de modo que os objetos (componentes dos softwares) possam se comunicar sem que o usuário perceba, ou seja, de forma totalmente transparente, mesmo que para isso seja necessário interoperar com outro software em um outro sistema operacional e em outra ferramenta de desenvolvimento. CORBA é um dos modelos mais populares de objetos distribuídos, juntamente com o **Distributed Component Object Model (DCOM)**, formato proprietário da Microsoft®.

A arquitetura CORBA define o ORB (**Object Request Broker**) como um barramento de objetos que permite aos objetos fazerem requisições e receberem respostas de objetos locais ou remotos. Os clientes não têm conhecimento dos mecanismos de comunicação, ativação ou armazenamento dos objetos servidores.

O ORB deixa que os objetos possam localizar os objetos com os quais pretendem se comunicar e, assim, invocar os serviços entre si. O ORB é muito mais sofisticado que as formas alternativas de **Middleware** cliente-servidor, tais como **Remote Procedure Call (RPC)**.

Usando o ORB, um cliente pode invocar um método do objeto servidor, o qual pode estar no mesmo processo, na mesma máquina ou em qualquer site do mundo. O ORB intercepta a chamada e é responsável por encontrar o objeto que pode implementar a requisição, passar os parâmetros, invocar o método e devolver os resultados. Para o objeto cliente não importa a linguagem ou compilador usado na implementação do servidor ou sua localização, o sistema operacional, o protocolo de transporte dos dados ou o tipo de rede. Para o cliente, basta conhecer a interface que o servidor tornou pública, a qual funciona como um contrato entre cliente e servidor.

As especificações das interfaces dos objetos em CORBA são sempre escritas em **Interface Definition Language (IDL)**, uma linguagem neutra que torna os componentes acessíveis por diferentes linguagens, ferramentas, sistemas operacionais e redes. Esta linguagem é puramente declarativa permitindo definir atributos, heranças entre classes, tratamento de exceções e os métodos suportados pela interface, incluindo os parâmetros de entrada, saída e seus tipos. A gramática IDL é um subset de C++ com palavras chaves adicionais para suportar os conceitos de distribuição, conforme o exemplo de Schmidt, Douglas(2000) apresentado na figura 17.

```

module Example {
    struct Date {
        unsigned short Day;
        unsigned short Month;
        unsigned short Year;
    }

    interface Ufo {
        readonly attribute unsigned long ID;
        readonly attribute string Name;
        readonly attribute Date FirstContact;
        unsigned long Contacts();
        void RegisterContact( Date dateOfContact );
    }
}

```

Figura 17. Modelo de IDL

A IDL, na verdade, é o que possibilita a interoperabilidade entre os diversos sistemas, vista a separação que é definida entre interface e execução. Veja ilustração na figura 18.

A interface de cada objeto é definida de forma bastante específica, enquanto sua execução (código fonte e dados) permanece oculta para o resto do sistema.

Ao contrário dos objetos tradicionais, os objetos em sistemas distribuídos possuem uma característica de dualidade: um estado dinâmico, tipicamente alocado em memória volátil e em tempo de execução, e um estado persistente, que não pode ser destruído após o encerramento do programa que os criou e que pode ser usado para reconstruir o estado dinâmico, devendo ser armazenado, portanto, em memória não volátil, seja em sistema de arquivos, seja em banco de dados. A arquitetura CORBA, para prover a persistência, define o **Persistent Object Service** (POS) como sendo responsável por armazenar o estado persistente dos objetos, utilizando quatro elementos: Objetos Persistentes (**Persistent Object** (POs)), Gerenciador de Objetos Persistentes (**Persistent Objects Manager** (POM)), Serviços de Persistência de Dados (**Persistent Data Services** (PDSs)) e Base de Dados (**Datastores**).

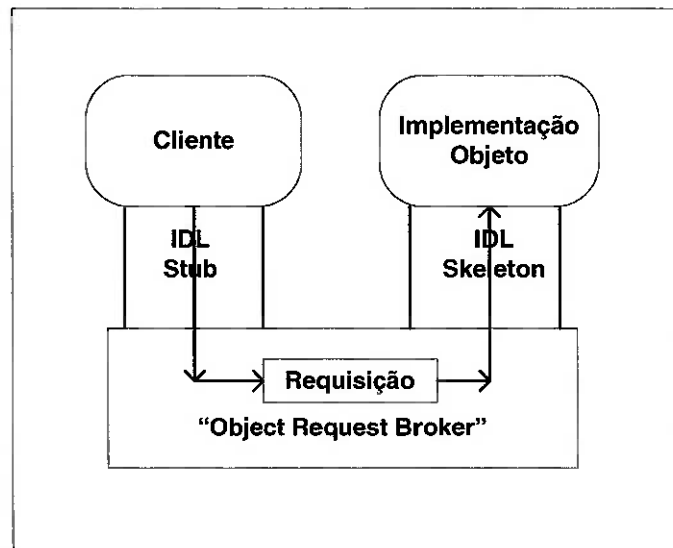


Figura 18 - Mostra como tudo fica junto, pelo menos dentro de um único processo: Como IDL define relações assim estritamente, o stub no lado do cliente não tem nenhum problema de engrenar perfeitamente com o Skeleton no lado do usuário, mesmo se os dois forem compilados em linguagens de programação diferentes.

3.8 Remote Function Call (RFC)

RFC (Chamada de Função Remoto ou Funções de Negócios Compartilhadas) apresentada por Schneider, Neureither(2005) é um protocolo de comunicação proprietário da SAP para execução de rotinas através de conexão TCP/IP ilustrada na figura 19.

Possuem parâmetros de entrada e saída com tratamento de exceção nos quais podem conter complexas estruturas e tipos de dados. Esses tipos de dados podem ser definidos livremente, possibilitando transferir tabelas, estruturas e parâmetros para ambos os lados.

RFC's suportam comunicações síncronas, assíncronas e chamadas transacionais.

RFC's transacionais possibilitam consolidar chamadas de módulos de funções como uma transação simples, estendendo a segurança de transações para sistemas remotos.

A RFC é o protocolo preferido para transferências de dados em modo síncrono, especialmente para comunicação entre o ERP e um sistema legado.

Deve-se implementar as rotinas comuns de modo compartilhado, tornando-as serviços para os sistemas.

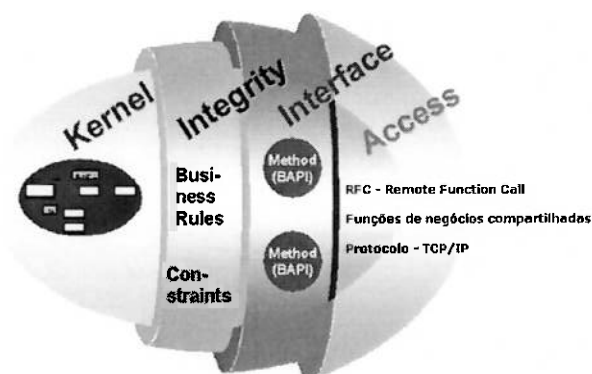


Figura 19. RFC como meio de acesso. Schneider, Neureither(2005)

Onde:

Business Rules ou Regras de negócios que possuem suas rotinas internas.

Constraints ou constantes que são utilizadas para a integridade do dado.

Method (BAPI) – Métodos padrões que são executados pelas funções remotas ao qual possuem seus parâmetros de E/S para efetuar mudanças ou buscas de dados.

3.9 EDI (Electronic Data Interchange)

EDI (**Electronic Data Interchange** ou troca eletrônica de dados) é o mesmo que troca estruturada de dados através de uma rede de dados qualquer. No mundo SAP uma forma específica de implementação de EDI é chamada de IDOC (**Intermediate DOCument**).

Segundo Turban et al(1999), o EDI pode ser definida como o padrão para movimentação eletrônica de documentos de negócio entre as empresas ou dentro das empresas. O EDI usa um formato de dados estruturado que efetua a carga automática de dados, ao qual permite que sejam transformados sem serem re-introduzidos, ou seja, carga direta de informações. Veja figura 20.

