



Sistemas ERP de Código Aberto: uma opção para as pequenas indústrias?

Ronaldo Ribeiro de Campos (Escola de Engenharia de São Carlos – USP – Depto de Engenharia de Produção) - ronaldodecampos@yahoo.com.br

Edson Walmir Cazarini (Escola de Engenharia de São Carlos – USP – Depto de Engenharia de Produção) - cazarini@sc.usp.br

Este trabalho propõe uma investigação descritiva nos sistemas ERP de código aberto com objetivo de se obter um levantamento de informações que possam auxiliar as pequenas indústrias na fase de seleção de um sistema ERP. São apresentados os conceitos de software livre e software de código aberto. É descrita uma evolução histórica dos conceitos ERP, privilegiando a visão da área de produção sobre esse tipo de sistema, e a busca das pequenas empresas em obter vantagens competitivas com a utilização da tecnologia de informação. Também são apresentados números que apontam para a quantidade crescente de projetos de software de código aberto. A seguir apresenta-se uma sugestão de três aspectos a serem pesquisados para auxiliar na escolha dos softwares: viabilidade de instalação, funcionalidades presentes e viabilidade de uso. A proposta é um estudo dentro de um referencial teórico qualitativo-positivista, com objetivo metodológico descritivo, utilizando-se como delineamento o levantamento, através da utilização de questionários e realização de testes de instalação e utilização. O resultado esperado é reunir informações que possam servir a pesquisadores da área de engenharia de produção, desenvolvedores de software e pequenos empresários interessados em sistemas ERP de código aberto.

Palavras-chave: ERP, pequenas empresas, software de código aberto, tecnologia de informação.

1. Introdução

Os softwares desenvolvidos sob o modelo de código aberto, não são mais exclusivamente dedicados a formar um conjunto de softwares que compõem o sistema operacional ou o ambiente gráfico. Esse limite foi ultrapassado e existe um número crescente de projetos de software dedicados a várias outras áreas: sistemas gerenciadores de banco de dados, servidores de página web, linguagens de programação de alto nível, gerenciamento de projetos e gestão empresarial, entre outros.

Os programas de código aberto apresentam vantagens em relação ao custo, menor tempo para correção de problemas e menor tempo para implementação de melhorias, quando comparados com os softwares proprietários.

As vantagens dos programas de código aberto, principalmente o custo, atraem a atenção das pequenas empresas que possuem recursos tecnológicos, e buscam uma forma de utiliza-lo para obter vantagens competitivas. Uma das aplicações que são apontadas como auxiliares na obtenção de vantagem competitiva é o ERP (Enterprise Resource Planning), e existem projetos de desenvolvimento deste tipo de programa sob os moldes do código aberto.

A dificuldade é identificar dentre o crescente número de projetos de código aberto, quais são os ERP que se destinam à utilização nas indústrias, quais são os requisitos necessários à sua instalação e quais são as funcionalidades presentes nesses sistemas.

Este trabalho considera a situação das pequenas empresas na busca de alcançar vantagens competitivas com a utilização da tecnologia, cogitando a utilização de sistema ERP de softwares de código aberto, e apresenta uma proposta de investigação descritiva de tais softwares com o objetivo de reunir informações que possam auxiliar pesquisadores, desenvolvedores de software e pequenos empresários a ter uma visão dos requisitos de instalação e funcionalidades presentes em tais softwares.

2. Softwares de código aberto

A idéia do desenvolvimento de software livre surgiu na década de 80, e seu objetivo inicial era desenvolver um sistema operacional que fosse totalmente portátil e livre. Livre sob aspecto do programa poder ser distribuído com seu código fonte, para que pudesse ser adaptado às características de cada organização em particular. Livre sob um conceito filosófico, mas que não significa necessariamente que por isso não seja necessário pagar para obtê-lo.

Com o passar do tempo, a idéia se difundiu e teve em particular uma derivação sob o aspecto de distribuição do código fonte do software, denominada de Código aberto (Open Source).

Embora os termos “Software Livre” e “Código Aberto” muitas vezes sejam utilizados como sinônimos, conforme retrata a Fundação do Software Livre na Europa – FSFE (2005), eles têm significados diferentes. O primeiro, de forma mais abrangente, preocupa-se além do código fonte do programa, com questões filosóficas que envolvem as conseqüências econômicas, políticas e sociais relacionadas com:

- Liberdade de executar o programa, independentemente de qual seja o propósito;
- Liberdade de poder estudar o programa e modifica-lo, de acordo com as necessidades próprias;
- Liberdade de redistribuir cópias do programa;
- Liberdade de aperfeiçoar o programa, e distribuir os aperfeiçoamentos, beneficiando toda a comunidade.

