

**Discussão de um Modelo Conceitual  
para  
*Enterprise Application Integration – EAI***

*Fernando Henrique Fernandez*

**Trabalho Final de Mestrado Profissional**

**Discussão de um Modelo Conceitual para**  
*Enterprise Application Integration – EAI*

**Fernando Henrique Fernandez**

**Março de 2004**

**Banca Examinadora:**

- **Prof. Dr. Ricardo Dahab (Orientador) – IC – UNICAMP**
- **Prof. Dr. Rafael F. Alves – FCMNTI – UNIMEP**
- **Prof. Dr. Luiz Eduardo Buzato – IC – UNICAMP**
- **Prof. Dr. Edmundo R. Mauro Madeira (Suplente) – IC – UNICAMP**

FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DO IMECC DA UNICAMP

Fernandez, Fernando Henrique

F391d      Discussão de um modelo conceitual para Enterprise Application  
Integration – EAI / Fernando Henrique Fernandez – Campinas, [S.P.:s.n.],  
2004.

Orientador : Ricardo Dahab

Trabalho final (mestrado profissional) – Universidade Estadual de  
Campinas, Instituto de Computação.

1. Software – Arquitetura. 2. Software – Compatibilidade. 3.  
Sistemas abertos (Computadores). I. Dahab, Ricardo. II. Universidade  
Estadual de Campinas, Instituto de Computação. III. Título.

# **Discussão de um Modelo Conceitual para**

## ***Enterprise Application Integration – EAI***

Este exemplar corresponde à redação final do  
Trabalho Final, devidamente corrigida e  
defendida por Fernando Henrique Fernandez  
e aprovada pela Banca Examinadora

**Campinas, 10 de Março de 2004.**

**Prof. Dr. Ricardo Dahab**  
**(Orientador)**

Trabalho Final apresentado ao Instituto de  
Computação da Unicamp, como requisito  
parcial à obtenção do título de Mestre em  
Computação, na área de Engenharia da  
Computação



**© Fernando Henrique Fernandez, 2004.  
Todos os direitos reservados**

## **Resumo**

A integração de aplicações é um item dentro do universo da tecnologia da informação com demanda em forte crescimento. A descentralização do processamento das aplicações, a transposição dos limites físicos das corporações sustentada pelo avanço tecnológico, bem como as constantes mudanças do ponto de vista dos negócios estimulam esse crescimento. A Discussão de um Modelo Conceitual para EAI (*Enterprise Application Integration*) explorada nesse trabalho tem por objetivo o estudo do problema da integração de aplicações, a partir do levantamento e esclarecimento dos seus requisitos e é concluída com a proposição conceitual de uma solução. Fazem parte do estudo os tipos de integrações conforme a camada onde estão os dados a serem integrados, são discutidos os métodos e a abrangência a serem considerados para realização do levantamento do projeto de integração, as várias estratégias e dispositivos utilizados para integração propriamente dita e finalmente é proposto um modelo conceitual que visa atender ao conjunto de requisitos de integração discutidos.

## ***Abstract***

*There has been a strong increase of the need for integration within the universe of Information Technology, spurred mainly by the decentralization of application processing, the transposition of physical limits of enterprises and the constant changes on business models. This work discusses and proposes a conceptual model for EAI – Enterprise Application Integration, having as starting point the study and clarification of its requirements, and concluding with the proposal of a conceptual solution. As part of this study different requirements are considered such as the types of layer-dependent data integration, the methods and scope of the integration project, and the several strategies and devices to be utilized in the integration itself. The resulting conceptual model aims at satisfying these requirements.*

## **Dedicatória**

Dedico este trabalho à Evelin,  
e aos meus filhos Vitor e Giovana,  
dos quais o apoio e o carinho foram e são fundamentais.

## **Agradecimentos**

Agradeço aos meus pais, José e Dulce, pelos ensinamentos tão preciosos que me ofereceram, sempre com paciência e persistência, sonhando juntos o meu futuro.

Ao Professor e grande mestre Fernando Antonio Vanini pela sua enorme colaboração e empenho na composição deste trabalho.

Ao Professor Ricardo Dahab, como orientador e pessoa, pela grande abertura, cordialidade e apoio oferecido em todos os momentos.

Aos amigos mestres, Rogério e José Antonio que na divisão de tarefas, conhecimentos e preocupações somaram esforços importantes ao longo do mestrado.

Aos Srs. Paulo R. Schincariol e Daniel Gavino da empresa Refrigerantes Xereta pelo incentivo e colaboração imprescindíveis.

Aos componentes da banca examinadora pela dedicação na revisão do trabalho.

A todos colegas de trabalho da Planin que colaboraram direta ou indiretamente com esse trabalho.

# Sumário

Resumo.....	vii
Abstract.....	vii
Dedicatória.....	viii
Agradecimentos.....	ix
Lista de Figuras.....	xii
Lista de Tabelas.....	xiii
<b>1 Introdução.....</b>	<b>1</b>
<b>1.1 O Ambiente de Software versus a Necessidade de Integração .....</b>	<b>1</b>
1.1.1 A tecnologia e o limite físico das corporações.....	1
1.1.2 Outros fatores relacionados à integração.....	4
<b>1.2 A evolução das “ferramentas” EAI e a evolução das necessidades de integração....</b>	<b>7</b>
<b>1.3 Proposta do Trabalho .....</b>	<b>9</b>
<b>1.4 Organização do trabalho .....</b>	<b>9</b>
<b>2 O escopo da integração.....</b>	<b>11</b>
<b>2.1 Descrevendo o Ambiente de Integração .....</b>	<b>11</b>
<b>2.2 Diferenciando a integração interna (EAI) e a externa (B2B) .....</b>	<b>13</b>
<b>2.3 Avaliando o tamanho do projeto de integração .....</b>	<b>15</b>
<b>3 Tipos e procedimentos de Integração realizados por camada.....</b>	<b>20</b>
<b>3.1 Integração das Plataformas Operacionais .....</b>	<b>20</b>
<b>3.2 Nível de Integração por Dados .....</b>	<b>20</b>
3.2.1 Definição .....	20
3.2.2 Etapas para integração por dados .....	23
<b>3.3 Nível de Integração pela Interface da Aplicação .....</b>	<b>28</b>
3.3.1 Definição .....	28
3.3.2 Os ERP’s – Arquiteturas, tipos de serviços e interfaces.....	30
<b>3.4 Nível de Integração por Métodos .....</b>	<b>32</b>
<b>3.5 Nível de Interface do Usuário .....</b>	<b>34</b>
<b>3.6 Avaliação da utilização dos níveis de integração .....</b>	<b>35</b>
<b>4 Tecnologias de Integração .....</b>	<b>37</b>
<b>4.1 Visão Geral.....</b>	<b>37</b>

4.1.1	Características do <i>Middleware</i> .....	37
4.2	<b>Orientação a Dados</b> .....	<b>39</b>
4.3	<b>Orientação a Mensagens</b> .....	<b>39</b>
4.4	<b>Chamadas de Procedimentos Remotos (RPC's)</b> .....	<b>40</b>
4.5	<b>Objetos Distribuídos (ORB's)</b> .....	<b>40</b>
4.6	<b>Monitores de Processamento de Transações</b> .....	<b>41</b>
4.7	<b>Servidores de Aplicação</b> .....	<b>42</b>
4.8	<b>Web Services</b> .....	<b>44</b>
<b>5</b>	<b>Impacto da Integração de aplicações</b> .....	<b>47</b>
5.1	Do ponto de vista técnico.....	47
5.2	Do ponto de vista dos negócios .....	50
<b>6</b>	<b>Premissas do Projeto de Integração</b> .....	<b>55</b>
6.1	Negócios , Integração e depois a Tecnologia. ....	55
6.2	A idéia é de Integração e não de Interligação .....	56
6.3	A integração deve ser imune à evolução do software e dos processos de negócio ..	57
6.4	A integração deve ser gerenciável.....	58
6.5	É necessário padronizar.....	59
6.6	Sequência das etapas do projeto de Integração .....	59
<b>7</b>	<b>Um Modelo Conceitual para EAI</b> .....	<b>61</b>
7.1	<b>O conceito do modelo</b> .....	<b>61</b>
7.1.1	Orientação a processos de negócio e gerenciabilidade.....	61
7.1.2	Construção da solução pela utilização de modelos; .....	63
7.1.3	Configuração da solução pela conjugação de componentes locais e remotos;.....	63
7.2	<b>Módulos da aplicação</b> .....	<b>65</b>
7.2.1	Desenho do modelo .....	65
7.2.2	Detalhamento do Modelo .....	66
7.3	<b>Características dos recursos tecnológicos a serem empregados</b> .....	<b>69</b>
<b>8</b>	<b>Conclusão</b> .....	<b>71</b>
<b>9</b>	<b>Bibliografia</b> .....	<b>73</b>
<b>10</b>	<b>Glossário</b> .....	<b>76</b>

## Lista de Figuras

Figura 1.1a – A Empresa Estendida – Adaptado de [Cherry – 2000] p.2 .....	3
Figura 1.1b – A Evolução da Tecnologia da Informação [Correa – 2001] p.2 .....	4
Figura 1.2 – Origens das Necessidades por EAI .....	6
Figura 2.1 – Hierarquia dos Sistemas .....	11
Figura 3.1 – Camadas de Dados das Aplicações .....	21
Figura 3.3 – Exposição de Interfaces nas Aplicações – API’s .....	29
Figura 3.3 – Nível de Processos .....	32
Figura 4.6 – <i>Middleware</i> Transaccional atualizando DBMS ´s heterogêneos .....	42
Figura 4.8 – Arquitura Básica dos <i>Web Services</i> – Adaptado de [W3CWS – 2002] seção 3.1 .....	45
Figura 4.1 – Comparativo dos impactos pós-EAI .....	49
Figura 5.2a – Visão do Processo de Integração do SCM .....	50
Figura 5.2b – Visão do Processo de CRM .....	54
Figura 6.4 – A Integração dentro do Processo de Negócio e do <i>Workflow</i> .....	59
Figura 7.1 – BAM – <i>Business Activity Monitoring</i> – Adaptado de [Hailstone – 2003] .....	62
Figura 7.2 – Modelo Conceitual para EAI .....	65

## **Lista de Tabelas**

Tabela 2.1 – Nível de complexidade das interfaces .....	18
Tabela 5.1 – Níveis CMM Adaptado de [ JSEPI – 2002] .....	56

# Capítulo 1

## 1 Introdução

### 1.1 O Ambiente de Software versus a Necessidade de Integração

No mercado as aplicações que realizam o papel de interligação e integração das aplicações são conhecidas como EAI (Enterprise Application Integration ou Integração de Aplicações Empresariais). EAI é o processo que direciona global e sistematicamente a integração [JBLN – 2001]. Segundo [Lublinsky – 2001] a demanda por EAI se deve principalmente à necessidade de integração originada da migração das aplicações monolíticas para um novo conjunto de aplicações formado por aplicações legadas, produtos de prateleira e aplicações com tecnologias emergentes para segmentos específicos. Rodando em um ambiente heterogêneo de *hardware* e *software* esse novo conjunto têm uma multiplicidade de aplicações cada uma com suas “ilhas de informação” dentro da infraestrutura da tecnologia de informação das corporações. A inexistência de aplicações com total abrangência sobre o negócio e mesmo a inviabilidade, sob vários ângulos, da troca das aplicações legadas por conjunto de aplicações unificadas coloca a integração de aplicações como um caminho inevitável [Butler – 2002]. De outro ponto de vista sua avaliação do retorno sobre o investimento demonstra ser mais rápido o caminho da integração de aplicações do que a substituição por aplicações integradas quando isso é possível. Podemos observar, na seqüência do texto, mais detalhadamente esses e vários outros fatores direcionando a necessidade por EAI.

#### 1.1.1 A tecnologia e o limite físico das corporações

Dentro do ambiente tecnológico atual, no mundo do *software*, as aplicações transcendem aos limites físicos das corporações devido à exigência do estabelecimento de relações “colaborativas” e “instantâneas” através de “caminhos digitais” que interligam as empresas ao mercado. Essas relações empresariais são de diversos tipos e na maioria das vezes o “caminho digital” utilizado é a Internet. Podemos ter como exemplos dessas aplicações:

- **Aplicações B2C** (*business to consumer*) – as empresas se comunicam diretamente com o seu consumidor final, disponibilizando ao mesmo, acesso à compra de seus produtos, informações comerciais (preços, prazos de entrega, processamento de pedidos) e técnicas

(informações do produto), serviço de atendimento ao consumidor e outros. Esses serviços normalmente podem ser encontrados pelo consumidor diretamente no *web site* da empresa;

- **Aplicações B2B** (*business to business*) – as empresas se comunicam com outras empresas clientes e fornecedoras na sua cadeia de suprimentos, trocando informações sobre pedidos de compras, entregas, informações financeiras e de crédito por exemplo.
- **Aplicações B2G** (*business to government*) – as empresas trocam informações sobre tributos, movimento fiscal e contábil, processamento de exportações e outras informações com órgãos governamentais.

Uma abordagem interessante [Cherry – 2000] apresentada na figura 1.1 descreve esse cenário, propiciado em parte pela web, como a “Empresa Estendida”. Seguindo esse ponto de vista uma empresa estendida é um negócio no qual os sistemas de informação operam dentro de uma arquitetura de aplicações distribuídas. Os principais componentes de uma empresa estendida são:

- **SCM** – *Supply Chain Management* – O gerenciamento da cadeia de suprimentos corresponde ao conjunto de aplicações responsáveis pela interligação de fornecedores, fabricantes, distribuidores e revendedores;
- **ERP** – *Enterprise Resource Planning* – O planejamento de recursos empresariais corresponde ao conjunto de aplicações integradas responsáveis pela sustentação interna das operações das empresas;
- **CRM** – *Customer Relationship Management* – O gerenciamento do relacionamento com o cliente corresponde ao conceito de unificação ou centralização da administração das interfaces de comunicação e das informações transacionadas em qualquer tipo de relacionamento com o cliente.

### “The Extended Enterprise”

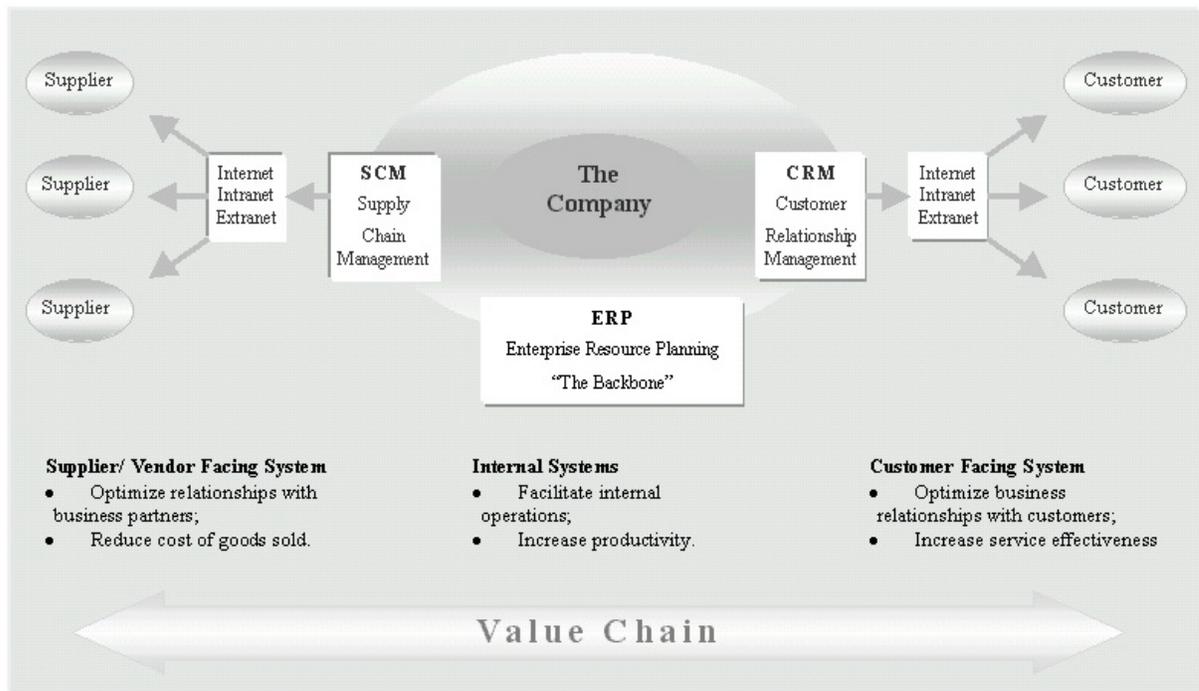


Figura 1.1a – A Empresa Estendida – Adaptado de [Cherry – 2000] p.2

Uma empresa torna-se estendida quando todos os participantes da sua cadeia de valor encontram-se completamente integrados. É interessante observar na figura 1.1b como a evolução da tecnologia ao longo das 4 últimas décadas [Correa – 2000] deixou mais complexa e ampla a atuação das aplicações comerciais, inicialmente com a automação departamental das operações, caminhando pela integração do planejamento empresarial e chegando aos processos extra empresa nos relacionamentos com parceiros e clientes.

**Evolução da Tecnologia da Informação:  
Agora é possível interagir com os fornecedores e clientes**

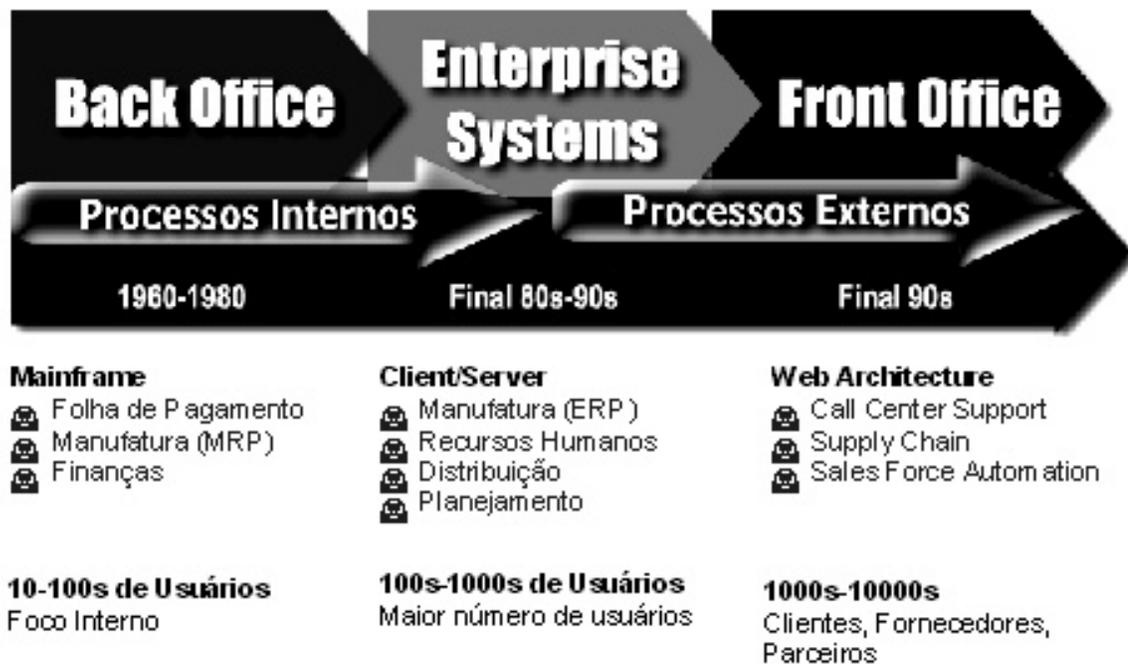


Figura 1.1b – A Evolução da Tecnologia da Informação [Correa – 2001] p.2

### 1.1.2 Outros fatores relacionados à integração

Além de observar o aparecimento desses tipos de aplicações que possibilitam às empresas ir além dos limites corporativos, podemos destacar uma série de outros fatores que causam novas necessidades de integração. Esses fatores afetam os processos de negócio e os sistemas baseados em tecnologia. Essas ocorrências têm transformado a área de tecnologia de informação das empresas num universo heterogêneo onde convivem aplicações legadas e aplicações de última tecnologia, de forma geograficamente distribuída. Esse universo heterogêneo e disperso criado demanda mecanismos para se realizar integração entre as aplicações. Alguns dos fatores que podemos citar são:

**Influência da ordem cronológica dos acontecimentos:**

- As épocas em que ocorrem as instalações ou atualizações de cada um dos tipos de aplicação são diferentes. Como exemplo podemos ter uma empresa que inicialmente possui aplicações desenvolvidas internamente e adquire em uma segunda época um pacote de gestão empresarial (ou como é conhecido no mercado um pacote *ERP – Enterprise Resource Planning*), que irá substituir parte dos sistemas legados (aplicações remanescentes com tecnologia ultrapassada). Em uma terceira época a empresa adquire no mercado, a partir de outro fornecedor e com nova tecnologia, um aplicativo para gerenciar o relacionamento com seus clientes (ou *CRM – Customer Relationship Management*). Mais fases de atualização ou implantação podem ser produzidas por aplicações menores, como as voltadas à automação industrial, as destinadas à automação da força de vendas e outras. As diferentes épocas para ocorrência de atualizações ou implantações de cada uma das aplicações podem ser originadas por eventualidades do negócio, inviabilidade da operacionalização conjunta de todas as mudanças, momentos financeiros ou da própria política nas organizações;
- Escolha de plataformas operacionais (S.O. – Sistemas Operacionais ou S.G.B.D. – Sistemas Gerenciadores de Bancos de Dados) distintas das anteriormente existentes por diversos fatores de conveniência;

**Fatos relativos a mudanças nas áreas de negócio:**

- Constantes fusões, aquisições de corporações e mudanças nos negócios – “em grandes empresas, as unidades locais desenvolvem muitas soluções de negócio. O isolamento de empresas semelhantes, as dificuldades de obter consenso e mesmo a competitividade entre as unidades de negócio desincentivam o compartilhamento de idéias e o desenvolvimento de soluções comuns [Cummins – 2002].” De outro lado a reestruturação dos negócios como aquisições, consolidações ou a criação de novas unidades de negócio obrigam a mudanças nos sistemas de informação, que são feitas da maneira mais conveniente e viável, em acordo com as possibilidades de cada situação em particular;

### Carência de planejamento com foco amplo em integração

- As aplicações em geral não contemplam ou normalmente não são construídas visando à integração em qualquer tipo de situação. As necessidades de integração atuais são resultado da construção de sistemas computacionais distribuídos sem planejamento ao longo de vários anos [Linthicum – 2000];

### Características ofertadas pelo mercado

- Pluralidade de fornecedores e padrões tecnológicos quase que diretamente proporcional à pluralidade de aplicações;
- Inexistência de um padrão consolidado de integração no mercado de *software*, também deixa em aberto a necessidade do uso de um "esquema comum" no desenvolvimento de aplicações, dotando as mesmas da capacidade de integração entre si e portanto viabilizando a total comunicação em qualquer ambiente em que o *software* seja instalado.