Além disso, Turban et al(1999) considera que o uso primário do EDI é transferir transações repetitivas de negócios, tais como: encomendas, faturas, aprovações de crédito e notificações de envio. Isto significa que o EDI hoje, contrariamente ao que muitos acreditam, não implica comunicação em tempo real.

O DISA (**Data Interchange Standards Association**) aponta os seguintes pontos fortes do EDI:

- É um padrão aberto e com fluxos de dados formalizados;
- Garante a troca segura de dados,
- Segura na perspectiva de que diferentes “**checksums**” garantem que os dados enviados sejam confiáveis.

O EDI geralmente fornece três serviços “chave” para trocas de dados aplicação-a-aplicação:

- Contexto: através do uso de documentos de negócio identificáveis.
- Semântica: um método para perceber o significado dos dados, usando dicionários de dados, segmentos e descrições dos conjuntos de transações.
- Sintaxe: através dos tipos de dados e regras padronizáveis, permite que os dados sejam empacotados em mensagens.

A EDI tradicional, do ponto de vista de uma empresa conectada à Internet, sofre de algumas limitações, o que reduz sua utilidade e eficiência em tais soluções. A seguir, serão apresentados os mais relevantes destas limitações:

- Implementação complexa - um problema associado a soluções EDI tradicional é que cada novo contato é único.
- Falta de flexibilidade - as relações baseadas em EDI ainda dominantes sofrem de fraca adaptação a novas tendências e desenvolvimentos. Isto é uma consequência que advém do fato do EDIFACT, que é o processo de reconciliar diferenças entre os diferentes dialetos de EDI e ter-se proposto como o padrão para todas as áreas de negócio.
- Penetração limitada - o preço para estabelecer uma ligação EDI de acordo com um padrão, como o EDIFACT, tem sido elevado.
- Padronização difusa - o esforço do EDIFACT para se fazer um padrão válido para todos os negócios, deixou-o com muitas variantes / subconjuntos.
- Uso limitado nas pequenas e médias empresas - as pequenas e médias empresas não são capazes de suportar os custos do EDI.

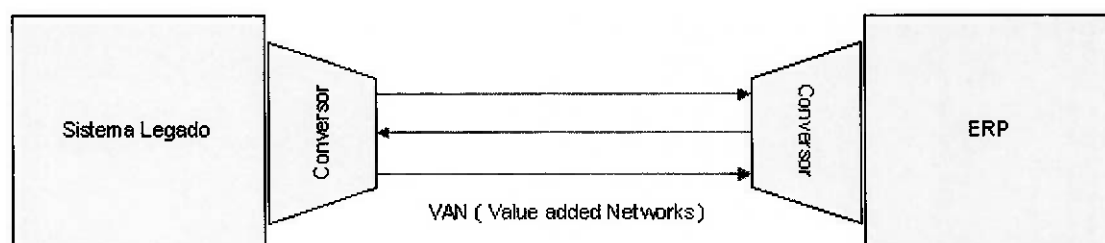


Figura 20 - Através das VAN (redes via contrato para disponibilizar maior segurança para funções e dados, é efetuado as trocas de estruturas de dados).

3.10 XML

XML é a abreviação de **Extensible Markup Language** (Linguagem extensível de formatação). Trata-se de uma linguagem que é considerada uma grande evolução na internet.

O XML é uma especificação técnica, desenvolvida pela W3C (**World Wide Web Consortium** - entidade responsável pela definição da área gráfica da internet), para superar as limitações do HTML, que é o padrão das páginas da Web.

A linguagem XML é definida como o formato universal para dados estruturados na Web. Esses dados consistem em tabelas, desenhos, parâmetros de configuração, etc. A linguagem, então, trata de definir regras que permitem escrever esses documentos de forma que sejam adequadamente visíveis ao computador.

A seguinte definição é apresentada pra XML (**Working Group W3C**):

Um módulo de software solicita serviços de um processador de XML que é usado para ler documentos XML e possibilitar acesso a seu conteúdo e estrutura. Supõe-se que um processador de XML está fazendo seu trabalho em nome de um outro módulo, chamado aplicação. Esta especificação descreve o comportamento requerido de um processador de XML nos termos de como os dados de XML devem ser lidos e a informação deve ser fornecida à aplicação.

Cada documento XML possui uma estrutura lógica e física. Fisicamente, os documentos são compostos de unidades chamadas entidades. Uma entidade pode referenciar outras entidades na criação dos documentos. Logicamente, o documento é composto de declarações, elementos, comentários, caracteres de referência e instruções de processamento, os quais são indicados explicitamente no documento.