O segundo preocupa-se com as maneiras de garantir que o código fonte será disponível aquele que queira utiliza-lo, bem como com a realização de modificações nesse código, e a forma que o programa será redistribuído. Para a Free Software Foundation (2004), o fato do código fonte ser distribuído junto com a aplicação é uma condição necessária para a existência do software livre, mas isso não significa que todos objetivos com relação à liberdade do usuário serão conquistados.

Os dois “movimentos” possuem órgãos que coordenam projetos de incentivo e regulamentação de suas definições e objetivos. A Fundação do Software Livre (FSF - Free Software Foundation) coordena o Software Livre, enquanto a Open Source Initiative (OSI), é a responsável pelo Código Aberto.

Os principais sites que funcionam como repositório de projetos destes tipos de sistema, adotam como forma de classificação de licenças a OSI (SOURCEFORGE, 2005) e (FRESHMEAT, 2005). Isso talvez se deva ao fato de que a OSI fornece uma espécie de certificação para as licenças por ela aprovadas, e também porque muitas das

licenças aceitas como sendo de software livre, são compatíveis com as licenças da OSI, conforme descrito pela própria OSI (2004).

As características que uma licença de software deve ter para ser considerada como “código aberto” segundo a OSI (2004), são as seguintes:

1 – Distribuição livre: A licença de distribuição não deve restringir a nenhuma das partes interessadas de vender ou ceder o software, bem como não deve cobrar direitos de propriedade ou outras taxas pela venda do programa.

2 – Código Fonte: o programa precisa incluir o código fonte, assim como deve possibilitar a distribuição do código fonte, e da forma compilada. Quando por algum motivo, o programa não tiver o código fonte distribuído junto com a aplicação, isso deve estar bem especificado, mas deve existir uma forma de se poder obter uma cópia dos fontes via internet, sem custo algum para o usuário.

3 – Trabalhos derivados: As licenças precisam permitir modificações e trabalhos derivados do programa original, disponibilizando as licenças sob os mesmos termos da licença do software no qual as modificações ou trabalhos derivados se basearam.

4 – Integridade do código fonte do autor: Somente é possível restringir a distribuição do código fonte em forma modificada se ela permitir a distribuição de atualizações junto com o código fonte original com o objetivo de modificar o programa durante a instalação.

5 – Não discriminar grupos ou pessoas: A licença não deve fazer discriminação a nenhuma pessoa, ou grupo de pessoas.

6 – Não discriminar campos de interesse: a licença não deve restringir o uso do programa a um campo de interesse específico, seja ele comercial ou não.

7 – Distribuição da licença: os direitos vinculados ao programa aplicam-se a todos aqueles aos quais o programa é distribuído, sem que seja necessária uma licença adicional para cada um deles.

8 - Licença não deve ser específica para um produto: Os direitos vinculados a programas não devem depender dos programas que fazem parte de uma distribuição em particular, ou seja, se os programas foram extraídos a partir de uma distribuição e usados ou distribuídos com os termos da licença do programa, todos aqueles a quem os programas foram redistribuídos devem ter os mesmos direitos concedidos aos que obtiveram o programa originário.

9 – A licença não deve restringir outro software: A licença não deve impor restrições a outro software que é distribuído juntamente com o software licenciado.

10 – Licença deve ser neutra em relação à tecnologia: Estabelece que a licença não deve estar vinculada a uma tecnologia, estilo ou interface.

O software de código aberto tem seus problemas. Nichols & Twidale (2002) aponta como um dos maiores problemas a usabilidade, pelo fato de serem elaborados normalmente sem a presença do usuário final, que passa a atuar como participante depois que o software é posto à prova. Ainda de acordo com Nichols & Twidale, a dificuldade dos desenvolvedores de software livre é geralmente não possuírem recursos para levantar as necessidades do usuário final.

As características do software de código também apresentam algumas vantagens:

- **Custo:** o usuário pode utilizar o software sem a necessidade de pagamento de direitos autorais ou licenças.
- **Tempo de correção de problemas e melhorias do sistema:** os projetos desse tipo de software são desenvolvidos por grupos de pessoas dispostas a resolver os problemas ou implementar soluções indicadas

pelos usuários. Assim, ao enviar uma mensagem via internet para o grupo responsável pelo desenvolvimento os problemas tendem a ser rapidamente resolvidos.

- **Suporte técnico:** os softwares de código aberto, normalmente possuem boa documentação, além da presença do grupo de desenvolvimento respondendo a dúvidas através da troca de mensagens via internet.

3. MRP, MPR II, MRP III e ERP

Os sistemas ERP são a evolução dos sistemas MRP II (Manufacturing Resource Planning), que por sua vez evoluíram a partir do MRP (Materials Requirements Planning). Os primeiros MRP datam do final da década de 60, mas é na década de 70 que sua utilização cresceu, conforme descreve Ritzman, Krajewshi, Moura (1989).

