

PADRONIZAÇÃO DA COMUNICAÇÃO ATRAVÉS DA TECNOLOGIA OPC

Adriano Pedrosa Puda
adriano@softbrasil.com.br
SoftBrasil Automação Ltda.

Abstract

Focusing on OPC technology, this article objectives clarify the main concepts from this communication standard, giving details about its main uses and benefits, finally showing that actually OPC can be much more than a simple connection tool between softwares and equipments.

Resumo

Tendo como foco principal a tecnologia OPC, este trabalho tem por objetivo esclarecer os principais conceitos deste padrão de comunicação, detalhando seus principais usos e benefícios, mostrando que atualmente o OPC pode ser muito mais do que um simples meio de interligação entre softwares e equipamentos. Também são abordadas algumas de suas limitações e as soluções mais utilizadas para resolver tais problemas.

Palavras chave: OPC, comunicação, driver, rede, ethernet, COM, DCOM, supervisorio, equipamento, controlador, padrão, DA, HDA, AE, UA.

1 - INTRODUÇÃO

Tipicamente existem inúmeros sistemas nas áreas de produção, incluindo sistemas SCADA/IHM's, SDCD's, LIMS (Laboratory Information Management Systems), MES (Manufacturing Execution System) além dos sistemas de manutenção e historiadores. Muitos destes sistemas tendem a ser aplicações isoladas umas das outras, ou em alguns casos - como os sistemas historiadores - possuem algum tipo de interface, mas no geral existe uma pequena integração entre eles. Décadas de protocolos de comunicação proprietários resultaram em sistemas desconexos dentro do próprio chão de fábrica. Ao expandir um sistema, tradicionalmente o usuário toma o caminho mais fácil e acaba agregando ferramentas do mesmo fornecedor inicial, mesmo que outro fornecedor possua soluções mais interessantes para sua necessidade. Somente assim se tem a garantia de total compatibilidade e possibilidade de troca de dados entre o sistema antigo e o novo.



Figura 1: A falta de conectividade dos sistemas tradicionais

Já no ambiente corporativo existem vários outros softwares que preenchem necessidades específicas como planejamento de produção, gerenciamento de custos, contabilidade, entre outros. Estes softwares podem ser fornecidos por um único fabricante ou então adquiridos separadamente. No entanto, de um modo geral, estes softwares tendem a possuir uma maior integração do que a existente nas áreas de produção.

Devido a aspectos culturais e de negócios da organização, as pessoas e sistemas envolvidos nestas áreas – corporativa e fabril - freqüentemente têm pouca interação, criando uma lacuna de comunicação. Como resultado desta lacuna as informações se tornam indisponíveis através da organização. Porém, em um ambiente competitivo como o encontrado atualmente, as informações tornaram-se essenciais, uma espécie de vantagem estratégica: sai na frente a organização que melhor conhece a si mesma e a seus clientes.

As tecnologias do padrão OPC (OLE for Process Control) permitem a integração dos dados de toda a empresa, sejam provenientes do chão de fábrica ou dos setores corporativos. Sendo um padrão aberto, o OPC separa os sistemas das dificuldades de comunicação, criando uma camada única e padronizada que permite a fácil integração de diversos sistemas, desde um simples instrumento de campo até os sistemas de ERP (Enterprise Resource Planning) e de Gestão Corporativa.



Figura 2: A camada de comunicação padronizada OPC

No item 2 (dois) deste trabalho é apresentado o conceito de OPC, explicando também sua evolução desde seu surgimento até os dias atuais. Já o item 3 (três) aborda algumas dificuldades quando se utiliza o OPC e apresenta soluções simples para se chegar à solução ideal, bem como a utilização do OPC para integração do chão de fábrica aos sistemas corporativos da empresa.

