

# INTEGRAÇÃO INDUSTRIAL: DANDO VIDA AO MIDDLEWARE (1)

Guilherme Camargo (2)

Haroldo Teixeira (3)

## Resumo

Com o crescimento do número de sistemas que suportam tanto o lado da planta quanto o lado da gestão, como ERP, SCADA, MRP, MES, PIMS, LIMS, CRM, etc, tornou-se necessário a elaboração de projetos voltados para a arquitetura de integração. A constante utilização de middlewares conhecidos do mercado fez inchar este tipo de componente, crescendo desordenadamente a famosa teia de aranha das arquiteturas. O objetivo deste trabalho é criar soluções em integração para os vários sistemas que compõem o processo siderúrgico, de forma eficiente e principalmente gerenciável. Como ponto de partida foi utilizado um pequeno problema, porém comum na maioria dos processos de expedição de produtos. A falta de sincronismo de tarefas coloca em risco várias tarefas do processo como emissão da nota fiscal, certificado de inspeção, marcação do produto, logística de transporte, etc, ocultando possíveis falhas dos subsistemas envolvidos. Para isto foi criado o seguinte cenário: utilização de middlewares (síncronos e assíncronos), gerenciadores de transações, banco de dados, drivers, camadas de software, máquinas de diferentes arquiteturas e sistemas operacionais, *workflow*, etc. Será adotada a metodologia de integrar várias funcionalidades no middleware, como workflow, regras de mensagens e acompanhamento (visibilidade) do processo. Serão apresentados as vantagens desta arquitetura e também questionar paradigmas como custo elevado, retorno a longo prazo e complexidade da tecnologia. Além da rapidez obtida na automação dos processos envolvidos, podemos incluir novas funcionalidades ao seu middleware como benchmark, BI, B2B, B2C, além de criar um cenário propício para mudanças estratégicas das siderúrgicas. Com o crescimento da concorrência no mercado de metalurgia, aquele que tiver sua infra-estrutura tecnológica integrada, irá conseguir responder de forma rápida e eficiente a variações do mercado.

**Palavras-chave:** integração, processos, middleware, automação.

(1) Trabalho a ser apresentado no VIII Seminário de Automação de Processo da ABM - Outubro de 2004 – Belo Horizonte – MG – Brasil

(2) Gerente de Soluções - CONTRASTE

(3) Gerente de Tecnologia - CONTRASTE

## 1 Introdução

As maiorias das empresas da área de metalurgia utilizam middlewares para comunicação de sistemas. Alguns são até subutilizados. Além da definição e modelos que o middleware pode formar, este recurso é aplicado pontualmente na automação do processo produtivo. É comum seu emprego apenas para elevar a informação em níveis verticais do sistema. Ou apenas fazer a conexão ponto a ponto de sistemas.

Mas pode-se agregar valor a esta camada de software. Com planejamento e projeto consistentes é possível realizar funções para facilitar os sistemas pilares como SCADA, MES, PIMS, ERP, PCP e Manutenção.

## 2 Objetivo

O objetivo deste trabalho é a elaboração de um novo modelo de middleware para gerar uma visibilidade do processo de manufatura das indústrias siderúrgicas. Este modelo será baseado no conhecimento do processo de produção, aproveitamento dos middlewares instalados e preparação do dado para informações de gestão.

O resultado será a mudança de paradigma, em que o middleware de função de operacional, passa a ser reconhecido como uma ferramenta estratégica para a empresa.

## 3 Definição do Conceito Middleware

Dando uma definição mais precisa, podemos dizer que “um middleware é um software de conectividade que consiste de um conjunto de serviços disponíveis que permite que múltiplos processos, executando em uma ou mais máquinas, interajam através de uma rede”. A Figura 1 mostra o uso do middleware.



Figura 1 – Uso do middleware

### 3.1 Paradigmas para a Construção de Middlewares

Para realizar a integração e o desenvolvimento de aplicações distribuídas o mais simples o possível, a maioria dos middlewares é baseado em algum modelo, ou

*paradigma*, para realizar a distribuição e comunicação. Basicamente são quatro esses paradigmas: Transaction Processing Monitors (TPs), Remote Procedure Calls (RPCs), Message-Oriented Middlewares (MOMs) e Object Request Brokers (ORBs).