O MRP foi um sistema criado com o objetivo de controlar o estoque e dar apoio às funções de planejamento de produção e compras (COLANGELO FILHO, 2001). Ainda de acordo com Colangelo Filho (2001), seus procedimentos eram baseados em listas de materiais e um plano de produção, que de forma geral não davam suporte ao planejamento de capacidade e custos. Em função das possibilidades oferecidas pelo tipo de hardware existente na época, a implantação acontecia de forma departamental, e não havia integração do MRP com outras aplicações usadas pela organização.

Os objetivos do MRP são um pouco mais detalhados por Corrêa, Gianesi (1996) quando define o objetivo dos sistemas de cálculo das necessidades de materiais como sendo permitir o cumprimento dos prazos de entrega dos pedidos do cliente com mínima formação de estoque, fazendo o planejamento de compras e de produção para que ocorram nas quantidades e nos momentos corretos.

Dentro desses objetivos, Ritzman, Krajewshi, Moura (1989) aponta como as metas mais importantes à manutenção de um baixo nível de estoques, e as atividades de atendimento ao cliente, que estão diretamente ligadas ao nível de controle de estoques, uma vez que estas são as duas metas mais visíveis traçadas pela Administração de Materiais.

Segundo Ritzman, Krajewski, Moura (1989), nessa fase a diretoria tinha seu próprio sistema para desenvolvimento de projeção de vendas, inventários, produção e lucros. Este “plano comercial” não estava interligado aos planos de produção e freqüentemente os planejadores da produção nem mesmo sabiam da existência de um plano comercial. Além disso, a contabilidade possuía seu próprio sistema, assim como as áreas de marketing e distribuição. Ainda assim, o uso da técnica de cálculo das necessidades de materiais popularizou-se e alguns pesquisadores perceberam que a mesma lógica poderia ser aplicada, com algum esforço adicional, para o planejamento de outros recursos de produção, além dos materiais. Foi a partir da necessidade dessa integração que o MRP expandiu, transformando-se no MRP II.

O raciocínio era que se as quantidades de recursos utilizados para a execução de todas as atividades de produção fossem conhecidas, seria possível calcular a quantidade total de recursos necessários ao atendimento dos pedidos e distribuir essas necessidades ao longo do tempo (CORRÊA, GIANESI, 1996).

Surgia então, na década de 80, o MRP II englobando aspectos financeiros, orçamentos e custeio de produção, mas apesar desta expansão ainda não havia uma

integração completa com os outros sistemas da empresas e o suporte completo e integrado ao processo de negócios ainda era difícil (COLANGELO FILHO, 2001).

Para Ritzman, Krajewski, Moura (1989), o MRP II pode ser usado também como um simulador, de forma a ajudar encontrar respostas para as perguntas do tipo “e se...”. Este tipo de análise, conhecido como “*What-If*” (O que – Se), torna possível ao usuário fazer mudanças em uma ou mais (dependendo do recurso utilizado) variáveis de entrada, e observar as mudanças ocorridas nos resultados. Na observação de Coles, Rowley (1996), este tipo de análise pode auxiliar os tomadores de decisão a responder questões do tipo: o que aconteceria com o lucro, se as vendas sofressem um aumento de 5 por cento?

No caso do MRPII, esta funcionalidade seria utilizada para simular certos planos de ação e previsões de demanda, respondendo questões relacionadas a necessidades e faltas de mão-de-obra, equipamentos, inventário, prazos de entrega e acúmulo de pedidos.

Conforme descreve Corrêa, Giansi (1996), o MRP II, ou planejamento dos recursos de manufatura, tem preocupações mais abrangentes do que o MRP, pois o objetivo do cálculo de necessidades é permitir o cumprimento dos prazos de entrega, mínima formação de estoque e planejamento de compras, tudo nas quantidades necessárias e no tempo necessário.

A fase de desenvolvimento do MRP II coincide com o período de adoção dos microcomputadores. Como o custo de utilização desses equipamentos era atrativo, a instalação era freqüentemente feita de forma departamental, fazendo com que as aplicações ainda continuassem sendo utilizadas de forma isolada, a exemplo do MRP, dificultando o suporte completo e integrado do processo de negócios (COLANGELO FILHO, 2001).

O MRP II foi utilizado ainda em conjunto com o “Just in time”, recebendo nessa fase, a denominação de MRP III.

Na definição de Ritzman, Krajewski, Moura (1989: 181), o “Just in Time” (JIT), “é uma filosofia que se concentra na eliminação de desperdício no processo de manufatura”. A filosofia JIT busca eliminar perdas, exigindo que a administração faça todo o possível para ter as peças certas no lugar certo, no tempo certo, e exatamente na quantidade certa. Dessa forma, o MRP III seria a integração do MRP com JIT, baseando-se em uma linha de raciocínio onde o JIT é utilizado para planejar o trabalho, mas produzir somente o que é consumido, enquanto o MRP é utilizado para o planejamento dos materiais.