2 – O PADRÃO OPC

2.1 – Introdução ao OPC

O OPC é um padrão de comunicação aberto, que tem por principal objetivo permitir a interoperabilidade vertical entre sistemas dentro de uma organização. A primeira versão funcional do OPC foi desenvolvida por volta de 1996, resultado do trabalho conjunto entre fornecedores de sistemas para automação industrial. Deste esforço conjunto surgiu a *OPC Foundation*, organização que define os padrões do OPC e que busca constantemente sua melhoria e evolução. Desde seu

surgimento há mais de 10 anos, novas especificações são elaboradas com o objetivo de agregar mais funcionalidades ao padrão OPC.

A sigla OPC significa *OLE for Process Control* ou OLE para Controle de Processos. Baseado nas tecnologias Microsoft OLE COM (Component Object Model) e DCOM (Distributed Component Object Model), o OPC é um conjunto comum de interfaces, métodos e propriedades de comunicação, agregados dentro de uma especificação padronizada e aberta para acesso público. Teoricamente qualquer pessoa com conhecimentos de programação pode desenvolver seus aplicativos OPC, basta acessar as especificações contidas no web site da OPC Foundation e desenvolver uma interface compatível.

Para ficar mais claro o que é o OPC podemos fazer uma analogia com um *driver* comum de impressora. Na época do MS-DOS, o desenvolvedor de um software como um editor de textos precisava desenvolver um *driver* de comunicação para cada uma das impressoras existentes no mercado: um para a Epson-FX, um para a HP LaserJet, entre outras impressoras. No mercado de automação industrial isso fica evidente até hoje. Se tomarmos como exemplo um software de supervisão, veremos que ele tem seus próprios *drivers* para cada um dos CLP's (Controladores Lógico Programáveis) existentes no mercado. No entanto estes *drivers* não podem ser utilizados em outro software de supervisão: este outro deve possuir seus próprios *drivers*, para os mesmos CLP's.

O Windows resolveu o problema dos *drivers* de impressoras ao incluir o suporte para impressão no próprio sistema operacional. A partir de então a impressora deveria possuir somente um *driver* para Windows, e não mais para cada um dos aplicativos utilizados. Este fato possibilitou uma redução de custos considerável para as empresas de software, que agora podiam manter o foco realmente no que interessa, ou seja, em sua solução.

Ao basear o OPC na tecnologia OLE, nativa do Windows, este mesmo benefício chegou à área industrial. Com o surgimento do OPC os desenvolvedores de sistemas de automação podem escrever servidores OPC para seus equipamentos, e os demais softwares (como os supervisórios) passam a ser clientes OPC. Desaparece a necessidade de se desenvolver inúmeros *drivers* de comunicação.

Enquanto o OPC permitiu aos fornecedores de automação reduzir seus custos de conectividade e assim manter o foco nas funcionalidades de sua solução, para os clientes o benefício foi a flexibilidade. Agora o usuário poderia escolher seus softwares com base nas funcionalidades, e não mais ficar dependente da dúvida "Este supervisório tem driver para meu equipamento?".

2.2 – A arquitetura cliente-servidor do OPC

O funcionamento do OPC é baseado na tradicional arquitetura cliente-servidor, conforme a figura abaixo:



Figura 3: Arquitetura cliente-servidor do OPC

O funcionamento desta solução é simples: um ou mais servidores fornecem dados para uma ou mais aplicações cliente.

Para melhor entender esta arquitetura tome-se como exemplo um cliente em um bar. O cliente solicita uma bebida ao garçom, que atende a esta solicitação e lhe serve uma bebida. O OPC funciona de maneira semelhante: uma aplicação cliente (como um software de supervisão) solicita um dado ao servidor OPC, que lhe atende e retorna com o dado solicitado.

O interessante do OPC é que uma aplicação cliente pode solicitar dados a um ou mais servidores OPC, e o inverso também é verdadeiro, um servidor OPC pode transferir dados a um ou mais clientes OPC. Portanto fica claro que o OPC possibilita uma variedade enorme de comunicações, basta que os aplicativos sejam compatíveis com OPC.

