

# A Adaptação de Áreas-Chave de Processo do SW-CMM ao Processo de Reengenharia Orientada a Objetos

Gizelle Sandrini Lemos  
Rosângela Ap. Dellosso Penteadó

Universidade Federal de São Carlos  
{gizelle,rosangel}@dc.ufscar.br

## Resumo

A reengenharia orientada a objetos de sistemas legados procedimentais pode ser realizada de diversas formas com o auxílio de diversos métodos e processos. Um deles é através de linguagem ou família de padrões. A preocupação com a qualidade do processo de desenvolvimento é constante nas empresas, porém o mesmo não ocorre com o processo de reengenharia. Nesse sentido, este artigo apresenta cinco padrões de processo que cuidam do planejamento e do acompanhamento de um processo de reengenharia orientada a objetos de sistemas legados desenvolvidos sem essas características. Esses padrões foram desenvolvidos com base na adaptação das áreas-chave de processo para alcance do nível 2 de maturidade do SW-CMM e incluídos no PRE/OO (Processo de Reengenharia Orientada a Objetos) que agrupa, além desses, outros quinze padrões relacionados à condução do processo de reengenharia. O PRE/OO é composto de sete *clusters* de padrões sendo que os relacionados à qualidade de processo visam trazer para o processo de reengenharia as diretrizes já utilizadas no processo de desenvolvimento de software. Um estudo de caso com sistema Delphi sem características orientadas a objetos é utilizado para exemplificar o processo sendo o sistema alvo re-implementado na mesma linguagem com características orientadas a objetos.

Palavras-Chave: qualidade de processo, reengenharia, padrões de processo, SW-CMM

## Abstract

The object oriented reengineering of procedural legacy systems can be accomplished in several ways, with the use of existent methods and processes. The use of a pattern language or pattern family is one way to make reengineering. The concern with quality in development process is constant in software organizations. However, the same does not occur with the reengineering process. In that sense, this article presents five process patterns that take care of tracking and planning the object oriented reengineering process in procedural legacy system. These patterns had been developed from the adaptation of the key process areas in the maturity level 2 process of the SW-CMM and enclosed in the PRE/OO (Object Oriented Reengineering Process), that groups these and other fifteen standards related to the conduction of the reengineering process. The PRE/OO is composed of seven patterns clusters, with two clusters being related to the quality process. These clusters aim to bring into the reengineering process directions already used in the software development process. A case study with a Delphi system implemented without object oriented characteristics is used to show the process. After employing PRE/OO the new version of Delphi system became object oriented.

Keywords: process quality, reengineering, process patterns, SW-CMM

## 1. Introdução

A qualidade de processo é atualmente um dos aspectos mais discutidos em se tratando de engenharia de software. Isso tem levado as empresas a adotarem regras e modelos para seus processos de desenvolvimento de software obtendo vantagens, principalmente a médio e a longo prazos, no que diz respeito à diminuição e à facilidade de manutenção dos softwares desenvolvidos em processos mais maduros. Porém, muitas organizações ainda enfrentam problemas com o uso e a manutenção de sistemas de software que ainda lhe são úteis porém

que foram desenvolvidos em linguagens obsoletas e em processos caóticos. Sistema legado é a denominação que recebe este tipo de sistema e que será utilizada neste trabalho.

Com o passar do tempo, a tarefa de realizar a manutenção torna-se mais complexa e mais cara e ainda esses sistemas tornam-se cada vez mais desorganizados devido às inúmeras tentativas de adaptações e inclusões de novas funcionalidades [1]. Sendo assim, para esse tipo de software as organizações têm três alternativas: mantê-los com a situação já descrita de desorganização e custos de manutenção cada vez maiores, redesenvolvê-los completamente ou realizar reengenharia tanto para aumentar sua manutenibilidade quanto para implementá-los em um paradigma mais atual com ou sem mudança de linguagem.

No caso de manter um software legado, apenas efetuando as manutenções para que o mesmo continue operando, pode-se citar problemas como a alocação de pessoal para essa tarefa que pode chegar a mais de 60% do esforço de uma organização, além da falta de sua documentação, comum nesses casos e que torna ainda mais crítica a situação [2].

A opção pelo re-desenvolvimento de um software legado também tem problemas associados. O fato de que o software tem as regras de negócios embutidas, que podem não estar documentadas e a possibilidade das pessoas que as dominam não estarem mais na empresa, faz com que o seu completo redesenvolvimento não seja confiável [3]. Outro fator a ser considerado para essa opção é o custo do re-desenvolvimento total do software, geralmente muito alto consumindo tempo e recursos que, na maioria das vezes, as empresas não dispõem [4].

A reengenharia ou parte dela, a engenharia reversa, são as formas que muitas organizações estão buscando para manter/refazer seus softwares, livrando-se das manutenções difíceis e da degeneração de suas estruturas. Por esse motivo, é importante que o resultado desse processo seja confiável.

Na última década, o fator qualidade com relação ao processo de desenvolvimento de software foi bastante estudado com a elaboração de vários modelos de melhoria e amadurecimento do mesmo, o que elevou sua capacitação para a geração de software. Assim, no processo de reengenharia a garantia da qualidade também pode ser encarada como mais uma etapa do processo.

Este trabalho está organizado da seguinte forma: na Seção 2 são comentados assuntos pertinentes ao aqui apresentado. Na Seção 3, o Processo de Reengenharia Orientada a Objetos é apresentado, sendo exibidos os padrões de qualidade e os padrões de processo agrupados nos sete *clusters* que o compõem. A Seção 4 mostra a aplicação dos padrões para qualidade do processo de reengenharia num estudo de caso e na Seção 5 são apresentadas as considerações finais.

## **2. Temas relacionados**

Chikofski e Cross [5] propõem uma padronização para os termos reengenharia, engenharia reversa e engenharia avante. Segundo os autores, a reengenharia de sistemas envolve basicamente duas etapas: alguma forma de engenharia reversa (para se alcançar um nível mais alto de abstração do sistema legado) e a engenharia avante ou reestruturação. Esse processo é indicado para os softwares que ainda têm alta utilidade, mas difícil manutenção. Métodos que proporcionem maior qualidade ao software devem ser utilizados nesse processo. Atualmente, o uso da orientação a objetos tem se mostrado como uma boa perspectiva tanto para o desenvolvimento do software quanto para sua posterior manutenção.