De acordo com Ritzman, Krajewski, Moura (1989), muitas empresas que implantaram o MRP negligenciaram outros elementos do sistema total, esquecendo-se de planejar e controlar o “lead-time” de seus produtos e a produção de suas fábricas. O MRP tem grande valor quando utilizado como um instrumento central de planejamento que assegura a disponibilidade de materiais explodidos, agrupados e compensados. O problema é que o MRP faz isso somente quando o processo de programação mestre é executado, especificando as quantidades como de um prazo de entrega, fornecendo informações sobre os componentes em grande número e em um único ponto no tempo.

Ritzman, Krajewski, Moura (1989) registra que o JIT resolveu o problema atingindo o planejamento de capacidade com um sistema de controle de entradas e saídas denominado Kanban, que é definido por Slack, Chambers, Johnston (2002: 494) como “*um método de operacionalizar o sistema de planejamento e controle puxado*”,

mas também avisa que o termo não deve ser usado como um sinônimo para JIT, nem como “controle JIT”.

Na década de 90, com a globalização criando um ambiente de negócios extremamente competitivo, o MRP sofreu nova ampliação, incorporando aspectos da área de Recursos Humanos e Finanças, prometendo maior agilidade e redução de custos. Com os novos recursos passou a ser chamado de ERP, ainda que alguns prefiram chamá-lo de ERM (Enterprise Resources Management), (Gerenciamento dos Recursos Empresariais) em função da grande amplitude que o escopo do sistema atingiu, abrangendo além do planejamento as atividades de execução e controle.

Acompanhando os progressos tecnológicos na área de hardware, o ERP é capaz de trabalhar em um ambiente cliente-servidor, com uma base de dados única e integrada. (COLANGELO FILHO, 2001).

A figura 1 demonstra a evolução dos sistemas ERP no tempo, e a abrangências das áreas de atuação durante sua evolução.

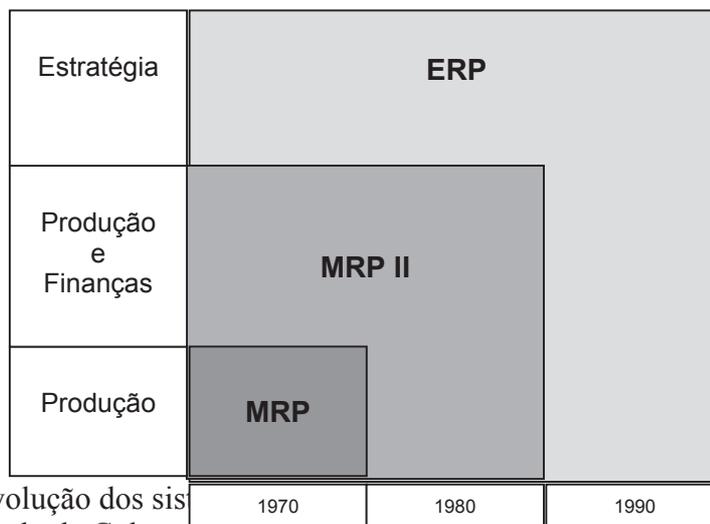


Figura 1 – Evolução dos sistemas ERP no tempo, e a abrangências das áreas de atuação durante sua evolução.
Fonte: Adaptado de Colangelo Filho, 2001: 21

Para Slack, Chambers, Johnston (2002: 474) o ERP é o “*último e provavelmente o mais significativo desenvolvimento da filosofia MRP básica*”. Apesar de ser fiel ao princípio herdado do MRP de explorar os reflexos que as mudanças em uma operação produzem em outras, o ERP possibilita que isso seja feito de uma forma muito mais ampla e em todas as partes da organização, refletindo as mudanças de uma parte da operação nos sistemas de planejamento e controle do restante da organização. Isso é possível porque o ERP trabalha com o conceito de ter uma base de dados que integra todas as partes da organização, possibilitando o trabalho no nível estratégico. Colangelo Filho (2001) explica que acompanhando os progressos tecnológicos das áreas de hardware e software, o ERP é capaz de trabalhar em um ambiente cliente-servidor, com uma base de dados única e integrada.

A figura 2 apresenta um esquema de funcionamento do ERP, baseando-se na utilização da base de dados integrada.