Abaixo está um exemplo de XML:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<INVCON01>
  <IDOC BEGIN="1">
    <EDI_DC40 SEGMENT="1">
      <TABNAM>EDI_DC40</TABNAM>
      <MANDT>100</MANDT>
      <DOCNUM>0000000075474542</DOCNUM>
      <DOCREL>620</DOCREL>
      <STATUS>30</STATUS>
      <DIRECT>1</DIRECT>
      <OUTMOD>2</OUTMOD>
      <IDOC TYP>INVCON01</IDOC TYP>
      <MESTYP>INVCON</MESTYP>
      <SNDPOR>SAPPC1</SNDPOR>
      <SNDPRT>LS</SNDPRT>
      <SNDPRN>PCO_100</SNDPRN>
      <RCVPOR>MAXIMO_PRT</RCVPOR>
      <RCVPRT>LS</RCVPRT>
      <RCVPRN>MAXIMO</RCVPRN>
      <CREDAT>2006-08-26</CREDAT>
      <CRETIM>13:33:20</CRETIM>
    </EDI_DC40>
    <EIICSL0 SEGMENT="1">
      <MATNR>M22120320</MATNR>
      <WERKS>104</WERKS>
      <DISPO>108</DISPO>
      <MTART>COMM</MTART>
      <MATKL>M</MATKL>
      <DISMM>V1</DISMM>
      <WAERS>EUR</WAERS>
      <MEINS>PCE</MEINS>
      <LABST>0.000</LABST>
      <LBKUM>14.000</LBKUM>
    </EIICSL0>
  </IDOC>
</INVCON01>
```

3.11 XML para EDI

As possibilidades do XML como uma substituição ou um complemento para o EDI tradicional são significativas ilustradas na figura 19. Algumas das características que fazem do XML uma alternativa atraente para comunicação empresarial eletrônica são:

- O XML é uma tecnologia capaz de executar comunicação empresarial pela Internet. E, ao mesmo tempo, as técnicas de comunicação anteriores não são ignoradas pelo XML, por causa da compatibilidade reversa “**backward compatibility**” para transformação de informação estruturada.
- O XML deixa seus utilizadores definirem o seu próprio dicionário usado para executar a troca de dados. Este é um grande passo a frente, comparado com a técnica de EDI tradicional, e impõe também grande exigência no desenvolvimento de padrões empresariais uniformizados. Se não for estabelecido nenhum padrão, o efeito poderá ser caótico.
- Como resposta ao perigo da não padronização, isto é, quando todos desenvolvem o seu próprio “padrão”, surgiram fortes iniciativas para desenvolver padrões que satisfizessem as necessidades empresariais. Exemplos são BizTalk abordadas no **Microsoft Miztalk Server** (2004) e RosettaNet encontrada no site da internet da organização(2006) .
- A tendência de XML é um passo promissor em direção à integração da aplicação-para-aplicação de um modo mais barato, mais fácil e mais flexível, comparado com o EDI tradicional. Este é um grande passo para um maior uso do comércio eletrônico **business-to-business** (B2B).
- Como consequência da possibilidade de definir os elementos na transação e do esforço de projetos como RosettaNet e BizTalk, o XML tem a possibilidade de ser interpretável por humanos e máquinas. XML pode transferir todos os tipos de dados, por exemplo, uma fatura, um pedido de cuidados médicos, estado de projeto, etc.

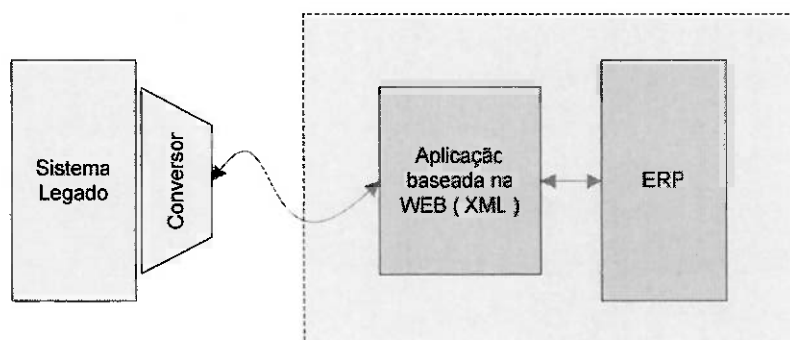


Figura 21 - EDI Baseado na WEB com XML

3.12 Web services

A definição a seguir, pertence a **Web Services Architecture Working Group W3C**.

Um **Web services** é uma aplicação de software identificada por um URI “**Uniform Resource Identifier**”, que é a Tecnologia de endereçamento pela qual URLs são criadas. Tecnicamente, http:// e ftp:// são subconjuntos específicos de uma URL., onde as interfaces e relacionamentos são capazes de serem definidos, descritos e localizados por artefatos XML e suportem interações diretas com outras aplicações de software utilizando mensagens baseadas em XML através de protocolos baseados em internet segundo Kaj Van de Loo (2005).

Estes serviços fornecem padrões para comunicação entre diferentes softwares com o objetivo de apresentar a informação dinamicamente ao usuário. Uma arquitetura padrão é necessária quando existe a necessidade de promover interoperabilidade e maior extensão entre as aplicações, assim como permitir que sejam combinadas a fim executar operações mais complexas.

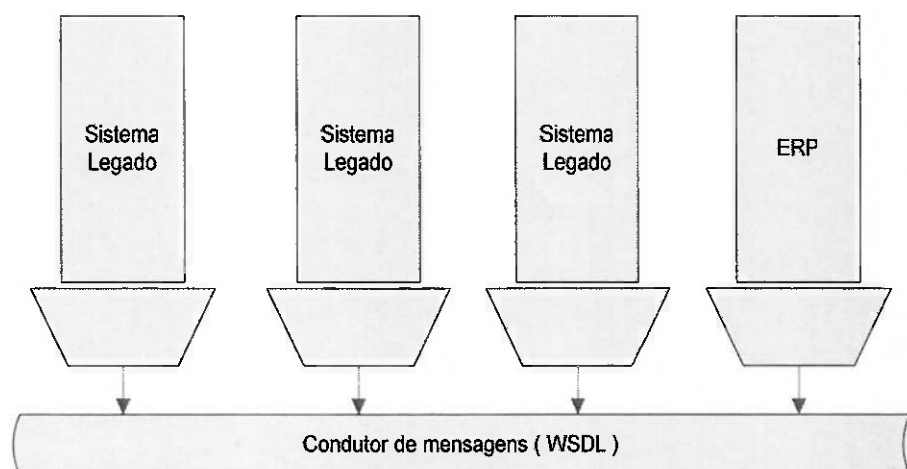


Figura 22 – Web Services

Para descrever os serviços de rede como um conjunto de pontos de comunicação operando com mensagens que possuem informações sobre documentos ou procedimentos no condutor, é definido segundo **Working Group W3C**, o WSDL.

WSDL “**Web Services Description Language**” (Figura 22) é um formato de XML usado para descrever o que um **Web services** pode fazer, onde está localizado no condutor de mensagens, e como invocar seus serviços. É baseado em XML e suporta de simples a complexas transações definidas pelas mensagens trocadas. Da perspectiva de interoperabilidade, WSDL define a interface para o **Web service**. Se estas interfaces forem bem definidas, as possibilidades de interoperabilidade aumentam.

3.13 Service Oriented Architecture (SOA)

As aplicações podem ser desenvolvidas como um conjunto de serviços, oferecendo relacionamentos bem definidos a seus usuários potenciais Brown et al(2002) . A descrição de um serviço na arquitetura SOA é essencialmente a descrição de mensagens que são trocadas através de softwares que solicitam e fornecem serviços, permitindo o fluxo de mensagens. Um serviço pode ser considerado como um conjunto de tarefas relacionadas, possuindo descrições que podem ser expressas formalmente em uma ou mais linguagens para descrição destes definidos pela **Working Group W3C**.

SOA permite que os sistemas legados publiquem suas funcionalidades, algumas vezes sem fazer mudanças significativas nos mesmos. As características do SOA, tornam a interoperabilidade mais eficaz e fácil, com o baixo acoplamento, interfaces publicadas, e modelo padrão de comunicação. Construir serviços para os sistemas existentes a fim de obter os benefícios da arquitetura SOA, entretanto, não é automático. De fato, tal migração pode representar uma tarefa complexa da engenharia.

O SOAP (**Simple Object Access Protocol**), segundo **Working Group W3C**, é um protocolo para troca de informações em ambientes descentralizados e distribuídos.

O SOAP fornece um mecanismo simples para a troca de informações estruturadas em um ambiente descentralizado e distribuído utilizando XML. O próprio SOAP não define nenhuma semântica da aplicação tal como um modelo ou uma semântica de programação específica para a execução; em vez disto, define um mecanismo simples para expressar a semântica da aplicação, fornecendo modelos de pacotes modulares e mecanismos para codificar os dados dentro dos módulos.

Isto permite que o protocolo SOAP seja utilizado em uma grande variedade de sistemas. De forma resumida, o protocolo SOAP é dividido em três partes:

- O envelope: que define uma estrutura total para expressar o que está em uma mensagem.
- As regras de codificação: define um mecanismo de serialização
- A representação de RPC do SOAP: define uma convenção que pode ser usada para representar as chamadas e respostas remotas.

Para que haja significado na utilização do **Web services**, é necessário fornecer informações, além de especificações técnicas do próprio serviço. Partindo dessa premissa, surge o **Universal Description Discovery & Integration (UDDI)**. O foco do UDDI, ilustrado na figura 23, é a definição de um conjunto de serviços que suportam a descrição e a localização (1) de negócios, de organizações e outros **Web services**, (2) se os **Web services** estão disponíveis e (3) das interfaces pelas quais estes serviços podem ser acessados. Baseado em um conjunto comum de padrões da indústria, que inclui