É importante ressaltar que o OPC não elimina o protocolo proprietário nativo do CLP ou equipamento de campo. O que acontece é que o servidor OPC “traduz” este protocolo proprietário para o padrão OPC. Portanto é necessário o desenvolvimento de um servidor OPC específico para cada um dos diferentes protocolos de comunicação existentes.

Algumas empresas como a canadense *Matrikon OPC* se especializaram neste desenvolvimento, escrevendo servidores OPC há muitos anos e possuindo em seu portfólio soluções para a grande maioria dos sistemas de automação conhecidos.

2.3 – A evolução do OPC

Inicialmente o OPC foi desenvolvido com o objetivo de solucionar o problema dos *drivers* de comunicação proprietários, trazendo um padrão para onde antes só existiam soluções customizadas. A primeira especificação OPC posteriormente veio a se chamar *Data Access* e, como o nome sugere, permitia somente a troca de dados em tempo real. O OPC Data Access é largamente utilizado em todo o mundo, é o mais comum de ser encontrado.

O OPC Data Access permite a resposta a perguntas como: “Qual o valor da variável X agora?”, “Qual o valor do setpoint de meu controlador agora?”

Porém logo surgiram outras necessidades, como o acesso a dados históricos (Qual foi o valor da variável x ontem?) e o acesso a alarmes e eventos gerados no sistema. Novas especificações surgiram para suprir estas e outras demandas

2.3.1 – OPC HDA – Historical Data Access

A especificação *OPC Historical Data Access*, ou OPC HDA, trata do acesso a dados históricos através do OPC. Com ele é possível acessar dados armazenados na memória de um equipamento como o CLP, ou então transferir dados de um software historiador para um software de supervisão ou até mesmo para uma planilha em Excel. O OPC HDA permite a utilização do padrão OPC em aplicações onde é necessária uma análise dos dados, com o objetivo de melhorar o processo. A vantagem de se utilizar o OPC em soluções deste tipo é que se torna fácil coletar dados de diversos equipamentos e sistemas diferentes, e consolidá-los em uma aplicação cliente única.



Figura 4: Exemplos de utilização do OPC HDA

2.3.2 – OPC AE – Alarms & Events

Já o *OPC Alarms & Events* ou simplesmente OPC AE permite a transmissão de alarmes e eventos, incluindo parâmetros como severidade do alarme, classificação, área de ocorrência, intervenções do operador, comentários, entre outros. Através do OPC AE é possível, por exemplo, alimentar uma base de dados centralizada contendo informações de alarmes e eventos de diversos sistemas

espalhado pelo chão de fábrica. Desta maneira o usuário pode entender melhor o comportamento anormal da planta e como certas anomalias se originam.

Outra utilização do OPC AE é a geração de um banco de dados com informações sobre a operação e principais eventos dos processos, assim atendendo normas exigidas em diversos setores como alimentos, bebidas, farmacêutico e energia elétrica.