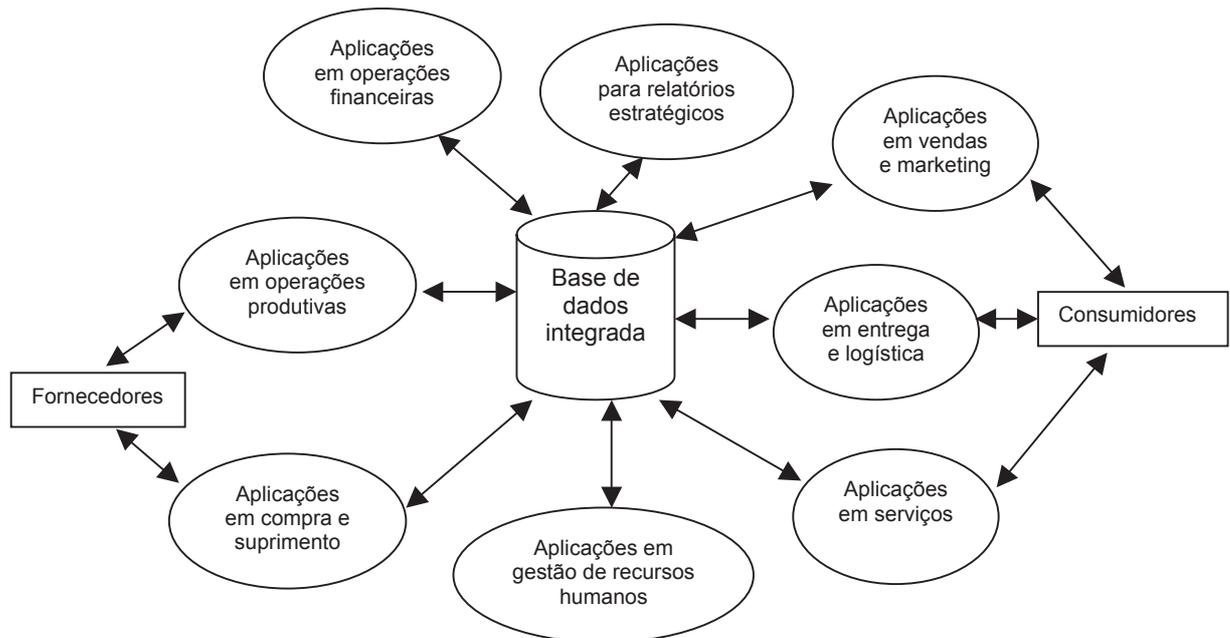


Figura 2 – Aplicações ERP acessam base de dados integrada

Fonte: Slack, Chambers, Johnston . 2002: 474

Mendes (2003) comenta que os sistemas ERP possuem grande abrangência funcional, procurando incorporar todas as atividades possíveis realizadas por uma organização, e fala da divisão do sistema em módulos. Um módulo corresponderia ao menor conjunto de funções que podem ser implementadas separadamente, e estão normalmente ligados a divisões departamentais da organização. Um ponto positivo dos módulos é que eles permitem que a empresa possa implementar somente os módulos de seu interesse, ou implementar o sistema em etapas. De forma genérica os módulos encontrados em sistemas disponíveis no mercado são os seguintes:

- Módulos relacionados a operações e gerenciamento da cadeia de suprimentos;
- Módulos relacionados à gestão financeira, contábil e fiscal;
- Módulos relacionados à gestão de recursos humanos;
- Módulos relacionados ao MRP II (planejamento de materiais, capacidade, gestão de demanda, compras e vendas)

4. A tecnologia de informação, pequena empresa e o ERP de código aberto

Apesar do fato de que Zimmerman, Scarborough (1994) apontarem para algumas desvantagens com relação à utilização da tecnologia de informação nas pequenas empresas, destacando: custo elevado em relação a baixa disponibilidade financeira das pequenas empresas; rápida obsolescência de equipamentos e dificuldade de alimentação do sistema com informações corretas, além da relutância dos funcionários por medo da perda de seus empregos e dos riscos para a saúde devido ao uso incorreto do computador por muitas horas, o fato é que o computador está presente no ambiente não só das pequenas, como também das micro empresas, pois de acordo com (SEBRAE-SP,

2003), 47 % das micro empresas possuem computador, índice este que se eleva para 61% quando se trata de indústrias, e chega a 81% quando se trata de empresas de pequeno porte.

O número de computadores se justifica pela tentativa de utilização dos recursos de tecnologia de informação como vantagem competitiva. De acordo com Zimmerer, Scarborough (1994), aquelas pequenas empresas que aprendem a explorar o uso da informação e tecnologia para atender melhor seus clientes, conseguem obter ganhos de competitividade, assim como as pequenas empresas que coletam e fazem uso da informação disponível para gerenciar suas operações de forma mais eficiente, também conseguem uma vantagem competitiva. A tecnologia está disponível para impulsionar a produtividade, controlar estoques e fluxo de caixa, identificar necessidades e preferências de clientes e antecipar problemas e oportunidades.

Conforme a observação de Beraldi, Escrivão Filho, Rodrigues (2000), existem organizações fornecedoras de software e hardware voltando-se para os problemas das pequenas empresas e oferecendo serviços personalizados a um menor custo. Dessa forma possibilitam às pequenas empresas acesso à recursos de tecnologia de informação.