Um aspecto relevante verificado na literatura foi a importância da engenharia reversa dentro do processo de reengenharia para o aumento da manutenibilidade de sistemas legados, pelo fato de se gerar toda a sua documentação, muitas vezes inexistente ou desatualizada.

Com a recuperação dessa documentação pode-se obter ganhos significativos de entendimento do software, auxiliando a realização da sua manutenção.

Segundo Ducasse e outros [9,10], quando um importante software legado não pode mais evoluir naturalmente para satisfazer as mudanças de requisitos, é, freqüentemente, submetido ao processo de reengenharia. Os autores explicam que padrões de reengenharia apresentam novas soluções para uma solução legada recorrente que já não é apropriada e descrevem como realizar a migração da solução legada à nova solução.

Demeyer e outros [11,12] criaram uma linguagem de padrões para atender sistemas legados orientados a objetos. Essa linguagem é utilizada durante a reengenharia de sistemas como alternativa para prover sua reestruturação.

Recchia e Penteadó [13] criaram a FaPRE/OO, uma Família de Padrões para a Reengenharia Orientada a Objetos de sistemas legados procedimentais. O termo reengenharia orientada a objetos, neste trabalho, refere-se ao processo de reengenharia de sistemas legados procedimentais para sistemas orientados a objetos. A FaPRE/OO é formada por quatro *clusters*, três com padrões de engenharia reversa e um com padrões de engenharia avante. O *cluster* é uma forma organizada de agrupar padrões relacionados à situações similares.

Os padrões de reengenharia propostos neste trabalho têm por objetivo comum o aumentar a facilidade de realização desse processo, dada a forma padronizada e bem documentada dos *clusters* que o descrevem. Desse modo, prevê-se a melhor aplicabilidade do processo de reengenharia por parte do engenheiro de software e, conseqüentemente, maior qualidade dos produtos gerados, ou seja, do software orientado a objetos resultante.

O processo de software definido pode ser entendido como o conjunto de ferramentas, métodos e práticas utilizados para desenvolver e manter o software e produtos associados [14]. O processo de software definido tem documentação que detalha o que é feito, quando, por quem, o que é utilizado e o que é produzido. Nesse sentido, a capacitação em desenvolvimento de software reflete a maturidade de uma infra-estrutura e da cultura relacionada aos métodos, práticas e procedimentos do desenvolvimento de software. Reflete, ainda, a qualidade do processo, pois se sabe que a qualidade dos produtos de software depende diretamente da qualidade do processo de desenvolvimento a eles associados.

Não existe na literatura um processo único, definido e documentado para a condução da reengenharia orientada a objetos e também não foram encontradas referências de trabalhos técnicos relacionando o processo de reengenharia com modelos de qualidade. Dessa forma, a partir dos trabalhos de Demeyer e outros [11, 12], Recchia e Penteadó [13] e Paulk e outros [16] observou-se que padrões podem ser elaborados para direcionar o engenheiro de software na condução da reengenharia por serem formas úteis e didáticas de descrever seus passos e atividades.

Dentre os modelos existentes relacionados à qualidade do processo de software, foi utilizado neste trabalho o SW-CMM (Capability Maturity Model for Software) [16], desenvolvido em uma estrutura de estágios de maturidade, níveis de melhoria crescente, visando alcançar um processo de software maduro.

Com exceção do Nível 1, cada nível de maturidade do SW-CMM é composto por áreas-chave de processo (Key Process Áreas - KPAs). Essas áreas-chave de processo constituem a primeira divisão sistemática dentro dos níveis de maturidade, descrevendo as metas que devem ser atingidas e as questões que devem ser tratadas para que se alcance essas metas e, assim, o nível de maturidade pretendido. As KPAs que devem ser implementadas para o alcance do nível 2 de maturidade [15] são: Gerenciamento de Requisitos; Planejamento do Projeto de Software; Garantia da Qualidade; Controle de Configuração; Acompanhamento e Supervisão do Projeto de Software e Gerenciamento de Sub-Contratos.

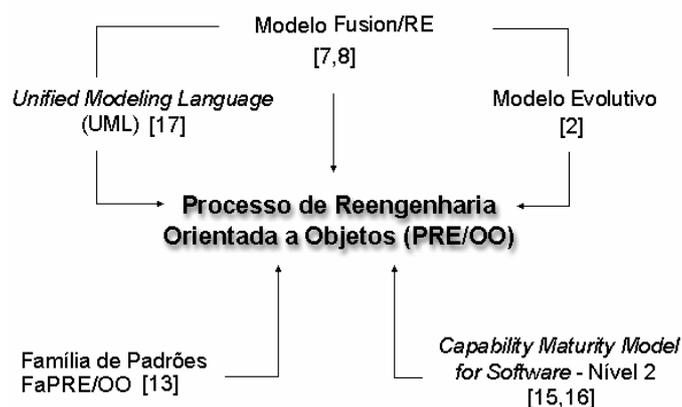
A qualidade de processo bastante referenciada, atualmente, em se tratando do processo de desenvolvimento de software, foi considerada neste trabalho pela importância de garantir que tanto o processo quanto o produto gerado, com a aplicação do processo de reengenharia, tenham qualidade. Para que isso fosse obtido, foram estudadas e adaptadas ao processo de reengenharia, as KPAs para alcance do nível 2 de maturidade do SW-CMM por tratarem de aspectos básicos de um processo e devido à sua relevância nesse caso.

### 3. PRE/OO (Processo de Reengenharia Orientada a Objetos)

#### 3.1. Origens do PRE/OO

O processo proposto para reengenharia orientada a objetos – PRE/OO, por meio de padrões, visa auxiliar o engenheiro de software na plena realização da reengenharia orientada a objetos sendo a sua elaboração realizada com base em (Figura 1):

- Fusion/RE – foram utilizados os passos do método, o qual foi proposto para realização da engenharia reversa orientada a objetos de sistemas legados procedimentais seguindo o modelo de processo seqüencial linear;
- UML (Unified Modeling Language) – foram utilizados os diagramas UML para documentação dos artefatos de trabalho;
- Modelo de Processo Evolutivo – utilizado com o objetivo de permitir ao engenheiro de software o retorno a passos anteriormente realizados do processo como forma de refinar os produtos gerados;
- FaPRE/OO – foram adaptados os padrões dessa família para uso na reengenharia orientada a objetos de sistemas implementados em Delphi sem características orientadas a objetos para sistemas com essas características;
- SW-CMM – foram adaptadas as KPAs para alcance do Nível 2 de maturidade ao processo de reengenharia, como forma de padrões para sua utilização durante todo o processo. A única KPA desse nível não utilizada foi Gerenciamento de Sub-Contratos.