HTTP, XML, e o SOAP, o UDDI fornece uma infra-estrutura interoperável para um ambiente de software baseado em **Web services** para serviços publicados e disponíveis externamente ou para serviços expostos somente dentro da organização.

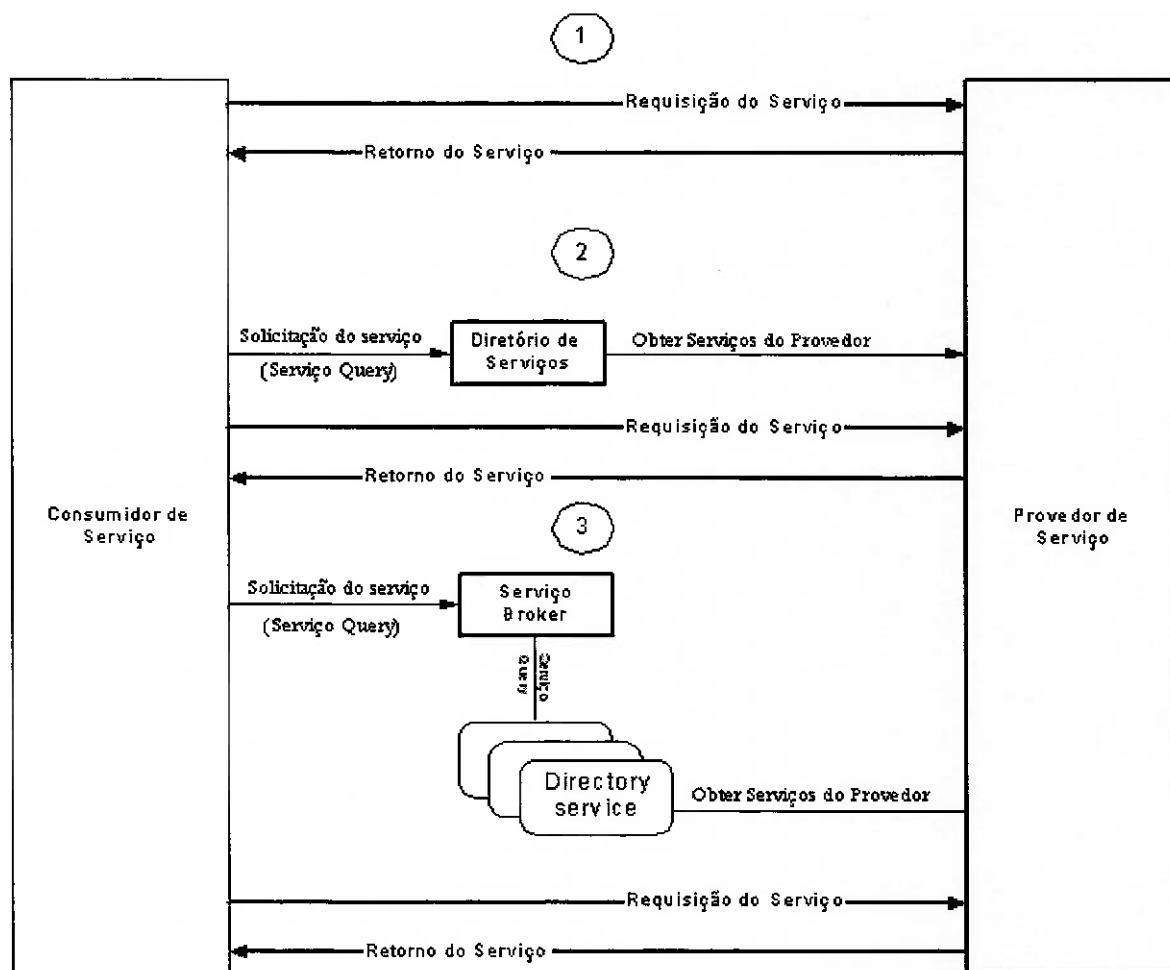


Figura 23 Diagramas entre provedor e consumidor de Serviços.

3.14 Enterprise Services Architecture (ESA)

Em particular, o conceito do ESA ou Arquitetura de serviços empresarial da SAP segundo Kaj Van de Loo(2005) habilita desenvolvedores integrarem aplicações heterogêneas e permitir seu uso simples, flexível, e independente da localização dos usuários finais.

O ESA proporciona um modelo para a concepção de soluções de negócio baseadas em serviços que sejam adaptáveis, flexíveis e abertas com um custo mínimo de propriedade ou TCO (**“Total cost of Ownership”**). Através do ESA as aplicações podem ser desenvolvidas sobre funcionalidades já existentes com o objetivo de aumentar o valor dos sistemas existentes e aumentar a automatização de novos processos de negócio.

O ESA estende a utilização dos **Web Services** para o ambiente empresarial.

Os serviços empresariais utilizarão a sintaxe e os modelos dos **Web Services** no sentido de implementarem vários requisitos de negócio como escalabilidade, robustez, segurança e facilidade de uso.

Na arquitetura orientada a serviço (via **Web services**) do ESA todas as aplicações são tratadas como serviços. A implementação técnica do SAP R/3 é baseada em uma combinação de tecnologias testadas, inovadoras, abertas e padronizadas.

4 Comparativos entre as interfaces

4.1 Introdução

Como se pode perceber, no momento em que a empresa necessita escolher por qual tecnologia de **Middleware** utilizar, ela encontrará no mercado diversas diferenças tecnológicas, siglas e nomes, mas se levar em conta algumas considerações importantes, este processo será facilitado. Cada uma das tecnologias tem suas características peculiares e nascem com um determinado propósito a cumprir. Será agora efetuado um comparativo entre as tecnologias abordadas neste trabalho.

4.2 Considerações ao comparar

Para efetuar esse comparativo serão considerados os fatores abaixo devido a serem os mais importantes no impacto do projeto de desenvolvimento e implementação do **Middleware**:

- Custos e necessidades de recursos (hardware, software e humanos)
- Tempo de total implementação
- Risco de adaptações das aplicações atuais para suportar a interface
- Complexidade para as atividades de **Sustaim** (suporte ao ambiente produtivo)
- Desempenho
- Tempo ao qual a interface ficara ativa
- Segurança
- Risco de impacto com as interfaces existentes.

Um resumo pode ser visto nas tabelas VII e VIII apresentada no item 4.4.

4.3 Importância desta comparação

A importância desta comparação é fazer com que a equipe do projeto leve em conta as tecnologias apresentadas neste documento devido aos diversos fatores já discutidos.

Os elementos aqui apresentados não possuem caráter científico ou comprovadamente eficaz para todas as situações, mas, entretanto, tem-se mostrado útil como mecanismos para a seleção de um **Middleware** em detrimento a outros. Neste ponto é importante esclarecer que se acredita não existirem metodologias científicas consolidadas para tal processo, dado a diversidade de situações que envolvem a implantação de um sistema ERP. Todavia o objetivo deste comparativo é demonstrar que tal procedimento é sempre útil e, sem dúvida, é uma estratégia ou um mecanismo efetivo para a seleção do melhor **Middleware** de ERP.

4.4 Comparativo

A tabela abaixo foi montada levando-se como base à implementação de uma interface entre um ERP da SAP/R3, que deverá disponibilizar os pedidos de compra que foram gerados para um sistema legado. O sistema legado deverá, por sua vez, ler essa informação e retornar para o ERP quais pedidos foram atendidos / efetuados. Este levantamento foi gentilmente cedido pela Empresa EDS (**Electronic Data System**) em novembro de 2006 e é utilizado mundialmente para tomada de decisões nos projetos que efetua em conjunto com a General Motors.

Atividade	Txt Files	RPC	DCOM	CORBA	RMI	RFC
Custos e Recursos	Baixo	Baixo	Médio	Médio	Médio	Baixo
Tempo de total implementação	Baixo	Médio	Médio	Médio	Médio	Baixo
Risco de adaptações das aplicações atuais para suportar a interface	Nenhum	Médio	Baixo	Baixo	Médio	Médio
Complexidade de Sustaim	Baixo	Baixa	Média	Média	Baixa	Baixa
Performance	Baixa	Alto	Médio	Médio	Alta	Alta
Segurança	Baixa	Média	Alta	Alta	Média	Média
Risco de impactar as interfaces existentes	Baixa	Baixo	Baixo	Baixo	Baixo	Nenhum

Tabela VII. "Quantidade" de esforço para a implementação do middleware

Atividade	EDI	XML	Web services	SOA	ESA
Custos e Recursos	Baixo	Médio	Alto	Alto	Médio
Tempo de total implementação	Baixo	Médio	Alto	Alto	Baixo
Risco de adaptações das aplicações atuais para suportar a interface	Médio	Nenhum	Nenhum	Nenhum	Médio
Complexidade de Sustaim	Alto	Médio	Médio	Médio	Médio
Performance	Alta	Média	Média	Média	Média
Segurança	Média	Baixa	Média	Média	Média
Risco de impactar as interfaces existentes	Alto	Baixo	Baixo	Baixo	Baixo

Tabela VIII. "Quantidade" de esforço para a implementação do middleware- continuação.

4.4 Prós e contras de cada solução

As tecnologias como RPC, RMI, RFC são similares por trabalharem de forma parecida e são modelos de componentes aceitos pelo mercado para interoperabilidade do tipo "**plug-and-play**", sendo estas a escolha para pequenas interfaces. Seu desenvolvimento e manutenção são relativamente simples e os projetos são de pequenos a médios prazos.

Os arquivos Textos ("**Text Files**") são amplamente utilizados por projetos em que a necessidade não é on-line, ou seja, os processamentos da informação são feitos em horários pré-definidos através de processamentos em batch. Existem diversos formatos de arquivos, por exemplo, CSV, TXT e DAT. Utilização de arquivos textos pode ser considerada como a forma mais simples de **Middleware** já que envolve baixo custo e esforço para a sua implementação. Normalmente o controle da segurança é feito nos programas que geram os arquivos e não no tráfego dos arquivos em si.

A tecnologia CORBA foi projetada desde o início como um infraestrutura multi-usuário de objetos distribuídos, prevendo interoperabilidade entre diferentes plataformas e linguagens oferecidas por uma variedade de fornecedores. Mas CORBA não especifica detalhes de implementação, não é uma aplicação, um ambiente de programação ou um mecanismo para implementação de interfaces de usuário. Em contraste, OLE/COM foi projetado como uma tecnologia monousuário para gerenciamento de documentos compostos, enfatizando a ligação e combinação de objetos e gerenciamento de eventos,

sendo definido para propagar mudanças de estado para múltiplas visualizações de um documento. Para isto o OLE/COM suporta funções para apresentação visual, negociação para apresentação do estado real entre aplicações e mecanismos para entrada de dados do usuário como “**drag-and-drop**” (Arrastar e soltar) e “**cut-and-paste**” (Copiar e colar) (Rosen, Michael; Curtis, David(1998).