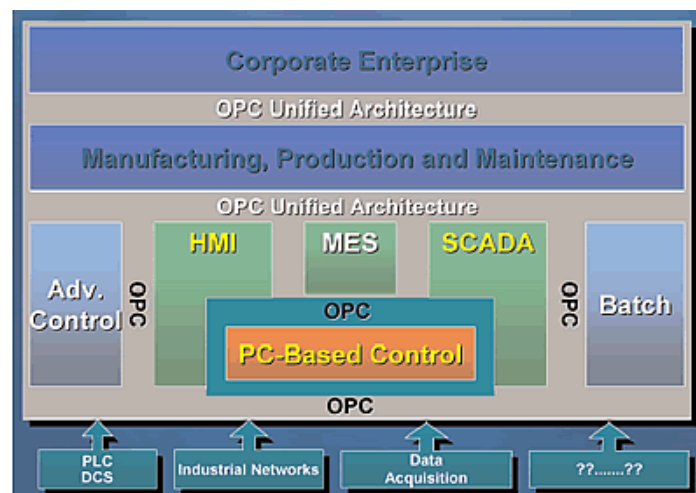
2.3.3 – OPC UA – Unified Architecture

A maior evolução do OPC está surgindo agora, com a especificação *OPC Unified Architecture* ou OPC UA. Este novo modelo OPC começou a ser desenvolvido em 2004 e as primeiras partes da especificação começaram a ser disponibilizadas em 2006. Esta nova especificação quer levar o OPC definitivamente além do ambiente industrial, tornando-se o padrão para sistemas tão distintos quanto um software de contabilidade e um controlador de campo.

Confirmando seu nome, o OPC UA irá unificar todas as especificações OPC, principalmente as OPC DA, HDA e AE. Ou seja, caso um equipamento ou sistema seja compatível com OPC UA, será possível a troca de quaisquer tipos de dados, sejam eles dados em tempo real, históricos ou alarmes.

O OPC UA também vem de encontro à constante evolução da informática e da Internet e além de ser multi-plataforma, também é baseado em tecnologias orientadas para a web como TCP/IP, HTTP, SOAP e XML, deixando o COM em segundo plano – assim como a própria Microsoft já o fez. Isto significa que será possível utilizar o OPC não só em computadores com Windows, mas também em soluções baseadas em outros sistemas operacionais como o Linux. Ou seja, o OPC poderá estar presente tanto em equipamentos compactos como computadores de mão, quanto em sistemas de grande porte como mainframes.

A especificação OPC UA está sendo gerada por partes, sendo que muitas delas já têm sua primeira versão finalizada. Dentro de poucos meses poderemos ver uma especificação OPC UA completa, levando o OPC a um novo patamar. O objetivo do OPC UA é ser o padrão dominante no mínimo pelos próximos dez anos.



OPC UA: Integração vertical de todos os sistemas da empresa

3 – OTIMIZAÇÃO DO OPC

O OPC vem passando por constante avanço ao longo dos anos, preenchendo cada vez mais espaço nas indústrias. No entanto algumas de suas características principais, que facilitaram seu surgimento e evolução, ironicamente hoje também são fatores limitantes ao seu pleno funcionamento. Alguns desenvolvedores acabaram por criar soluções que complementam o OPC, possibilitando ultrapassar algumas barreiras inerentes à sua tecnologia.

3.1 – DCOM, OPC e Tunneling

Como vimos anteriormente, o OPC é baseado nas tecnologias Microsoft OLE COM (Component Object Model) e DCOM (Distributed COM). Quando o servidor e o cliente OPC estão instalados no mesmo computador, o OPC utiliza o COM para estabelecer a comunicação entre ambos. Geralmente não há problemas nesta configuração, o COM é de fácil configuração, envia e recebe dados a altas velocidades e raramente apresenta problemas.

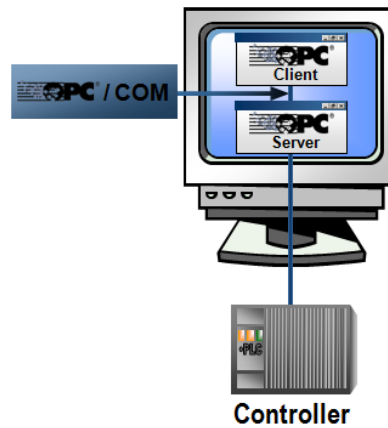


Figura 6: OPC e COM

Porém quando o servidor e cliente OPC estão instalados em computadores diferentes dentro de uma rede, o OPC passa a utilizar o DCOM. Surgido em 1996, com foco no ambiente de TI, o DCOM é uma extensão do COM, com foco na comunicação entre objetos em sistemas distribuídos. O DCOM atendia bem aos requisitos daquela época, basicamente o funcionamento dentro de LANs (Local Area Networks).