**Figura 1. Fontes utilizadas para elaboração do PRE/OO**

A seção seguinte descreve a organização do Processo de Reengenharia Orientada a Objetos. São apresentados os *clusters* que compõem o processo e a finalidade de cada um.

### 3.2. Organização do PRE/OO

O PRE/OO está organizado em sete *clusters* de padrões, como mostra a Figura 2, divididos da seguinte forma:

- padrões para a melhoria da qualidade do processo de reengenharia orientada a objetos:
  - *cluster 1* – Preparar e Planejar o Processo de Reengenharia: trata da preparação e do planejamento, sendo realizado anteriormente ao início do processo de reengenharia;
  - *cluster 2* – Melhorar a Qualidade do Processo de Reengenharia: trata do acompanhamento da aplicação dos padrões dos *clusters* 3 a 7 a fim de se acompanhar a elaboração dos produtos de trabalho gerando produtos com qualidade.
- padrões para a realização da reengenharia orientada a objetos:
  - *clusters* 3 a 5, contêm os padrões que descrevem a realização da engenharia reversa;
  - *clusters* 6 e 7, contêm os padrões que descrevem a realização da engenharia avante.

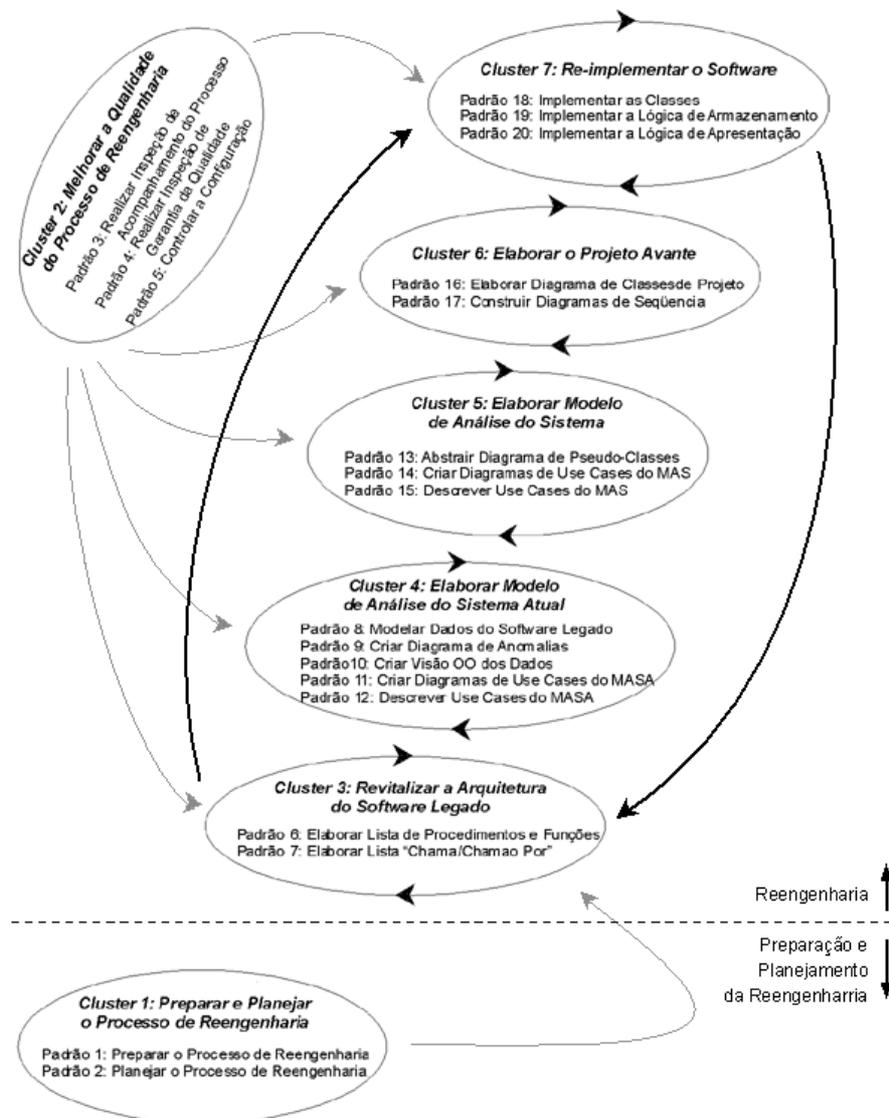


Figura 2. Clusters de padrões do PRE/OO

O processo é iniciado com a aplicação do *cluster 1*, com a definição e o planejamento do processo de reengenharia para o software legado em questão. A seguir são aplicados os padrões dos *clusters 3 a 7*, intercalados pelos padrões do *cluster 2*, para acompanhamento, verificação e controle dos resultados.

Pelo fato do processo ser realizado de forma evolutiva, pode-se repetir sua aplicação a partir de qualquer um dos *clusters 3 a 7* e, cada padrão, pode ser aplicado quantas vezes for necessário. Na medida em que o processo evolui são aplicados os padrões do *cluster* posterior ao realizado, porém pode ser necessário retornar a *clusters* anteriores para sanar dúvidas que surgirem.

O processo de reengenharia, descrito neste trabalho sob a forma de *clusters* de padrões, segue o formato proposto na Linguagem de Padrões de Engenharia Reversa e Reengenharia [11,12] e que também foi utilizada na Família de Padrões para Reengenharia Orientada a Objetos [13]. Cada padrão é composto pelos seguintes itens: **Número, Nome, Intuito, Problema, Contexto, Influências, Solução, Avaliação, Justificativa, Exemplo, Usos Conhecidos, Padrões Relacionados e Produtos Obtidos.**

Os *clusters* relacionados ao processo de reengenharia orientada a objetos têm o objetivo de prover o entendimento de um sistema legado procedimental para, posteriormente, gerar um modelo de análise orientado a objetos (engenharia reversa – *clusters 3 a 5*) correspondente. A partir do modelo de análise é elaborado o modelo de projeto para que o sistema legado seja implementado em linguagem orientada a objetos (engenharia avante – *clusters 6 e 7*).