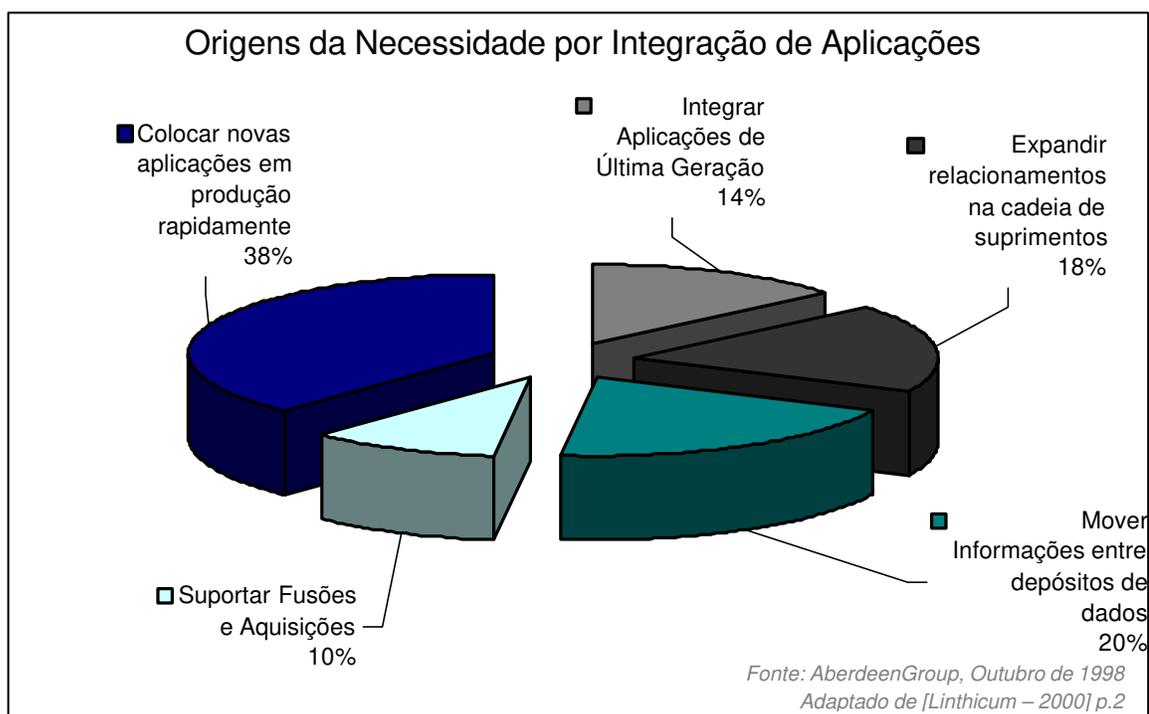


Figura 1.2 – Origens das Necessidades por EAI

Na figura 1.2 podemos observar dados sobre a proporcionalidade da influência de vários fatores sobre a necessidade de integração. A partir destas considerações e da propensão a um ambiente cada vez mais heterogêneo, surge naturalmente o crescimento da demanda por aplicações específicas de *software* que realizem o papel de integração. As aplicações precisarão sempre de “pontes” ou “tradutores” para viabilizar a troca de informações entre si.

## **1.2 A evolução das “ferramentas” EAI e a evolução das necessidades de integração.**

Segundo Nigel Stokes [Stokes – 2001] “o objetivo da estratégia EAI é integrar e criar um fluxo perfeito dos processos de negócio entre diferentes aplicações e unidades de negócio e ao mesmo tempo permitir a colaboradores, tomadores de decisão, e parceiros de negócio prontamente acessar as informações da empresa e dos clientes não importando onde as mesmas estejam localizadas”.

O papel das tecnologias focadas em EAI está diretamente relacionado à supressão de diferenças entre dados de diferentes fontes e padrões tecnológicos.

Reverendo um breve histórico sobre as ferramentas EAI percebemos que ao longo do tempo os diferentes tipos de EAI foram assumindo características e papéis diferentes. Inicialmente as ferramentas de integração, tanto como as aplicações de negócios, tinham um escopo mais limitado. As aplicações EAI inicialmente tinham a capacidade de realizar somente a integração entre bases de dados, sendo que posteriormente surgiram aplicações de integração mais sofisticadas. Hoje por exemplo aplicações como aquelas baseadas em servidores de aplicação monitoram as trocas de dados, mensagens, etc. e até gerenciam a segurança, a disponibilidade e a qualidade dos serviços. Avaliando a partir do ponto de vista dos desenvolvedores da integração, inicialmente os mesmos eram obrigados a trabalhar na camada de rede ou no mais baixo nível para conseguir a integração das aplicações. Em um segundo momento surgiram mecanismos conhecidos como *middleware*, que têm como papel principal esconder as diferenças tecnológicas entre as aplicações (sistema operacional, rede, etc.) minimizando as dificuldades de programação. Espera-se para um terceiro momento mecanismos inteligentes e adaptáveis que levem a programação próxima a zero [Linthicum – 2000]. Mesmo considerando que ocorreu um grande processo de evolução de vários pontos de vista, ele é ainda insatisfatório.

Podemos atribuir a evolução das ferramentas EAI a um conjunto de fatores históricos. Inicialmente a evolução ocorreu no campo da tecnologia de informação. Contribuíram para isso o barateamento e a melhoria do *hardware*, a disponibilização de recursos tecnológicos para a maior conectividade e também a melhoria nos processos de desenvolvimento de *software*. O reflexo desse conjunto de melhorias foi o avanço significativo das aplicações corporativas ao longo do tempo o que ocasionou novas exigências de integração. Esses fatos, avanços que criam novas necessidades, vêm ocorrendo a todo o momento e se repetindo em um movimento cíclico em que a própria tecnologia alavanca a quebra de seu paradigma contextual. Desde a “informática departamental”, baseada em aplicações isoladas, aos sistemas “monolíticos” de gestão empresarial (ou *ERP – Enterprise Resource Planning*) e chegando finalmente às atuais aplicações que suportam esse “mercado digital” (*e-marketplace*), a tecnologia nova rompe com os padrões anteriores. Cada vez que uma nova aplicação é incorporada forma-se um novo “conjunto de aplicações”, as tentativas de ajuste das aplicações a esse novo cenário buscam a adequação da convivência (comunicação) entre as mesmas. Podemos perceber ciclos nos quais as aplicações estão em um contexto formando um conjunto de aplicações ou um sistema único. No momento seguinte surgem novidades tecnológicas ou nos negócios que ampliam ou desarrumam esse contexto anterior. O sistema, formado pelo conjunto de aplicações, passa a ser uma parte ou um subsistema nesse ambiente “territorialmente ampliado” e naturalmente precisará de novas “habilidades de comunicação” para se recompor.

Referindo-se exclusivamente ao desenvolvimento de novas aplicações [Cummins – 2002] descreve também a evolução da tecnologia, por si só, como fator que contribui para a fragmentação de sistemas de informação. Quando novos sistemas são desenvolvidos é utilizada a tecnologia mais recente, já os sistemas pré-existentes criados com tecnologia mais antiga não podem ser simplesmente atualizados e implantados novamente o que leva automaticamente ao isolamento das aplicações. Os motivos do isolamento são devidos à incompatibilidade tecnológica dos padrões técnicos de cada aplicação, antiga e nova, que não se comunicam entre si. Também a incompatibilidade funcional ocorre com as transações ou processos das novas aplicações que possuem funções ou determinado fluxo de operações que não podem ser conjugados com as aplicações anteriores para formar o novo processo de negócio.

“Grupos de analistas como Gartner tem sugerido que estamos caminhando para o paradigma da empresa com latência zero, onde qualquer aplicação (ou transação) tem acesso à outra aplicação ou depósito de dados instantaneamente e sem restrição” [Linthicum – 2000].

### **1.3 Proposta do Trabalho**

A abordagem proposta neste trabalho é a da construção de um modelo conceitual de *software* voltado a EAI (Enterprise Application Integration). A troca de informações prevista no escopo do trabalho engloba dados formatados, documentos comerciais, requisições de serviços e outras tantas. Os processos de integração propostos, de um modo geral, se encarregam de fazer a tradução dos dados entre as partes comunicantes, oferecendo serviços que suportem as transações. O domínio dos processos de negócio sobre o qual essa aplicação de integração atuaria vai além dos processos internos, até os processos realizados com parceiros, consumidores, instituições governamentais ou outras unidades de negócios de uma corporação. O domínio da integração deve coincidir com o domínio de todos os processos de negócio dentro da organização, a partir da observação da extensão dos aspectos funcionais e não dos limites físicos da estrutura corporativa.

A consideração a respeito das funcionalidades que devem ser oferecidas pelo *software*, passa por entendê-lo como uma caixa de ferramentas onde estão disponíveis funções para quaisquer dos tipos de integração conhecidos. Ela se propõe também a disponibilizar a facilidade necessária a uma eventual manutenção com acréscimo de novas funcionalidades. Concluindo, defende-se a idéia de completude, maleabilidade e facilidade de manutenção que uma aplicação com essa vocação deveria ter para satisfazer o seu papel “integrador” dentro desse universo extremamente heterogêneo e mutante.

### **1.4 Organização do trabalho**

O trabalho inicia-se definindo o problema de integração (Capítulo 1), segue esclarecendo qual o escopo completo a ser tratado (Capítulo 2), as estratégias e métodos que podem ser utilizados para a abordagem e as tecnologias disponíveis (Capítulos 3 e 4), enfatizando o impacto causado pela integração e discriminando algumas premissas que devem ser consideradas nesses projetos

(Capítulos 5 e 6). Ao final é então sugerida uma visão conceitual para concepção de uma ferramenta EAI (Capítulo 7). O Capítulo 8 conclui o trabalho e o Capítulo 9 contém a bibliografia consultada. Por se tratar de um trabalho que envolve um grande número de conceitos e acrônimos, no Capítulo 10 está disponibilizado um glossário que apresenta definições para esses termos.

## Capítulo 2

### 2 O escopo da integração

#### 2.1 Descrevendo o Ambiente de Integração

O papel da tecnologia é o de produzir meios para suportar as atividades do negócio. A tecnologia não pode representar, na maioria dos casos, os fins ou objetivos do negócio. Para efeito de avaliar e definir o escopo da integração é obrigatório que levemos em consideração o escopo do negócio como ponto de partida na atividade de dimensionamento do projeto.

Quando olhamos um problema de integração exclusivamente do ponto de vista técnico e de forma isolada, sem levar em conta os processos de negócio periféricos, o problema na maioria das vezes parece simples e a solução óbvia. É comum, nesses casos, vislumbrar-se uma solução paliativa originada dessa visão parcial. Avaliar pontualmente o problema técnico pode resultar em uma decisão errada de investimento. Em curto espaço de tempo visões parciais poderão gerar resultados insatisfatórios e para evitar isso é necessário observar o contexto dos processos de negócio onde o problema técnico de integração está inserido.



Adaptado de [Cummins –2002] p.54

Figura 2.1 – Hierarquia dos Sistemas

Sugerimos então utilizar a estratégia para definição do escopo do projeto de integração adotando o escopo dos processos de negócio como delimitador.

Na figura 2.1 [Cummins – 2002] apresenta uma visão hierárquica que combina de forma interessante o aspecto organizacional ao tecnológico. Podemos observar pela Figura 2.1 e pelas definições subseqüentes, que há hierarquicamente conjuntos compostos de unidades de *software* combinadas de maneira sistêmica e organizada, com o objetivo de atender a estrutura organizacional.

Subindo a hierarquia temos:

#### **Componentes de aplicação**

- São unidades de *software* que podem ser agrupadas para produzir funcionalidades através de diversos tipos de implementação de *software* com diversas tecnologias;

#### **Aplicações de negócios**

- São módulos de *software*, onde cada qual têm funcionalidades específicas para atender a funções específicas de negócio. As aplicações de negócio são suportadas pela infraestrutura computacional e acessam bancos de dados da organização. Uma aplicação de negócio que não esteja vinculada a uma forma de construção monolítica poderá atender a diversos processos de negócio;

#### **Processos de negócio**

- São conjuntos de atividades que atendem a determinada área e que idealmente devem ser administrados com recursos de gerenciamento de fluxo de trabalho para que fiquem visíveis e gerenciáveis. Alguns processos de negócio também poderão estar disponibilizados para várias partes da empresa;

#### **Domínios de Sistema de Negócio**

- Em inglês *BSD – Business System Domain*, são conjuntos de processos e aplicações de negócio que compartilham componentes mantendo consistência entre o estado desses componentes. Um BSD tem a característica de estar situado em um só local. Um BSD pode estar atuando em um mesmo local com outro BSD, eles juntos formam um domínio do ponto de vista de segurança. Esse domínio único, do ponto de vista de segurança, está baseado em uma estrutura de rede local (*LAN – Local Área Network*) e a

mesma deve ser provida de determinado grau de isolamento. Um BSD pode também exercer o papel de controlador dos processos de negócio de outro BSD via integração;

### **Domínios corporativos**

- São conjuntos de BSD's que se integram através de uma infraestrutura apropriada provendo comunicação e coordenação às funções de negócios distribuídas pelas várias unidades organizacionais da empresa;

### **A Empresa Virtual**

- A empresa virtual, diferentemente da visão tradicional de empresa, incorpora além da visão interna da organização e suas unidades de negócios, a visão externa da empresa com seus clientes e parceiros de negócios. A diferença que é percebida na empresa virtual é que os processos envolvidos que estão na face externa têm maior isolamento e são mais difíceis de gerenciar, pois sobre eles não há domínio ou propriedade. Na face externa devem estar disponíveis recursos, normalmente baseados em Internet, para prover facilidades, segurança e qualidade em um ambiente para troca de informações e execuções de transações comerciais.

Essa visão demonstra a abrangência do problema da integração. O escopo do projeto de integração está delimitado em torno das informações dos processos de negócio envolvidos e da infraestrutura tecnológica necessária para suportar as transações da empresa virtual integrada. Os requisitos do projeto de integração serão obtidos a partir da exploração desse território. O conceito de [Cummins – 2002] sobre a composição hierárquica da empresa virtual coloca uma estrutura básica interessante para abrangência do EAI, onde o mesmo deve se estender desde os processos internos até a cadeia de suprimentos e outros tipos de B2B, aos aplicativos de relacionamento com clientes (CRM) e ao atendimento de consumidores via Internet (B2C).

## **2.2 Diferenciando a integração interna (EAI) e a externa (B2B)**

O processo de integração de aplicações até recentemente era subdividido por vários autores em duas partes, uma relativa à integração realizada internamente na empresa e a outra externamente. Essas subdivisões eram conhecidas por dois acrônimos: EAI – *Enterprise Application Integration* que se referia a integração intra-empresa e B2B – *Business to Business*

que se referia a integração extra-empresa ou entre parceiros de negócios. Nesta dissertação, pela própria conceituação de escopo já colocada, percebemos como inevitável a junção das duas partes, EAI e B2B, no processo de integração. A unificação dos conceitos no EAI é fundamental ao atendimento das características de integração demandadas pelo negócio. Os processos de negócio atendidos pelas aplicações não se limitam à empresa fisicamente ou institucionalmente. Para integrar é necessário abranger completamente a extensão dos processos de negócio.

[Linthicum – 2002] descreve a ausência de aplicações de mercado que conjuguem as duas funcionalidades de forma completa, em uma discussão sobre as tendências das aplicações EAI e B2B oferecidas e pontua as características para os dois tipos de integração como se segue:

#### **Características das tecnologias EAI**

- A tecnologia EAI é intrusiva e em muitos casos a aplicação a ser integrada é alterada para atender aos requisitos da integração;
- As ferramentas utilizadas no EAI, a tecnologia e os padrões de desenho dos sistemas origem e destino requerem um conjunto de recursos coesos e acoplados como por exemplo integração orientada a eventos, métodos de interligação e outros baseados em mensagens;
- O gerenciamento da integração envolve a semântica das aplicações e transações em curto espaço de tempo e , na maioria dos casos, com comportamento síncrono.

#### **Características das tecnologias B2B**

- A integração não pode ser invasiva e deve ter baixo acoplamento;
- A tecnologia B2B não pode ser intrusiva, pois nem sempre se tem o domínio de todas as aplicações envolvidas;
- O gerenciamento envolve a semântica dos negócios e as transações podem tomar um largo espaço de tempo, assumindo características de comportamento assíncrono.

### **Conclusão sobre as abordagens – EAI e B2B:**

As tecnologias tradicionais de EAI carecem de meios de segurança, confiabilidade e estabilidade para se comportar adequadamente em um ambiente externo (B2B), nesse caso mais disperso e heterogêneo. Também necessitam ter características mais acentuadas em relação a baixo acoplamento, para poderem atuar sem implicar necessariamente na imposição de adaptação às aplicações integradas.

Também é fácil concluir que para um cenário interno de integração de aplicações, havendo uma documentação mínima, há domínio sobre os aspectos semânticos da informação. Integrar as aplicações intra-empresa pressupõe ter mais facilidades de entendimento sobre os significados e especificação do uso das informações, pois há propriedade sobre elas. Para o caso de uma integração com vários parceiros de negócio, existe a necessidade de ser definido um padrão semântico para as transações, que seja adotado e compreensível a todos, bem como prever e prover adaptações para múltiplos padrões tecnológicos.

### **2.3 Avaliando o tamanho do projeto de integração**

O tamanho do projeto EAI [Levine – 2002] influencia o volume de recursos financeiros e de tempo a serem despendidos no projeto. A condução do projeto de integração está baseada na estimativa inicial realizada na fase de planejamento. Ela será seguida, utilizada para controle do andamento do projeto ou mesmo corrigida, quando observadas incompatibilidades durante a execução. Compreender claramente ou assertivamente o tamanho envolvido no escopo, significa realizar uma ação essencial para garantir o sucesso do projeto.

As dificuldades no desenvolvimento das estimativas a respeito da abrangência e recursos envolvidos estão diretamente ligadas à imprevisão criada pelo conjunto de partes desconhecidas a serem integradas. As partes não foram criadas para serem integradas e na maioria dos casos não há documentação ou domínio da sua especificação.

[Levine – 2002] sugere algumas etapas para orientar a criação de estimativas sobre os volumes envolvidos no escopo do projeto:

“(…)

- criar uma lista das interfaces;
- analisar as interfaces;

- estimar o esforço a ser despendido baseado na complexidade da interface;
- resumir todas as estimativas;
- revisar as estimativas e ajustar; (...) ”.

A criação das listas de interfaces se resume ao agrupamento das mesmas por categorias e subcategorias por tipo de integração a partir do levantamento do projeto. As interfaces podem ser classificadas como segue:

- a. **Migração de Dados** (*Data Migration*) – estes são os tipos de interfaces criadas exclusivamente para migração das informações entre os bancos de dados da aplicação desativada e a recém instalada. Essas interfaces têm o papel de realizar a carga dos bancos de dados das novas aplicações no momento de coloca-las em produção e se dividem em duas subcategorias:
  - Migração de Dados Mestres (*Master data Migration*) – transferência envolvendo os dados “estáticos” como cadastros por exemplo;
  - Migração de Dados Transacionais (*Transaction Data Migration*) – transferência de dados envolvendo transações de negócio como por exemplo dados relativos a pedidos de compra , venda, etc;
  
- b. **Pontos de Integração** (*Integration Points*) são os tipos de interfaces que estarão ativas no pós-implantação ou que simplesmente realizarão o papel de integração das transações do ambiente de produção. Subdividem-se em:
  - Pontos permanentes de integração (*Permanent integration points*) – são interfaces que permanecerão definitivamente ativas no ambiente de produção;
  - Integração temporária (*Temporary integration*) – são pontos de integração provisórios relativos a projetos executados por etapas, ao longo do tempo conforme o projeto vai sendo concluído esses pontos de integração são eliminados;
  - Múltiplos pontos centrais de integração (*Multiple master integration points*) – são interfaces relativas a cópias, réplicas, de bancos de dados que devem ser continuamente sincronizados. Esses tipos de integração podem ser derivados de

pontos de integração que são criados no início do projeto para a carga dos bancos de dados (*Data migration*);

A estratégia de divisão das interfaces em categorias e subcategorias separa por tipos os problemas ou situações a serem tratados pela integração. Em certas situações os dados simplesmente serão migrados em uma única vez, dados que corresponderão a transações permanentes entre as novas aplicações e as aplicações legadas. Pode também haver dados que, devido à adoção da estratégia de implantação por fases para as novas aplicações, deverão ser temporariamente integrados. Integração entre bancos de dados que são ciclicamente replicados ou sincronizados no ambiente de produção dentro de um intervalo de tempo definido.