### 3.1.1 Transaction Processing Monitors (TPs)

A tecnologia Transaction Processing (TP) Monitor provê ao ambiente cliente/servidor distribuído a capacidade para desenvolver, rodar e gerenciar aplicações com transações de forma eficiente e confiável.

A tecnologia TP monitor controla aplicações de transação e realiza computações de negócios e atualizações em banco de dados. Ela pode prover serviços de aplicações para milhares de clientes em um ambiente cliente/servidor distribuído. Tal tecnologia realiza isso através da multiplexação de pedidos de transações de clientes (por tipo) sobre um número controlado de rotinas de processo que suportam serviços particulares. Estes eventos são mostrados na Figura 2.

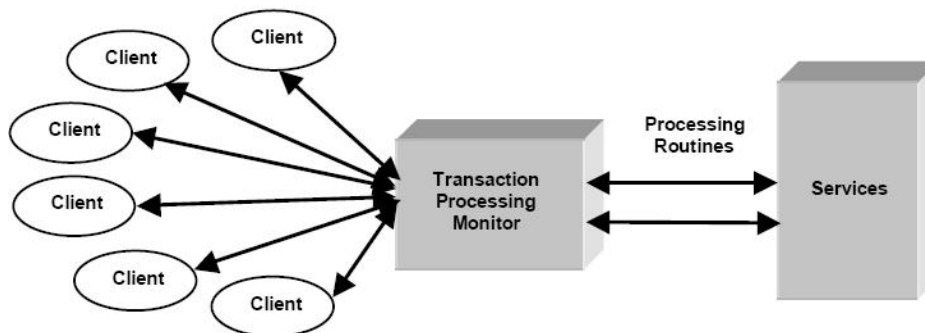


Figura 2 – Tecnologia de Monitor de transação de processo

A tecnologia TP monitor mapeia vários pedidos de clientes através de rotinas de serviços de aplicação para melhorar a performance do sistema. Tal tecnologia (locada com um servidor) pode também realizar as transições lógicas para o cliente. Isto reduz o número de atualizações requeridas por estas plataformas de clientes. Além disso, essa tecnologia inclui várias características de gerenciamento, tais como reinício de processos falhos, balanceamento dinâmico de carga e obriga a consistência de dados distribuídos. Essa tecnologia é escalável pela adição de mais servidores.

### 3.1.2. Message-Oriented Middlewares (MOMs)

Message-Oriented Middleware (MOM) é uma infraestrutura cliente/servidor que aumenta a interoperabilidade, portabilidade e flexibilidade de uma aplicação, permitindo que tal aplicação seja distribuída sobre múltiplas plataformas heterogêneas. Ela reduz a complexidade do desenvolvimento de aplicações que funcionam sobre múltiplos sistemas operacionais e protocolos de rede através do isolamento do desenvolvedor da aplicação de detalhes dos vários sistemas operacionais e interfaces APIs de rede que são usados na construção do sistema. O modelo é mostrado na Figura 3:

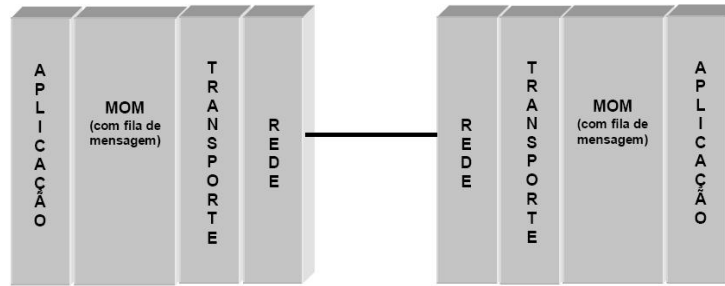


Figura 3 – Middleware orientado a mensagem

O MOM aumenta a flexibilidade de uma arquitetura por habilitar que aplicações troquem mensagens com outros programas sem ter de conhecer em qual plataforma ou processador a outra aplicação reside dentro da rede. As mensagens acima mencionadas podem conter dados formatados, pedidos de ações, ou ambos. Normalmente, o sistema MOM provê uma fila de mensagem entre os processos interoperantes, assim se o processo destino estiver ocupado, a mensagem é armazenada temporariamente até que ela possa ser processada. O MOM é tipicamente assíncrono e peer-to-peer, mas muitas implementações suportam também passagem síncrona de mensagens.