É nesse ponto que os sistemas ERP passam a ser vistos como uma das soluções que podem auxiliar na busca das “vantagens competitivas”. Segundo PEQUENAS EMPRESAS GRANDES NEGÓCIOS (1998) apud Mendes (2003), os ERP economizam em média, 35% dos custos empresariais, através do planejamento de compras que atinge diretamente a aquisição de materiais, e ainda de acordo com o mesmo artigo, em se tratando de uma pequena empresa, essa economia pode ser ainda maior.

A adoção de sistemas ERP não é um processo simples, que atinge apenas uma área da organização, ele envolve questões de processos, tecnologia, pessoas e as conseqüentes mudanças organizacionais ocorridas (MENDES, 2003).

Para Corrêa (1998), o sucesso na adoção de um ERP se inicia na sua seleção, período em que se realiza uma análise de adequação de funcionalidades com o objetivo de verificar se as particularidades de cada organização são atendidas pelo sistema, e é justamente nesse ponto que se iniciam as dificuldades com relação à seleção de um sistema ERP de Código aberto.

Os softwares de código aberto não mais são exclusivamente dedicados a programa que visavam garantir a existência de um sistema operacional completo, ou voltados à área científica e acadêmica. Existe um crescente número de projetos de software dedicados à diversas áreas, incluindo gestão de empresas. Uma rápida pesquisa no mês de Janeiro/2005 nos principais sites, onde podem ser cadastrados os projetos de software, nos mostra o seguinte:

- O site “Source forge”, o maior repositório de projetos do mundo, (SOURCE FORGE, 2005), registrava a quantia de 58.484 projetos sob licença de Código aberto (OSI), dos quais 3.672 ligados à categoria Escritório/Negócios, e dentre esses, 1.169 pertencentes à subcategoria Finanças, que é onde se encontram nesse site, os ERP.
- O site “Freshmeat”, outra fonte de pesquisa sobre softwares livres, possuía 26.623 projetos sob licença OSI, sendo 1.861 cadastrados na categoria Escritório/Negócios, com 561 pertencentes à subcategoria Finanças.

O primeiro passo é a necessidade de identificar dentre os softwares classificados na subcategoria Finanças, quais aqueles que são descritos como sendo ERP, e depois identificar quais os ERP destinados à utilização em indústrias. Com o passar do tempo, a sigla ERP pela abrangência de seu significado acabou servindo de classificação para uma determinada categoria de software, assim é possível encontrar descrições de programas ditos “ERP para comércio”, ou “ERP para serviços”. Esse fato é comentado por COLANGELO FILHO (2001).

As diferenças entre um tipo e outro estão no objetivo de cada tipo de atividade que o ERP se dispõe a auxiliar, o que irá influenciar diretamente nos módulos que compõem o sistema. Um “ERP para indústria”, é o que obedece a evolução do MPR, e assim terá módulos destinados ao controle de matéria-prima, composição de produtos e cargas de máquinas, entre outros que abrangem funcionalidades de planejamento e controle de produção, módulos que não são necessários por exemplo a um “ERP de comércio”.

5. Metodologia

Diante do objetivo da proposta do trabalho que é a de apresentar um levantamento, análise e avaliação da viabilidade de instalação e funcionalidades presentes em softwares de código aberto classificados como ERP e que possam ser utilizados nas pequenas indústrias, de forma a reunir informações para auxiliar o processo de seleção do software, esse estudo foi classificado como tendo um referencial teórico qualitativo-positivista, que de acordo com Martins (1996), não aceita uma realidade diferente daquela que não seja a realidade dos fatos que possam ser observados.

Quanto ao objetivo metodológico, este foi definido como descritivo em concordância com a definição de Dane (1990) quando explica que o objetivo de uma pesquisa descritiva é capturar as características de um objeto, uma pessoa ou um evento em um determinado período no tempo através da coleta de dados.

Como delineamento, foi escolhido o “Levantamento”, que na opinião de Gil (1999) é adequado para estudos descritivos, sendo indicado por exemplo para estudos do tipo censo, onde a identificação de características é o que se deseja conhecer. Para tanto serão utilizados como instrumentos de coleta questionários e a realização de testes de instalação e utilização dos softwares selecionados.

Estabelecido o método, o próximo critério definido foi quanto à seleção de amostras. Descobrir se o projeto existente está em um nível de desenvolvimento maduro o suficiente para atender às necessidades básicas a que se propõem, uma vez que o fato de um projeto estar cadastrado não significa que ele está pronto para ser utilizado. O desenvolvimento do projeto ainda pode estar na fase de planejamento e nem sequer existir uma aplicação a ser testada. Para ser considerado um elemento válido para a amostra, é necessário que:

- software tenha um tipo de licença que seja aprovada pelo OSI, como sendo um software de código aberto;
- software deve conter os módulos dos ERP, que faça com o mesmo seja considerado “destinado à indústria, ou seja, deve conter módulos destinados à matéria-prima, composição de produtos, plano mestre de produção e requisição de materiais, estando descartados os ERP construídos com o objetivo de atender as atividades de serviço ou comércio.