Na seção seguinte são comentados os *clusters* que tratam da garantia de qualidade no processo de reengenharia (*clusters 1 e 2*).

### 3.3. Padrões de qualidade de processo do PRE/OO

Segundo Fiorini e outros [14], determinar os requisitos em relação ao processo de software é, em última análise, entender exatamente o quê deve ser feito e o quê se espera receber como resultado. Adotando-se um paralelo para o processo de reengenharia, há a necessidade da determinação das atividades e dos produtos de trabalho que devem ser gerados. O conjunto dos produtos de trabalho, sejam resultados simples ou compostos a serem desenvolvidos ao longo do processo, compõe os requisitos do processo de reengenharia.

As diferenças entre um projeto e outro relacionadas à complexidade do software legado, ao domínio da aplicação e às fontes de dados auxiliares disponíveis, tornam necessário o levantamento dos requisitos. A partir da obtenção das informações sobre o quê deve ser feito, segue a determinação de como o processo de reengenharia será realizado, visando a elaboração do planejamento do projeto. A partir dessa necessidade originou-se o *cluster 1* – Preparar e Planejar o Processo de Reengenharia composto pelos padrões:

- Padrão 1: Preparar o Processo de Reengenharia;
- Padrão 2: Planejar o Processo de Reengenharia.

As bases para a construção desse *cluster* de padrões foram as KPAs 1 e 2 do Nível 2 de maturidade do SW-CMM, respectivamente. O Gerenciamento de Requisitos (KPA 1) permite a definição dos requisitos, ou seja, das necessidades em relação à reengenharia do sistema legado. A partir do Planejamento do Projeto (KPA 2) é possível prover os prazos e custos para o cumprimento dessas necessidades. Maiores detalhes desse *cluster* são fornecidos na próxima seção.

O *cluster 2*, composto por três padrões relacionados ao acompanhamento do processo, é aplicado durante a realização da reengenharia orientada a objetos com a utilização e a adaptação das KPAs: Acompanhamento e Supervisão do Projeto de Software, Garantia da

Qualidade e Gerenciamento de Configuração. A única KPA do Nível 2 de maturidade não utilizada foi o Gerenciamento de Sub-Contratados, pelo fato desta não estar relacionada diretamente ao processo. O *cluster* 2, Melhorar a Qualidade do Processo de Reengenharia, agrupa os seguintes padrões:

- Padrão 3: Acompanhar o Progresso do Processo de Reengenharia;
- Padrão 4: Realizar Inspeção de Garantia da Qualidade;
- Padrão 5: Controlar a Configuração.

O padrão 3 – Acompanhar o Progresso do Processo de Reengenharia visa o acompanhamento do projeto, complementando o planejamento, feito anteriormente, de forma a mantê-lo sempre atualizado. Esse padrão foi elaborado com base na KPA Acompanhamento e Supervisão de Projeto.

O padrão 4 – Realizar Inspeção de Garantia da Qualidade visa a realização de inspeções de garantia da qualidade nos produtos elaborados durante o processo, tomando por base a KPA Garantia da Qualidade. Segundo Rocha e outros [18], as inspeções são atividades voltadas para a garantia da qualidade que podem ser aplicadas ao longo do processo para a revisão de vários produtos de trabalho, com a vantagem de poderem ser aplicadas assim que o produto é elaborado.

Completando a utilização das KPAs do Nível 2 de maturidade do SW-CMM, foi criado o padrão 5, Controlar a Configuração, aplicado ao longo do processo de reengenharia, adaptando o que é recomendado na KPA Gerenciamento de Configuração com relação ao controle de versões e o gerenciamento de *baselines*.

O estudo de caso apresentado na próxima seção ilustra o uso dos padrões do PRE/OO sendo o foco dado nos dois primeiros *clusters* e apenas comentados os que se referem ao processo de reengenharia em si.

#### 4. Estudo de Caso

Para exemplificar o uso dos padrões de qualidade de processo do PRE/OO, *clusters* 1 e 2 da Figura 2, um software legado procedimental, denominado *ControleGlobal*, implementado em Delphi [21] sem características orientadas a objetos foi submetido ao processo de reengenharia [20]. O software possui em torno de 20.000 linhas de código e apesar do ambiente ter *Object Pascal* como linguagem de programação, permitindo a implementação totalmente de acordo com a orientação a objetos, esse paradigma não foi seguido sendo o software desenvolvido de forma procedimental. Esse sistema está inserido no domínio de gestão de recursos de negócios e foi desenvolvido há alguns anos por um dos autores deste trabalho, encontrando-se ainda em uso.

O processo de reengenharia orientada a objetos do software legado iniciou-se com a aplicação do *cluster* 1, de forma a prover a contextualização do projeto em questão seguido pelo planejamento das atividades contidas no PRE/OO.

O Quadro 1 ilustra o documento resultante da aplicação do padrão 1 – Preparar o Processo de Reengenharia. Para a preparação do processo foram definidos os limites do projeto, a partir do entendimento de seu escopo (Exame da Situação). A determinação dos “Produtos de Trabalho a Serem Elaborados” considera os produtos que serão elaborados pela aplicação dos padrões 3 a 7. As “Atividades a Serem Realizadas” correspondem aos padrões necessários à elaboração dos produtos anteriormente listados.