CORBA é um padrão aberto da indústria que facilita a invocação remota de métodos através de interface públicas. No entanto, ainda não é um modelo de componentes aceito pelo mercado para interoperabilidade do tipo **plug-and-play**. No mundo Microsoft, o CORBA seria equivalente ao DCOM como substrato de comunicação.

Conforme RAJ, Gopalan(2000) embora se possa comparar as tecnologias de **Middleware** CORBA e DCOM, deve-se ter o cuidado de realizar comparações apenas entre tecnologias concorrentes. Por exemplo, a versão 3.0 do CORBA adicionou o modelo **CORBA Component Model (CCM)**, mas ainda se encontrava em fase de pré-especificação na ocasião da redação deste trabalho.

Já as tecnologias como SOA, ESA, EDI, XML e **Web services** são iniciativas independentes, algumas lideradas por grandes empresas como Microsoft, IBM, Oracle e SAP e outras por evangelistas da padronização. É muito cedo ainda dizer qual o formato final. Tudo indica, pela experiência anterior, que o padrão a ser estabelecido deve ser um padrão de fato e, portanto, terminar por obrigar empresas a abandonarem quaisquer tentativas de apropriação destes.

Estes padrões tendem a convergir rapidamente e, com isto, o estabelecimento de um padrão de fato. Conseqüentemente, uma massificação deste tipo de **Middleware** irá levar a uma redução dos seus custos de implementação. Esta redução sensível do custo aliado à simplicidade deste **Middleware** tende a fazê-lo de maior aceitação e, portanto, acelerando a sua adoção pelo mercado. Em se confirmando esta tese, poderá num futuro não muito distante ser comprovadamente a maneira mais eficiente e eficaz de se integrar ERP com sistemas legados. O importante a saber é que essas tecnologias são as tecnologias em foco hoje, com exceção do EDI, e que ainda estão surgindo novidades. Se por um lado, as implementações normalmente dependem de um prazo maior, de recursos e de treinamentos, devido ainda serem novas no mercado, por outro, elas vêm se

mostrando cada vez mais sólidas e mostrando-se muito eficiente na sua implementação e utilização.

5 Conclusão

As implementações de sistemas ERP são complexas e requerem consideráveis recursos de tempo e dinheiro – o que exige uma série de cuidados para que não sejam ultrapassados prazos e orçamentos. Se a implantação não for planejada corretamente, trará mais problemas que benefícios.

Dentre os problemas encontrados em relação à implementação de ERP, incluem-se a resistência à mudança das pessoas que fazem parte da organização, a necessidade de modificação de processos existentes nas organizações e o alto custo associado à implementação de um produto ERP.

Em relação a esses problemas, qualquer que seja a solução indicada, deve-se considerar as características comportamentais da organização ao implementar um ERP adequado.

Tendo em vista que as consequências das falhas num projeto de implementação de um ERP são prejudiciais, devido ao grande custo financeiro do projeto e ao grau de esforço exigido durante o projeto, deve-se fazer um bom planejamento do projeto antes de iniciar o processo de implementação.

Se por um lado, entende-se e a literatura comprova que muitos benefícios podem ser auferidos na adoção de um sistema ERP, porém os esforços e custos são altos e enfrentam-se grandes desafios. Por outro lado, o constante amadurecimento das soluções ERP, a constante qualificação dos usuários e a constante evolução da tecnologia, tendem cada vez mais a reduzir seus riscos, desafios e custos e aumentar seus benefícios, tornando os sistemas ERP fundamentais para a gestão das empresas modernas.

Dentre as diversas dificuldades na implantação de um ERP, processuais, comportamentais, recursos, organizacionais, culturais, etc., quase todos intangíveis ou de difícil mensuração, as tecnológicas são as mais simples e uma boa prática é a escolha do **Middleware** adequado. Integrar sistemas de grande porte, através de um conjunto de serviços independentes utilizando a arquitetura SOA, mostra-se entre os mais estáveis e

com perspectiva de crescimento, como pode ser visto no caso do ESA da SAP, oferecendo relacionamentos bem definidos.

5.1 TRABALHOS FUTUROS

Como trabalhos futuros, consideram-se a necessidade de desenvolver e aprimorar cada tecnologia abordada e suas respectivas técnicas de desenvolvimento / implementação.

Além dos conhecimentos tecnológicos são importantes também os conhecimentos processuais e de negócios que estão envolvidos na implementação do ERP e suas formas de **Middleware**.

Os aspectos relativos à mensuração do atendimento aos atributos de qualidade por estas táticas por constituir-se também como mais um trabalho a ser realizado.

Por fim, o Estudo aprofundado da Arquitetura de serviços empresariais (ESA) pode ser levado em conta devido ao grande investimento por parte da SAP e consecutiva aceitação pelo mercado mundial.

LISTA DE REFERÊNCIAS

SOUZA, Cesar. **Sistemas ERP no Brasil : (Enterprise Resource Planning) : Teoria e Casos**. 1 ed. Ed.Atlas, 2003. 368p.

CORRÊA, H.L. **Planejamento, Programação e Controle da Produção – MRP II / ERP: Conceitos, Uso e Implantação**. 2 ed. Ed.Atlas, 1999. 412p.

CLIFFE, S. **ERP Implementation**. Boston, Harvard Business Review, 1999.

DAVENPORT, Thomas. **Living with ERP**. CIO Magazine. 01/12/1998.

NAH, F.F.H.; LAU, J.L.S.; KUANG, J., “**Critical factors for successful implementation of enterprise systems**”. Business Process Management Journal, Vol. 7 No. 3, p 285-296, 2001.

DAVENPORT, Thomas. **Putting the Enterprise into the Enterprise System**. Boston, Harvard Business Review, Jul/Ago de 1998.

ROBEY, Daniel . **Learning to Implement Enterprise Systems: An Exploratory Study of the Dialects of Change**. MIT Center for Information Systems Research, Georgia, 2000.

SOUZA, Cesar; ZWICKER, Ronaldo. **Ciclo de Vida de Sistemas ERP**. Caderno de Pesquisas em Administração. São Paulo, FEA/USP, v.1, nº 11, p.46-57, 1º trimestre 2000.

Projedata. Disponível em : <<http://www.projedata.com.br/utilidade.htm>>. Acesso em 15 jul.2006.

SCHNEIDER, Neureither ; SCHLINDWEIN, Schungel: **The ABAP Developer's Guide to Java 2005 – SAP Press**

F. Heinemann; C.Rau: **Web Programming with SAP Web Application Server 2004** - SAP Press

SILVA Muniz, Paulo Sergio; **Tecnologia de implementação de Sistemas de Software** – Notas de Aula do curso de TS ano 2006.- PECE USP

DAVENPORT, Thomas. **Mission Critical: realizing the promise of enterprise system**. Boston, MA: Havard Business School Press, 2000.

SMITH, H. Jeff, KEIL, Mark e DEPLEDGE, Gordon. **“Keeping num as the project goes number”**, Journal of management information System 19, n.2, outono de 2001.

KEIL, Mark e ROBEY, Daniel **“Blowing the whistle on troubled software projects”**, Communications of the ACM 44, n.4 abr. 2001.

ALTER, Steve e GINZBERG, Michael. **“Managing uncertainty in MIS implementation”**, Sloan Management Review 20, n.1, outono 1978.

FRANZ, Charles e ROBEY, Daniel. **“An investigation of user-led system design : rational and political perspectives”**, Communications of the ACM 27, dez. 1984.