Identificados os projetos em estágio de possível utilização, torna-se necessário verificar quais são os requisitos para que ele esteja pronto para ser utilizado. Com base no objetivo estabelecido para a pesquisa, bem como descrição de Haberkorn (1999) das rotinas genéricas que fazem parte da análise de softwares de gestão ERP, foram estabelecidos 3 grupos de variáveis de pesquisa: viabilidade de instalação, funcionalidades e viabilidade de uso. Cada uma delas será apresentada em detalhes a seguir:

Viabilidade de instalação: alguns softwares são desenvolvidos para utilização em ambiente web, o que implica na instalação de programas auxiliares, provavelmente um programa servidor de páginas web e um gerenciador de banco de dados. Torna-se importante aqui, determinar quais são os recursos de hardware mínimos para que o software seja instalado; verificar se existe a necessidade de instalação de software adicional e se existe uma ferramenta que auxilie a instalação do software.

Funcionalidades: uma vez que a aplicação esteja devidamente instalada é possível verificar quais são as funcionalidades que ela apresenta com relação à usabilidade, tipos de relatórios, gráficos e flexibilidade de configuração. Com relação a cada um desses itens, deseja-se abordar as seguintes questões:

Integração com banco de dados:

- software é capaz de obter dados de mais de um tipo de banco de dados ou sistema gerenciador de banco de dados? (Isso se torna particularmente importante, caso a empresa já possua dados cadastrados e precise aproveitá-los).
- É possível criar ou alterar campos e tabelas no banco de dados padrão do sistema para adaptá-lo às necessidades da empresa?
- Pode-se alterar as telas do software com relação aos campos de dados?
- software é capaz de gerenciar dados de empresas diferentes (por exemplo: matriz e filiais)?

Relatórios e gráficos

- Quais são os relatórios oferecidos pelo software, de acordo com sua instalação padrão?
- O software possui ferramentas que permitam criar novos relatórios ou alterar relatórios existentes de acordo com a necessidade da empresa?
- Quais são os formatos sob os quais os relatórios são apresentados? Somente na tela, apenas na impressora, é possível gravar os relatórios em arquivos? Em que formato de arquivo os relatórios podem ser gravados?
- É possível gerar gráficos com as informações? Que tipos de gráficos?

Recursos de integração com outros softwares

- software possibilita comunicação com outros softwares, por exemplo os softwares chamados de “pacotes de escritório” (do tipo Office (Word, Excell,))?
- Permite integração com software de e-mail ou algum tipo programa de envio de mensagens?

Segurança

- Quais são os recursos de segurança oferecidos pelo software?
- Existe senha para acesso ao software?

- Existem níveis de acesso diferentes, com acesso restrito a diferentes áreas?
- A que tipo de segurança as informações gerenciadas pelo software podem ser impostas (criptografia, segurança oferecida pelo gerenciador de banco de dados, segurança oferecidas pelo sistema operacional)?
- De que forma é estabelecido o sistema de backup dos dados? O próprio software gerencia o backup ou é necessário uma rotina externa ao programa?

Viabilidade de uso: Existe ainda a proposta de avaliar a viabilidade de utilização do software. A análise desta variável procura identificar o relacionamento homem-máquina (usuário – sistema) que o software pode proporcionar. Este aspecto do software faz parte de uma área chamada de usabilidade. A definição de usabilidade possui diferentes aspectos e abordagens, para Nielsen (1993), nesta definição devem ser incluídos ao menos os seguintes atributos: facilidade de aprendizado, eficiência de uso, retenção, mínimo de erros e satisfação.

Há diferentes formas de se avaliar a usabilidade, conforme registra Barcelos (2001). Uma das técnicas é a chamada “avaliação Eurística”, que segundo a descrição de Nielsen (1993), baseia-se na observação das seguintes características: interface tão simples quanto possível; caixas de diálogos com expressões às quais o usuário esteja habituado; fácil visibilidade de objetos, ações e opções; clareza quanto ao significado de palavras ou ações; informações constantes ao usuário sobre o que está sendo feito pelo software; facilidades para que o usuário tenha o controle do sistema; atalhos para facilitar a interação com usuários mais experientes; boas mensagens de erro e oferecimento de documentação.

De forma geral, o objetivo desta variável, é responder questões como:

Interface com o usuário:

- As telas do software possuem um visual agradável, de fácil ‘navegação’ e compreensão para o usuário?
- As informações que aparecem nas telas estão bem posicionadas e possuem tamanho de fonte suficiente à sua compreensão?
- O software apresenta suporte a várias línguas diferentes?

Documentação

- O software possui uma documentação que detalhe sua forma de instalação e funcionamento?
- De que forma essa documentação pode ser acessada? Diretamente nas telas de operações do software, no momento em que a dúvida surgir? Em formato de manuais?
- É possível encontrar alguma forma de suporte à instalação, treinamento ou problemas com o software? De que forma?