A análise das interfaces e definição da complexidade envolvida, segundo [Levine – 2002], pode ser realizada através da medição de alguns fatores que podem ser classificados conforme sua complexidade em três níveis: alto, médio e baixo. A seguir na tabela 2.1, que trata o nível de complexidade das interfaces, podemos observar a aplicação dessa classificação:

Tabela 2.1 – Nível de complexidade das interfaces

Fator a Analisar	Métrica	Nível de Complexidade		
		Baixo	Médio	Alto
Mapeamento dos dados entre fontes e destinos	quantidade de campos	0 até 50	51 até 100	Mais que 100
	estrutura de dados	<i>single flat format</i>	<i>multiple flat format</i>	<i>hierarchical</i>
	conversão de dados	mapeamento campo-a-campo com conversão simples	combinação de campos com cálculos simples	múltiplos cálculos com dados derivados
	fluxo de dados	entrada única para saída única	entrada única para múltiplas saídas e vice-versa	múltiplas entradas para múltiplas saídas
Disponibilidades de metadados e gerador de formato de dados	capacidade de geração pelo <i>software</i> envolvido	geração automática	construção manual para mapeamento de dados de baixa complexidade	construção manual para mapeamento de dados de média e alta complexidade
Interfaces do legado e das novas aplicações	nível de mudanças exigidas	configuração ou pequenos ajustes	necessita de algumas alterações	precisa ser criada ou extensamente alterada
	influência do ciclo de atualização	<i>batch</i>	<i>near-time e event-driven</i>	<i>real-time</i>
Sistema Operacional	necessidade de configuração	modificação da configuração	criação de novas configurações	requisitos especiais
	influência do ciclo de atualização	<i>batch</i>	<i>near-time e event-driven</i>	<i>real-time</i>
Controle de erros	Necessidade de informação e interferência no processo.	parar a integração após o erro e editar o item de dados	indicar a regra que foi violada e se possível prosseguir com o processamento	realizar checagens cruzando os campos, identificar múltiplas regras violadas e sugerir modificações

Após a análise da complexidade das interfaces, deve-se fazer uma revisão nas interfaces extremamente complexas ou simples para verificar eventuais super ou sub-estimativas no levantamento [Levine – 2002] . O estabelecimento de critérios para definição dos níveis de complexidade das partes do projeto é importante para medir de forma mais objetiva o tamanho do problema abordado.

No tratamento da complexidade extra-empresa além dos pontos mencionados cabe a verificação de alguns pontos adicionais:

- Quais são as expectativas em relação ao tempo durante o qual o processo de integração permanecerá ativo para fundamentar uma análise de custo versus benefício;
- Quantos serão os parceiros de negócio que participarão, para analisar as possibilidades de utilização e aceitação de padrões, viabilizando a integração “em série”;
- Qual o grau de independência que os processos internos poderão ter em relação aos externos para que as interfaces se mantenham fracamente acopladas;
- Quem será encarregado e qual meio será utilizado na monitoração dos erros de processo;
- Qual o impacto de custos que poderá ser suportado pelos parceiros.

Visto que a integração extra-empresa não poderá ser invasiva e tendo em mente que o domínio técnico sobre aplicações e dados é nulo, o processo exigirá um esforço e um conjunto de ações diferenciados do processo de avaliação interno.

## **Capítulo 3**

### **3 Tipos e procedimentos de Integração realizados por camada**

Nesse capítulo serão discutidos os diversos níveis ou as camadas nas quais a integração pode ser realizada. Cada camada oferece diferentes possibilidades de integração, com variações nos custos, recursos e na forma de implementação.

#### **3.1 Integração das Plataformas Operacionais**

A integração entre as plataformas é o pré-requisito para os outros níveis de integração [JBLN – 2001]. Os sistemas operacionais compõem o ambiente ou a infraestrutura onde as aplicações atuam. A tarefa de obter interoperabilidade entre sistemas operacionais encontra seu maior grau de dificuldade nos sistemas operacionais mais antigos, aos quais não há mais suporte do fabricante ou nos casos em que a atualização pode provocar instabilidade.

#### **3.2 Nível de Integração por Dados**

##### **3.2.1 Definição**

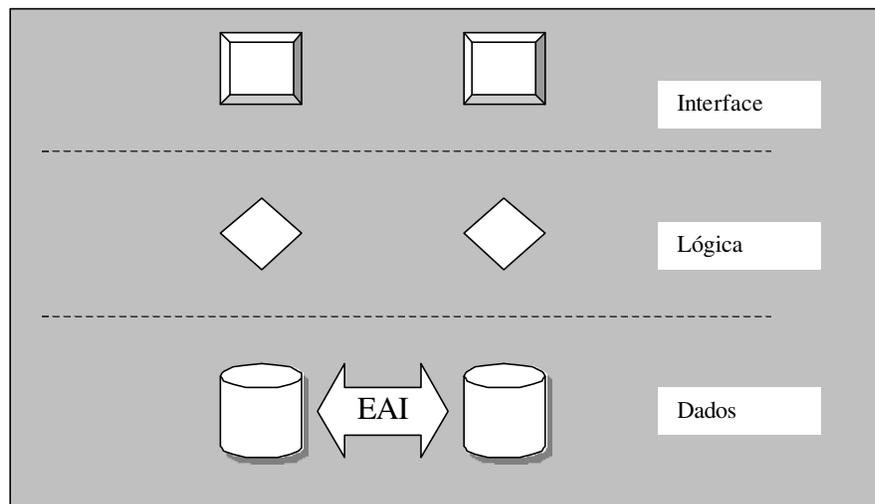
A integração realizada na camada de dados [Linthicum – 1999] é aquela que pressupõe integração através da troca de dados direta entre depósitos de dados sem envolver alteração na lógica das aplicações. Frequentemente é o ponto inicial da integração de aplicações [JBLN – 2001]. É a mais fácil de ser realizada porque não afeta diretamente as aplicações. O mercado oferece ferramentas e técnicas para sua realização e os fornecedores de bancos de dados disponibilizam sempre interfaces para o acesso aos dados e sua estrutura, ao contrário dos fornecedores de aplicações. Outro grande benefício deste tipo de integração é que o mesmo é suportado por muitas ferramentas e técnicas de EAI [Lublinsky – 2001].

A integração ao nível dos dados deixa de ser totalmente simples se levada em conta a necessidade de envolver:

- O vasto conjunto de tecnologias que podem compreender os bancos de dados envolvidos no escopo da integração;

- A complexidade do fluxo de dados corporativo contraposta à ausência de documentação detalhada e atualizada de lógica e regras de processos e dados;
- A necessidade de transformação de dados.

O processo de integração é simplificado quando as aplicações têm as camadas de dados, lógica e interface com o usuário desacopladas (Fig 3.1), tornando possível trabalhar somente com a primeira camada.



Adaptado de [Linthicum – 1999]

Figura 3.1 – Camadas de Dados das Aplicações

A camada de dados independente das camadas de lógica e interface proporciona uma integração simples, mais barata e com menos riscos. Não é necessário, nesses casos, adicionar ao processo da integração o processo de manutenção de *software* – levantamento, desenvolvimento, testes, etc.. Com a lógica das aplicações “fora” do processo de integração o caminho torna-se mais curto.

Antes de abordar a tarefa em si de integração de dados é interessante olhar para os dados sob dois aspectos, o semântico e o sintático. Em um estudo sobre a sintaxe e semântica dos dados [MacGoveran – 2002] define a integração de dados como a tarefa de tornar os dados acessíveis sintática e semânticamente de uma aplicação para outra aplicação. Normalmente os aspectos

sintáticos (formato) dos dados são mais considerados no processo de integração do que os semânticos (significados e regras de consistência da informação) na busca por compatibilizar origens e destinos. Seguindo sua tipificação podemos observar os dois aspectos:

### **Aspecto sintático**

- Envolvem o formato e a estrutura dos dados;
- Caracterizado pelo processo de conversão de um ou mais campos origem para um ou mais campos destino distribuídos ambos entre múltiplas tabelas e bancos de dados;
- Passa pela dificuldade de conversão de tipos de dados diferentes (eg. Numérico para caracter) ou mesmo envolvendo compatibilidade de precisão (eg. Representações numéricas com ponto flutuante). Em escala erros de precisão podem representar montantes significativos ou inconsistência de dados entre as aplicações;
- Não existe imunidade à diferenças na conversão devido à existência de tipos incompatíveis;
- A composição ou decomposição de dados em uma conversão de muitos para muitos campos pode afetar semânticamente a informação, violando a sua integridade.

### **Aspecto Semântico**

- Envolve o significado dos dados para o negócio;
- Após a conversão sintática os dados devem manter suas propriedades semânticas originais;
- A identificação semântica está ligada às operações de negócio que o produzem e utilizam;
- É necessário que a especificação da utilização da informação seja disponibilizada, o papel que ela representa e as restrições de integridade associadas, de forma explícita em um repositório com essa finalidade. Isso dificilmente é realizado nas organizações, pois normalmente essas regras estão dispersas nas linhas de códigos das aplicações ou inseridas nos bancos de dados;
- O conteúdo semântico do modelo de dados deve ser originado do modelo de dados da organização;

- É necessária a observância dos aspectos semânticos dos dados. A tarefa de integração se tornará muito repetitiva devido à necessidade de correções cíclicas nos processos de transformação que são semânticamente inconsistentes.

Concluindo o cruzamento das duas definições, o aspecto semântico agrupa o significado, a integridade e o valor da informação para o negócio. O aspecto sintático indica o formato que a informação tem na sua fonte e em seu destino, estabelecendo então a necessidade de tradução ou adaptação entre os formatos. Deve-se levar em conta que ao converter sintaticamente a informação, a mesma pode perder o seu significado comprometendo o resultado final da integração. Um dos principais problemas que ocasionam diferentes visões dos dados é relativo ao fato de que cada aplicação modela, dimensiona e entende cada dado pela sua visão particular, em detrimento da visão do todo da empresa [JBLN – 2001]. Isso acaba isolando as informações e dificultando a junção das mesmas entre diversas aplicações.

### **3.2.2 Etapas para integração por dados**

A integração ao nível dos dados demanda o levantamento e a análise de uma série de propriedades dos mesmos no ambiente pré-integração para que seja possível estabelecer o conhecimento sobre a matéria-prima básica do processo de integração. Ao realizar o processo de integração essas propriedades dos dados devem ser explícitas para garantir a execução e o resultado do processo.

Podemos observar que vários dos autores que abordam o assunto são unânimes em vincular a presença das dificuldades do processo de integração ao nível de qualidade e abrangência do modelo de dados da organização. Citando ainda os aspectos semânticos dos dados, que a priori podem ser os de mais difícil apuração, [MacGoveran – 2002] trata o assunto fazendo um comparativo no qual estabelece altos custos para criação de um modelo de dados durante o processo de integração. Recomenda-se a idéia da implantação de uma disciplina na organização que produza um trabalho incremental, mantendo atualizados e independentes os modelos conceituais, físicos e lógicos dos dados. A tarefa de criar essa disciplina e mantê-la ao longo do tempo pode poupar esforços e custos de maior vulto no momento da integração, produzindo uma relação custo versus benefícios muito mais interessante.

[Linthicum – 1999] trata a tarefa da integração por dados como tendo três etapas de preparação: identificação dos dados, catalogação dos dados e criação do modelo de metadados da organização

### **3.2.2.1 Identificação dos dados**

O dados segundo [Linthicum – 1999] estão dispersos pela organização e é necessário avaliá-los quanto aos aspectos técnicos e de negócio. Seguindo a abordagem de montagem do escopo de integração apresentada no capítulo I, podemos determinar a cobertura dos dados e a abrangência do modelo de dados da organização. Devemos percorrer de forma descendente a visão hierárquica da empresa virtual, de acordo com os componentes da figura 2.1 da página 10, até os componentes da aplicação. O modelo dos dados envolvidos, pode ser projetado conforme [Linthicum – 1999] utilizando-se de ferramentas com recurso de engenharia reversa sobre bancos de dados para acelerar essa fase, auxiliando nas tarefas de montagem do dicionário de dados, que deverá conter:

- A razão e a existência dos diversos elementos de dados;
- A determinação do proprietário de cada informação;
- O formato e os parâmetros de segurança com o papel lógico e físico da estrutura de dados;

Neste momento de identificação de dados percebe-se claramente o impacto da manutenção de um modelo de dados completo, pois em muitos casos serão encontrados dicionários de dados apresentando vários padrões diferentes; para certos bancos de dados os dicionários não existirão.

### **Identificação dos dados – definições de integridade**

- Obter as regras e procedimentos da construção do banco de dados;
- Obter as regras de integridade que por qualquer motivo foram implementadas na lógica das aplicações em detrimento aos bancos de dados;
- Observar os casos em que na verdade não existe um suporte de gerenciador de banco de dados e sim pseudo banco de dados com acesso direto à informação (nesse caso haverá necessidade do *middleware* escolhido oferecer suporte para o tratamento dessa situação);

Incluindo novamente a visão semântica de [MacGoveran – 2002], os métodos para obtê-la podem ser detalhados pelas seguintes ações:

- Recorrer ao repositório onde estão especificadas as definições semânticas – o que raramente se encontra totalmente disponível;
- Examinar o código fonte dos dados:
  - o É uma tarefa onerosa, tanto em relação ao tempo como ao custo envolvido, que pode gerar má interpretação e confusão pois as definições estão distribuídas;
  - o Existe a dificuldade de separar as restrições de integridade do negócio do algoritmo dos módulos de código e obter uma visão geral e hierárquica;
  - o Implica em regras de negócio de difícil abstração e obtenção, devido ao fato de a visão de seqüência e a prioridade de execução dos vários módulos estarem embutidas no código das aplicações. Essas regras em um nível superior implicam na compatibilidade transacional;
  - o Há restrições de integridade que são exclusivas de determinadas transações ou subtipos de dados e sua aplicação não pode ser generalizada.
- Entrevistar pessoas que podem fornecer o significado das informações
  - o É uma tarefa que normalmente oferece resultados pouco confiáveis;
  - o Os entrevistados podem não ter a visão completa sobre o significado das informações para o negócio;
  - o A comunicação entre o pessoal de negócios e o pessoal de tecnologia pode resultar em interpretações inadequadas, conforme já abordado por especialistas em extração de requisitos. Poucos profissionais de tecnologia terão habilidades para consolidar as múltiplas definições dos homens de negócio;
  - o Os homens de tecnologia e de negócio presentes no momento do desenvolvimento da aplicação podem não fazer mais parte do quadro de colaboradores da organização, pois hoje o *turnover* é alto;
  - o Aplicações de fornecedores externos podem fugir ao domínio do projetista da integração;
  - o A memória humana é falha e isso deve ser observado, pois mesmo que haja acesso aos desenvolvedores e homens de negócio que conceberam a aplicação, poderá não haver recuperação satisfatória das informações.

### **Identificação dos dados – latência das informações**

Outro ponto tratado por [Linthicum – 1999] na identificação dos dados é a definição da latência das informações, ou seja, os vários níveis de urgência com que cada informação deve ter suas atualizações propagadas. Esta pode ser classificada em três categorias:

- 1) Tempo real – *Real-time*
  - A necessidade do dado é em tempo curto ou imediata;
  - Caracterizada pelas aplicações como a de divulgação de preços de ações e processos que dependem de transações on-line. As organizações que precisam do tempo real podem ser as mais beneficiadas pelo resultado do EAI;
  - A implementação dessa integração requer cuidados devido ao fato de criar um grande número de acessos, tomando mais a capacidade dos gerenciadores de bancos de dados e da infra-estrutura e influenciando na performance.
- 2) Tão rápido quanto possível – *Near-time*
  - A necessidade do dado pode ser relacionada a intervalos de tempo, havendo a necessidade de se estabelecer qual intervalo é mais adequado;
  - Caracterizado pelas aplicações *off-line* que recebem remessas de dados em intervalos de tempo;
  - Nesse caso existe também a necessidade de observância de performance pois a frequência dos ciclos e o volume envolvido podem revelar impactos significativos;
- 3) Sem repetição – *No-time*
  - O dado somente é atualizado na sua criação;
  - Oferece menos riscos à performance pois o processo é menos contundente;

### **Identificação dos dados – formato dos dados**

O formato dos dados implica basicamente na estrutura usada para representar a informação (referido por [MacGoveran – 2002] como a parte sintática) e [Linthicum 1999] distingue basicamente as propriedades relativas ao tamanho, tipo de dados (numérico, carácter, etc) e forma de armazenamento utilizada (binário, texto, etc).

### 3.2.2.2 Catalogação dos Dados

Após entender os bancos de dados do ponto de vista lógico e físico é necessário catalogar os dados para obter os metadados que fluem pelo domínio do problema. Os metadados são a base do nível de dados no EAI [Linthicum – 1999\_2].

O catálogo de dados representa o dicionário de dados mestre da organização onde todas as informações estão consolidadas. É uma especificação que envolve grande volume de tempo para ser confeccionada e deve conter:

- Informações sobre os sistemas;
- Definições de segurança;
- Qualificação dos proprietários das informações e dos processos correlatos;
- Mecanismos de comunicação utilizados;
- Aspectos de integridade;
- Formato de dados.

O catálogo também deve apontar para eventuais redundâncias existentes.

A catalogação dos dados promove então a explicitação requerida das informações sobre os dados do domínio do problema e segundo [Linthicum – 1999\_2] apesar de não ser o modelo de metadados da organização nele estão contidas as potenciais soluções do problema. Após a obtenção do catálogo dos dados, o modelo de metadados pode ser construído. O modelo de metadados contém a base para estabelecer a troca de mensagens, referências para localização de dados, regras, lógica transacional, etc.. O modelo de metadados também será necessário nos outros níveis de integração (métodos, interface de usuário, interface de aplicação), pois ele contempla a visão consolidada dos dados que é necessária em todos os casos.

Segundo [Linthicum – 1999\_2] podemos produzir dois modelos de metadados, o modelo lógico e o modelo físico. O modelo lógico é uma forma de representação independente da visão física e serve de base para a visão dos arquitetos desde o ponto macro até os detalhes. O principal componente do modelo conceitual é o diagrama entidade relacionamento, que independente da metodologia aplicada. É o meio mais utilizado para descrever as entidades com seus relacionamentos e atributos. O uso de ferramentas pode auxiliar o processo de documentação e a própria criação do banco de dados do EAI. O modelo físico é mais difícil de ser criado, pois depende da compatibilidade dos bancos envolvidos (relacional, orientado a objeto, etc). O

modelo físico compreende o desenho do banco, a organização dos arquivos, os métodos de acesso empregados e o esquema de indexação. Na prática significa criar um banco mapeando todos os outros encontrados na organização.

Em resumo, o trabalho no nível de dados significa obter como resultado final um banco de dados conceitualmente único para a organização, facilitando a interpretação e a administração da informação. A normalização em busca de extinguir as redundâncias existentes não é uma tarefa simples pois implica na maioria dos casos em alteração das aplicações existentes no ambiente de integração, o que sempre se quer evitar. Pode-se conseguir grandes melhorias com diminuições parciais de redundância, aumento da consistência das informações a partir da equivalência do estado de atualização entre as mesmas e o aumento da confiabilidade dos dados. Por ser um processo complexo e extenso é interessante que ele faça parte da rotina da tecnologia de informação da organização sendo gradualmente fomentado e não como um grande e único evento dentro do processo de integração em si. Esse processo é possível e desejável tecnicamente, mas nem sempre viável do ponto de vista econômico.

Como vimos até aqui, uma das divisões do problema e da solução de EAI [Linthicum – 2000] está no nível do modelo de dados e a outra, explorada nas próximas seções, está no nível do modelo de negócios. O nível do modelo de negócios pode ser subdividido em três partes:

- nível de interface de aplicação;
- nível de métodos;
- nível de interface do usuário.

### **3.3 Nível de Integração pela Interface da Aplicação**

#### **3.3.1 Definição**

A integração pela interface da aplicação [JBLN – 2001] é um tipo de integração de mais alto nível onde uma aplicação utiliza funcionalidades disponibilizadas ou expostas por outras aplicações, o que provoca reutilização do código das aplicações. Segundo [Linthicum – 2000] as interfaces de aplicação são interfaces disponibilizadas pelos desenvolvedores das aplicações. As interfaces podem permitir acesso a processos de negócio, aos dados ou a ambos. As interfaces das aplicações criam então a possibilidade de compartilhamento de dados e serviços. Os recursos

encapsulados na aplicação passam a ter disponibilidade de acesso pelas interfaces, eliminando a necessidade de ações intrusivas. Essa definição pode ser comparada em [Lublinsky –2001] ao nível de integração por mensagens – *Message-Level EAI* – onde o mesmo considera que esse tipo de integração produz uma troca mais controlada de informações que o nível de integração por dados, contudo é mais invasivo por exigir modificação nas aplicações.