#### 4.2.3 Object Request Brokers (ORBs)

Object Request Broker, ou simplesmente ORB, é uma tecnologia que gerencia a comunicação e a troca de dados entre objetos. Em outras palavras, o ORB provê interoperabilidade em sistemas de objetos distribuídos. Ele permite a construção de sistemas pelo agrupamento de objetos que se comunicam entre si através dele. Os detalhes da implementação do ORB geralmente não são importantes para os desenvolvedores de sistemas distribuídos. Os desenvolvedores devem se preocupar apenas com os detalhes da interface do objeto. A tecnologia ORB permite a comunicação de objetos entre diferentes máquinas, diferentes softwares e diferentes fornecedores. Um ORB provê um diretório de serviços e auxilia a estabelecer conexões entre clientes e estes serviços. Esta definição pode ser ilustrada na Figura 4.

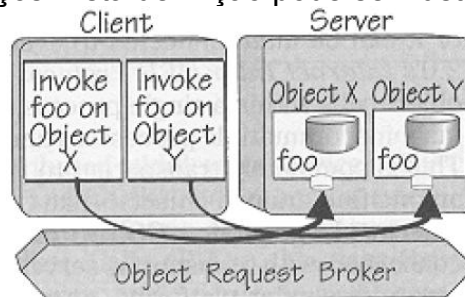


Figura 4 – Middleware Object Request Broker

Os principais padrões para computação distribuída de hoje e que utilizam a tecnologia ORB são:

- Common Object Request Broker Architecture (CORBA);
- Distributed Component Object Model (DCOM);
- Java Remote Method Invocation (Java RMI);

### 3.1.2. Remote Procedure Calls (RPC)

O modelo de RPC é semanticamente semelhante a chamada local de procedimento, no que consiste na passagem de controle e parâmetros para outro programa, nos seguintes aspectos:

- Ao chamar um programa, o chamador coloca os argumentos em uma localização bem determinada;
- O controle é então transferido à seqüência de instruções que constituem o corpo do procedimento;
- O corpo do procedimento é executado em um ambiente de execução recém criado que inclui copia dos argumentos dados pelas instruções de chamada
- Após o termino da execução o controle retorna ao ponto de chamada possivelmente com um resultado.

O RPC é uma extensão deste processo na qual o procedimento chamado está executando em uma maquina remota, num espaço de endereçamento diferente (eventualmente pode ser no mesmo computador). O mecanismo de IPC presta um papel fundamental neste serviço, viabilizando a transferência de mensagens entre os processos. A figura abaixo apresenta o esquema geral de RPC:

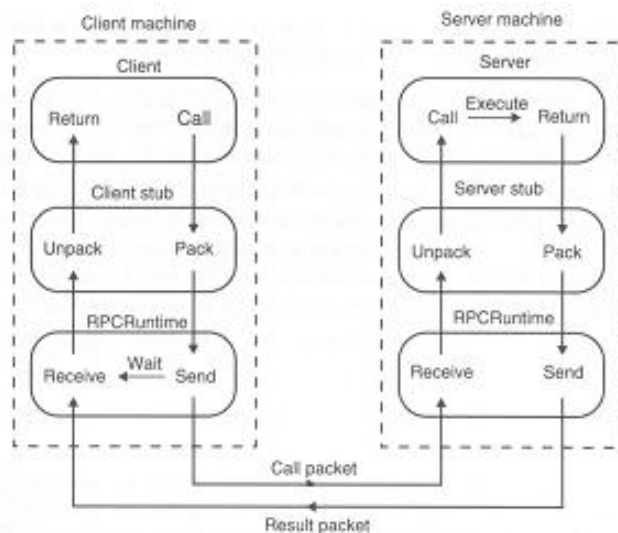


Figura 5 – Implementação do mecanismo RPC

## 4 Cenário

Utilizando os conceitos vistos acima poderemos idealizar um novo modelo que possa ser aplicado em situações usuais, em empresas que já utilizam a ferramenta middleware. A idéia inicial é incorporar novas funções a esta camada software com objetivo de solucionar problemas de integração dos processos de produção. Vamos imaginar o seguinte cenário - o processo de expedição de placas de aço, que envolve hipoteticamente os seguintes sistemas:

1. **MCC**: Sistema de Informação e controle, composto por SCADA, PIMS e MES, da máquina de corrida contínua que informa o fim da produção de placas de aço;
2. **PCP**: Sistema de controle e planejamento da produção, responsável pelo sequenciamento da produção e controle do estoque físico;
3. **Expedição**: Sistema de Informação que a partir dos lotes de placas identificados realiza a expedição do produto e realiza a entrada destes dados no sistema;
4. **MainFrame**: Sistema legado, responsável pela geração dos códigos de identificação e pela emissão do certificado de qualidade do produto ;
5. **Identificação**: Sistema de identificação das placas do estoque. Não possui entrada de dados no sistema;
6. **ERP**: Sistema de Gestão Corporativa responsável pelo estoque contábil e emissão de notas fiscais.