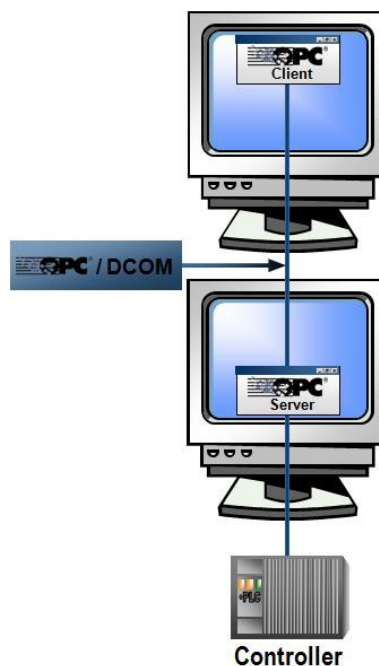


Figura 7: Funcionamento do OPC com DCOM

Com o surgimento e popularização da Internet na década de 90, as demais tecnologias acabaram por evoluir rapidamente a fim de atender uma infinidade de novas demandas. A própria Microsoft acabou desenvolvendo uma nova plataforma de desenvolvimento, o .Net, em resposta ao Java da Sun Microsystems e também como forma de combater a onda de vírus e invasões remotas que acompanharam a evolução da Internet. O DCOM deixava a desejar frente às novas demandas da tecnologia, e ainda por cima não estava acessível aos programadores para que os mesmos pudessem melhorar superar tais deficiências.

Dentre as limitações do DCOM podemos mencionar:

- Dificuldade de se trabalhar em WANs (Wide Area Network): redes com diferentes usuários, senhas e domínios são um problema para o DCOM, que exige configurações detalhadas a fim de funcionar corretamente.
- *Timeout* demasiadamente longo: em caso de simples oscilações na rede, o DCOM pode levar vários minutos até reestabelecer a conexão.
- Dificuldade de se trabalhar com *firewalls*: O DCOM inicia a comunicação através de uma porta TCP/IP e em caso de encontrar algum impedimento, utiliza outras portas aleatoriamente até conseguir estabelecer a conexão. Em uma época em que o próprio Windows já vem com *firewall* instalado para aumentar a segurança, a utilização do DCOM acaba obrigando o usuário a manter diversas portas TCP/IP abertas, ou até mesmo a desativar o *firewall* por completo, prejudicando criticamente a segurança do sistema.

Tais limitações são um grande empecilho em aplicações industriais onde, por exemplo, alguns sistemas obrigatoriamente devem ser instalados em domínios diferentes, a velocidade de comunicação é fator essencial e a segurança contra *hackers* e invasões aumenta a cada dia.

Algumas vezes os problemas do DCOM podem ser solucionados com muitas horas de trabalho e políticas corporativas que respeitem o ambiente industrial. Outra abordagem para eliminar estes problemas por inteiro é utilizar a tecnologia de *Tunneling*. Neste caso, um software chamado OPC Tunneller é colocado em cada uma das pontas da comunicação OPC, por exemplo, um no computador do servidor e outro na estação cliente OPC. Os dois objetos *Tunneller* se comunicam com seus OPC locais através do COM, confiável e rápido. Após isto os dois OPC Tunneller estão livres para trocar dados entre si através de outras tecnologias mais apropriadas para as necessidades da aplicação, como o TCP/IP, HTTP, HTTPS, XML, etc. Problemas com usuários, senhas e domínios diferentes são automaticamente anulados. O desempenho da rede já não tem tanta interferência sobre o OPC, uma vez que o timeout é configurável no OPC Tunneller. E o problema mais comum encontrado atualmente que é o de se trabalhar com OPC e *firewalls* ao mesmo tempo passa a ser resolvido, uma vez que o OPC Tunneller utiliza sempre a mesma porta de comunicação TCP/IP. Esta porta é configurável pelo usuário, portanto basta escolher a mais adequada e liberá-la no *firewall*.