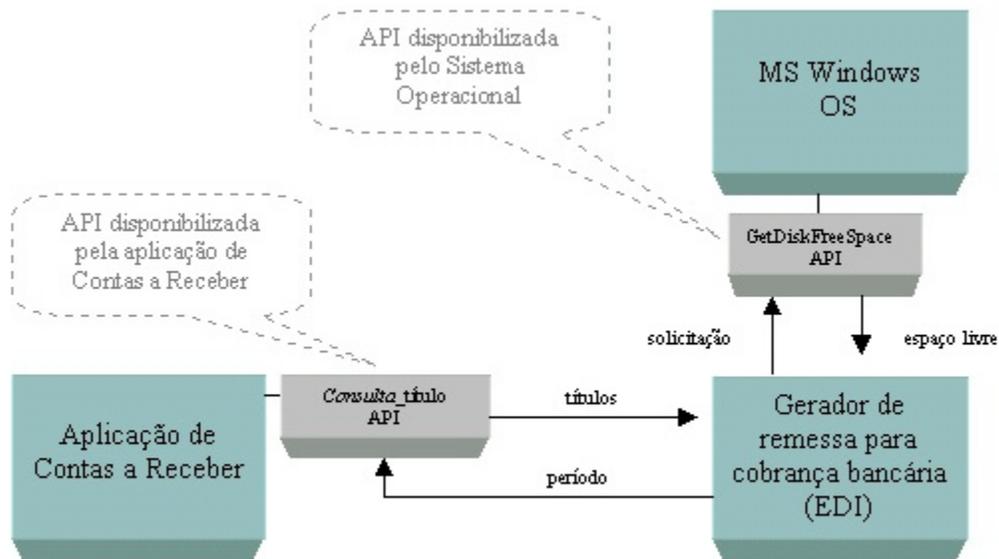


Figura 3.3 – Exposição de Interfaces nas Aplicações – API's

API's – *Application Programming Interfaces* – são mecanismos criados e disponibilizados para que outros desenvolvedores possam utilizar serviços como os de bancos de dados, sistemas operacionais, sistemas de automação, servidores de aplicação e outros. Segundo [JBLN – 2001] as API's permitem uma alternativa de integração de mais alto nível, onde uma aplicação pode utilizar uma funcionalidade existente em outra aplicação. As API's se conectam e acessam os recursos, podendo ser utilizadas no desenvolvimento de aplicações e integração. As API's expõem, os recursos encapsulados na aplicação. As API's mais completas e flexíveis tendem a facilitar o trabalho de integração de aplicações. A ordem de amplitude e flexibilidades desse mecanismo de *software* é diretamente proporcional à facilidade do processo de integração via interface. As API's completas e flexíveis, além de facilitar o processo de integração, provocam

uma drástica redução de custos. As aplicações que expõem suas funcionalidades via API's, como exemplo da figura 3.3, além de criar facilidades de integração promovem também a reutilização dessas funcionalidades.

### 3.3.2 Os ERP's – Arquiteturas, tipos de serviços e interfaces

Os ERP's fazem parte, ou a maior parte, dos casos de integração de aplicações. A maioria dos ERP's atuais fornece interfaces habilitando o acesso a serviços e dados. O fornecimento das interfaces, por si só, não implica sempre em garantia de atendimento das necessidades de integração. Alguns dos pacotes ERP's oferecem interfaces com padrões abertos (JavaBeans, CORBA, etc.), alguns oferecem soluções proprietárias e outros simplesmente não oferecem nada, dificultando a integração. Nesse último caso, o não fornecimento de interfaces provoca um encarecimento em torno de 50% dos custos relativos à parte de integração no momento de implantação do pacote [Linthicum – 2000].

#### 3.3.2.1 As arquiteturas dos pacotes ERP's, segundo [Linthicum – 2000], são de três tipos:

- Arquitetura centralizada – onde dados, lógica e interface com o usuário estão centralizadas na mesma máquina (normalmente a situação *mainframe* versus terminais). A manutenção é mais fácil e a integração também devido à unicidade de processamento;
- Arquitetura de duas camadas – *two tier* – é conhecida como a arquitetura cliente/servidor. Nesse caso as três partes (interface, lógica e dados) estão fisicamente dispostas em dois locais, o cliente e o servidor. A camada de dados e em alguns casos, a lógica de negócios, está localizada no servidor. A interface com o usuário normalmente fica no cliente; em alguns casos, a lógica de negócios também pode ficar. A maior concentração da lógica no servidor indica um “cliente leve” – *thin client*. A maior concentração de lógica no cliente indica um “cliente pesado” – *fat client*.
- Arquitetura de três camadas – *three tier* – as três partes ficam totalmente separadas. Há um servidor para os dados, um servidor de aplicações para a lógica e o cliente com

a interface do usuário. Nesse caso a comunicação das camadas é separada, a parte de dados se comunica com a parte lógica que se comunica com a interface. Normalmente não ocorre a ligação direta entre dados e interface com o usuário.

### 3.3.2.2 Outras arquiteturas dos pacotes ERP's

Além das arquiteturas citadas existem outras arquiteturas com camadas adicionais. Segundo [Linthicum – 2000], alguns ERP's , por exemplo, utilizam-se de servidores *web* para distribuir *thin clients* através de *browsers*. Os pacotes, via servidores de aplicação e utilização de API's, enviam conteúdo das transações a *web servers* que as transformam em páginas HTML e as enviam aos clientes. Esse método de distribuição acaba criando uma camada intermediária entre a interface do cliente e a lógica.

### 3.3.2.3 Tipos de serviços disponibilizados pelos ERP's: [Linthicum – 2000]

- Serviços de Negócio – são interfaces para qualquer lógica de negócios. Na prática fornecem acesso aos dados do pacote adicionando controle de integridade;
- Serviço de Dados – são interfaces para criar um caminho direto aos bancos de dados físicos e lógicos. Elas podem variar entre si quanto ao conjunto de funcionalidades oferecido, contudo o seu uso normalmente é direcionado à extração de dados. Os mecanismos disponíveis podem ser com ou sem controle de integridade.

Esses dois serviços podem ser fornecidos agrupados em formato de objetos. Esses objetos nem sempre são objetos distribuídos padrão (DCOM, JavaBeans, CORBA) e sim objetos proprietários.

### 3.3.2.4 Tipos de Interfaces: [Linthicum – 2000]

#### **Interfaces com todos os serviços – *full service interface***

- Fornecem acesso ao nível dos serviços de negócio;
- Fornecem acesso ao nível de serviço de dados;
- Fornecem acesso ao nível de objetos;

#### **Interfaces com serviços limitados – *Limited-service interface***

- Normalmente fornecem acesso a um dos níveis;
- O acesso ao nível normalmente abrange um conjunto limitado de serviços;

#### **Interfaces controladas – *controlled interfaces***

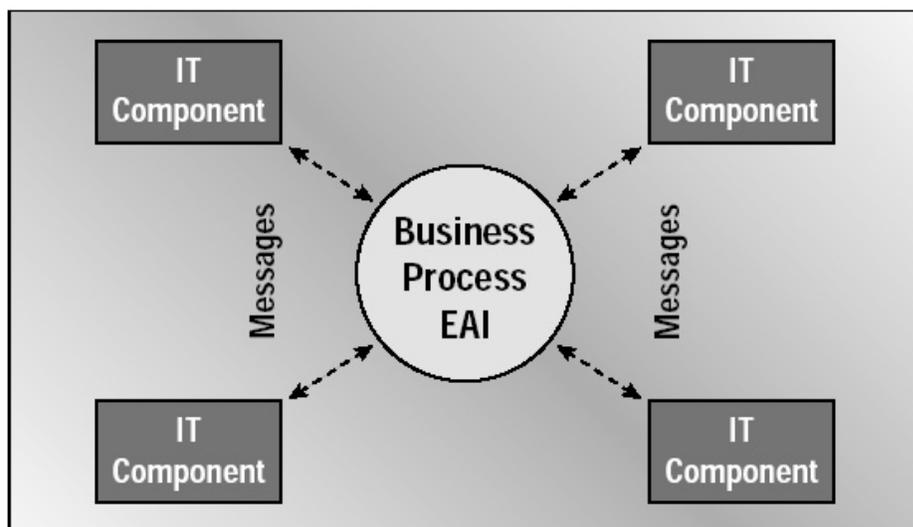
- Fornecem um número extremamente limitado de funcionalidades com características proprietárias e fechadas.

#### **Exemplo de algumas outras interfaces direcionadas a mercados específicos:**

- SWIFT (*Society for Worldwide Interbank Financial Communication*) – é um padrão de comunicação que oferece uma série de serviços para indústria financeira internacional;
- FIX (*Financial Information Exchange*) padrão desenvolvido para troca de dados de forma segura, utiliza uma espécie de dialeto XML o FIXML;
- HL7 (*Health Level 7*) – padrão desenvolvido para troca eletrônica de dados no mercado de saúde, também utilizado na indústria biomédica e farmacêutica.

### **3.4 Nível de Integração por Métodos**

Segundo [Linthicum – 2000] o nível de integração por métodos ou *method level EAI* ocorre pelo compartilhamento dos processos de negócio. Os processos de negócio podem ser compartilhados por servidores de aplicação ou via objetos distribuídos.



Process Level EAI – Fonte [Lublinsky – 2001] p.3

Figura 3.3 – Nível de Processos

[Lublinsky – 2001] classifica esse nível de integração como nível de processos – *Process-level EAI* – e considera o mesmo como uma forma orquestrada de troca de mensagens, onde algum tipo de *middleware* atua como um mecanismo de controle de execução dos processos. “O *Process-level EAI* vê a empresa integrada como um conjunto de processos inter-relacionados”.

É interessante observar que essa técnica tem como espinha dorsal o reuso e este constitui a espinha dorsal do EAI.

Esse paradigma de integração por um lado se evidencia como a melhor solução EAI, contudo essa é a solução mais invasiva. Para se ter o melhor resultado para essa solução as aplicações do projeto acabam por ter seu código alterado. É um processo muito caro que exige o re-desenvolvimento das aplicações do projeto. A análise de viabilidade dos integradores quanto a utilização desse tipo de integração passa por verificar se o valor que ele agrega ao negócio não será ultrapassado pelos custos envolvidos com o seu emprego. Alguns conceitos a serem definidos dentro desse nível de integração [Linthicum – 2000]:

- Processo – “para fins de EAI, um *processo* é qualquer regra ou pedaço de lógica que existe dentro da empresa e que tem um efeito no *como* a informação é processada.” O grande problema não reside na identificação dos processos e sim na heterogeneidade de tecnologia sobre as quais estão baseados;
- Cenários – para implementar o *method-level EAI* é necessário entender os processos da empresa para escolher a melhor maneira de agir. É necessário avaliar os cenários dos processos quanto a regras, lógica, dados e objetos:
  - Regras – *rules* – são o conjunto de condições existentes com o objetivo de controlar o fluxo de informações;
  - Lógica – *logic* – que é a seqüência de instruções de um programa. A lógica pode ser por processamento seqüencial, seleção (tomada de decisão), iteração (repetição);
  - Dados – *Data* – As informações compartilhadas entre as aplicações do ambiente de integração;
  - Objetos – *objects* – são a combinação de dados encapsulados em um objeto acompanhados de métodos que agem sobre eles.

### 3.5 Nível de Interface do Usuário

A integração via interface do usuário normalmente é utilizada quando não há outra possibilidade de integração devido à tecnologia da aplicação ser proprietária. Em outros casos devido à inviabilidade em termos de custo quando a tecnologia é muito antiga e não há uma relação custo/benefício positiva em função da alteração da aplicação ser grande e onerosa. As principais características da integração pelo nível de interface do usuário segundo [Linthicum – 2000] são:

- É o esquema de integração mais primitivo ou improvisado;
- É o mais limitado, ficando restrito às características da interface. Contudo é o que apresenta maior facilidade por não ser intrusivo com relação à aplicação integrada;
- A interface de integração corresponde à própria interface de usuário, onde os dados são capturados e é aproveitada a lógica a eles associada;
- Essa modalidade limitada de integração, em muitos projetos, é a única alternativa tecnicamente viável. Mesmo assim, com essas limitações, pode apresentar funcionamento adequado;
- Não se pode esperar um alto desempenho com esse método. Deve-se levar em consideração que a interface de usuário não foi criada com essa finalidade.

Passos para a integração pela interface do usuário [Linthicum – 2000]:

#### **Entendimento do domínio da integração**

Independente da simplicidade deste método de integração é necessário fazer o mapeamento dos dados e da lógica das aplicações com objetivo de entender o significado, comportamento, origem e destino das informações envolvidas em cada processo. Essa parte da integração pela interface do usuário normalmente é difícil, pois as aplicações sobre as quais ela é aplicada normalmente são de tecnologia fechada. É necessária a compreensão detalhada da lógica da aplicação para obter uma reversão dos dados até o seu estado primário, pois muitas vezes os dados obtidos na interface do usuário já são derivados ou calculados .

### **Catálogo dos Dados**

Deverão ser relacionados em documentação padronizada, os itens de dados por tela, com o título e descrição do campo, acompanhados da lógica de processamento e dos bancos de dados utilizados.

### **Mapeamento das telas**

Além de catalogar os dados é necessário também indicar o posicionamento de cada dado na tela em que o mesmo poderá ser encontrado. A extração dos dados das telas poderá ser realizada de duas formas:

- Forma estática – os dados são localizados e extraídos conforme sua posição física na tela (referências de linha e coluna);
- Forma dinâmica - os dados são localizados utilizando-se para isso condições lógicas, tais como utilizar o título de um campo para localiza-lo. Essa técnica é mais interessante por ser mais imune a alterações que modificam a posição física dos dados nas telas.

Ao processar a extração das informações das telas o programa que realiza a extração ocupa ou simula o papel do usuário da aplicação provocando e recebendo a saída de tela. Os dados podem então ser recuperados de duas formas:

- Leitura da tela como texto – *screen-as-data* – nesse caso as telas são capturadas pelo programa como texto e a aplicação de integração interpreta o texto para obter os dados pretendidos;
- Leitura da tela como objeto – *screen-as-objects* – essa forma de leitura da tela facilita em caso do ambiente de integração suportar tratamento a objetos como objetos JavaBeans ou ORB's.

## **3.6 Avaliação da utilização dos níveis de integração**

É necessário, em quaisquer dos níveis de integração que se trabalhe, entender os processos de negócio da organização. A finalidade, a interação com outros processos e o ciclo de vida dos processos (tempo de conclusão da transação) são pontos de análise fundamentais à avaliação do grau de integração necessário à adequação e otimização do processo. Segundo [Linthicum –

2000] a necessidade desse entendimento é compreendida em muitos casos após a implementação do processo de integração, momento no qual os resultados são efetivamente percebidos.

## Capítulo 4

### 4 Tecnologias de Integração

#### 4.1 Visão Geral

As tecnologias de EAI são sustentadas por mecanismos que possibilitam a uma entidade (uma aplicação ou banco de dados) se comunicar com outra ou com outras entidades, de acordo com [Linthicum – 2000]. Esses mecanismos são conhecidos como *middleware*. Em termos práticos o *middleware* faz o trabalho de supressão das diferenças entre as partes comunicantes facilitando ou viabilizando a troca de informações. A diversos tipos de *middleware* como segue:

- Orientados a Dados
- Orientados a Mensagens – MOM – *Message Oriented Middleware*
- Chamadas de Procedimentos Remotos – RPC – *Remote Procedure Call*
- Objetos Distribuídos – ORB’s – *Object Request Brokers*
- Monitores de Processamento de Transações – *TP Monitor*
- Servidores de Aplicação – *Application Servers*

Os tipos de *middleware* serão abordados mais detalhadamente nas próximas seções desse capítulo.

##### 4.1.1 Características do *Middleware*

Os modelos de *middleware* segundo [Linthicum – 2000] são de dois tipos: lógico e físico.

O modelo lógico é aquele que trata o movimento da informação pela empresa do ponto de vista conceitual. A discussão desse tipo de *middleware* segundo o autor requer a comparação de alguns tipos de configuração:

- Configuração um-para-um – *one-to-one* – versus muitos-para-muitos – *many-to-many*. A configuração ponto-a-ponto (um-para-um) é o tipo de configuração onde duas aplicações são “isoladamente” interligadas através de um canal único e direto. Como está limitada à interligação de no máximo duas aplicações, essa configuração não dispõe de uma camada própria para abrigar lógica e efetivamente não é uma solução de EAI. Exemplos desse tipo de *middleware* são os MOM como o MQSeries e RPC’s como o DCE. A configuração muitos-para-muitos é uma

tendência, pois atende a necessidade de EAI, oferecendo integração entre muitas aplicações. São exemplos dessa categoria os *application servers* e *TP Monitors*.

- O funcionamento síncrono – *synchronous* – versus o funcionamento assíncrono – *asynchronous*. No funcionamento assíncrono o processamento das aplicações interligadas pelo *middleware* está desvinculado, ou seja, as requisições feitas por uma aplicação são atendidas em uma determinada faixa de tempo que pode não ser imediato e a aplicação requisitante não interrompe o seu processamento para aguardar o atendimento. No funcionamento síncrono existe dependência de resposta ou atendimento para que o processamento prossiga. O funcionamento assíncrono é mais apropriado para EAI, principalmente pela independência de processamento que é permitida entre as aplicações.

O modelo físico é aquele que trata o movimento da informação como realmente acontece [Linthicum –2000] e para abordá-lo é interessante observar os modelos de trocas de mensagens existentes que podem ocorrer de diversas formas:

- Com ou sem conexão – as trocas de mensagens podem ocorrer através do estabelecimento de uma “sessão de comunicação” , onde as aplicações estariam conectadas ou de forma desconectada;
- Comunicação direta – é um tipo síncrono de atuação onde a mensagem é passada pelo *middleware* diretamente da aplicação requisitante para a requisitada;
- Comunicação intermediada por fila de mensagens – é o processo onde existe um gerenciador de filas de mensagens onde são depositadas as mensagens pelas aplicações requisitantes para serem retiradas pelas aplicações requisitadas e vice-versa, em um momento futuro de forma assíncrona. O processamento das aplicações participantes do processo de integração via fila de mensagens é independente, contudo podem ser estabelecidos limites de tempo ou regras para atendimento;
- Comunicação por publicação e assinatura – é um tipo de comunicação intermediada ou indireta onde as aplicações não precisam se conhecer. As aplicações que têm informações para “publicação”, disponibilizam essas mensagens em um intermediário que se encarrega desse serviço. Aplicações interessadas em receber

essas mensagens se “inscrevem ou assinam” uma lista de entrega de mensagens controlada pelo intermediário. Toda vez que mensagens são publicadas os assinantes da lista as recebem;

- Comunicação por requisição e resposta – é realizada através de *middleware* que comporta essa função. Esse padrão é mais utilizado por *message brokers* e *application servers*;
- Comunicação negociada – nesse tipo de comunicação o *middleware* precisa controlar o estado das regras de negócio, para poder negociar transações dependentes de estado com as aplicações.

Os diferentes comportamentos das tecnologias de *middleware* irão atender às disparidades de problemas e necessidades existentes no ambiente de integração. Poderemos verificar vantagens e desvantagens em relação às funcionalidades de cada tecnologia, contudo essas variações do potencial de cada uma deverão ser avaliadas em função de cada problema específico de integração.

## 4.2 Orientação a Dados

As tecnologias orientadas a dados basicamente utilizam uma única interface abstrata que permite acesso a diversos tipos de bancos de dados [JBLN – 2001]. Através desse tipo de *middleware* uma aplicação pode acessar diversos tipos de bancos de dados ao mesmo tempo utilizando uma interface única, o que possibilita a integração dos dados com facilidade [Linthicum – 2000]. Algumas das tecnologias existentes são:

- JDBC *Java Database Connectivity* e JDO *Java Data Objects* – plataforma Java;
- ODBC *Open Database Connectivity* e ADO *Active Data Objects* – plataforma Microsoft.

## 4.3 Orientação a Mensagens

O *middleware* orientado a mensagens – MOM é um tipo de tecnologia apropriada para comunicação orientada a eventos, que deixa transparente as diferenças entre aplicações, plataformas e protocolos envolvidos oferecendo API's padrões para seu acesso [Linthicum – 2000] e [JBLN – 2001]. Baseado na recepção, armazenamento e envio de mensagens aplica um

método assíncrono de trabalho que suporta falhas e não interrompe o processamento das aplicações. Os MOM não são por si só uma solução completa de EAI, mas podem ser utilizados para transportar mensagens para intermediadores de mensagens como os *Message Brokers* que no caso tem funções complementares que permitem transformar mensagens, controlar o fluxo de informações, estabelecer regras de negócios, monitorar o acontecimento de eventos e gerenciar a distribuição de mensagens.

#### **4.4 Chamadas de Procedimentos Remotos (RPC's)**

As chamadas de procedimentos remotos são mecanismos de funcionamento síncrono tipo cliente/servidor, no qual o cliente realiza uma chamada – *request* – e fica aguardando o servidor enviar uma resposta – *reply* – [JBLN – 2001]. É um mecanismo que têm problemas de performance tanto do ponto de vista da aplicação que aguarda enquanto o processamento é concluído, quanto do ponto de rede onde o volume de comunicação realizado afeta o desempenho exigindo uma rede com alta velocidade [Linthicum – 2000]. Apesar de ser um método fácil de ser utilizado pode ser problemático em situações onde o processamento pode ser interrompido por qualquer tipo de falha de ambiente ou das aplicações.