O fluxo deste processo é esboçado na figura 6, onde a seqüência de mensagens está entre parênteses:

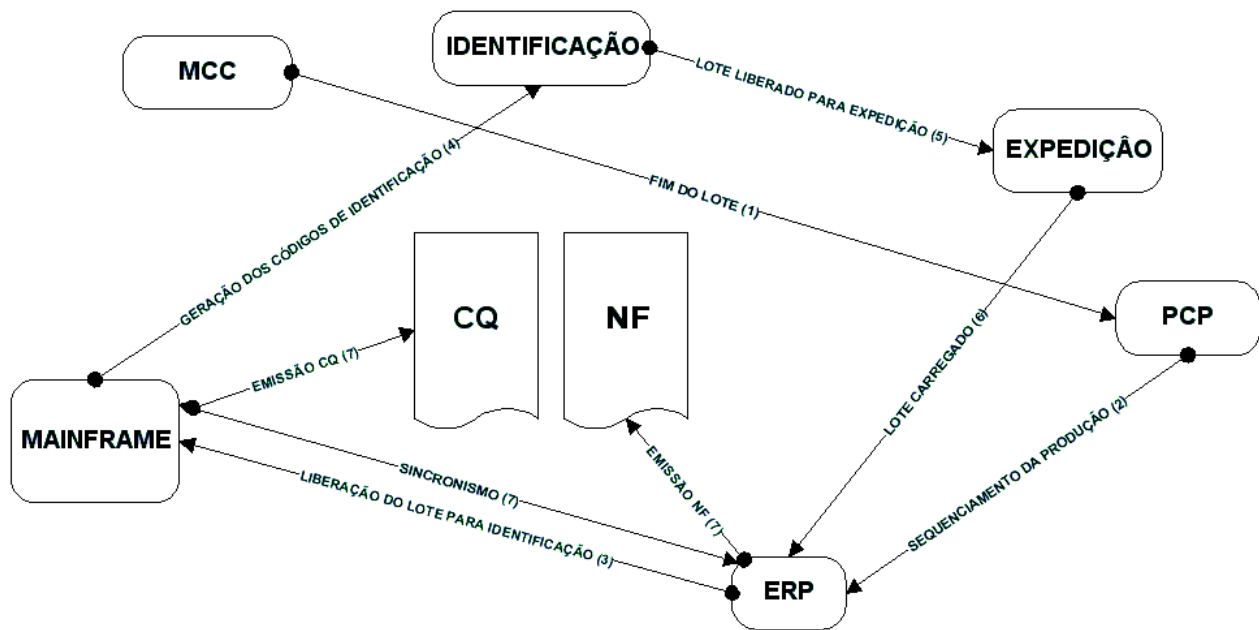


Figura 6 – Fluxo de um processo de expedição

Este fluxo lógico é composto por middlewares e drivers para viabilizar a comunicação entre os sistemas, que é representado na figura 7:

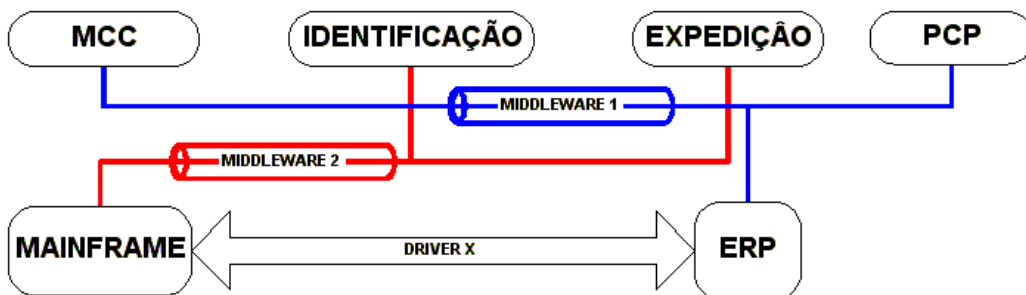


Figura 7 – Arquitetura lógica

**Middleware 1** : Produto de mercado que utiliza apenas o modelo MOM de distribuição.

**Middleware 2** : Produto de mercado que utiliza além do modelo MOM de distribuição a tecnologia ORB.

**Driver X** : Driver específico de mercado, com característica *peer-to-peer*.

Obs.: Com exceção do driver X, ambos middlewares possuem APIs ou serviços de objetos disponíveis.