### Quadro 1. Resultado do levantamento de atividades realizado para o estudo de caso

LEVANTAMENTO DE ATIVIDADES	
<b>Projeto</b> ControleGlobal	<b>Data de Criação do Documento</b> 21/11/2001
<b>Versão do Documento</b> 1.0	<b>Responsável pela Criação do Documento</b> Engenheiro de Software 1
<b>Exame da Situação</b> <p>O software em questão foi desenvolvido para prover a gerência das seguintes áreas de uma empresa de pequeno porte responsável pela comercialização de artigos para vestuário:</p> <ul style="list-style-type: none"><li>estoque: cadastro, entrada e saída de mercadorias, cálculo do preço de venda médio e do ganho com relação a cada mercadoria cadastrada; possibilidade de associação de mercadorias a grupos; impressão de vários relatórios e de etiquetas com códigos de barras para cada mercadoria;</li><li>vendas e trocas efetuadas: cálculo de vários tipos de planos de pagamento (à vista ou em crediário), movimentação do estoque com relação à entrada e saída de mercadorias, cálculo da comissão dos vendedores e do valor de impostos; impressão de notas fiscais;</li><li>clientes cadastrados: confecção e impressão de mala direta, possibilidade de diferenciação de clientes em especiais e com restrições de crédito;</li><li>crediários: controle de pagamento de parcelas, controle de parcelas em atraso, a vencer; impressão de relatórios periódicos de atrasados;</li><li>fechamento mensal: controle de custos e lucros com relação às vendas, saídas de estoque e pagamentos aos vendedores;</li><li>caixa diário da empresa: fechamento diário e controle das entradas e saídas do caixa, controle da origem da cada movimentação (venda, troca, retirada de caixa).</li></ul> <p>Cada uma das funcionalidades citadas foi desenvolvida em um módulo independente e os dados são armazenados em tabelas do banco de dados relacional MySQL.</p>	
<b>Itens Disponíveis</b> <ul style="list-style-type: none"><li>Descrição de todas as tabelas de dados (scripts SQL constantes de arquivos de dados);</li><li>Código-fonte completo;</li><li>Programa executável.</li></ul>	
<b>Produtos de Trabalho a Serem Elaborados</b> <p>No processo de reengenharia, ter-se-á a elaboração do software orientado a objetos, sendo gerados, durante todo o processo, os seguintes produtos:</p> <ul style="list-style-type: none"><li>Lista de Procedimentos e Funções;</li><li>Lista "Chama/Chamado Por";</li><li>Lista de Anomalias;</li><li>Modelo Entidade-Relacionamento;</li><li>Lista de Tabelas e Chaves;</li><li>Diagramas de Pseudo-Classes;</li><li>Diagrama de Classes MAS;</li><li>Diagramas de Use Cases (MASA e MAS);</li><li>Descrições dos Use Cases;</li><li>Lista de Correspondência de Métodos Anômalos;</li><li>Lista de Mapeamento MASA x MAS;</li><li>Lista de Mapeamento dos Use Cases;</li><li>Diagrama de Classes de Projeto;</li><li>Diagramas de Seqüência;</li><li>Código-fonte Orientado a Objetos.</li></ul>	
<b>Atividades a Serem Realizadas</b> <p>Dada a necessidade da elaboração de todos os produtos de trabalho que compõem o PRE/OO, serão aplicados os 15 padrões que compõem os <i>clusters</i> 3 a 7.</p>	

O padrão 2 foi aplicado após a elaboração do artefato Levantamento de Atividades produzido durante a aplicação do padrão 1. O planejamento do processo é realizado de forma a dispor as atividades documentadas, a partir da aplicação do padrão 1, de acordo com o tempo disponível para a realização do PRE/OO. Neste estudo de caso, o tempo considerado é de 280 horas. Assim, realizou-se a divisão de tempo entre os *clusters* de padrões a serem aplicados, considerando-se os diversos produtos de trabalho a serem elaborados. Essa distribuição de tempo disponível para aplicação dos diversos padrões foi feita manualmente, sem a utilização de métricas de esforço, considerando o número e a complexidade dos produtos obtidos com a aplicação de cada padrão, considerando-se somente a disponibilidade e a experiência do engenheiro de software na empresa. Havendo disponibilidade, métricas de

esforço podem ser utilizadas para fornecer valores mais precisos na distribuição do tempo.

Informações sobre a equipe responsável pela realização do PRE/OO bem como os papéis de cada integrante foram omitidos pelo fato do processo ter sido realizado por apenas um engenheiro de software.

Em seguida, foram definidos os itens de configuração controlados durante o PRE/OO. Outros dados documentados foram os referentes às inspeções de garantia da qualidade e às inspeções de acompanhamento do processo, com o objetivo de garantir sua realização com qualidade, como ilustra o Quadro 2. Nesse quadro estão detalhadas somente as atividades para o passo 1 deste estudo de caso, sendo que o planejamento foi realizado para todos os passos (*clusters* 3 a 7) constantes da reengenharia ou que serão realizados de acordo com o estipulado no Quadro 1, em “Atividades a Serem Realizadas”.

**Quadro 2. Visualização parcial do Plano para Realização da Reengenharia**

PLANO PARA REALIZAÇÃO DA REENGENHARIA	
<b>Projeto</b> ControleGlobal	<b>Data de Criação do Documento</b> 21/11/2001
<b>Versão do Documento</b> 1.0	<b>Responsável pela Criação do Documento</b> Engenheiro de Software 1
<b>Dados Gerais do Processo de Reengenharia</b>	
<b>Tempo Disponível para o Processo</b> 70 dias. 4 horas de trabalho/dia. Total: 280 horas	
<b>Data Estimada para Finalização do Processo de Reengenharia:</b> 22/02/2002	
<b>Itens de Configuração</b>	
<ul style="list-style-type: none"> <li>• Lista de Procedimentos e Funções;</li> <li>• Lista “Chama/Chamado Por”;</li> <li>• Lista de Anomalias;</li> <li>• Modelo Entidade-Relacionamento;</li> <li>• Lista de Tabelas e Chaves;</li> <li>• Diagramas de Pseudo-Classes;</li> <li>• Diagrama de Classes MAS;</li> <li>• Diagramas de Use Cases (MASA e MAS);</li> </ul>	<ul style="list-style-type: none"> <li>• Descrições dos Use Cases;</li> <li>• Lista de Correspondência para Métodos Anômalos;</li> <li>• Lista de Mapeamento MASA x MAS;</li> <li>• Lista de Mapeamento do Use Cases;</li> <li>• Diagramas de Seqüência;</li> <li>• Diagrama de Classes de Projeto;</li> <li>• Código-fonte Orientado a Objetos.</li> </ul>
<b>Inspeções de Garantia da Qualidade</b>	
As inspeções de garantia da qualidade para prover a conformidade dos produtos de trabalho elaborados devem ser realizadas ao fim da elaboração de cada produto de trabalho, em cada uma das voltas a serem dadas durante o processo, que ocorre de forma evolutiva.	
<b>Inspeções de Acompanhamento e Supervisão do Projeto</b>	
Uma inspeção de acompanhamento do processo de reengenharia comparando os resultados obtidos com os dados estimados deve ser realizada ao final de cada um dos <i>clusters</i> 3 a 6. Itens a serem inspecionados:	
<ul style="list-style-type: none"> <li>• Produtos de trabalho a serem elaborados</li> <li>• Tempo gasto na conclusão do passo</li> <li>• Inspeções de garantia da qualidade realizadas e resultados obtidos</li> </ul>	
<b>Passo 1 – Revitalização da Arquitetura (cluster 3)</b>	
<b>Produtos de Trabalho</b>	
<ul style="list-style-type: none"> <li>• Lista de Procedimentos e Funções</li> <li>• 1 dia para conclusão (3 hs de trabalho)</li> <li>• Alvo de inspeção de garantia da qualidade (0,5 h de trabalho)</li> <li>• Recurso computacional necessário: editor de textos</li> <li>• Lista de “Chama/Chamado por”</li> <li>• 6 dias para conclusão (24 hs de trabalho)</li> <li>• Alvo de inspeção de garantia da qualidade (5,5 hs de trabalho);</li> <li>• Recurso computacional necessário: editor de textos.</li> </ul>	