#### **4.5 Objetos Distribuídos (ORB's)**

Tecnologia que gerencia e suporta a comunicação entre objetos distribuídos e componentes [JBLN – 2001] na maioria das vezes de forma síncrona: ORB's são produtos complexos que acabam escondendo a complexidade das plataformas e do ambiente distribuído onde se localizam, causando a impressão de que os componentes estão localmente presentes. Os ORB's são um meio excelente de compartilhar componentes em ambientes de computação distribuída [Linthicum – 2000]. Devido à estrutura de funcionamento esconder localização dos objetos o processamento local e o remoto são invocados da mesma forma. Os objetos distribuídos estão recebendo novas funcionalidades para suportar funcionamento assíncrono baseado em mensagens, isso vem ocorrendo tanto com o padrão CORBA da OMG, como com o padrão COM da Microsoft [Linthicum – 2000]. Devido às características apresentadas pela tecnologia, em alguns momentos a performance pode ser prejudicada pelo funcionamento síncrono e por determinadas funcionalidades estarem disponíveis remotamente. É um padrão que apresenta

grande independência de ambiente, facilita o desenvolvimento e o reuso de lógica e está evoluindo através das novas funcionalidades acima citadas, que podem suportar as necessidades de EAI. A tecnologia também é atrativa nos problemas de integração, porque adicionalmente as suas propriedades de interoperabilidade, a sua estrutura componentizada facilita a “montagem” da solução.

#### **4.6 Monitores de Processamento de Transações**

Baseados no conceito de transações os *TP Monitors* fazem o gerenciamento de performance e o controle de segurança, são tolerantes a falhas e podem causar uma diminuição nas licenças necessárias para acesso a banco de dados, intermediando o acesso entre os cliente e o banco. São parte de uma tecnologia de *middleware* mais antiga e que precedeu os servidores de aplicação [JBLN – 2001]. É um tipo de *middleware* que geralmente não suporta o processamento orientado a eventos [Linthicum – 2000]. Sua tecnologia de processamento utiliza-se de chamadas de procedimentos remotos para realizar a comunicação entre as aplicações [JBLN –2001]. Por ser baseado em transações é um meio confiável para realização da integração em ambientes complexos, heterogêneos e distribuídos. A garantia que é fornecida pelo método de processamento torna o método apropriado para trabalhos de missão crítica. A tecnologia de monitores de transação é antiga como já abordado, mas seu método de garantia de transação atende a uma necessidade sempre atual. As aplicações mantêm controle de suas transações, garantindo a obrigatoriedade entre a conclusão ou o retorno ao estado inicial da transação. Quando uma única transação utiliza para sua execução várias aplicações, acaba englobando “sub-transações” que podem ser realizadas por aplicações distribuídas em ambientes heterogêneos e, nesses casos, o controle dessa transação integrada fica atribuído ao *TPMonitor*. Esse caso pode ser observado no exemplo apresentado na figura 4.6 adaptado de [Linthicum – 2000], supondo uma atualização que deve ser propagada para vários DBMS’s heterogêneos.

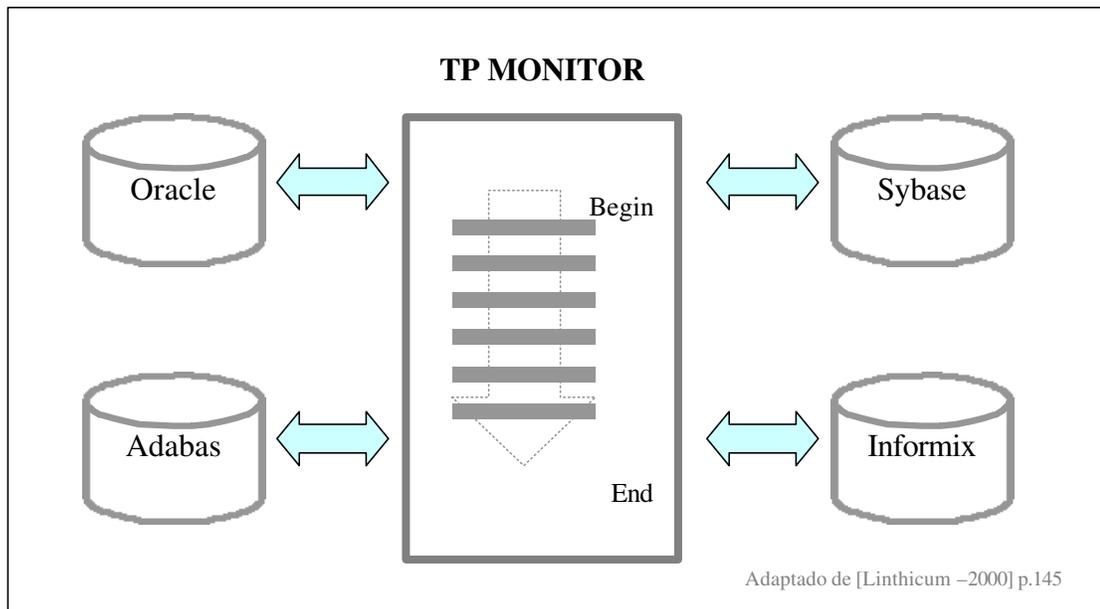


Figura 4.6 – *Middleware* Transacional atualizando DBMS 's heterogêneos

Monitores de transação podem se utilizar, de forma combinada, de outros tipos de *middleware* além dos RPC's, como os MOM .

#### 4.7 Servidores de Aplicação

Os servidores de aplicação fazem parte do conjunto das mais recentes tecnologias para EAI. Eles integram o conjunto de *middleware* até agora apresentado, formando um ambiente de integração gerenciável do ponto de vista de segurança, controle de distribuição da carga do processamento e utilização de recursos [JBLN – 2001]. Os servidores de aplicação já foram criados com foco em transações baseadas na Internet e sua capacidade de abrigar aplicações como os conectores dos pacotes de *software* e vários tipos de *middleware* direcionam naturalmente essa tecnologia para EAI, segundo [Linthicum – 2000]. Os servidores de aplicação são ferramentas de integração que além de poder conjugar outros *middleware* [BMT – 2002], estão adotando padrões de integração como SOAP, WSDL, RossetaNet, OAGI, J2CA e funcionalidades tais como BPM, *Workflow* e TPM ou PRM. Todos a seguir descritos:

- Padrões
  - SOAP – *Simple Object Access Protocol* – Protocolo para troca de informações em ambiente distribuído;
  - WSDL – *Web Services Description Language* – Linguagem de definições, baseada em XML, para descrição de *Web Services* (assunto da próxima seção);
  - J2CA – Padrão para construção de aplicações e adaptadores em J2EE;
- Organizações que desenvolvem padrões
  - RossetaNet – Consórcio independente focado no desenvolvimento de interfaces padrões para o comércio eletrônico;
  - OAGI – Consórcio para construção de uma estrutura de interoperabilidade de aplicações e desenvolvimento de processos repetíveis de distribuição de conteúdo de alta qualidade;
- Funcionalidades
  - BPM – *Business Process Management* – Gerenciamento de processos de negócio;
  - *Workflow* – Controle do fluxo de trabalho;
  - TPM – *Trading Partner Management* – Gerenciamento de parceiros de negócio;
  - PRM – *Partner Relationship Management* – Gerenciamento de relacionamento com parceiros.

As diversas tecnologias de *middleware* existentes, foram adquiridas pelas empresas ao longo do tempo para resolução de problemas específicos, criando um ambiente repleto de soluções de integração com tecnologia proprietária que não se comunicam. A evolução das tecnologias, consolidada nos servidores de aplicação que adotam padrões abertos e funcionalidades de gerenciamento de processo, traz a promessa ou mesmo a possibilidade da integração completa das aplicações (A2A) e dos negócios (B2B e SCM, B2C e CRM, TPM ou PRM). Segundo [Linthicum – 2000] a imprensa trata muitas vezes os servidores de aplicação e os monitores de processamento de transação como sendo o mesmo tipo de aplicação. Alguns monitores de

processamento de transação estão se atualizando e aderindo às funcionalidades voltadas a ambientes distribuídos e trabalhando em conjunto com vários outros tipos de *middleware*, o que claramente ajuda a confundir os dois conceitos. Os monitores de processamento de transação – *TP Monitors* – são a geração antiga dos servidores de aplicação.

## 4.8 Web Services

*Web Services* constituem um padrão de fornecimento de serviços, que é independente de plataforma, linguagem de programação e localização do processamento (local ou remoto). Os *Web Services* correspondem a uma forma dinâmica e padronizada de disponibilizar serviços de forma aberta para troca de dados e processos entre empresas ou mesmo dentro de suas próprias unidades de negócio. *Web Services* são baseados totalmente em padrões abertos. Podemos explicar o funcionamento dos *Web Services* – [JBLN – 2001] e [W3CWS – 2002] – através das seguintes características:

- Especificação dos Serviços – Pode ser descrita utilizando-se a WSDL – *Web Services Description Language* – que é uma linguagem descritora baseada em XML. A descrição aborda basicamente três pontos: quais funcionalidades tem o serviço, onde ele está localizado e como pode ser acionado;
- Conteúdo das Mensagens Serviços – Os *Web Services* contém mensagens autodescritivas escritas em XML – *Extensible Markup Language* com dados e documentos ou requisição de operações, de forma independente de linguagens de programação.
- Publicação e Descoberta – As especificações dos *Web Services* podem ser publicadas, caso sejam serviços abertos (públicos), utilizando-se UDDI – *Universal Description, Discovery and Integration* – que são basicamente listas ou diretórios onde podem ser registrados os serviços para posteriormente serem pesquisados e localizados.
- Transporte – A infraestrutura para os *Web Services* serem processados na Internet se dá sobre o protocolo SOAP – *Simple Object Access Protocol*, que é um protocolo que oferece independência de plataforma, independência de linguagem de programação, define uma forma padrão de transmitir dados em formato XML e também possibilita uma forma de realizar chamadas de procedimentos remotos – RPC's. O protocolo SOAP normalmente é utilizado sobre o protocolo HTTP;

- Funcionalidades - Os *Web Services* também podem oferecer tratamentos como segurança, gerenciamento e monitoração da qualidade dos serviços.

Segundo [JBLN – 2001] e [W3CWS – 2002] são três os principais papéis dos participantes dentro do contexto de funcionamento dos *Web Services*: O fornecedor dos *Web Services* – *Service Provider*, o requisitante do serviço – *Service Requestor* e o publicador do serviço – *Service Registry (Discovery Agency)*, demonstrados na Figura 4.8.

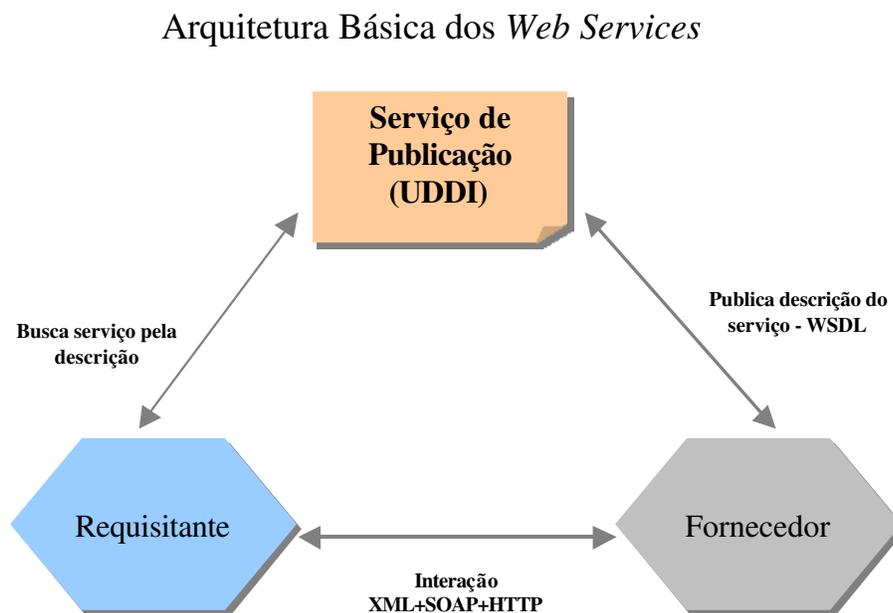


Figura 4.8 – Arquitetura Básica dos *Web Services* – Adaptado de [W3CWS – 2002] seção 3.1

Nesse contexto os participantes podem assumir mais do que um papel. Descrevendo o processo básico: o fornecedor (*Provider*) coloca a descrição do serviço (*Web Services*) no diretório de publicação (da *Discovery Agency*), o requisitante (*Requestor*) retira as

informações do diretório e passa a saber o formato da mensagem, as características do serviço e sua localização, obtendo condições para a interação com o fornecedor do serviço.

Existem algumas variações possíveis no processo básico dos *Web Services*, das quais destacam-se:

- Fusão do papel do publicador com o papel do cliente (requisitante), possibilitando a interação direta. O serviço é publicado diretamente no cliente;
- Presença de um novo agente, mudando o fluxo da comunicação, para intermediar a interação entre o requisitante e o provedor do serviço. O intermediário fica com um papel duplo no fluxo, de recepção e envio das mensagens. Sua atuação é restrita ao roteamento ou outras operações que não envolvam propriamente o conteúdo das mensagens;
- Ocupação do papel de requisitante pelos próprios *Web Services* que podem requisitar outros *Web Services* na sua implementação.

A interação pode se dar através do envio de uma ou múltiplas solicitações por parte do requisitante ou até mesmo com o envio de processos de negócio completos. O processamento das solicitações pode se estabelecer de forma síncrona ou assíncrona.

A configuração descrita até agora para *Web Services* é a convencional, pressupondo o seu uso na Internet onde naturalmente o protocolo HTTP irá dominar totalmente, contudo pode-se esperar outras aplicações em redes privadas que não utilizem esse protocolo. Ainda segundo [W3CWS – 2002] outras tecnologias poderão surgir e substituir, por exemplo, o uso do protocolo SOAP e da linguagem WSDL .

O termo “*Web Services*” não consta na lista de *middleware* apresentada, pois não é propriamente um componente ou um conjunto de componentes de *software* voltados à integração e sim uma forma padronizada de disponibilizar serviços através da *web*.

## Capítulo 5

### 5 Impacto da Integração de aplicações

A integração de aplicações [Cummins – 2002] apresenta um valor limitado enquanto não ocorre a sinergia entre as aplicações e enquanto o número de pontos integrados é pequeno, impedindo o retorno em escala. O impacto que a integração de aplicações promove nas corporações, positivo ou negativo, deve ser cuidadosamente avaliado em função do envolvimento que essas mudanças “integradoras” têm sobre os processos de negócio.

#### 5.1 Do ponto de vista técnico

Em busca do aumento de retorno sobre o capital ou diminuição de custos, as empresas exercem pressão acentuada pela atualização das aplicações e por outras mudanças que racionalizem os processos de negócio. Esses movimentos deixam iminente a necessidade por recursos de integração. As empresas fornecedoras de tecnologia de integração, tendo em vista a forte demanda por esses recursos, têm oferecido facilidades de implementação e criado expectativas de resultados, que na prática não têm ainda comprovação técnica ou efetivamente não se realizaram. Por outro lado vários especialistas vem discutindo, elaborando e propondo técnicas para condução desse processo, com o objetivo de reduzir a distância entre expectativas e resultados. Para se ter uma idéia de alguns números envolvidos com a questão, segundo [Linthicum – 2000], as participantes da lista das 1000 maiores empresas da revista Fortune gastam em torno de U\$ 100 bilhões por ano com os esforços de integração e o Instituto de pesquisas Forester Research estima em torno de 30% os gastos da tecnologia da informação direcionados a integração de aplicações. A apresentação desses números nos leva a pensar de forma cuidadosa sobre o impacto desse segmento da tecnologia da informação, que vem tomando corpo e importância a cada dia no universo das organizações, no meio técnico e no mercado de *software*.

[Laroia – 2002] cita e analisa alguns benefícios oferecidos pelo mercado como:

- A eliminação da redundância entre repositórios de dados;
- A integração das aplicações isoladas nos departamentos com facilidades para absorção de custos;
- Redução dos custos com tecnologia;

- Melhoria de performance no processamento de transações e processamento batch;
- Melhoria no gerenciamento dos processos de negócio.

Na prática observa-se que após a implantação haverá eliminação de alguns dados, sob este ponto de vista redundantes, contudo algumas aplicações que utilizam bancos de dados isolados continuarão utilizando esses dados, mas os mesmos passarão a ser também disponibilizados a outras aplicações. Eventualmente a própria aplicação EAI poderá manter dados intermediários, tecnicamente redundantes, em seu próprio banco de dados. Um dos aspectos importantes a se observar é que não se pode confundir a integração dos dados com a fusão dos mesmos, pois isso não ocorrerá integralmente após a implantação do EAI.

Extraíndo uma conclusão entre o que o mercado promete e os fatos, observamos que a integração das aplicações departamentais (*stovepipes*) acontece com a aplicação de soluções EAI. Os aspectos relativos a custo e facilidades de integração normalmente frustram as expectativas, por diversos motivos, como segue:

- necessidade de aquisição de diversos pacotes de *software* para se realizar a integração. Entre eles o próprio *software* EAI, adaptadores para viabilizar extração ou exportação de dados;
- contratação de consultores externos, disponibilização de recursos humanos internos;
- a construção do modelo de integração através da especificação de formato, regras de transação e ciclos de envio e recebimento são tarefas onerosas e complexas.

A tarefa de prever ou estimar custo, tempo e outros recursos em um projeto EAI não é nada fácil como verificamos na abordagem da definição do escopo e tamanho do projeto. Em função da complexidade envolvida espera-se que ocorra uma variação muito grande de comportamento entre vários projetos e uma efetiva dificuldade de realizar o previsto. Pode-se concluir que os baixos custos e facilidades prometidos pelos fornecedores de tecnologia na maioria dos casos serão um objetivo de difícil alcance.

A única mudança ou aumento percebido na velocidade de processamento de transações, segundo [Laroia – 2002], ocorre somente após a implementação no ajuste fino da aplicação EAI e não pela sua implementação. Normalmente a implementação de um projeto EAI muda as

características da concorrência das transações pelos recursos tecnológicos disponíveis em dois aspectos, volumes de transações e o momento de ocorrência das transações. A iniciação de uma atividade de integração irá gerar um aumento no fluxo de dados existente. Um exemplo: em alguns casos a transferência de um processamento batch noturno para tempo real diurno pode gerar um volume impactante, tomando excessivamente os recursos disponíveis e eventualmente afetando aplicações de missão crítica. Em outros casos poderá ocorrer o processo inverso onde podemos ter como resultado o equilíbrio do uso de recursos. Os dois casos, I e II, podem ser observados na figura 4.1.

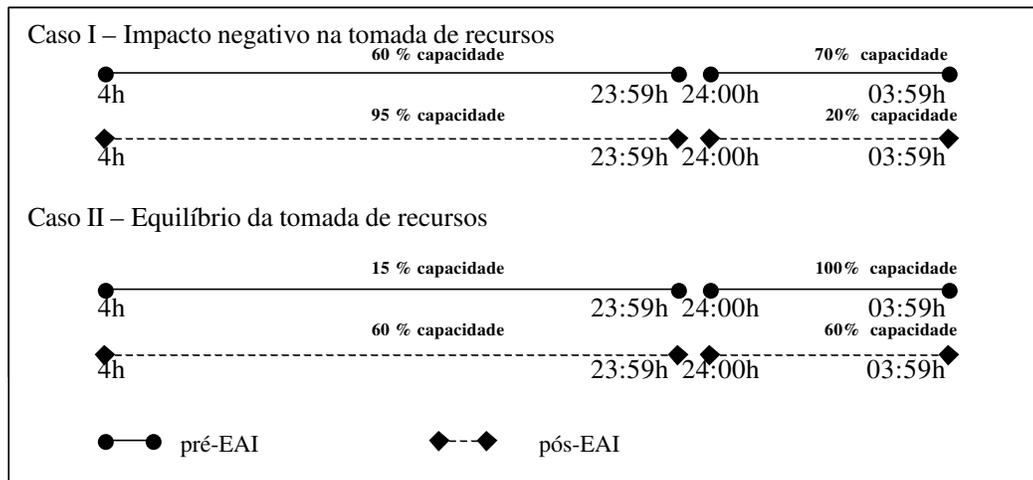


Figura 4.1 – Comparativo dos impactos pós-EAI

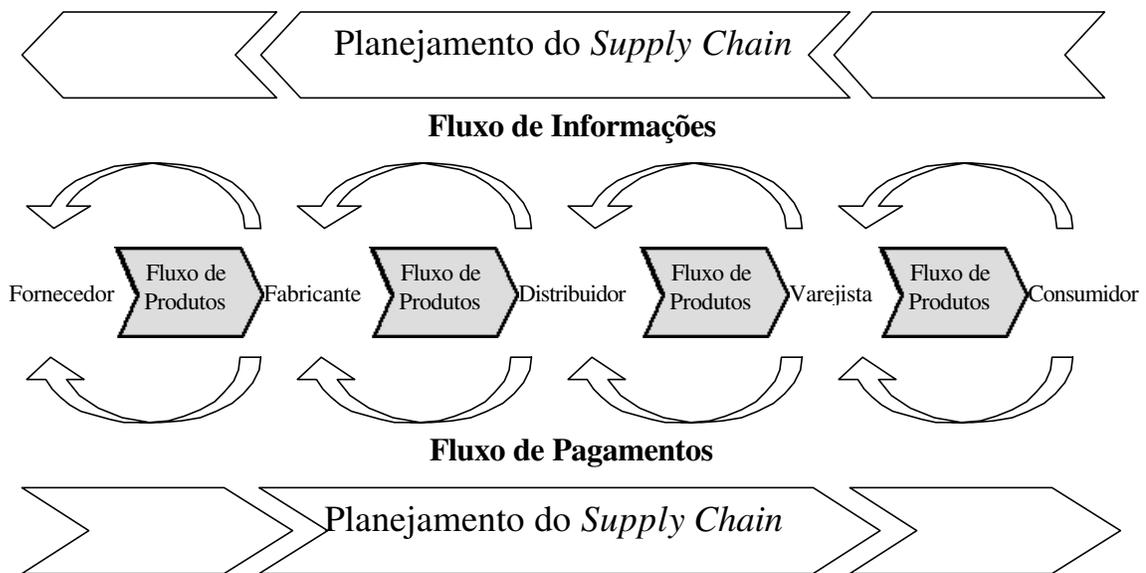
Segundo [Cherry – 2000] enquanto as soluções de integração tradicionais baseadas, por exemplo, em *middleware* tratam cenários independentes e transações mais simples, o conceito de EAI é diferente. EAI permite o gerenciamento do relacionamento de integração de múltiplas aplicações formando um verdadeiro e completo processo de negócio. Nesse paradigma diferenciado de integração podemos citar como resultados do projeto:

- Eliminação do conceito de ilhas de automação ou integração propiciada por soluções isoladas. A integração é centralizada e gerenciável;
- A integração não está mais totalmente incorporada na aplicação, é um elemento solto ou independente;

- O processamento orientado a eventos – *event driven process* - é de mais alto nível, permitindo a atuação de um analista de negócio ao invés de um desenvolvedor no processo de integração da aplicação;
- A automação dos processos de negócio ou *business process automation* possibilita a configuração dinâmica de novos processos de negócio.