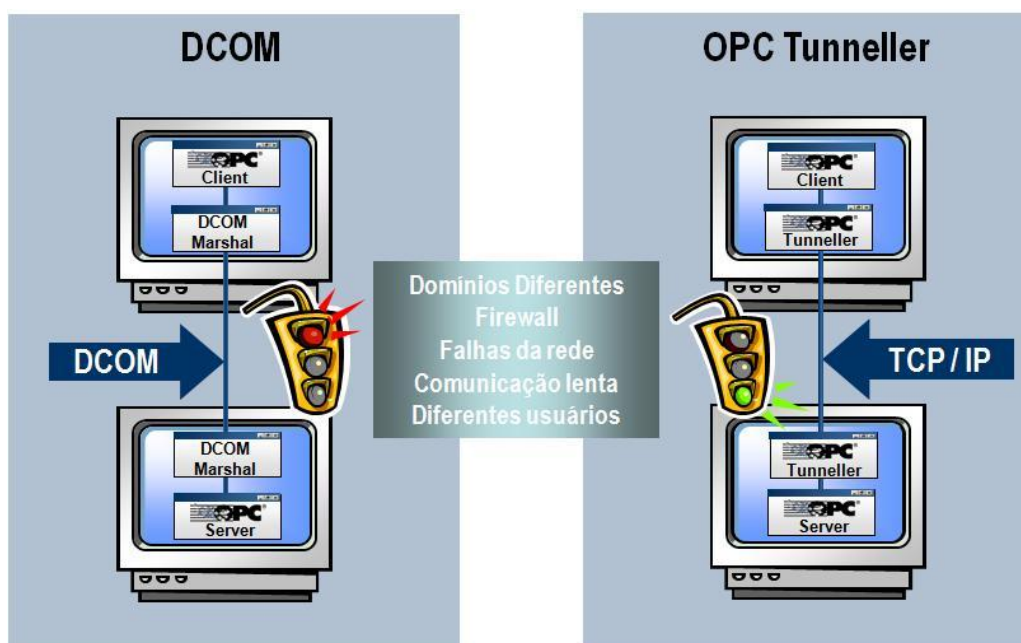


Figura 8: Substituindo o DCOM pelo OPC Tunneller

Em resumo o OPC Tunneller tem duas funções básicas: transferir dados da maneira mais fácil, segura e confiável possível para o outro componente OPC Tunneller, e traduzir todos os dados para o padrão OPC novamente, tornando a comunicação mais consistente. Todas as dores de cabeça geralmente associadas ao DCOM são aliviadas. Basta especificar um endereço IP e uma porta de comunicação e assim estabelecer a comunicação em rede utilizando OPC.

3.2 – Redundância e OPC

Em muitos processos a segurança e confiabilidade são fundamentais. Os motivos para tanto podem ser diversos, tais como:

- Prevenção contra acidentes e fatalidades em sistemas críticos. Exemplos: indústrias químicas e petroquímicas
- Minimizar paradas de produção e/ou quebras de equipamentos caros e que demandam muito tempo para reparo. Exemplos: indústrias siderúrgicas e de mineração
- Impossibilidade de se interromper a prestação de serviços sob a penalidade de sofrer multas e prejuízos na imagem. Exemplos: empresas de geração, transmissão e distribuição de energia elétrica
- Necessidade de se manter um histórico de dados confiável para facilitar a reastreabilidade e identificação de falhas. Exemplos: indústria automobilística.

Uma solução bastante utilizada na indústria para o aumento da confiabilidade e segurança é a redundância, que basicamente consiste na utilização de dois ou mais sistemas iguais. Desta maneira, caso um dos sistemas apresente problemas, o outro estará pronto para entrar em operação e assumir suas funções. A redundância pode ser utilizada em diversos níveis dentro de um sistema de automação industrial, ou seja, pode-se encontrar CLP's com CPU's redundantes, softwares de supervisão em redundância, redes redundantes, etc.