JOSHI, Kailash. **“A Model of users”** perspective on change: the case of information system technology implementation”, MIS Quarterly 15, n.2, jun. 1991.

KEEN, Peter W. **“Information system and organizational change”**, Communications of the ACM 24, jan. 1981.

Bancroft, Nancy H., Seip, Henning e Sprengel, Andrea (1998), **Implementing SAP R/3: How to introduce a large system into a large organization** (2. edição). Greenwich: Manning.

Slater, Derek (1999). "An ERP package for you... and you... and even you". CIO Magazine, 15/02/99.

Bingi, Prasad, Sharma, Maneesh K. e Godla, Jayanth K. (1999). "Critical issues affecting na ERP implementation". Information Systems Management, 1999, vol 16, no. 13, pp 7-14.

W. Noffsinger, R. Niedbalski, M. Blanks, N. Emmart, **Legacy object modeling speeds software integration**, CACM 41 (12) (1998) 80 -- 89. ACM Press

Keil, Mark, Mann, Joan e Rai, Arun. "Why software projects escalate: an empirical analysis and test of four theoretical models", MIS quarterly 24, n. 4, dez. 2000.

Kenneth C. Laudon, Jane P. Laudon. "Sistemas de informações Gerenciais" Administração da empresa digital. 5 Edição, 2005.

Irani, Zahir e Love, Peter E. D. "The propagation of technology management taxonomies for evaluating investments in information systems" Journal of Management information Systems 17, n.3, inverno 2000-2001.

L. Roberts. **The ARPANET and computer networks**. In A. Goldberg, editor, *A History of Personal Workstations*, pages 141-171. Addison-Wesley, Reading, MA, 1988.

ISC Internet Domain Survey - <http://www.isc.org/index.pl?/ops/ds/> - (acesso em 30/10/2006).

IPM - Impacta Pesquisa Periódica de Mercado -
<http://www.impacta.com.br/ipm/ipm.asp> (acesso em 30/10/2006)

Turban E., Mclean, W., Wetherbe, J., **Information Technology for Management: Making Connections for Strategic Advantage**, John Wiley & Sons, Inc., 1999.

W3C Extensible Markup Language (XML) 1.0 (Third Edition);
<http://www.w3.org/TR/2004/REC-xml-20040204>

Kaj van de Loo; **Enterprise Services Architecture** - SAP UK; 26 April 2005 ;
<https://www.sdn.sap.com/irj/sdn/weblogs?blog=/weblogs/>

W3C Web Services Description Language (WSDL) 1.1;
<http://www.w3.org/TR/2001/NOTE-wsdl-20010315>

Brown, A; Johnston, S.; & Kelly, K. **Using Service-Oriented Architecture and Component-Based Development to Build Web Service Applications**. Rational Software Corporation, 2002.
<http://www-106.ibm.com/developerWorks/rational/library/4860.html>.

W3C Web Services Architecture; August, 2003;
<http://www.w3.org/TR/2003/WD-ws-arch-20030808/>

W3C Simple Object Access Protocol (SOAP) 1.1; May 2000;
<http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

CLEVERLEY, James. **COM Plus Watch: COM Developer's Eagle Eye on the future of COM and DCOM**. [S.l.]: COMDeveloper, 1999. Disponível por WWW em:
<http://www.comdeveloper.com/complus/> (12 mar. 2006).

KRUGLINSKI, David. J. **Inside Visual C++**. 4.ed. Redmond, Washington: Microsoft Press, 1997. 986p. p.555-597. ISBN 1-57231-565-2.

ROY, Mark; EWALD, Alan. **Inside DCOM**. [S.l.]: Miller Freeman, Inc., Apr. 1997. Disponível por WWW em: <http://www.dbmsmag.com/9704d13.html> (12 mar.2006).

SZYPERSKI, Clemens. **Component Software: Beyond Object-Oriented Programming**. New York: Addison-Wesley, 1997. 411p. p.132-144; 178-217. ISBN 0-201-17888-5.

HORSTMANN, M.; KIRTLAND, M. **DCOM Architecture**, 1997. Disponível por WWW em: http://www.microsoft.com/library/backgrnd/html/msdn_dcomarch.htm (13 mar. 2006).

SUN MICROSYSTEMS, <http://java.sun.com/j2se/1.5.0/docs/guide/rmi/relnotes.html>

OBJECT MANAGEMENT GROUP. **What is CORBA**, 1999. Disponível por WWW em: <http://www.omg.org/corba/whatiscorba.html> (7 mar. 2000).

ORFALI, Robert; HARKEY, Dan; EDWARDS, Jeri. **Instant CORBA**. New York: Wiley & Sons, 1997. 313p. p.3-28. ISBN 0-471-18333-4.

SCHMIDT, Douglas. **Overview of CORBA**, 19 Jan. 2000. Disponível por WWW em: <http://www.cs.wustl.edu/~schmidt/corba-overview.html> (7 mar. 2000).

RAJ, Gopalan Suresh. **DCOM, CORBA, Java/RMI: A Step by Step Comparison**. Disponível por WWW em: <http://www.execpc.com/~gopalan/misc/compare> (15 jun. 2000)

SAMPAIO, Cleuton. **Soa e Web Services em Java** Rio de Janeiro: Brasport, 2006. p.01. ISBN 85-7452-267-8.

RosettaNet web site. Disponível em : <<http://portal.rosettanet.org/cms/sites/RosettaNet/>>.
Acesso em 06 dez.2006.

STEPHEN Mohr, SCOTT Woodgate. **Microsoft Biztalk Server 2004 Unleashed**.
SAMS 2004. Microsoft Corporation.

EDS – Electronic Data System do Brasil Ltda. www.eds.com.

ANEXO A

ARTIGOS ELETRÔNICOS

Gestão São Paulo, 30 de Setembro de 2004

ERP ainda é desafio em algumas empresas de grande porte

Ricardo Cesar

O que Petrobras, Votorantim Cimentos, Companhia Vale do Rio Doce, Sabesp e Furnas têm em comum? Todas são grandes corporações, todas têm faturamento na casa dos bilhões de reais e nenhuma delas conta com um pacote de Enterprise Resource Planning (ERP). Pode soar estranho que empresas desse porte tenham passado incólumes pela onda dos ERPs da década de 90 e sobreviveram até hoje – em alguns casos muito bem – sem um dos alardeados sistemas de gestão que dominam o mercado. Mas é exatamente isso que acontece – e essas companhias são apenas alguns exemplos.

Tudo indica, no entanto, que a lista encolherá rapidamente. Todas elas sem exceção estão em processo de implementação de um pacote ou já começaram a sondar possíveis alternativas. Uma pesquisa da International Data Corporation (IDC), que acaba de ser realizada no Brasil, confirma essa movimentação. O trabalho mostra as tendências de investimentos em diversos segmentos e os ERPs aparecem com destaque em setores como utilidades, governo e companhias de médio porte. “Este ano, muitas empresas implementarão projetos de gestão que ficaram de lado em 2002 e 2003”, afirma Ivair Rodrigues, gerente de pesquisas de TI da IDC Brasil. “Um vento de otimismo faz com que os projetos sejam desengavetados. Sempre digo que é errado imaginar que a fase de grandes implementações de ERP terminou.”

O analista conta que as áreas em que sistemas de ERP podem trazer maiores ganhos de produtividade, sobretudo para o governo, são gestão de contratos e projetos, módulos de planejamento financeiro e processos de compras e armazenamento de materiais. No segmento de manufatura, a maioria das empresas entrevistadas que estão priorizando implementações de ERP investirá mais em 2004 do que no ano passado.

Até que ponto os ERPs são indispensáveis para a gestão corporativa? A resposta varia de empresa para empresa. O diretor da prática de SAP da BearingPoint, Paulo Sérgio Rodrigues, acredita que as companhias que conseguiram se manter até hoje sem um ERP estão em sua maioria na área de commodity e freqüentemente trabalham com poucos produtos, contam com um canal de escoamento único e com processos de produção simplificados.

O consultor da Accenture, Ronald Munk, lembra que a decisão de adotar ou não um

pacote também passa pela saúde financeira da companhia. “Quando a empresa está ganhando dinheiro com os processos atuais e os sistemas que o CIO tem em casa estão dando conta do recado esses processos, não há um elemento interno ou externo que force a companhia a sair desse conforto”, diz.