## 5.2 Do ponto de vista dos negócios

As novas tecnologias a todo o momento transformam os processos tradicionais no mercado, decretando a compactação das tarefas, do tempo ou a extinção de ações que após a evolução de um processo se tornam obsoletas ou desnecessárias. A integração é um causador direto de custos mais baixos, maior satisfação com a execução do processo e garantia de qualidade nos processos.



Adaptado de [Franco – 2001] p.243

Figura 5.2a – Visão do Processo de Integração do SCM

SCM – *Supply Chain Management* ou Gerenciamento da Cadeia de Suprimentos segundo [Cherry – 2000] são pacotes de *software* que tem o papel de integrar ou interligar os participantes da cadeia de suprimentos (fornecedores, fabricantes, distribuidores, revendedores e operadores de logística entre outros). A tecnologia SCM representa a seqüência evolutiva das aplicações MRP,

MRPII e ERP. Essas tecnologias representam a convergência entre administração do negócio e tecnologia, pois correspondem ao emprego da tecnologia em um sentido integrado e ao mesmo tempo integrando o negócio. A cobertura sobre os processos de negócio, exercida pela tecnologia, abrange toda a empresa e agora, com adição dos pacotes SCM, toda a cadeia de suprimento na qual a empresa está situada.

A troca de informações entre os participantes da cadeia de suprimentos estende a visibilidade para a tomada de decisões. Através dessa visibilidade é possível perceber os fatos ocorridos externamente à empresa numa faixa de tempo próxima ao real. A visão estendida em tempo real leva à decisão mais rápida e todos os participantes da cadeia recebem como reflexo maior velocidade, menor custo e menor necessidade de recursos. O planejamento de atividades para os participantes da cadeia de suprimento torna-se mais fácil. As necessidades dos clientes finais requisitadas em um dos extremos da cadeia (a demanda) e a disponibilidade de fornecimento de materiais oferecida pelos fornecedores no outro extremo (suprimento), passam a fazer parte do bloco das informações conhecidas pela atividade de planejamento.

Em [Yantra – 2001] podemos observar claramente os impactos da integração na cadeia de suprimento. Esse caso aborda a viabilização da integração de processos entre os prestadores de serviço de logística, seus clientes e os clientes de seus clientes. A tecnologia disponibilizada recentemente permite a visão extra-empresa. Essa visão faz com que a empresa passe a perceber os seus prestadores de serviço como braços ou extensões de si mesma. Nesse caso, para avaliar a tecnologia influenciando os negócios ou os negócios passando a ser atendidos pela tecnologia, podemos fazer um cruzamento entre as demandas do negócio e os resultados esperados.

**As demandas do mercado de logística atual do ponto de vista dos negócios exigem:**

- Atendimento a cadeia de suprimentos onde o mix de produtos está em constante mutação necessitando capacidade de adaptação;
- Como os canais de vendas são múltiplos, a demanda por serviços logísticos pode oscilar significativamente em curto espaço de tempo, o que exige respostas rápidas;
- Os clientes do serviço de logística desejam que seus clientes sejam tratados de forma personalizada pelo prestador de serviços. O prestador deve ter condições de “diferenciar” os clientes do seu cliente;

- O prestador de serviço deve ter capacidade de se conectar com facilidade (flexibilidade) a múltiplos clientes sem ter custos de personalização;
- A prestação de serviço deverá oferecer gerenciamento do estoque do cliente;
- A visibilidade da cadeia de suprimento deve ocorrer em tempo real;

A partir de todas essas demandas fica claro o papel da integração de aplicações como fator de indispensável utilização.

**Como resultado ou impacto final do emprego da tecnologia na integração podemos esperar:**

- Aumento da visão dos participantes;
- Otimização do tempo de entrega ou reposição;
- Diminuição dos estoques na cadeia de suprimentos;
- Diminuição dos custos da operação e aumento de ganhos;
- Satisfação e retenção de clientes.;

A utilização da tecnologia focada em integração trará vantagem competitiva para os fornecedores de serviços de logística e participantes da “cadeia de suprimento integrada”. Por essa abordagem [Yantra – 2001] pode-se concluir que o emprego de tecnologia para fornecedores de *commodities*, como no caso dos prestadores de serviço de logística, pode adicionar vantagem competitiva ao negócio e com essa diferenciação propiciar maior retorno nas operações.

As aplicações para gerenciamento do relacionamento com clientes – CRM – *Customer Relationship Management*, segundo [Cherry –2000] são aplicações projetadas para “facilitar a captação, consolidação, análise e distribuição por toda a corporação dos dados de clientes e possíveis clientes”. Os processos cobertos pelo CRM se estendem, além dos processos notadamente comerciais, a todos os processos internos ou externos nos quais ocorrem contatos com clientes e possíveis clientes.

Os dois focos básicos de uma aplicação CRM segundo [Cherry – 2000] são:

1. Possibilitar a empresa uma melhor identificação, controle do contato e aquisição de novos clientes através das seguintes funcionalidades:
  - a. Gerar perfis de clientes e segmentos de mercados;

- b. Acompanhar a execução das campanhas de marketing realizada via diversos canais de contato com clientes;
  - c. Controlar negociações;
  - d. Direcionar os esforços da empresa, através de uma melhor identificação de clientes, para um público alvo que proporcione maior retorno;
2. Alavancar o relacionamento com os clientes atuais:
- a. Identificando possibilidades da realização de vendas cruzadas;
  - b. Aumentando a retenção de clientes através da aplicação de instrumentos de pós-vendas;
  - c. Analisando padrões de comportamento para tentar antecipar serviços de suporte ao cliente além das vendas;

Os canais de contato mais comuns, conforme figura 5.2b com os clientes, onde ocorrem entradas e saídas de informações do relacionamento, que são centralizados pelo CRM são:

- Centrais de contato telefônico – *call centers* – para atendimento de clientes, consumidores, representantes comerciais nos serviços de atendimento, suporte técnico dos produtos e televendas;
- Acessos via *website* da empresa para serviços de vendas (*e-commerce*), reclamações, sugestões e suporte técnico dos produtos;
- Malas-diretas via eletrônica (*e-mail*) ou postadas no correio convencional;
- Coletores de pedidos eletrônicos – *handhelds* – ou outros computadores portáteis da força de vendas (*SFA – Sales Force Automation*);
- Pontos de vendas dos parceiros de negócio;
- Outros parceiros de negócio como operadores de logística, transportadores, distribuidores, etc.

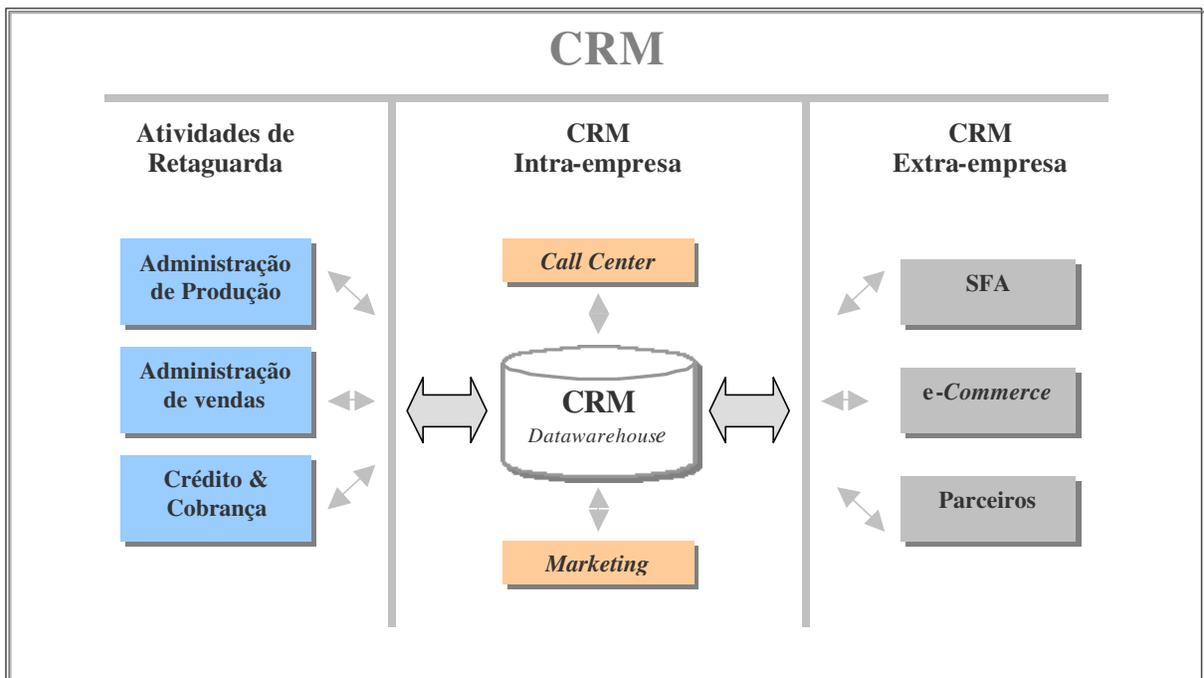


Figura 5.2b – Visão do Processo de CRM

O desafio dos pacotes de *software* de CRM está em aglutinar as informações que transitam por todos esses pontos, consolidando diversas fontes de informação geograficamente distribuídas e oferecendo informação em tempo quase real. A partir da integração dos canais de relacionamento com os clientes é possível explorar o potencial e melhorar a retenção da carteira atual de clientes ou adquirir novos clientes mais habilmente.

Segundo [JBLN – 2001] “pela perspectiva dos negócios , EAI é a vantagem competitiva que uma empresa adquire quando todas as suas aplicações estão integradas em um sistema de informações único, capaz de compartilhar informações e suportar o fluxo de trabalho do negócio”.

## Capítulo 6

### 6 Premissas do Projeto de Integração

#### 6.1 Negócios , Integração e depois a Tecnologia.

As empresas traçam seu planejamento com o objetivo de formular ou decidir a construção do seu próprio futuro. Os objetivos a serem atingidos são apontados com base nas possibilidades ou potencialidades que o mercado e a empresa têm entre si. Olhando a tecnologia em um sentido passivo, o planejamento estratégico define quais tecnologias deverão ser adquiridas e/ou empregadas para sustentar as iniciativas futuras do negócio. Percebendo de outra forma a tecnologia, em especial a de integração de aplicações, pode mudar o comportamento (canais, velocidade, processos, etc.) com o qual os negócios são realizados. A estratégia empresarial pode fazer com que os negócios sejam alterados com finalidade de aproveitar as possibilidades propiciadas por novas tecnologias. Como argumenta [Collins – 2002] a tecnologia pode representar o papel de um acelerador no desenvolvimento dos negócios. Em contrapartida, assim como a presença de tecnologia pode alavancar os negócios, em um segmento de negócios que avança de forma diretamente dependente de tecnologia, a sua falta será um diferencial competitivo negativo para a empresa, podendo até vir a representar o motivo de sua exclusão do mercado.

Atrelar o plano de negócios ao plano de tecnologia pode parecer óbvio mas, na prática em muitos casos isso não acontece. A integração de aplicações é em muitas ou na maioria das situações deixada para trás, como uma das últimas ações a serem realizadas. Ampliam-se os processos de negócio, adquirem-se tecnologias mais atualizadas e então por último a tecnologia voltada à integração de aplicações é lembrada. O planejamento estratégico dos negócios pode passar a perceber ou adotar a integração de aplicações como um dos recursos iniciais para a mudança nos processos de negócios. A partir dessa consideração, a tecnologia destinada à integração de aplicações passa a se encaixar no planejamento estratégico corporativo de forma sincronizada aos outros esforços para alcance dos objetivos empresariais.

Segundo [JSEPI – 2002], “o sucesso da implementação de um projeto de integração de soluções está fundamentalmente baseado no alinhamento entre negócios e tecnologia em várias

dimensões”. É interessante avaliar essa colocação de forma estreitamente conjugada à que está colocada no parágrafo anterior. Se a mudança nos processos de negócio ocorre sem a devida atualização ou mesmo construção do projeto de integração, a mesma acaba por provocar a desestruturação da integração tecnológica e também da corporativa. Os níveis de evolução entre integração e processos de negócio devem ter uma determinada compatibilidade, ou seja, não há como ter um nível de integração de aplicações alto se o processo de negócio não tiver um nível de maturidade compatível e vice-versa. Essa compatibilidade deve ser mantida no decorrer das mudanças que acontecem nos negócios. Podem ser comparados os níveis de evolução dos processos de negócio e de integração empregando-se uma analogia com as definições CMM (*Capability Maturity Model*) como na tabela a seguir:

Tabela 5.1 – Níveis CMM Adaptado de [ JSEPI – 2002]

	<b>Aspectos a serem mapeados</b>		
<b>Nível CMM</b>	Maturidade dos Processos de Negócio	Maturidade do Projeto de Integração	Maturidade do uso de Modelos ou Metadados
<b>1</b>	Independente	Inicial	Estático
<b>2</b>	Compartilhado	Repetível	Ativo
<b>3</b>	Contínuo	Sistêmico	Dinâmico
<b>4</b>	Gerenciado	Sustentável	Preditivo
<b>5</b>	Em evolução	Em Otimização	Em aprendizagem

Além dos eventuais riscos de não atrelar a visão da tecnologia aos objetivos de negócio, o desequilíbrio dos níveis de maturidade dos processos de negócio, desenvolvimento de projetos de integração, uso de modelos ou metadados irá provocar perdas de esforços nos projetos de integração ou de negócio.

## **6.2 A idéia é de Integração e não de Interligação**

A idéia de integração pressupõe garantir todos os comportamentos sistêmicos necessários à formação de um todo, ou seja, o conjunto de necessidades dos processos de negócio atendidos pelas aplicações a serem integradas deve ser satisfeito. Normalmente quando há necessidade de alterações nas aplicações os usuários são chamados a participar do desenvolvimento [Cummins –

2002], pois são eles que fornecem os requisitos dos processos de negócio. As linhas de código das aplicações traduzem em linguagem informática as ações dos processos de negócio. No momento de executar a integração das aplicações espera-se que o “código da integração” deva refletir as ações, ou nesse caso interações, dos processos de negócio.

“Para obter soluções completas utilize uma abordagem orientada a processos” [JSEPII – 2002]. Como postula o autor, a integração tem várias dimensões incluindo-se nelas: dados, processos, plataforma, rede, empresa, localidade, empregados, clientes e produtos entre outros. Na implementação da integração uma dessas dimensões pode assumir um papel de maior evidência. Costumeiramente isso ocorre com a dimensão dos dados. Sem prescindir de quaisquer das dimensões, os processos constituem a parte central ou fundamental de qualquer projeto de integração. O processo de negócio é a dimensão que arrazo a existência de todas as outras dimensões citadas. Quando se afirma que a orientação da integração deve ser voltada a processos, isso indica que deverão ser satisfeitas, através dos recursos empregados na integração, todas as prerrogativas do processo de negócio, desde o alcance dos objetivos do processo até o atendimento dos seus requisitos funcionais como segurança, qualidade, velocidade e etc..

### **6.3 A integração deve ser imune à evolução do software e dos processos de negócio**

Segundo [Cummins – 2002] o custo e o risco de substituição de componentes de *software* está vinculado ao seu tamanho e ao seu escopo de atuação. É então necessária uma arquitetura de sistemas que facilite mudanças ou expansões nas funcionalidades existentes através da atualização, troca ou aquisição de novos componentes de *software*. Como descrito em toda dissertação, as mudanças nos processos de negócio e na tecnologia são variáveis a serem consideradas. Tecnologias “autoconfiguráveis” como BPA (*Business Process Automation*), XML (Extended Markup Language), WS (*Web Services*) são promessas de maior flexibilidade e alternativas para a sustentação das mudanças de tecnologia e de negócios.

O projeto de integração deve contemplar a adoção de ferramentas de integração que prevejam mudanças, assim como deve criar uma norma para que as novas aplicações contemplem a idéia de “componentização” e tenham requisitos de integração como parte de seu escopo inicial. Segundo [JSEPIII – 2002] normalmente os aspectos relativos à integração são considerados após

a aquisição ou desenvolvimento de novas aplicações, o que adiciona um alto custo no ciclo de vida das aplicações. O projeto de integração também deve prever manutenção cíclica da tecnologia empregada para que a mesma não se torne obsoleta precocemente. Os valores do orçamento de tecnologia são limitados [JSEPV – 2003] e para que as interfaces possam sofrer manutenção elas devem apresentar flexibilidade e ser gerenciáveis. Observando que determinados projetos apresentarão um grande número de pontos de integração, as interfaces de integração devem ter recursos ou características como:

- recursos de autoteste;
- ser “registráveis” publicamente;
- ser “descobertas” automaticamente;
- acoplamento dinâmico em tempo de execução;

#### **6.4 A integração deve ser gerenciável**

A integração será tecnicamente gerenciável se houver centralização do controle dos processos, ou seja, todos os pontos de integração visíveis, mensuráveis e controlados. O fluxo de trabalho e os processos de negócio, na verdade clientes do processo de integração, devem ter disponibilizada a visibilidade do ponto de vista de etapa ou tarefa, correspondente ao seu grau mínimo de granularidade. Podemos supor uma composição para a pilha de integração, formada por: processos de negócio, tarefas de *workflow* e pontos de integração, como a apresentada na figura 6.4. O gerenciamento dos processos de negócio, bem como do fluxo de trabalho, ocorre sobre os alicerces do gerenciamento dos pontos de integração, uma vez que a integração constitui a infra-estrutura para esses processos. A integração gerenciável atende aos processos de negócio e ao fluxo de trabalho, entre outros, através da provisão e do controle dos aspectos de segurança, qualidade de serviços, garantia das transações efetuadas, persistência.

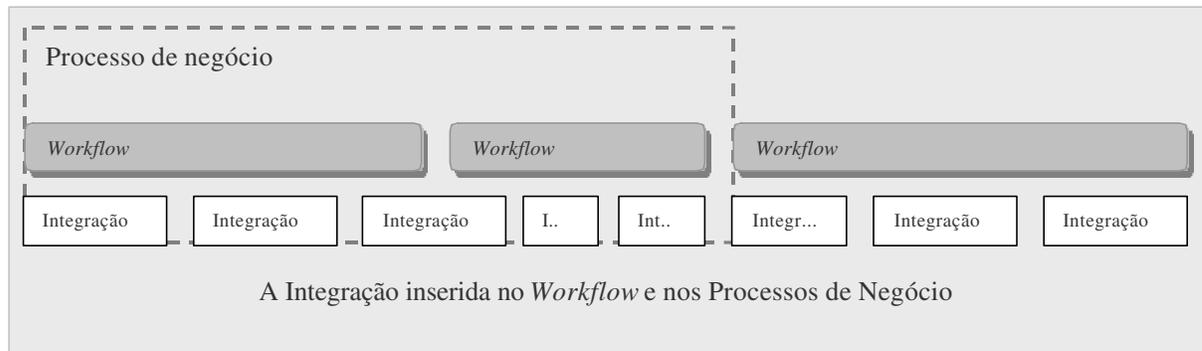


Figura 6.4 – A Integração dentro do Processo de Negócio e do Workflow

## 6.5 É necessário padronizar

Tendo em vista o próprio foco dessa tecnologia e as exigências naturais impostas pela tarefa de integração, o estabelecimento de padrões para esse fim é extremamente necessário. Não se espera que um projeto de integração substitua ou elimine completamente as disparidades entre os recursos tecnológicos do ambiente de integração. É necessário, além de unificar os esforços de integração das corporações em um projeto consolidado, estabelecer ou homologar os tipos de recursos que sempre serão utilizados como:

- Ferramentas de modelagem e documentação de dados, processos, etc.;
- Fornecedores de aplicações e componentes de aplicação;
- Linha tecnológica que será adotada na integração;
- Critérios para desenvolvimento e aquisição de novas aplicações para a empresa;

A padronização não irá garantir a homogeneidade do ambiente, contudo irá garantir a evolução ou organização da integração de aplicações da corporação. Estando estabelecidos os padrões de documentação, aquisição e outros, ocorrerá naturalmente a tendência à queda no custo e no tempo da integração, originada pelo maior aproveitamento dos recursos.