No entanto, apesar da grande importância da redundância para aumento da segurança e confiabilidade dos sistemas, a especificação OPC não aborda este tema de maneira satisfatória, abrindo uma grande lacuna na utilização do OPC. Para um sistema ser redundante não basta instalar dois softwares e/ou equipamentos iguais, é necessário um gerenciamento entre os mesmos com o objetivo de não aumentar o fluxo de dados na rede e ao mesmo tempo manter uma base de dados completa e confiável, sem duplicações que podem levar à dúvida.

Muitas vezes tenta-se instalar dois servidores OPC em paralelo como objetivo de aumentar a segurança. No entanto o resultado é que ambos passam a funcionar em paralelo, aumentando o fluxo de dados na rede e deixando dúvidas sobre qual deles tem o dado de melhor qualidade.

Soluções de terceiros foram desenvolvidas para preencher a lacuna da redundância no OPC. Também conhecido com *OPC Redundancy Broker*, este gerenciador de redundância sempre escolhe somente um OPC Server para realizar a comunicação, mantendo o outro em estado de espera. O OPC Server *backup* somente entra em cena em caso de falha no OPC Server primário, assumindo suas funções e garantindo a continuidade e confiabilidade da comunicação OPC.

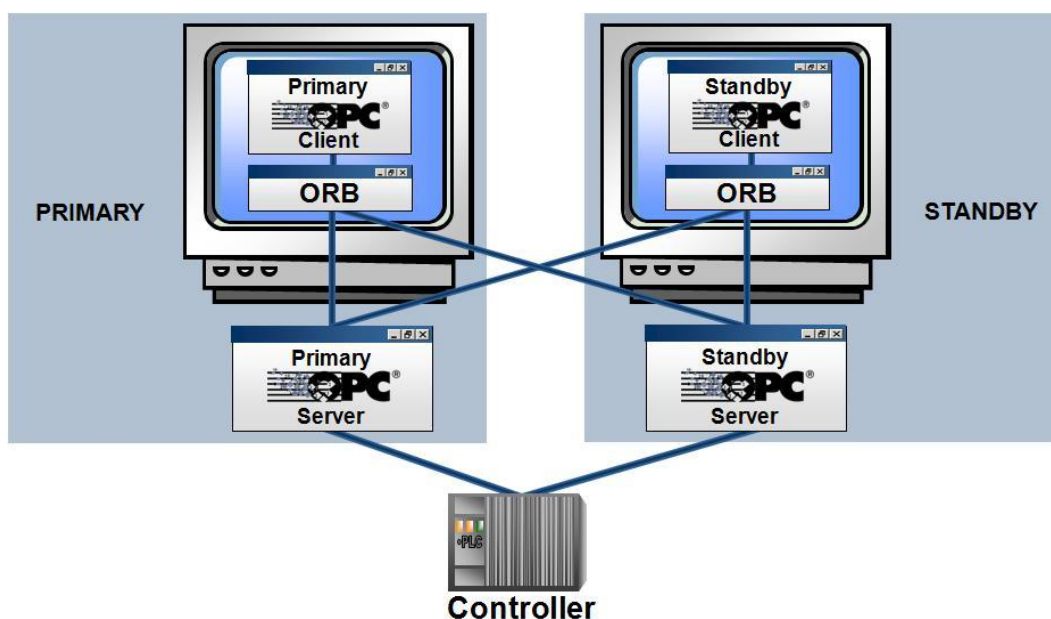


Figura 9: Redundância em OPC através do OPC Redundancy Broker

3.3 – OPC e Sistemas Corporativos

Inicialmente o OPC foi desenvolvido para o ambiente de chão de fábrica. Porém a evolução trouxe novas necessidades, como a troca de informações com sistemas de gestão. Através disso é possível, por exemplo, que uma ordem de produção gerada em um sistema de planejamento de produção seja levada diretamente aos sistemas de automação, permitindo uma agilidade muito maior nas operações. Conforme o andamento da produção, os sistemas de automação devolvem os dados de processo aos softwares de gestão, dando aos gestores informações em tempo real. Assim uma equipe de vendas pode saber, em tempo real, se pode aceitar determinado pedido ou não, dependendo do andamento da produção. A contabilidade pode calcular os custos de produção em tempo real. O departamento de compras pode saber qual fornecedor tem a melhor relação custo/benefício, em função do rendimento da matéria-prima no processo.