Pode-se ver que os dois middlewares não se comunicam, dependendo do driver X para que ambos os processos se comuniquem. Com base nisso, qualquer anormalidade neste componente comprometerá o processo para expedição de placas. Outro ponto é a necessidade de sincronismo de dois eventos emissão de NF e emissão de CQ. É preciso criar garantias que ambos os eventos serão executados pelos sistemas heterogêneos.

Com sistema em produção estas necessidades não serão percebidas de imediato. Serão percebidas em duas situações distintas:

1. Na ocorrência de um problema em produção, quando será difícil diagnosticar onde ocorreu o problema.
2. No momento de inclusão de um sistema ou algum novo atributo no processo, que nos levará o impacto nos seis sistemas.

## 5 Arquitetura de Integração

Para resolver estas questões será preciso focar em arquitetura. O projeto de arquitetura será determinado pelo desenho do processo de expedição em questão. Será levada em consideração a estrutura física e lógica do cenário projetado.

O modelo será baseado em uma plataforma de integração onde todos os middlewares e drivers estarão convivendo harmoniosamente, respeitando os limites de sua arquitetura. Simplificando seria organizado conforme a figura 8:

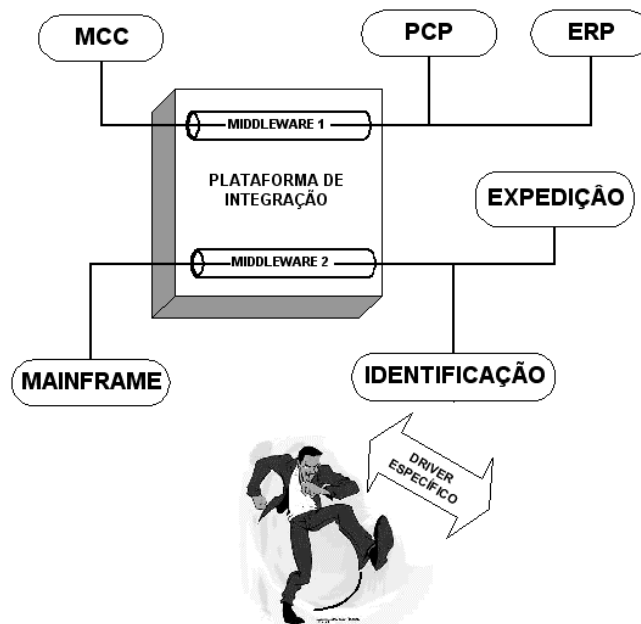


Figura 8 – Arquitetura do modelo de integração

A plataforma de integração seria baseada nos modelos TP e RPC, utilizando as API dos middlewares existentes. O mais importante na arquitetura será adoção do conceito ao invés de uma ferramenta ou um produto, modelando uma solução as necessidades do processo. Com isso serão mudados dois paradigmas :

**Operacional para Estratégico** – O posicionamento desta camada será diferente dos middlewares padrões, de essencialmente operacional para uma visão estratégica do processo de negócio ou processo de produção.

**Transparência para Visibilidade** – A principal característica do middleware que é transparência será mudada para visibilidade do processo, se tornando uma importante informação na tomada de decisão.

## 6 Funcionalidades

Esta plataforma possuirá três funcionalidades, conforme representado na figura 9:

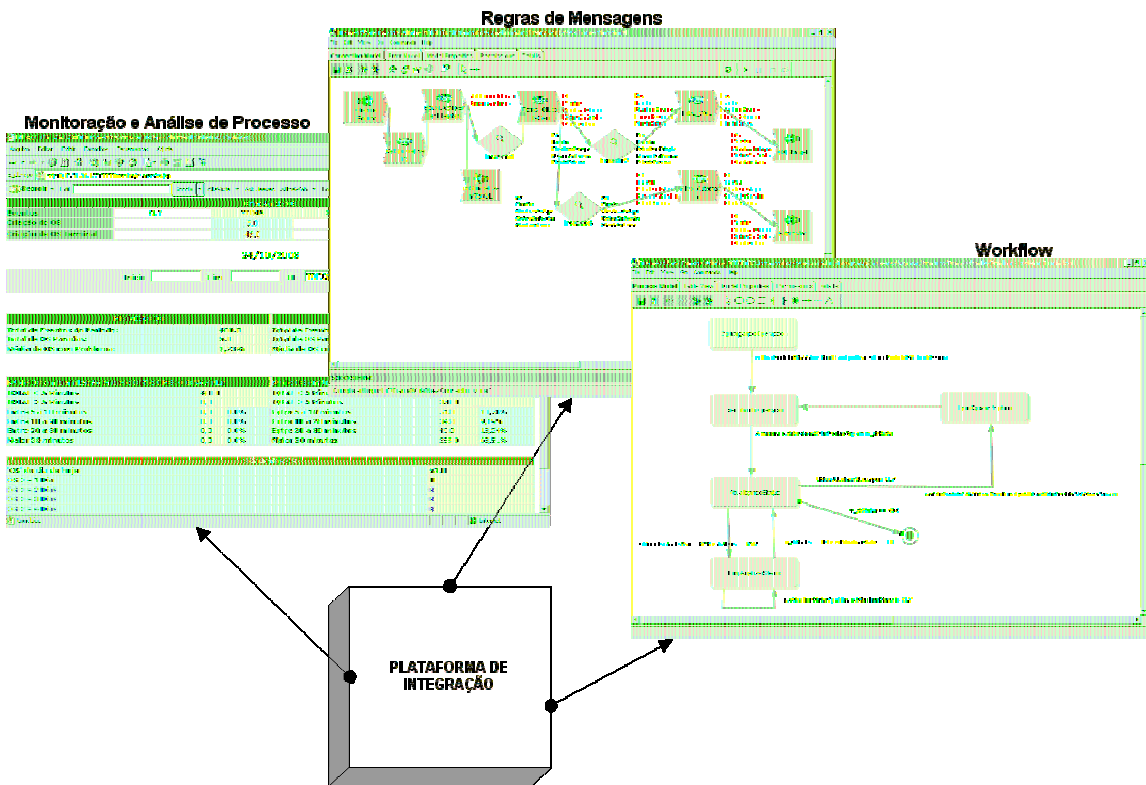


Figura 9 – Funcionalidades da plataforma

### ✓ Regras de mensagens

As mensagens que chegam ou saem precisam ser traduzidas e conduzidas conforme o padrão definido pelo analista do processo de integração. Estas regras são armazenadas em um banco de dados criando uma maior acessibilidade do sistema;



### ✓ Workflow

O desenho do fluxo do processo é realizado nesta ferramenta que utiliza as informações provenientes das regras de mensagens. Desenhando este fluxo, o processo de expedição torna-se explícito ao analista, fornecendo uma completa visibilidade do negócio.

### ✓ Monitoração do processo

Através das funcionalidades anteriores, é possível criar interfaces de seu processo. Estas interfaces serão responsáveis pela análise e monitoração do processo de expedição. No caso da expedição das placas de aço, podemos orientar o processo a corrida, que é o ponto inicial do fluxo. Através do número da corrida, podemos acompanhar todo processo de expedição em *real time*. Esta é visibilidade requerida pelo cliente. Com isso, é possível a rápida tomada de decisão e rastrear o andamento do lote de placas. Isso gera funcionalidades como controle estatístico, qualidade e eficiência, fornecendo o ambiente para o maior objetivo desta funcionalidade: otimização do processo.

Observe que a solução técnica pode utilizar um produto de mercado ou se basear em uma arquitetura proprietária, customizada sobre algum produto ou tecnologia, como J2EE ou WebServices. No caso de J2EE, é padronizado o uso de diversos mecanismos de comunicação entre os sistemas. Já no caso de Webservices, estamos utilizando uma plataforma que torna cada sistema um fornecedor ou consumidor de serviços, encapsulados em componente Java. Teríamos então uma arquitetura no seguinte formato apresentado na figura 10.