## 6.6 Sequência das etapas do projeto de Integração

Alguns cuidados deverão ser tomados no sequenciamento das etapas do projeto de integração. A integração têm características particulares e sua complexidade tende a gerar dificuldades desde o levantamento do problema até execução do projeto. As dificuldades podem

impactar significativamente no custo e no tempo estimados, assim como levar ao insucesso, frustrando as expectativas projetadas. Segundo [LSMK – 2003] “a finalidade de um projeto EAI é economizar ou efetivamente ganhar dinheiro para uma empresa a curto prazo e criar uma vantagem competitiva a longo prazo”. O planejamento do projeto de integração é o primeiro passo para buscar garantir o atendimento das expectativas mais comuns de retorno. A escolha da seqüência de etapas do projeto, a proporção de tempo dedicada a cada etapa dentro do total do projeto podem representar problemas de aumento de custo ou retardamento do retorno sobre o investimento de várias formas:

- A relação entre o tempo a ser gasto com planejamento e o tempo de implantação deve se aproximar respectivamente de uma relação de 70% por 30%. Projetos nos quais o tempo de implantação ultrapassa largamente a faixa dos 30% do total do projeto tendem a apresentar excesso de custo principalmente originados de retrabalho;
- A etapa de aquisição dos aplicativos de *software* que serão utilizados no momento em que o projeto de integração estiver rodando deve ser postergada ao máximo, coincidindo se possível com o ponto de testes e implantação. Casos nos quais a aquisição ocorre no início do projeto representam o emprego de capital sem retorno e também pagamento desnecessário de taxas de manutenção de *software* por mais tempo;
- Após iniciado o projeto de integração, eventuais prorrogações sempre irão representar custo adicional. Se a prorrogação ocorre após a aquisição do *software* como visto no item anterior o problema com custo fica mais ampliado. Também podemos entender que pelo mesmo motivo o retorno sobre o investimento será retardado.
- Após as fases de levantamento do domínio, análise e criação do projeto de integração deve ser realizada uma reavaliação do impacto da integração. O projeto de integração é um projeto de mudança, após sua implantação os processos de negócio não serão mais os mesmos, novas regras de negócio surgirão a partir da implantação dessas mudanças. Tendo em mãos o levantamento dos dados, interfaces, dos processos de negócio com suas regras e objetivos pré-implantação é possível avaliar o impacto das mudanças pós-implantação.

## Capítulo 7

### 7 Um Modelo Conceitual para EAI

#### 7.1 O conceito do modelo

Os predicados necessários ao atendimento do universo dos problemas de EAI, estão ligados a dois fatores principais do “conceito da aplicação”: flexibilidade e foco de abordagem do problema. Em função desses dois pontos, as funcionalidades devem atender aos os seguintes quesitos:

- Orientação a processos de negócio;
- Gerenciamento de *software* atrelado ao de negócios;
- Construção da solução pela utilização de modelos;
- Configuração da solução pela conjugação de componentes locais e remotos;
- Capacidade de mudanças dinâmicas de comportamento, em tempo de execução.

##### 7.1.1 Orientação a processos de negócio e gerenciabilidade

O foco da abordagem do problema é orientação a processos de negócios, utilizando modelos com a tarefa de representar os dados e os processos de negócios vinculados a “meta-definições” e estabelecendo o caminho do plano conceitual até a implementação. A utilização de modelos<sup>1</sup> orientados a processos, significaria definir uma árvore de processos e tarefas que derivem uma implementação combinando: serviços (SOA), componentes (*middleware*), API's e dados (transacionados por exemplo em formato XML). Construção baseada no conceito de gerenciabilidade, onde se pode estipular parâmetros de comportamento, medir e controlar a execução através de ferramentas de administração centralizadas. Sob muitos aspectos as definições aqui colocadas como requisitos para o modelo conceitual vão coincidir com funções apresentadas por outros conceitos divulgados no mercado. Por exemplo, como abordado em [Hailstone – 2003] e apresentado na figura 7.1, nas camadas que se superpõem ao EAI, o BPA – *Business Process Automation* – têm o conceito de agregar componentes de *software* para montar processos de negócio ou automação de processos para realização de tarefas, e o BAM a possibilidade de gerenciamento dos processos automatizados.

---

<sup>1</sup> MIC – *Model-Integrated Computing* – abordada na seção 7.1.2

TYPICAL INTEGRATION TECHNOLOGY STACK

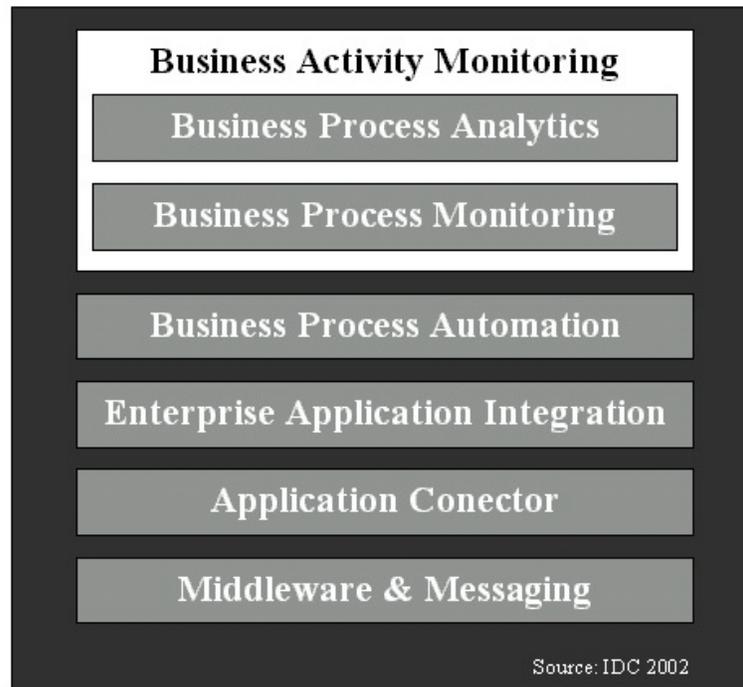


Figura 7.1 – BAM – *Business Activity Monitoring* – Adaptado de [Hailstone – 2003]

Outro conceito próximo da definição do BAM, o BPM – *Business Process Management* abordado em [Delphi – 2003], tem como principal diferencial em relação ao BAM, a inclusão de atividades humanas nos passos ou tarefas na definição dos processos além das tarefas automatizadas, essa característica é pouco comum ou quase inexistente no BAM. As características dos problemas e soluções relativos à integração de aplicações e gerenciamento de processos de negócio se tocam em muitos pontos coincidentes. Em [Hailstone – 2003] o mesmo afirma que a plataforma EAI fornece muito da tecnologia necessária para o BPA, em [Delphi – 2003] nas pesquisas de opinião abordando o BPM e seus requisitos, todos os questionamentos colocados têm na maior porcentagem das respostas solicitações coincidentes diretamente com os problemas EAI (serviços de integração, controle de transação, integridade, regras de negócio, etc.). A idéia aqui apresentada é de um conceito de EAI orientado a processos, onde a diferença básica com relação a BPM e outros conceitos similares está na disposição das camadas, portanto não caberia ao BPM a estruturação ou o desenho dos processos de negócio e sim ao EAI. Como

principal função ao BPM caberia o gerenciamento, através da análise e do monitoramento das métricas de desempenho e ou qualidade dos processos.

### **7.1.2 Construção da solução pela utilização de modelos;**

O emprego de “meta-definições” para o desenho de processos, definição das regras e sintaxe e semântica dos dados tem como motivação:

- Criar um banco de dados (ou metadados) que registre de forma independente da implementação (código das aplicações, plataforma operacional, *hardware* e etc.), o conhecimento sobre a constituição, o comportamento e os objetivos dos processos de negócio e de seus componentes. Serão transformadas em metadados as definições de: desenho do processo (árvore de tarefas automatizadas e humanas), regras de negócio, configuração dos componentes de *software* utilizados (tipo de comunicação, interface, qualidade do serviço, etc.), definição da sintaxe e semântica dos dados;
- Possibilitar a utilização de ferramentas integradas ou orientadas a modelos, MIC – *Model-Integrated Computing*. MIC é um paradigma de desenvolvimento que fornece um ambiente de modelagem aonde os desenvolvedores podem gerar um modelo integrado, contemplando desde uma visão total da aplicação em desenvolvimento até a visão da definição dos detalhes de integração entre os componentes que estão sendo utilizados [NSG – 2002]; o modelo desenhado e configurado permite a geração do código fonte das aplicações. Esse paradigma seria aplicado à automatização da geração do código das novas aplicações de integração e automação, configuração de componentes pré-existentes, criação de bancos de dados virtualmente unificados, etc.;

### **7.1.3 Configuração da solução pela conjugação de componentes locais e remotos;**

A proposta apresentada é de um *framework* onde componentes podem ser conectados ou pode-se estabelecer um ambiente de comunicação entre componentes de *software* como *middleware*, *Web Services* ou mesmo servidores de aplicação e outras aplicações de *software* para integração, gerenciamento de processos de negócio, *workflow* e etc..

- Disponibilizar (publicar) os recursos de *software* próprio e utilizar recursos de terceiros, via padrão SOA – *Service Oriented Architecture*, como, por exemplo, *Web Services*. Isso

possibilitaria a aplicação utilizar outros serviços ou rodar várias instâncias de si mesma e ainda assim o conjunto ser gerenciável. A possibilidade de acionar outras aplicações via SOA agregaria também o conceito de total interoperabilidade.

## 7.2 Módulos da aplicação

### 7.2.1 Desenho do modelo

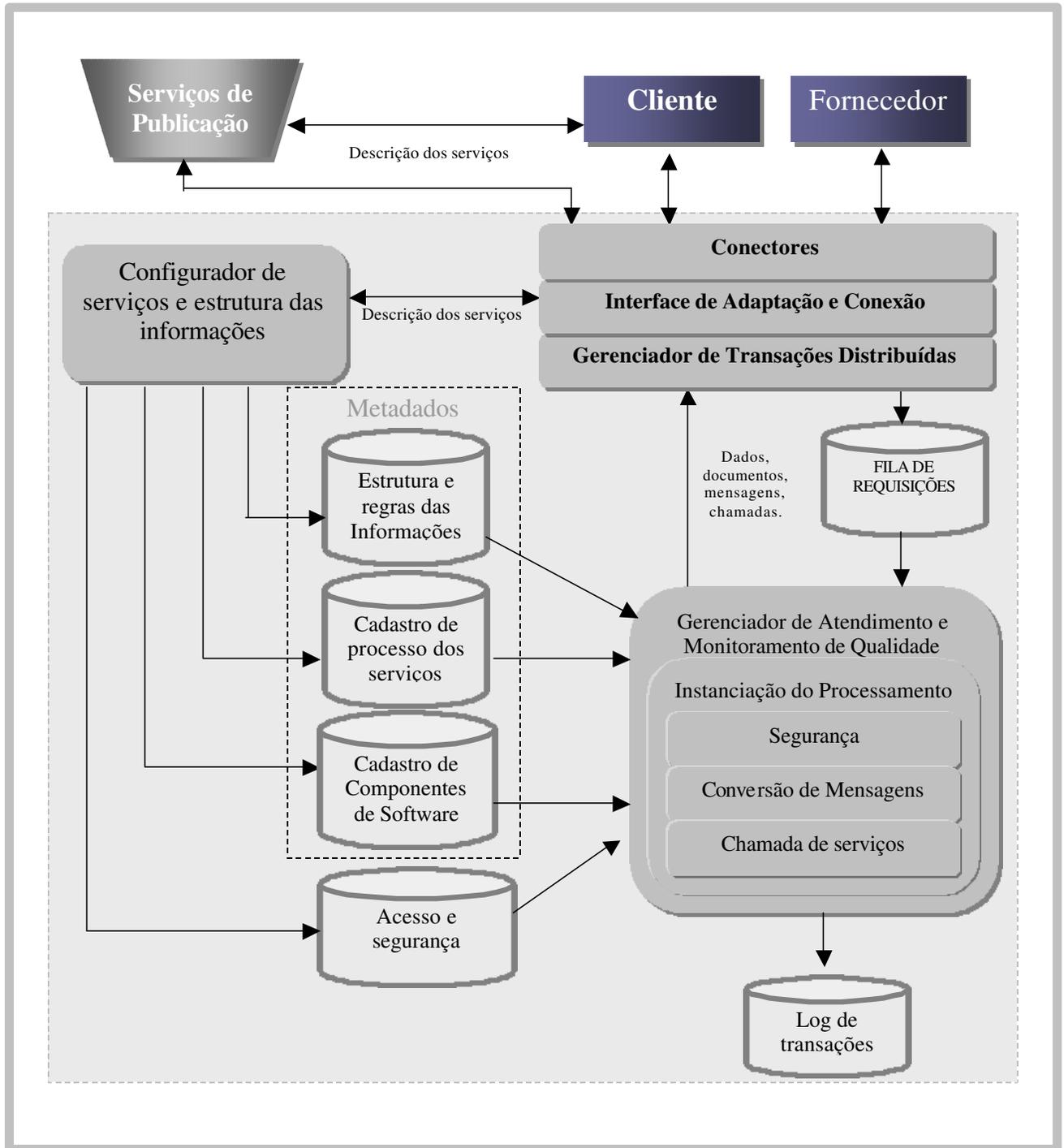


Figura 7.2 – Modelo Conceitual para EAI

### 7.2.2 Detalhamento do Modelo

- Serviços de Publicação – Serviços de Diretório de Publicação Local ou Remota utilizando padrão UDDI
- Cliente e Fornecedor – Aplicações, Servidores de Aplicação, RPC's, Banco de Dados, API's, usuários, *hardware* com aplicações de software ou mesmo outras instâncias do próprio EAI podem interagir com a aplicação. Em conformidade com o contexto apresentado, a diversidade de *hardware* e *software* do ponto de vista de EAI não permite, ao se imaginar uma solução única, limitar os tipos de clientes e fornecedores de informação
- Conectores – Biblioteca de componentes de *software* e objetos (ORB's) com os papéis, entre outros, de monitorar o recebimento de chamadas, acontecimento de eventos e possibilitar a interoperabilidade com diversos padrões de *software*. Os adaptadores podem ser serviços criados ou gerados pelo Configurador de Serviços a partir da orientação a modelos (MIC). A interoperabilidade pode determinar a aceitação, entre outros de:
  - ORB's – CORBA, COM;
  - API's proprietárias de pacotes de *software*;
  - API's personalizadas;
  - Escritores e Leitores de *Web Services*;
  - Conectores de Bancos de Dados como ODBC, JDBC;
  - Serviços de Mensagens (JMS, MSMQ, MQSeries);
- Interface de Adaptação e Conexão – módulo destinado a realizar a entrada e saída de requisições do EAI operando como um *hub* no papel de unificar o atendimento desses múltiplos padrões mencionados, distribuindo (roteirizando) e recebendo as requisições dos acopladores de *software*. Após receber, realizar o reconhecimento do conteúdo das mensagens, verificar a validade dos dados e rotular as requisições, deposita as mesmas na fila de requisições. As respostas ou requisições recebidas internamente do gerenciador de atendimento são encaminhadas aos adaptadores adequados, para serem remetidas externamente. Os serviços cadastrados podem indicar em sua configuração a necessidade

de disponibilização de adaptadores, entre outros para: perceber o acontecimento de eventos, aguardar o recebimento de mensagens, checar serviços de distribuição de mensagens ciclicamente, checar o status de aplicações ou bancos de dados.

- Gerenciador de Transações Distribuídas – módulo destinado ao controle de transações distribuídas que precisam de confirmações múltiplas de conclusão para efetivação das transações distribuídas;
- Configurador de serviços e estruturas de informações – Através da utilização de metadados, faz a criação de serviços (ou conjunto de serviços formando processos de negócio), estabelece direitos de acesso aos serviços, roteamento estático, especifica a estrutura das informações de entrada e saída do serviço.
  - Serviços – são criados serviços próprios e catalogados serviços de terceiros. Especificação, entre outros, das regras de negócio, nível de privacidade e acesso, qualidade exigida (velocidade, persistência), tipo de processamento (síncrono, assíncrono) e padrões técnicos do serviço (plataforma, *middleware*, banco de dados). Possibilidade de composição de serviços com conjunto de serviços, onde a finalização de um dispara a execução de outro. Diversos tipos de ferramentas podem ser utilizados para o desenho de processos ou, por exemplo, descrição de *Web Services*. Os serviços internos do EAI podem ser publicados como *Web Services* em diretórios de publicação ou pode haver acesso direto a serviços para clientes.
  - Estrutura das Informações – especificação da estrutura dos dados fonte e destino para cada serviço. Registro das restrições lógicas de sintaxe dos dados. Montagem de um dicionário semântico (registrando o significado das informações). Outras possíveis funções auxiliares:
    - Mapeamento entre tabelas, registros e campos de bancos de dados, através de filtros SQL – prevendo sempre a composição um para muitos e vice-versa;
    - Geração de scripts para criação, por exemplo, de triggers para disparar tarefas em bancos de dados;

- Interpretação e geração, entre outros de estrutura de informações, documentos, dados em XML;
  - Conversão de tipos de dados.
- Gerenciador de Atendimento e Monitoramento de Qualidade – Módulo de gerenciamento que controla o atendimento de recepção, processamento e respostas ao fluxo de requisições, monitorando a qualidade dos serviços. O Gerenciador além de oferecer visão de nível mais alto através da medição do comportamento da execução das etapas dos processos , também engloba as funções técnicas como:
- Instanciação do Processamento – Por atender em determinados casos a tarefas síncronas ou mesmo havendo a possibilidade de distribuição do processamento, determinadas requisições , como RPC´s por exemplo, poderão ser processadas em várias camadas para evitar interrupção de processamento ou remotamente utilizando em ambas as Chamadas de Serviços .
  - Segurança – deverá ser controlada a segurança pelo menos nos aspectos de acesso aos serviços e preservação do sigilo do conteúdo das mensagens no transporte, o que respectivamente pressupõe controle dos níveis de acesso aos serviços bem como uso da criptografia para o corpo das mensagens.
  - Conversão de Mensagens – com base na estrutura das informações definidas para fonte e destino dos serviços, a função de conversão traduz os formatos e padrões entre as partes comunicantes;
  - Chamadas de serviços – um mesmo serviço pode estabelecer abaixo dele próprio uma série de outros serviços encadeados formando uma árvore de serviços. Esses serviços podem ser realizados gerando requisições internas dentro do próprio EAI ou através de chamadas a um fornecedor de serviços, que realizará um papel semelhante ao do EAI. As chamadas de serviço correspondem efetivamente ao processamento de requisições. Esse processamento é realizado conforme a sintaxe determinada na especificação dos serviços do ponto de vista de escolha do componente de software a ser utilizado, tipo de processamento (síncrono/assíncrono), limite de tempo de

atendimento e conforme a especificação de segurança atribuída. A decisão de processar local ou remotamente fica a cargo da instanciação da aplicação sendo a sua função simplesmente de empacotamento do serviço e retorno à fila de requisições para saída da requisição.

□ Metadados

- Estrutura e regras das informações – Banco de dados contendo a definição de localização física, tipo, formato, semântica, faixas de validade e regras de integridade das informações;
- Cadastro de processos dos serviços – Especificação da seqüência lógica de etapas dos processos, componentes de software utilizados para cada processo, regras de transação, mapeamento entre as estruturas de informação de clientes e fornecedores de serviços;
- Cadastros de componentes de software – Especificação dos componentes de software com descritivo de suas funcionalidades, compatibilidades com plataformas operacionais, interfaces e parâmetros de processamento.

□ Dados

- Fila de Requisições – Banco de dados com o registro das requisições em andamento;
- Acesso e Segurança – Banco de dados com o registro das restrições de acesso de clientes e a fornecedores de serviço;
- Log de Transações – Banco de dados com o registro das transações realizadas, tais como: Movimento das requisições atendidas, transações não efetivadas, histórico de comportamento das aplicações e histórico de clientes e fornecedores.

### **7.3 Características dos recursos tecnológicos a serem empregados**

A construção da aplicação acompanhará algumas políticas baseadas no conceito EAI aqui apresentado, sendo elas:

- ❑ Independência da plataforma operacional cliente e fornecedora de serviços;
- ❑ Atuação planejada para acontecer desde numa rede local até a Internet;
- ❑ Previsão da necessidade de comunicação com outros serviços de integração, tais como: Microsoft Biztalk, IBM MQSeries, entre outros;
- ❑ Previsão de integração com pacotes de *software* como ERP's, CRM's e outros;
- ❑ Adoção de padrões abertos, por exemplo: XML, *Web Services*, CORBA;
- ❑ Definição de arquitetura com característica de baixo acoplamento e processamento distribuído;
- ❑ Tratamento de recuperação a falhas, oferecendo garantias em caso de quedas do EAI, as requisições devem permanecer íntegras;
- ❑ Reutilização de componentes para compor múltiplas soluções combinadas;

## 8 Conclusão

Pode ser observado durante toda a dissertação que o domínio do problema do ponto de vista da engenharia de *software* abrange tanto a visão extra-empresa como a intra-empresa. Uma das formas de olhar o problema é entender que os processos externos, imediatamente anteriores e imediatamente posteriores ao negócio, hoje fazem parte do escopo de um projeto de tecnologia da informação devido a visão possibilitada pelos recursos tecnológicos atualmente disponíveis. Considerar o fluxo de informação levando em conta a periferia da empresa (fornecedores, clientes, parceiros) é fundamental do ponto de vista dos negócios, pois habilita os fluxos de troca com o mercado, ambiente de inserção da empresa. Outra forma de olhar o problema é enxergar os processos de negócios e não as aplicações ou a empresa em si. Os processos de negócio formam uma teia de ações conjugadas, não são delimitados territorialmente em relação à empresa, à um *software* específico ou um ponto específico qualquer. Os processos de negócio compõem, de maneira situacional e também dinâmica, essa teia de ações destinada a atender determinados objetivos de negócio. Os objetivos de negócio ou a finalidade dos processos de negócio, elementos que norteiam a existência dos mesmos, contém requisitos a serem satisfeitos. Os requisitos da aplicação nascem então dos requisitos originados nos objetivos do processo de negócio.