<b>Tempo Necessário para Conclusão do Passo</b>	
<ul style="list-style-type: none"> <li>• 7 dias ou 27 hs de trabalho – revitalização da arquitetura</li> <li>• 2 dias ou 6 hs de trabalho – inspeções para garantia da qualidade</li> <li>• 1 dia ou 1 h de trabalho – inspeção de acompanhamento de projeto ao término da realização desse passo</li> <li>• Tempo total resultante (estimado): 27hs + 6hs + 1h = <b>34hs ou 9 dias</b></li> </ul>	
<b>Número de Inspeções</b>	
Inspeções de Garantia da Qualidade: 2	Inspeções de Acompanhamento e Supervisão: 1
<b>Passo 2 – Recuperação do Modelo de Análise do Sistema Atual (cluster 4)</b>	
...	
<b>Passo 3 – Obtenção do Modelo de Análise do Sistema (cluster 5)</b>	
...	
<b>Passo 4 – Elaboração do Projeto Avante (cluster 6)</b>	
...	
<b>Passo 5 – Re-implementação do Software (cluster 7)</b>	
...	
<b>Totalizações</b>	
...	

Após a aplicação dos padrões 1 e 2 (*cluster 1*) iniciou-se a etapa de engenharia reversa com a aplicação do padrão 6 que resulta em uma lista com o mapeamento dos procedimentos e funções do software legado. Após a elaboração dessa lista, ocorreu sua inspeção, com a aplicação do padrão 4 – Realizar Inspeção de Garantia da Qualidade, do *cluster 2*. A inspeção de garantia da qualidade da Lista de Procedimentos e Funções foi realizada manualmente neste estudo de caso.

Antes da inspeção foram verificados quais os aspectos seriam analisados. De acordo com o tempo disponível para essa inspeção o artefato foi inspecionado e os resultados obtidos documentados para posterior realização das ações corretivas. No Quadro 3 é exibido o resultado da aplicação do padrão 4 para o produto obtido com aplicação do padrão 7 (Lista de Procedimentos e Funções).

**Quadro 3. Visualização do documento de inspeção da Lista de Procedimentos e Funções**

INSPEÇÃO DE GARANTIA DA QUALIDADE	
<b>Projeto</b> ControleGlobal	<b>Data de Criação do Documento</b> 22/11/2001
<b>Item Alvo da Inspeção de Garantia da Qualidade</b> Lista de Procedimentos e Funções – Versão 1.0	<b>Responsável pela Inspeção</b> Engenheiro de Software 1
<b>Inspeção de Garantia da Qualidade N.º: 1</b>	
<b>Aspectos Analisados</b>	
<ol style="list-style-type: none"> <li>1. Nomes dos módulos</li> <li>2. Nomes dos procedimentos e funções</li> <li>3. Classificação dos procedimentos e funções</li> <li>4. Comparação entre o número de procedimentos e funções no código e na lista analisada</li> </ol>	
<b>Diferenças Encontradas</b>	
<ol style="list-style-type: none"> <li>1. Nomes dos módulos: OK</li> <li>2. Nomes dos procedimentos e funções: <ul style="list-style-type: none"> <li>• Nome do Procedimento 180 – AjustaValOrcamento</li> <li>• Nome do Procedimento 203 – OrcTocasPageManagerGetNextPage</li> </ul> </li> <li>3. Classificação dos procedimentos e funções: <ul style="list-style-type: none"> <li>• Classificação do Procedimento 5 – Display Hint: Ev para Pv</li> </ul> </li> <li>4. Comparação entre o número de procedimentos e funções no código e na lista analisada: OK</li> </ol>	
<b>Ações Corretivas</b>	
<ul style="list-style-type: none"> <li>• Criação da Versão 1.1 da Lista de Procedimentos e Funções com as correções necessárias</li> </ul>	

A Lista de Procedimentos e Funções foi corrigida, segundo as necessidades verificadas na inspeção e, então, submetida ao controle de configuração. Para isso, foi aplicado o segundo padrão do *cluster 2* (padrão 5), chamado Controlar a Configuração. O Quadro 4 ilustra a Lista de Controle de Configuração ao final da aplicação do *cluster 3*, propiciando ao engenheiro de software o controle de alterações das diversas versões necessárias de cada produto de trabalho gerado. Nesse Quadro podem ser visualizados todos os artefatos inspecionados e suas respectivas versões obtidas após a realização do processo e das inspeções.