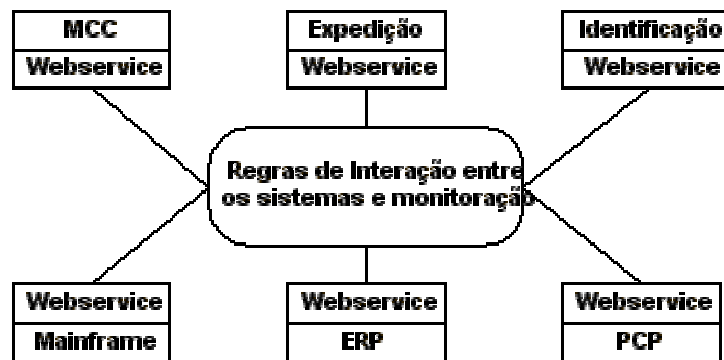


Figura 10 – Arquitetura Webservices

Deste modo, o webservice serve como camada de comunicação entre as aplicações e um módulo central em Java, onde é descrito o processo, sendo que esta infra-estrutura faz o papel de middleware.

Além disso, podemos estender este conceito não apenas para integração entre aplicações dentro da empresa, mas para outros parceiros de negócio. Com isso, poderíamos repassar informações sobre o andamento do processo de expedição do aço para o cliente solicitante, ou solicitar material para fornecedores, integrando a cadeia produtiva como um todo.

## 7 Conclusão

A tecnologia muda, se transforma, se adapta, porém os conceitos sempre permanecem. O diferencial é utilizar da melhor forma possível, os recursos e conceitos da tecnologia de informação. Atualmente, é comum os fabricantes de softwares reposicionarem seus produtos para “atender” seus clientes.

Foi visto também que atualização da tecnologia pode ser realizada sem impactos no processo, sem abandonar as ferramentas conhecidas e preparar condições para migrações futuras. Dentro das vantagens percebidas podemos citar:

- Redução dos custos de integração de novas aplicações;
- Redução dos custos de manutenção das aplicações existentes;
- Permitir e agilizar uma mais fácil migração para novas tecnologias e conceitos de negócio;
- Redução dos custos das fusões e aquisições, já que permitem uma mais rápida integração dos sistemas;
- Potenciam a partilha de informação entre as aplicações, já que tornam mais fácil e barata a sua integração;
- Automatizam processos de negócio ou de produção em real-time, criando uma maior capacidade de resposta a eventuais alterações;
- Tornam viável a distribuição das funcionalidades do processo entre diferentes sistemas;
- Permitem uma efetiva comunicação inter-empresa;
- Asseguram um ponto de controle bem definido.

# INDUSTRIAL INTEGRATION: GIVING LIFE TO THE MIDDLEWARE<sup>(1)</sup>

*Guilherme Camargo (2)*  
*Haroldo Teixeira (3)*

## **Abstract**

With the growth of the number of systems that in such a way support the side of the plant how much the side of the management, as ERP, SCADA, MRP, MES, PIMS, LIMS, CRM, etc, became necessary the elaboration of projects come back toward the integration architecture. The constant use of middlewares known of the market, made to swell this type of component disorderedly, growing the celebrity spider web of the architectures. The objective of this work is to create solutions in integration for the some systems that compose the siderurgical process, of efficient and mainly manageable form. As starting point a small problem, however common was used in the majority of the processes of expedition of products. The lack of synchronism of tasks places in risk some tasks of the process as emission of the bill of sale, certified of inspection, marking of the product, logistic of transport, etc, occulting possible imperfections of the involved subsystems. For this the following scene was created: use of middlewares (synchronous and asynchronous), managers of transactions, data base, drivers, layers of software, machines of different architectures and operational systems, workflow, etc. The methodology will be adopted to integrate some functionalities in middleware, as workflow, rules of messages and accompaniment (visibility) of the process. The advantages of this architecture will be presented and also to question paradigms as raised cost, return in the long run and complexity of the technology. Beyond the rapidity gotten in the automation of the involved processes, we can include new functionalities to its middleware as benchmark, BI, B2B, B2C, besides creating a propitious scene for strategically changes of the siderurgical ones. With the growth of the competition in the metallurgy market, that one that will have its integrated technological infrastructure will go to obtain to answer of fast and efficient form the variations of the market.

**Keywords:** integration, process, middleware, automation

(1) Trabalho a ser apresentado no VIII Seminário de Automação de Processo da ABM - Outubro de 2004 – Belo Horizonte – MG – Brasil

(2) Gerente de Soluções - CONTRASTE

(3) Gerente de Tecnologia - CONTRASTE