Apesar de ainda não abordado em detalhes na especificação OPC DA – o que deve ocorrer na nova especificação OPC UA, existem soluções customizadas para a integração do OPC com sistemas corporativos.

Um exemplo é o *OPC Generic Data Access* da empresa Matrikon OPC, que permite a troca de dados bidirecional com bancos de dados relacionais. Um sistema de automação de chão de fábrica pode enviar dados diretamente para um ERP baseado em um banco de dados como Oracle ou SQL Server, e este sistema pode devolver dados aos sistemas de produção.

Já o *Nlink OPC to SAP* da Junot Systems é uma interface direta entre o OPC e o ERP mais famoso mundialmente, o R/3 da SAP. É uma solução pronta e que minimiza a criação de códigos em linguagens de programação.

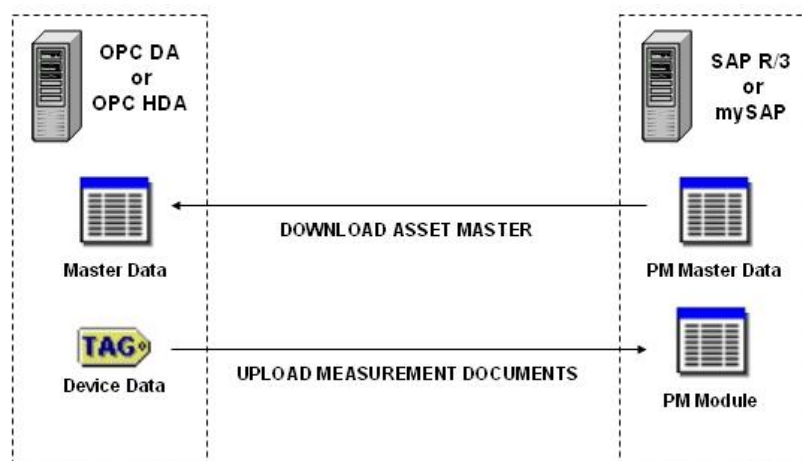


Figura 10: Integração do OPC com SAP R/3

4 – CONCLUSÃO

O OPC surgiu como uma simples resposta aos drivers de comunicação proprietários e em pouco mais de 10 anos acabou por se tornar um padrão altamente difundido na indústria. Com o OPC é possível criar uma camada de comunicação padronizada, que integra facilmente todas as informações industriais. Com o avanço da especificação OPC UA o OPC quer estar presente em sistemas muito além do chão de fábrica, levando a comunicação de dados a um nível de integração sem precedentes.

Agradecimentos

Agradecemos às empresas e organizações que têm auxiliado na difusão e melhoria das soluções OPC, principalmente a Matrikon OPC Inc. e a OPC Foundation.

Referências Bibliográficas

www.matrikonopc.com

www.opcfoundation.com

www.kepware.com

www.opc-to-sap.com

www.pt.wikipedia.org

www.softbrasil.com.br

Dados do Autor

Adriano Pedroso Puda

Gerente de Produtos

SoftBrasi Automação Ltda.

Av. Dr. Cardoso de Melo nº 1.855, cj. 32 bloco 2

CEP 04548-903 – São Paulo – SP

Telefone (11) 3488-8787

e-mail: Adriano@softbrasil.com.br