**Quadro 4. Lista de Controle de Configuração após a aplicação do *cluster 3***

LISTA DE CONTROLE DE CONFIGURAÇÃO			
<b>Projeto</b> ControleGlobal		<b>Responsável pelo Controle de Configuração</b> Engenheiro de Software 1	
<b>Passo do PRE/OO</b> Revitalização da Arquitetura			
<b>Data Criação</b>	<b>Nome do Documento</b>	<b>Versão</b>	<b>Origem</b>
22/11/2001	Lista de Procedimentos e Funções	1.0	Revitalização da Arquitetura
22/11/2001	Lista de Procedimentos e Funções	1.1	Inspeção de Garantia da Qualidade: N.º 1
23/11/2001	Lista "Chama/Chamado Por"	1.0	Revitalização da Arquitetura
28/11/2001	Lista "Chama/Chamado Por"	1.1	Revitalização da Arquitetura
03/12/2001	Lista "Chama/Chamado Por"	1.2	Inspeção de Garantia da Qualidade: N.º 2

Ao final da aplicação dos padrões do *cluster 3* foi originada a primeira *baseline* do projeto (Quadro 5), com os produtos elaborados e inspecionados durante a condução do passo. A seguir, foi aplicado outro padrão do *cluster 2* (padrão 3) – Acompanhar o Progresso do Processo de Reengenharia, com o objetivo de verificar se o Plano para Realização da Reengenharia estava sendo cumprido de forma efetiva. O Quadro 6 ilustra o documento criado após a inspeção. Na aplicação desse padrão no estudo de caso não houve necessidade da aplicação de Ações Corretivas, pois pelos Resultados Obtidos pode-se notar que o tempo estimado foi suficiente para a realização das atividades pretendidas. Caso isso não tivesse ocorrido, algumas ações deveriam ser realizadas e estariam listadas nesse item.

**Quadro 5. *Baseline* com produtos inspecionados após a aplicação do *cluster 3***

BASELINE DE PROJETO		
<b>Projeto</b> ControleGlobal		<b>Data de Criação da <i>Baseline</i></b> 03/12/2001
<b>Passo do PRE/OO</b> Revitalização da Arquitetura		<b>Responsável pelo Controle de Configuração</b> Engenheiro de Software 1
<b>Descrição</b> <i>Baseline</i> 1, contendo o conjunto dos itens de configuração aprovados após as inspeções de garantia da qualidade, os quais descrevem a revitalização da arquitetura do software legado.		
<b>Itens de Configuração</b>	<b>Versão</b>	<b>Data de Criação</b>
Lista de Procedimentos e Funções	1.1	22/11/2001
Lista "Chama/Chamado Por"	1.2	03/12/2001

**Quadro 6. Inspeção de acompanhamento realizada após a aplicação do *cluster* 3**

INSPEÇÃO DE ACOMPANHAMENTO DO PROCESSO	
<b>Projeto</b> ControleGlobal	<b>Data da Criação do Documento</b> 03/12/2001
<b>Passo do PRE/OO</b> Revitalização da Arquitetura	<b>Responsável pela Criação do Documento</b> Engenheiro de Software 1
<b>Inspeção de Acompanhamento N.º 1</b>	
<b>Itens Analisados</b> 1. Produtos de Trabalho Elaborados 2. Tempo Gasto na Elaboração dos Produtos de Trabalho 3. Inspeções de Garantia da Qualidade Realizadas 4. Tempo Gasto para Conclusão do Passo	
<b>Resultados Obtidos</b> 1. Todos os produtos de trabalho que compõem a revitalização da arquitetura foram gerados 2. O tempo gasto para a elaboração dos produtos de trabalho condiz com o estimado no Plano 3. Foram realizadas todas as inspeções de garantia da qualidade necessárias 4. O tempo gasto nas inspeções de garantia da qualidade condiz com o estimado no Plano	
<b>Ações Corretivas</b> -	

Os demais passos do PRE/OO, correspondentes aos *cluster* 4 a 7 são realizados da mesma maneira, intercalando-os com a aplicação dos padrões do *cluster* 2, com o objetivo de garantir a qualidade do processo de reengenharia. Detalhes do processo podem ser encontrados em [22].

A preparação e o planejamento prévios do processo permitem, durante a sua aplicação, o acompanhamento por meio de inspeções, o que mantém sua condução sempre de acordo em termos de alocação de tempo, recursos e pessoal. Antes da utilização de cada padrão dos *clusters* 3 a 7, devem ser revisados o Plano para Realização da Reengenharia (Quadro 2) e os tempos e os recursos estimados para a elaboração dos produtos de trabalho recuperados para a aplicação do padrão de forma organizada. O tempo estimado é utilizado como guia, sendo redimensionado de acordo com a necessidade e, caso seja preciso, reportado durante as inspeções de acompanhamento do processo.

A aplicação do *cluster* 2, Melhorar o Processo de Engenharia Reversa, relacionado à qualidade do processo deve ser contínua, realizada ao longo da aplicação de todos os padrões que compõem os *clusters* 3 a 7 e, ainda, após o encerramento do PRE/OO (em futuras manutenções ou evoluções do software). Dessa forma, garante-se que os produtos de trabalho e a documentação da qualidade como, por exemplo, a Lista de Controle de Configuração, não se tornam desatualizados.

O controle de configuração permite a qualquer momento durante o processo, o retorno às versões mais atualizadas dos produtos de trabalho e à realização de alterações de forma controlada. Já, o estabelecimento de *baselines* possibilita que conjuntos de produtos de trabalho corretos e inspecionados sejam utilizados, quando necessários, durante a aplicação dos padrões posteriores pelo engenheiro de software.

A descrição do PRE/OO em forma de padrões facilita a sua realização dada a modularidade e a organização das informações necessárias, aspecto relevante em vista de sua extensão. Essa divisão facilita, ainda, a utilização individual de padrões, por ser possível retornar de qualquer ponto para que se detalhe produtos específicos por ele gerados.

## 5. Considerações Finais

Este trabalho propôs um processo de reengenharia orientada a objetos (PRE/OO) definido na forma de vinte padrões, distribuídos em sete *clusters* compondo a engenharia reversa, a engenharia avante e a qualidade global do processo. Para exemplificar a utilização do PRE/OO foi conduzida a reengenharia de um software legado procedimental desenvolvido utilizando-se o ambiente Delphi sem a preocupação em considerar os recursos orientados a objetos nele existentes.

A necessidade da criação de um processo de reengenharia orientada a objetos que cuidasse dos aspectos de garantia da qualidade foi percebida pela falta de referências ao assunto na literatura especializada bem como pela falta de informações de engenheiros de software para a condução desse processo. A maioria das empresas não tem documentação de seus sistemas e quando optam pela reengenharia a fazem sem a preocupação com a qualidade do processo e/ou do produto. Muitas vezes, a falta de planejamento para a realização desse processo leva a produtos inferiores aos que previamente existiam.