Mas o consultor faz uma ressalva: é preciso ter visão de longo prazo. Se o CIO estiver avaliando apenas as necessidades atuais, a empresa pode estar à beira do abismo sem que isso seja notado. Esse é o grande risco de quem está em uma situação confortável. Saber a hora certa de mudar é fundamental, ainda que a companhia em questão esteja na liderança do seu setor.

Esse talvez tenha sido um dos maiores méritos da Companhia Vale do Rio Doce (CVRD). A mineradora está implementando, desde fevereiro de 2003, um sistema de ERP da Oracle que deve ir ao ar em outubro. O projeto faz jus ao tamanho da Vale e tem orçamento de US\$ 55 milhões, envolvendo uma equipe de 350 pessoas. Trata-se de um sistema único que atenderá até 14 mil usuários espalhados por 40 empresas do grupo.

Por volta de 1996, a CVRD se deparou com o dilema de preparar o emaranhado de sistemas de suas múltiplas empresas para o bug do milênio ou adotar um pacote de ERP. A companhia optou por realizar um gasto mínimo e manter os sistemas vigentes. “Naquela época havia uma disputa muito grande por recursos no mercado. Se fizéssemos o projeto em 1999, gastaríamos duas vezes e meia a mais do que estamos gastando hoje”, calcula João Luiz Nogueira, diretor de desenvolvimento de tecnologias da CVRD.

Mas, então, por que não postergar ainda mais? A consolidação das empresas do grupo levou a uma situação em que a CVRD ficou com diversos sistemas fazendo as mesmas funções. “Precisávamos acelerar e não tínhamos mais motor para fazer isso. O que tínhamos estava no limite”, diz Nogueira. “Fomos privilegiados em relação ao tempo da implementação. Não estamos agindo nem muito cedo, quando tudo estava excessivamente caro, nem muito tarde, diz Nogueira.”

Hora certa ou errada

O tempo certo para implementar um pacote de gestão, no entanto, não é igual para todas as empresa. A Sabesp, por exemplo, suspendeu a contratação do ERP que começou a ocorrer no ano passado. “Aquele projeto morreu”, afirma Fernando Menezes, CIO da companhia de saneamento. “A Sabesp tem ferramentas desenvolvidas internamente. Não dependemos do ERP para fazer nossa gestão. Todo benefício tem um custo. E ERP tem um custo alto.”

No caso da Yamaha, que inaugurou seu sistema de gestão em dezembro de 2003, o prazo correto talvez tenha passado. Pelo menos essa é a avaliação da gerente de sistemas da subsidiária nacional da empresa, Gisele Mayumi Ikeda. “Se eu pudesse escolher, teria feito a implementação há três anos”, diz. “A Yamaha não chegou a perder competitividade, mas estaríamos mais tranquilos e hoje poderíamos ter dado passos mais

largos se já estivéssemos com um ERP.”

A partir de 2002, quando a Yamaha Brasil passou por um período de acelerada expansão, os sistemas começaram a apresentar problemas. “Já estávamos no gargalo”, conta Gisele. A opção por um pacote da Oracle foi a saída encontrada. O projeto, que compreendeu toda a operação nacional, foi implementado em apenas sete meses e passa neste momento por uma fase de ajustes para eliminar alguns problemas.

A situação da Yamaha ilustra bem os prós e contras das empresas que demoraram alguns anos a mais para instalar seus ERPs (veja tabela ao lado). A companhia gastou menos, mas poderia estar em uma situação mais confortável se tivesse implementado o sistema um pouco antes.

Quanto mais antigos são os sistemas proprietários, mais caro é o suporte e o custo de manutenção. Assim como aconteceu com a CVRD, chegará um momento em que o custo de manter o sistema acenderá a luz vermelha no orçamento do departamento de TI e empurrará a companhia para um pacote. Isso não é inevitável, pois a empresa pode optar por desenvolver novos sistemas customizados ou atualizar seus software. Mas é uma tendência.

“Quando optamos por um ERP, analisamos a possibilidade de desenvolver sistemas customizados ou adquirir um pacote. O que mais pesou na decisão foi o tempo, porque os pacotes estão prontos”, diz Nogueira, da CVRD. Como se vê, não é necessariamente um mau negócio adiar a implementação de um ERP. Não há cedo ou tarde, mas a hora errada ou certa – e esta só o CIO de cada companhia pode dizer quando chega.

[Computerworld - Edição 402 - 11/02/2004]

27/02/2004

Disponível em

<http://computerworld.uol.com.br/AdPortalV3/adCmsDocumentoShow.aspx?Documento=27545>

Acessado em 02/10/2005

Petrobras: ERP de US\$ 240 milhões estréia em outubro

Ricardo Cesar

No dia 4 de outubro, a maior implementação de sistema de gestão empresarial já realizada na América Latina irá ao ar. A iniciativa do SAP da Petrobras, conhecida internamente na empresa como “projeto sinergia”, começou a ser desenhada no final dos anos 90, enfrentou contestação de fornecedores rivais que perderam a concorrência, foi interrompida em 2001 e agora promete finalmente sair do papel. Ao longo desse período, a estatal investiu US\$ 240 milhões - cerca de R\$ 720 milhões, atualmente - no ERP que terá 22 mil usuários em todo o território nacional.

A implementação propriamente dita começou em maio de 2000, com um grupo de profissionais da Petrobras e a consultoria Ernst & Young do Brasil, que posteriormente foi adquirida pela KPMG Consulting e se tornou Bearing Point. O plano inicial previa uma implementação em três anos, mas logo o escopo aumentou e passou a incluir a implantação da BR Distribuidora.

Em 2001 o escopo do projeto já estava bastante diferente, incluindo companhias que a estatal abriu no exterior. As implementações na Petrobras e na BR Distribuidora vinham sendo realizadas em paralelo, mas em outubro de 2001 ficou claro que as necessidades das duas empresas eram muito diferentes e decidiu-se paralisar o projeto da Petrobras para dar prioridade ao projeto da BR Distribuidora.

A implementação da BR Distribuidora terminou em julho de 2002 e se estabilizou até o final do ano. Em janeiro de 2003, o atual gerente do projeto sinergia, José Carlos da Fonseca, assumiu seu cargo com a missão de implementar o ERP na Petrobras. Cerca de 30% do trabalho estava concluído, parte porque já havia sido implementado na estatal antes de a iniciativa ser interrompida, parte porque foi possível replicar algumas customizações do sistema implementado na BR Distribuidora.

Ao longo de 2003, Fonseca montou uma estratégia de colocar o sistema no ar onde era possível. O passo seguinte foi testar a solução em um ambiente controlado. Assim, foi costurado um acordo com a petroquímica Refap, empresa independente integrante do sistema Petrobras. Em 5 de julho o ERP estreou na Refap com sucesso, contemplando 500 usuários, em uma versão compacta da solução da Petrobras.

Atualmente com 784 pessoas trabalhando no projeto, entre funcionários da Petrobras, Bearing Point, SAP e outras empresas que desenvolvem algumas customizações, o ERP finalmente está pronto para sua estréia. Na conta dos investimentos entram também 700 mil horas / homem de treinamento e 300 multiplicadores (professores).

A estatal acredita que o sistema trará ganhos em redução de estoque, logística e outros benefícios que podem totalizar um retorno de US\$ 490 milhões até 2007 em um cenário

conservador e até US\$ 650 milhões em uma perspectiva mais agressiva, justificando os pesados investimentos.

Ricardo Cesar 26/08/2004

Disponível em