EAI existe para suprir a defasagem da visão da tecnologia, que cria aplicações para atender as necessidades mutantes dos processos de negócios, e tem um resultado fragmentado. Nos últimos tempos a tecnologia quase tem conseguido oferecer uma gama de recursos compatíveis com as necessidades de negócio. Durante o amadurecimento da tecnologia, o *software* ou tudo aquilo que se pensou, projetou e desenvolveu foi feito com base na visão limitada pela própria limitação da tecnologia. Hoje a tecnologia pode atender mais que o departamento da empresa, ela pode processar a informação em tempo real e via internet em qualquer lugar. As aplicações de *software* que se estendem além das corporações, as que se atualizam diariamente impondo novas vantagens competitivas; os negócios que se encolhem, que se especializam e se globalizam, compõem um cenário realmente muito dinâmico e rápido, que cobra uma revisão dos métodos para se pensar e desenvolver tecnologia. Uma aplicação EAI orientada a processos gerenciáveis, se

comportando como um *framework* para composição de aplicações através da junção de componentes interoperáveis: é uma proposta preocupada com mudança e integração.

Preservar o conhecimento e o investimento utilizando metadados para processos, dados e configuração de *software*, assim como ter a camada de interoperabilidade proposta pelo padrão SOA, parece ser o caminho propício para sustentar tecnicamente o *framework* proposto.

Como tópicos a serem pesquisados em trabalhos futuros, alguns realmente poderiam complementar o estudo sobre integração de aplicações. São eles:

- Em engenharia de software – extração de requisitos voltada à integração de aplicações;
- BPML – processos auto-descritivos – definição de processos de negócio;
- Aprofundamento técnico no EAI proposto – esclarecer a combinação das tecnologias propostas e mesmo a implementação do modelo;
- MIC – desenvolvimento orientado a modelos – teste dessa ferramenta CASE para o desenho e implementação da solução;
- Comparação do modelo apresentado com as soluções de integração do mercado – obtenção de uma visão detalhada das funcionalidades das ferramentas disponíveis no mercado, cruzadas com a proposta realizada.

## 9 Bibliografia

- [[BMT – 2002](#)] Roger Bodamer, Ashish Mohindroo, Tina Tang. *Application Servers: Changing the Face of Enterprise Application Integration*.  
<http://www.eaijournal.com>, Outubro 2002
- [[Butler – 2002](#)] Martin Butler. *What’s Better: Application Integration or Integrated Applications?*.  
<http://www.eaijournal.com>, Novembro 2002
- [[Cherry – 2000](#)] Cherry Tree & Co. “*Extended Enterprise Applications*” – *Spotlight Report*  
[www.cherrytreeco.com](http://www.cherrytreeco.com), Janeiro 2000.
- [Collins – 2002] Jim Collins. *Empresas feitas para Vencer*. Campus, 2002.
- [Correa – 2001] Antonio Carlos Corrêa. *Neogrid – A2A – Mantel – EAI. Enterprise Application Integration – Integração A2A e B2B*  
[www.neogrid.com.br](http://www.neogrid.com.br), Março 2001.
- [Cummins – 2002] Fred A. Cummins. *Integração de Sistemas – EAI Enterprise Application Integration*. Arquiteturas para Integração de Sistemas e Aplicações Corporativas. Campus, 2002.
- [[Delphi – 2003](#)] Delphi Group. *BPM2003 – Market Milestone Report*  
[www.fuego.com](http://www.fuego.com), Abril 2003.
- [Franco – 2001] Carlos F. Franco Jr.. *e-Business*. Tecnologia de Informação e Negócios na Internet. Atlas, 2001.
- [Hailstone – 2003] Rob Hailstone. *Integration Strategies: The Start of Convergence*.  
<http://www.idc.com>, Janeiro 2003.
- [JBLN – 2001] Matjaz B. Juric, S. Jeelani Basha, Rick Leander, Ramesh Nagappan. *Professional J2EE EAI*. Wrox Press Ltd., UK, 2001
- [[JSEPI – 2002](#)] John Schmidt. *EAI Principles, Part I*  
<http://www.eaijournal.com>, Outubro de 2002

- [JSEPII – 2002] John Schmidt. *EAI Principles, Part II*  
<http://www.eaijournal.com>, Novembro de 2002
- [JSEPIII – 2002] John Schmidt. *EAI Principles, Part III*  
<http://www.eaijournal.com>, Dezembro de 2002
- [JSEPV – 2003] John Schmidt. *EAI Principles, Part V*  
<http://www.eaijournal.com>, Fevereiro de 2003
- [Laroia – 2002] Ankur Laroia. *EAI Facts and Fiction: A Look at the Promises and Realities of EAI Tools*  
<http://eai.ebizq.net>, Março 2002.
- [Levine – 2002] Russell Levine. *Estimating the Scope of your EAI Project*  
<http://www.eaijournal.com>, Julho 2002.
- [Linthicum – 2000] David S. Linthicum. *Enterprise Application Integration*. Information technology series. Addison Wesley, 2000.
- [Linthicum – 1999] David S. Linthicum. *Data Level Integration (Part I)*.  
<http://eai.ebizq.net/>, Janeiro 1999.
- [Linthicum – 1999\_2] David S. Linthicum. *Data Level Integration (Part II)*.  
<http://eai.ebizq.net/>, Fevereiro 1999.
- [Linthicum – 2002] David S. Linthicum. *Enterprise Evolution: Where EAI Meets B2B*.  
<http://eai.ebizq.net/>, Março 2002
- [LSMK – 2003] Leo Sayavedra, Michael Kuhbock. *The High Cost of Postponing Integration Deployments*  
<http://ebizq.net/>, Setembro 2003.
- [Lublinsky – 2001] Boris Lublinsky. *The Ultimate EAI Implementation*.  
<http://www.eaijournal.com>, Fevereiro 2001.
- [MacGoveran – 2002] David McGoveran. *Data Integration [I](#), [II](#), [III](#), [IV](#), [V](#), [VI](#), [VII](#), [VIII](#), [IX](#)*.  
<http://www.eaijournal.com>, Junho 2002 a Fevereiro de 2003

- [NSG – 2002] Nambor W.B. Natarajan, Douglas C. Schmidt, Aniruddha Gokhale. *Using Model-Integrated Computing to Compose Web Services for Distributed Real-time and Embedded Applications* <http://www.cs.wustl.edu>, Junho 2002.
- [Stokes – 2001] Nigel Stokes. *EAI and Beyond: A Multi-Level Flow Model*. <http://www.eaijournal.com>, Maio 2001
- [Yantra – 2001] Yantra Corporation. *The Impact of Technology on Outsourced Logistics*. <http://www.ebizq.net/>, Abril 2001.
- [W3CWS – 2002] W3C - World Wide Web Consortium. *Web Services Architecture* <http://www.w3c.org/TR/2002/WD-ws-arch-20021114/>, Novembro de 2002.

## 10 Glossário

Termo	Definição
API	<i>Application Programming Interface</i> – Interface de programação de uma aplicação corresponde à exposição de um conjunto de funções por um módulo de software. Através de uma API um software oferece acesso as suas funcionalidades a outros módulos de software. O método de exposição via API's, por suas características, pode ser útil para realização de integração entre as aplicações.
B2B	<i>Business-to-Business</i> – De negócio para negócio, esse termo se refere as transações realizadas entre duas ou mais organizações. É um novo estilo de realizar negócios praticado via Internet. Focado em velocidade, integração e redução de custos para troca de informações possibilita oferecer novos serviços voltados a retenção de clientes.
B2C	<i>Business to Consumer</i> – Similar ao B2B o termo no caso se aplica as transações realizadas entre uma empresa e um consumidor individualmente, analogamente o varejo via Internet.
BAM	<i>Business Activity Monitoring</i> – Monitoramento das atividades de negócio - é um termo divulgado pelo Gartner Group, se refere à análise e apresentação de informações, em tempo real ou histórico, do desempenho das operações e transações do negócio.
<i>Batch Process</i>	Processamento em lotes – é um termo de sistemas referente ao tipo de processamento onde uma transação não é processada no momento do seu acontecimento. Normalmente são acumuladas transações, em lotes, para posterior processamento.
BPM	Business Process Management – Gerenciamento dos processos de negócio é o conceito de agrupar e estruturar as tarefas humanas e automatizadas de uma forma que possam ser medidas e controladas.
<i>Business Process</i>	Processos de negócio são conjuntos de tarefas típicas de uma organização como, por exemplo, venda de produtos, recebimento de pedidos e entregas.

Termo	Definição
	Um processo de negócio depende de uma série de recursos, dentre eles, instalações, pessoas e recursos de tecnologia. Uma organização é movida por uma série de processos de negócio interdependentes.
BPA	<i>Business Process Automation</i> – Automação de processos de negócio corresponde à utilização de recursos tecnológicos, de forma dinâmica, combinando funções de negócio complexas em uma organização.
Cadeia de Valor	Cadeia de valor ou <i>value chain</i> é um conjunto seqüencial de atividades que direta ou indiretamente as empresas aplicam para transformar os recursos que entram em saídas com valor adicionado para seus clientes externos. As saídas assim como as entradas podem ser produtos e serviços.
CMM	<i>Capability Maturity Model</i> – é um conceito que envolve um padrão formal de especificações dos estágios de evolução em determinadas áreas de competência na engenharia de software. Desenvolvido pelo <i>Software Engineering Institute</i> (SEI) com o objetivo de melhorar o processo de desenvolvimento de software e aumentar a qualidade das aplicações, o modelo oferece um referencial evolutivo às boas práticas em software..
COM	<i>Component Object Model</i> é um modelo de programação orientado a objetos da Microsoft, que define o relacionamento entre os objetos dentro de uma única aplicação ou entre aplicações. No modelo COM o software cliente acessa um objeto (ou seus métodos) através da sua interface. Dois padrões Microsoft baseados em COM são OLE e ActiveX.
CORBA	<i>Common Object Request Broker Architecture</i> é um padrão definido por uma organização chamada OMG ou <i>Object Management Group</i> . A especificação CORBA foi desenvolvida com o objetivo de proporcionar interoperabilidade entre aplicações com independência da localização, linguagem de desenvolvimento das mesmas, dos computadores e sistemas operacionais sobre os quais estão rodando.
CRM	<i>Customer Relationship Management</i> – O gerenciamento do relacionamento

Termo	Definição
	com clientes é um sistema de informações utilizado para planejar, organizar e controlar as atividades desde o pré-vendas até o pós-vendas. Envolve acompanhamento, em todos os canais, do comportamento de todas as interações que são realizadas com o cliente, objetivando o aprendizado das suas preferências de compra e atendimento e sua retenção.
DBMS	<i>Database Management System</i> – Sistema Gerenciador de Banco de dados é uma aplicação utilizada par criar, manter e prover acesso a bancos de dados. O DBMS esconde os detalhes físicos do banco de dados mostrando uma visão lógica da estrutura das informações.
DCE	<i>Distributed Computing Environment</i> – É um conjunto de serviços que permitem a criação, uso e manutenção de aplicações cliente/servidor em ambientes heterogêneos. Entre os serviços, estão presentes funcionalidades como chamadas de procedimentos remotos (RPC's), segurança ( <i>Security Services</i> ) e funcionalidades para ambiente distribuído.
EAI	<i>Enterprise Application Integration</i> ou Integração de Aplicações Empresariais é um termo de computação relacionado aos planos, métodos e ferramentas que agregam eficiência aos sistemas de informações através da unificação sistêmica da visão das informações.
<i>e-Commerce</i>	<i>Electronic commerce</i> ou comércio eletrônico compreende as transações comerciais envolvendo empresas e indivíduos, operacionalizadas por meio eletrônico. Este tipo de comércio se caracteriza por diminuir as limitações físicas encurtando a distância entre comprador e vendedor.
EDI	Electronic Data Interchange ou Troca Eletrônica de Dados é um tipo operação estabelecida pela comunicação direta entre computadores, que trocam informações em um formato padrão. Através dessa operação é possível às empresas realizar transações comerciais trocando documentos sem intervenção humana no processo.
ERP	Enterprise Resource Planning ou Planejamento de Recursos empresariais é

Termo	Definição
	um conjunto de aplicações de software que abrangem a parte interna de uma empresa. Automatizam, por exemplo, desde o departamento de recursos humanos, a área financeira, industrial e comercial de forma que os processos internos fiquem integrados.
<i>Event Driven Process</i>	<i>Event Driven Process</i> ou Processo orientado a eventos é um tipo de comportamento de alguns programas que no momento que estão rodando ficam aguardando a ocorrência de algum evento para tomar determinadas ações. Esse tipo de processo está presente em algumas das funções de sistemas operacionais, aplicações de robótica, software de aviação e alguns tipos de <i>middleware</i> , entre outros.
<i>Framework</i>	É uma estrutura que tem por função agregar uma série de componentes, tais como objetos, para implementar determinadas funções. Essa estrutura pode ligar dinamicamente esses componentes, promovendo sua interoperabilidade para que o conjunto formado possa ser utilizado para determinada aplicação.
<i>Hub</i>	Denominação atribuída a equipamento ou software que realiza principalmente a centralização do recebimento e encaminhamento de informações.
JAVA	Linguagem orientada a objetos desenvolvida pela empresa Sun Microsystems que é aceita pelo mercado como um padrão.
J2EE	Java 2 Platform Enterprise Edition (J2EE™) é uma plataforma Java que oferece suporte adicional principalmente ao desenvolvimento de aplicações multicamadas.
JAVA BEANS	Java Beans são componentes de software reutilizáveis que desempenham funções simples e podem ser agrupados ou combinados para criar aplicações mais complexas. A arquitetura Java Beans fornece um meio padrão para os objetos se identificarem e trocar informações, podendo rodar em qualquer plataforma operacional. Os objetos Java Beans podem se conectar a objetos como CORBA, ActiveX e outros padrões.

Termo	Definição
JDBC	Java <i>DataBase Connectivity</i> é um <i>middleware</i> , que permite a programas Java acessar dados dos bancos de dados relacionais, oferecendo uma interface de acesso padrão SQL. É similar ao ODBC, padrão para o qual está sendo disponibilizada sua integração. A interface JDBC permite que o mesmo programa Java acesse qualquer banco de dados que aceite o padrão.
LAN	<i>Local Area Network</i> ou rede local é o termo que se refere a redes de computadores, situadas em uma área geograficamente pequena e normalmente única.
<i>Legacy System</i>	O sistema legado é termo utilizado para descrever sistemas de informação com tecnologia ultrapassada, sobre os quais o único investimento realizado é o de manutenção de segurança.
<i>Logistics</i>	Logística é a função de fornecer ou distribuir materiais e produtos adequadamente em relação a tempo, quantidades e locais.
<i>Mainframe</i>	Termo que representa um paradigma em computação. Tipicamente se refere a computadores muito grandes, que estão instalados em um local diferenciado, tem alta capacidade de memória e processamento. Característicos da época em que a maior parte do processamento era realizado em <i>batches</i> e havia a centralização de sistemas multiusuários com terminais burros. Os <i>mainframes</i> têm sido, ao longo do tempo, substituídos pelo paradigma cliente/servidor.
<i>Metadata</i>	<i>Metadata</i> ou metadados se refere, de uma forma geral, as informações relativas a determinados dados, tais como: O que são os dados (significado), onde eles estão, qual a sua origem, qual o seu formato original, como são coletados e quais são suas regras de validade.
<i>Middleware</i>	Software que tem por finalidade facilitar a comunicação entre aplicações. O <i>middleware</i> fornece API's pelas quais as aplicações requisitam serviços e controlam a transmissão de dados.
MOM	<i>Message-Oriented Middleware</i> é o <i>middleware</i> orientado a mensagens. O

Termo	Definição
	MOM cria ou recebe mensagens que são armazenadas e disponibilizadas em uma fila. Quando o programa destinatário está pronto para receber e processar a mensagem, o mesmo a retira da fila. Isso exige ao programa cliente e ao programa fornecedor das mensagens a necessidade de se conectarem direta e continuamente.
MQSeries	É uma família de produtos da IBM inicialmente focada em serviços de fila de mensagens.
MRP	<i>Material Requirement Planning</i> – Planejamento de necessidade de materiais é um termo utilizado nas empresas nos anos 80 que foi substituído pelo termo ERP. Corresponde a um sistema computadorizado utilizado para determinar a quantidade de materiais e de tempo necessários à execução de um processo de produção.
MRP II	Algumas definições para MRP II são idênticas a definição anterior acrescidas do sufixo “II”. Outra versão seria <i>Manufacturing Resource Planning</i> (MRPII) ou Planejamento das necessidades de Fabricação. Adicionalmente ao MRP, essa versão expandida atende também a outras funções de planejamento relacionadas à fabricação nas áreas de compras, vendas, custo e programação de produção, entre outras.
MSMQ	Microsoft <i>Message Queue</i> – Produto Microsoft voltado a serviço de mensagens.
ODBC	<i>Open Database Connectivity</i> é um padrão aberto de API’s com função de acesso a banco de dados. Uma mesma aplicação via ODBC pode acessar dados de diferentes formatos. ODBC é baseado no padrão conhecido como <i>SQL Call–Level Interface (CLI)</i> .
OMG	O <i>Object Management Group</i> é uma organização sem fins lucrativos, aberta a novos associados, que produz e mantém especificações para indústria de computação, entre elas exemplos como CORBA e UML. A utilização de suas especificações é livre de qualquer ônus.

Termo	Definição
ORB	<i>Object Request Broker</i> é um tipo de <i>middleware</i> que facilita a interoperabilidade, e estabelece uma plataforma única, na qual objetos podem trocar dados e serviços, estando no lado cliente ou no lado servidor.
PRM	<i>Partner Relationship Management</i> – Gerenciamento de relacionamento com parceiros.
<i>Real-time Process</i>	É o processamento, ao contrário do <i>batch</i> , que é realizado no momento em que a transação acontece.
RPC	<i>Remote Procedure Call</i> ou chamada de procedimentos remotos é um modelo de programação que estabelece um padrão de comunicação cliente/servidor. É um tipo de comunicação tipicamente síncrono onde o cliente envia a mensagem e interrompe o seu processamento, enquanto aguarda a resposta do servidor.
SCM	<i>Supply Chain Management</i> ou Gerenciamento da Cadeia de Suprimento corresponde ao uso da tecnologia para auxiliar a manter atualizadas as informações sobre cada um dos participantes da cadeia e disponibiliza-las aos outros, orientando o funcionamento da cadeia no balanceamento entre suprimento e demanda. O objetivo é movimentar os produtos, da origem ao ponto de consumo, no menor tempo possível e ao menor custo.
SFA	<i>Sales Force Automation</i> ou Automação da Força de Venda corresponde ao uso da tecnologia para facilitar o trabalho de vendedores externos. Esse tipo de aplicação normalmente abrange funcionalidades tais como: gerenciamento de contatos, coleta de pedidos, informações de clientes incluindo comportamento nas compras e retornos da empresa relativos a acompanhamento de metas, comissão e correio eletrônico.
SOAP	<i>Simple Object Access Protocol</i> é um protocolo baseado em XML. SOAP foi originalmente desenvolvido para trabalhar com aplicações distribuídas se comunicando sobre HTTP através do sistema de segurança das empresas ( <i>firewalls</i> ). Esse protocolo define o uso de XML e HTTP para acessar

Termo	Definição
	serviços.
SQL	<i>Structured Query Language</i> é uma linguagem de computação utilizada para criar, manipular e analisar bancos de dados relacionais.
<i>Supply Chain</i>	A cadeia de suprimento corresponde à interligação de todos os pontos que participam no fluxo de materiais, produtos e serviços e informações desde os fornecedores de matérias-primas até os consumidores dos produtos.
<i>Web</i>	A abreviação para “ <i>World Wide Web.</i> ” A web é uma rede global onde computadores servidores oferecem acesso a arquivos formatados em HTML. Esses documentos formatados em HTML, ou páginas Web, podem conter texto, imagens, vínculos com outras páginas e arquivos ( <i>hyperlinks</i> ), entre outros. A Web, lançada em 1993, utiliza a Internet como meio de transmissão, mas constantemente as duas são confundidas como uma só.
<i>Workflow</i>	É uma ferramenta da tecnologia da informação que utiliza os sistemas de informação para controlar o fluxo de trabalho entre indivíduos e departamentos. Em um sistema de Workflow, um evento dentro do fluxo de trabalho (atividades e documentos) automaticamente leva a outro e a ocorrência de ações dispara mensagens para determinados participantes do fluxo.
WSDL	Web Services Description Language é uma linguagem baseada em XML. WSDL permite a documentos XML definir as interfaces para os Web Services.
UDDI	Universal Description, Discovery and Integration é uma especificação baseada em XML para registro e catalogação dos negócios e dos serviços que estes fornecem em seus <i>websites</i> . Esse serviço de publicação foi projetado para dar a possibilidade aos aplicativos de software de automaticamente descobrir serviços na <i>web</i> e automaticamente interagirem com eles, realizando transações dinamicamente.

---

Termo	Definição
XML	<i>Extensible Markup Language</i> é uma linguagem autodescritiva similar a HTML. Tanto XML quanto HTML contém uma metalinguagem para descrever o conteúdo de páginas e arquivos. Enquanto HTML descreve as páginas XML descreve o conteúdo em termos de dados. Um arquivo XML pode ser processado por um programa como dados.

---