Dessa forma, optou-se por utilizar as KPAs do nível 2 de maturidade do SW-CMM para a realização de um processo de reengenharia orientada a objetos. A utilização dessas KPAs foi possível por tratarem de aspectos básicos de processos em geral como a definição de requisitos, o planejamento do processo, a gerência de configuração e as inspeções, preocupações existentes também na reengenharia. Com isso pretende-se que o processo de reengenharia orientada a objetos, PRE/OO, seja realizado com qualidade proporcionando um resultado final também satisfatório.

A elaboração do PRE/OO utilizou as diretrizes do método Fusion/RE e o ambiente de desenvolvimento Delphi com os recursos da linguagem *Object Pascal* de forma que o engenheiro de software não tenha gastos de tempo com o aprendizado de uma nova linguagem de programação. Sua escrita ocorreu na forma de padrões para facilitar o entendimento por parte de outros engenheiros de software que venham a utilizá-lo.

Acredita-se que a utilização de padrões relacionados à melhoria da qualidade favoreça a geração de um produto com a mesma funcionalidade que a do sistema legado e que em todos os pontos do processo possam ser acompanhados os esforços com ele gastos.

Como trabalho futuro pretende-se medir a dificuldade de manutenção antes de depois de software serem submetidos ao PRE/OO de forma a quantificar o aumento da manutenibilidade alcançado. Além disso, pretende-se verificar a possibilidade da adaptação de outras KPAs ao PRE/OO como, por exemplo, a Engenharia do Produto de Software, KPA do Nível 3 do SW-CMM v1.1.

## Referências Bibliográficas

- [1] **Tilley, S.** *A Reverse Engineering Environment Framework*. Software Engineering Institute, CMU/SEI-98-TR-005. 1998.
- [2] **Pressman, R. S.** *Software Engineering: A Practitioner's Approach. Fifth Edition*. McGraw-Hill Higher Education. 2001.
- [3] **Quinaia, M. A.;** Sanches, R. *Reengenharia de Software*. Relatório Técnico nº 84, ICMC-USP – São Carlos. 1999.
- [4] **Olsem, M. R.** *An Incremental Approach to Software Systems Re-engineering*. Software Maintenance: Research and Practice. Volume 10, páginas 181-202. 1998.

- [5] **Chikofsky, J. E.; Cross, J. H.** *Reverse Engineering and Design Recovery: A Taxonomy*. IEEE Software. Vol. 7, n.º 1, páginas 13-17. 1990.
- [6] **Coleman, D.; Arnold, P.; Bodoff, S.; Dollin, C.; Gilchrist, H. and Hayes, F.** *Object-Oriented Development - The Fusion Method*. Prentice Hall. ISBN 0133388239. 1994.
- [7] **Penteado, R. A. D.** *Um Método para Engenharia Reversa Orientada a Objetos*. Tese (Doutorado em Física Computacional). Instituto de Física de São Carlos, Universidade de São Paulo – São Carlos/SP. 1996.
- [8] **Penteado, R. A. D.; Masiero, P. C.; Cagnin, M. I.** *An Experiment of Legacy Code Segmentation to Improve Maintainability*. III European Conference on Software Maintenance and Reengineering – IEEE. Amsterdã, Holanda. Páginas 111-119. 1999.
- [9] **Ducasse, S.; Richner, N.; Nebbe, R.** *Two Reenginerring Patterns: Eliminating Type Checking*. 4<sup>th</sup> European Conference on Pattern Languages of Programming and Computing. Paul Dyson (Ed.). UVK Universitätsverlag Konstanz GmbH, Konstanz, Germany. July, 1999.
- [10] **Ducasse, S.; Richner, N.; Nebbe, R.** *Type Checking Elimination: Two Object Oriented Reengineering Patterns*. 6<sup>th</sup> Working Conference on Reverse Engineering). IEEE Computer Society Press. Pages 157-168. 1999.
- [11] **Demeyer, S.; Ducasse, S.; Nierstrasz, O.** *A Pattern Language for Reverse Engineering*. 5<sup>th</sup> European Conference on Pattern Languages of Programming and Computing. Pages 189-208. July, 2000.
- [12] **Demeyer, S.; Ducasse, S.; Nierstrasz, O.** *Tie Code and Questions: a Reengineering Pattern*. 5<sup>th</sup> European Conference on Pattern Languages of Programming and Computing. Pages 209-217. July, 2000.
- [13] **Recchia, E. L.; Penteado, R. A. D.** *FaPRE/OO: Uma Família de Padrões para Reengenharia Orientada a Objetos de Sistemas Legados Procedimentais*. II SugarloafPLOP. Itaipava, Rio de Janeiro. Agosto, 2002.
- [14] **Fiorini, S. T.; Staa, A.; Baptista, R. M.** *Engenharia de Software com CMM*. Ed. Brasport, Rio de Janeiro. 1998.
- [15] **Paulk, M.C.; Weber, C. V; Garcia, S.; Chrissis, M. B.; Bush, M.** *Key Practices of the Capability Maturity Model, Version 1.1*. Software Engineering Institute, CMU/SEI-93-TR-25. 1993.
- [16] **Paulk, M. C.; Curtis, B.; Chrissis, M. B.; Weber, C. V.** *Capability Maturity Model for Software, Version 1.1*. Software Engineering Institute, CMU/SEI-93-TR-24. 1993.
- [17] **OMG.** *Unified Modeling Language Specification*. Object Management Group. Version 1.3. June, 1999.
- [18] **Rocha, A. R. C.; Maldonado, J. C.; Weber, K.C.** *Qualidade de Software Teoria e Prática*. 1ª Edição – Prentice Hall. São Paulo. 2001.
- [19] **Rational Corporation.** *Unified Modeling Language*. <http://www.rational.com/uml> – Acessado em 01/05/2001.

- [20] **Lemos, G. S.; Penteado, R. A. D.** *Reengenharia do Software ControleGlobal para Lojas*. Documento de Trabalho nº 1. Departamento de Computação, Universidade Federal de São Carlos. Fevereiro, 2002.
- [21] **Inprise Corporation.** Borland Delphi 5 – Developer’s Guide. 1999.
- [22] **Lemos, G. S.** PRE/OO – Um Processo de Reengenharia Orientada a Objetos com Ênfase na Garantia da Qualidade. Dissertação (Mestrado em Ciência da Computação). Universidade Federal de São Carlos – São Carlos/SP. 2002. Disponível em: <http://www.svconsultoria.com.br/pessoal/gizelle>