<http://computerworld.uol.com.br/AdPortalV3/adCmsDocumentoShow.aspx?Documento=29016>

Acessado em 02/10/2005

Gestão São Paulo, 30 de Setembro de 2004

Vale do Rio Doce estréia ERP de US\$ 55 milhões

Ricardo Cesar

Entra em funcionamento hoje (01/07), o projeto de ERP da Companhia Vale do Rio Doce. A fornecedora Oracle e o braço de consultoria da IBM comemoram juntos o “go live” de um dos maiores projetos de sistemas de gestão já realizados no Brasil.

Por sua envergadura, a empreitada lembra as grandes implementações da época da febre dos sistemas de gestão, na segunda metade de década de 90. As semelhanças param por aí. Ao contrário do que se acostumou ver naquela época, o roteiro foi seguido à risca e não estourou nem em prazo nem em orçamento.

Os números impressionam: a iniciativa contou com verba de US\$ 55 milhões e nos momentos-chave mobilizou uma equipe de até 500 profissionais. O sistema irá ao ar simultaneamente em 41 empresas do grupo, atendendo a mais de 14 mil usuários, e em seguida deve ser estendido a outras 30 empresas nas quais a CVRD tem participação.

Como a Vale conseguiu colocar um projeto desse porte no ar em tão pouco tempo? Para responder a esta questão, é preciso esclarecer antes uma outra: como a Oracle foi selecionada para o projeto. A mineradora começou a estudar mais a fundo as possibilidades de implementar um pacote de ERP em 2001. No ano seguinte, elaborou um plano de viabilidade econômica, que contou com participação ativa do novo diretor de desenvolvimento de tecnologias, João Luiz Nogueira, que veio para a empresa após 13 anos na Accenture trazendo na bagagem a experiência de quem implementou diversos grandes ERPs ainda em atividade no Brasil.

Larry Ellison na linha

Naquela ocasião, a escolha da Vale estava entre a SAP e a Oracle, que, devido à robustez dos sistemas e pela capacidade de mobilizar recursos, mostravam condições de ganhar o projeto. Um dos segredos da mineradora para realizar a difícil empreitada dentro dos parâmetros estabelecidos em contrato foi a consciência de seu peso como cliente –

enorme para o mercado nacional, muito grande até mesmo para os padrões norte-americanos. O fornecedor que demonstrasse compreender isso ganharia muitos pontos na disputa.

A fatura começou a ser definida em favor da Oracle quando a secretária do presidente da Vale, Roger Agnelli, anunciou uma ligação importante: o senhor Larry Ellison estava na linha. O principal executivo da Oracle deixou claro que considerava essencial ter a Vale como cliente e que ele mesmo cuidaria para que não faltassem recursos e empenho para que o projeto fosse executado com sucesso em um prazo reduzido. Seguiram-se alguns e-mails de Ellison e até mesmo um filme de marketing em que o CEO da fornecedora falava sobre a importância da Vale.

“Esse envolvimento dos níveis executivos nos deu muita segurança e foi determinante para fecharmos o acordo”, afirma Nogueira, que hoje tem contato direto com dois vice-presidentes mundiais da software house que se reportam sem intermediários a Ellison. “A flexibilidade e o apetite da Oracle em ter a Vale como cliente pesou muito. Não há nada que pedimos a eles que não conseguimos”, diz. Em dezembro de 2002 foi assinado o contrato, com orçamento-base previsto de US\$ 55 milhões que, garante Nogueira, está sendo milimetricamente mantido. “Se não fosse assim, hoje eu não estaria ocupando o meu cargo.”

Contagem regressiva

Em janeiro de 2003 a Vale começou a treinar uma equipe de profissionais próprios que participaria da implementação, que se iniciou efetivamente dois meses depois. Apesar do planejamento minucioso, ocorreram alguns imprevistos. Durante o período do projeto, a CVRD envolveu-se em aquisições, fusões e trocas de controle acionário, o que obrigou a equipe de TI a mudar a rota em pleno vôo. “Tivemos que lidar com os desafios de um grupo em constante crescimento. De repente aparecia uma outra empresa enorme que passava a fazer parte da Vale e tinha de ser inserida no projeto”, conta Nogueira.

Por outro lado, antes mesmo de estreitar o ERP trouxe alguns benefícios. O principal foi o redesenho de processos e a instituição de melhores práticas gerenciais, uma iniciativa que surgiu em função do sistema, mas que começou a render frutos muito antes dele entrar no ar. Exemplo disso é o setor de aquisições e suprimentos, que aprimorou suas práticas de recebimento e expedição de materiais. Outro efeito colateral benéfico foi o projeto de ensino a distância da companhia, que surgiu para treinar os usuário na plataforma Oracle. Agora esse conhecimento em e-learning está sendo usado para a formação de diversas competências da companhia, por meio de uma área que a Vale chamada de “universidade corporativa”.

A partir deste mês, começa a operar a primeira – e principal – fase do projeto, que envolve a parte de finanças, controladoria, suprimentos e orçamento. Em outubro, inicia-se a segunda etapa, com os módulos de manutenção, comercial e recursos humanos, a serem concluídos até o fim de 2004, menos de dois anos após o pontapé inicial da

implementação. Pelos cálculos da Vale, o projeto vai se pagar em meados de 2007 – apenas três anos a partir da estréia da primeira fase. “É um prazo agressivo para uma empresa desse tamanho”, diz Nogueira. “Isso só é possível devido ao nível de comprometimento e patrocínio da nossa diretoria executiva e do presidente da Vale.”

De imediato, a primeira fase do ERP já trará um ganho expressivo: a visibilidade das informações do grupo, que até então ocorria em ciclos mensais, passa a ocorrer em ciclos diários. O sistema de gestão oferece a base de procedimentos, processos e tecnologia integrada para enxergar a operação diariamente, o que garantirá mais agilidade na gestão da empresa e nas tomadas de decisões. Quem disse que ERP não gera vantagem competitiva?

Cronograma do projeto

1996 – A CVRD se deparou com o dilema de preparar os sistemas legados de suas múltiplas empresas para o bug do milênio ou adotar um pacote de ERP já compatível. A companhia optou por realizar um gasto mínimo e manter os sistemas vigentes, que precisavam de poucos ajustes.

2001 – O tema de um pacote de ERP voltou à baila dentro da empresa, desta vez de forma decisiva. Ainda se cogitou promover uma atualização em grande escala dos sistemas proprietários legados. Porém, dois fatores – o tempo necessário para fazer essa atualização e a necessidade de instituir melhores práticas de gestão – fizeram a escolha recair sobre um pacote.

2002 – A Vale realizou um detalhado estudo de viabilidade econômica e iniciou o processo de seleção do fornecedor. No fim do ano, a Oracle foi escolhida.

2003 – Em janeiro foi iniciado o treinamento dos funcionários na plataforma Oracle e em março começou a implementação.

2004 – No dia 1º de julho vai o ar a primeira e principal fase do ERP, com os módulos de finanças, controladoria, suprimentos e orçamento atendendo a 41 empresas do grupo. Em outubro, começa a segunda etapa do projeto, que envolve as partes de manutenção, comercial e recursos humanos e será concluída até o fim do ano.

|Computerworld - Edição 411 - 23/06/2004|

Disponível em

<http://computerworld.uol.com.br/AdPortalV3/adCmsDocumentoShow.aspx?Documento=28478>

Acessado em 02/10/2005