O trabalho de filtrar dentre os projetos de software de código aberto, aqueles que estão qualificados como ERP aplicáveis à pequenas indústrias, descobrir quais estão em um estágio de desenvolvimento maduro o bastante para serem usados e ainda identificar quais são os requisitos para que eles possam ser colocados em operação e as funcionalidades existentes é uma tarefa que economizaria tempo e dinheiro para a pequena empresa que está à procura de um software desse tipo, pois conforme descreve Mendes (2003), as pequenas empresas jamais cogitam projetos milionários, com prazos

de conclusão de dois anos ou mais, como fazem as grandes empresas, quanto o se trata da adoção de um ERP.

Resultado

O resultado esperado deste estudo é uma análise realizada de acordo com os critérios estabelecidos (viabilidade de instalação, funcionalidades e viabilidade de uso) de uma amostra de sistemas ERP de código aberto que possam ser utilizados em pequenas indústrias, de forma a reunir informações que venham a auxiliar no processo de escolha do software a ser implantado.

Espera-se ainda que as informações possam servir a pesquisadores da área de engenharia de produção e desenvolvimento de software .

Conclusão

As pequenas indústrias que desejam aproveitar-se do ERP como ferramenta de vantagem competitiva, podem encontrar softwares de código aberto que respondam às suas necessidades?

A resposta a este questionamento é fruto de um estudo que exige o levantamento de informações referentes aos projetos mais maduros de software de código aberto. Contudo a sugestão do presente trabalho promove o auxílio em apenas uma das fases de adoção de um ERP: a seleção. Esta é uma das fases de um complexo processo, restando ainda outros aspectos que podem ser estudados sobre o relacionamento ERP de código aberto e as pequenas empresas.

Referências Bibliográficas

COLES, S. ROWLEY, J. *Spreadsheet modeling for management decision making – Industrial Management & Data System*. p.17-23, MCB University Press, 1996

CORRÊA, H.C *ERP's: por que as implantações são tão caras e raramente dão certo? In: Simpósio de administração da produção, Logística e operações industriais, 1998, São Paulo. Anais... São Paulo: FGV. p. 288-300., 1998*

CORRÊA, H.L., GIANESI, I.G.N, (1996) . *Just in time, MRP II e OPT – Um enfoque estratégico 2ª edição. São Paulo: Atlas, 1996*

DANE,F.C. *Research methods*. Belmont-CA:Brooks/Cole, 1990

FREE SOFTWARE FOUNDATION *O que é software livre*. Disponível em: <http://www.gnu.org/philosophy/free-sw.pt.html>. Acesso em 28/12/2004.

FREE SOFTWARE FOUNDATION EUROPE *Porque Falamos de Software Livre*. Disponível em <http://fsfeurope.org/documents/whyfs.pt.html>. Acesso em 29/12/2004.

FRESHMEAT. *About Freshmeat*. Disponível em: <http://freshmeat.net/about/>. Acesso em 03/01/2005.

GIL, A. C. *Métodos e técnicas de Pesquisa Social*. São Paulo: Atlas, 1999.

HABERKORN, E.. *Teoria do ERP – Enterprise Resource Planning*. São Paulo: Makron Books, 1999.

MARTINS, G.A. *Epistemologia da pesquisa em Administração*. Tese (Livre docência) - Faculdade de Economia, Administração e Contabilidade. Universidade de São Paulo, São Paulo. 1995.

MENDES, J. V. *Avaliação de sistemas ERPs como ferramenta de mudança organizacional nas pequenas e médias empresas – um roteiro auxiliar*. Tese de doutorado, Escola de Engenharia de São Carlos, Universidade de São Paulo. 2003

NICHOLS, D.M. & TWIDALE, M. B. *The usability of open source software*. Disponível em: http://firstmonday.org/issues/issue8_1/nichols/index.html. Acesso em 05/01/2005.

NILSEN, J. *Usability Enginnering*. Londres: Academic Press Limited, 1993.

OPEN SOURCE INITIATIVE. *The Open Source Definition – version 1.9* – Disponível em <http://www.opensource.org/docs/definition.php>. Acesso em 29/12/2004.

RITZMAN, L., KRAJEWSKI, L., MOURA, R.. *MRP, MRPII, MRP III (MRP + JIT com Kanban)* – São Paulo: IMAM, 1989.

SOURCE FORGE. *What is Sourceforge.net?* Disponível em: <http://sourceforge.net/docman/>. Acesso em 03/01/2005.

SEBRAE- SP. *A informatização nas MPES paulistas* – Disponível em <www.sebraesp.com.br>. Acesso em 01/09/2003.

ZIMMERER, T. W., SCARBOROUGH, N.M. *Essential of small business management*. New York: Macmillan College Publishing Company. Cap. 16 p. 441 – 458, 1994.