

Publicação do Processo de Software no ImPProS: Um Ambiente de Implementação Progressiva do Processo de Software

Sandro Ronaldo Bezerra Oliveira
Universidade da Amazônia – UNAMA
Centro de Ciências Exatas e Tecnologia
Av. Alcindo Cacela, 287, 66060-902,
Belém – PA – Brasil
Fone: (+55 91)40093000
e-mail: sandro@cci.unama.br

Alexandre Marcos Lins de Vasconcelos
Universidade Federal de Pernambuco - UFPE
Centro de Informática
Caixa Postal 7851, 50732-970
Recife – PE – Brasil
Fone: (+55 91)21268430
e-mail: amlv@cin.ufpe.br

RESUMO

Ambiente de Engenharia de Software Centrado no Processo tem como um dos seus propósitos prover que as fases do ciclo de vida de um processo de software (definição, simulação, execução e avaliação) possam ser automatizadas. Na fase de Definição do Processo é feita uma análise das características organizacionais, de projetos e de produtos de software para a composição do processo. Assim, faz-se necessário a existência de métodos e padrões que provejam a publicação deste processo de software a ser difundido e ajude na sua institucionalização. Este trabalho apresenta a estrutura das duas formas de publicação providas pelo ImPProS [13] a fim de suportar esta necessidade.

PALAVRAS-CHAVE: Processo de Software, Ambiente de Desenvolvimento de Software, XML, Documentação.

ABSTRACT

PSEE - Process-centered Software Engineering Environment has one of its intentions to provide that phases of the software process life cycle (definition, simulation, enacting and evaluation) can be automatized. In Process Definition phase an analysis of organizational, software projects and products characteristics for composition of process is made. This way, it becomes necessary the existence of methods and patterns that provide the documentation this software process to be spread out and helps in its institutionalization. This paper presents the structure of two documentation ways provided by ImPProS [13] in order to support this need.

KEYWORDS: Software Process, Software Development Environment, XML, Documentation.

1. INTRODUÇÃO

A Tecnologia de Processo de Software surgiu em meados da década de 1980 e representou um importante passo em direção à melhoria da qualidade de software através de mecanismos que proporcionam o gerenciamento automatizado do desenvolvimento de software. Diversas teorias, conceitos, formalismos, metodologias e ferramentas surgiram nesse contexto, enfatizando a descrição formal do modelo de processo de software, para que possa ser automatizado por um ambiente integrado de desenvolvimento de software [15]. Os modelos de processos de software descrevem o conhecimento de uma organização e, portanto, modelos que descrevem experiências bem sucedidas devem ser continuamente disseminados em diferentes projetos a partir de um meio que permita a sua publicação. Apesar da importância desse tópico, atualmente apenas uma pequena porção do conhecimento produzido durante o desenvolvimento de software é documentado para ser usado em novos projetos.

A prática da documentação de processos de software é uma técnica que tem sido, ao longo do tempo, o “calcanhar de Aquiles” de muitas organizações de Tecnologia da Informação. Tão distante para atingir quanto para manter a documentação. O que parece ser um sonho impossível. Da perspectiva da Tecnologia da Informação, realizar a documentação é fator crítico para o sucesso. Processos devem ser bem documentados, acessíveis e repetitivos. Serviços e recursos devem ser reusáveis ou compartilháveis [17].

A investigação conduzida nesse trabalho tem como intuito principal aumentar o nível de automação fornecido na publicação de processos de software, apoiando a fase de definição do processo de software e a sua disseminação para todos os projetos organizacionais.

Assim, o objetivo deste trabalho recai no estabelecimento das estruturas de documentação do processo de software; e objetivos e importâncias destas estruturas no contexto da implementação de um processo de software. Tudo isso deve ser aliado à definição do processo por níveis de caracterização definida ao ambiente de implementação de processo de software, já que este permite uma definição do mesmo de forma particionada.

Este ambiente de implementação de processo de software, proposta por [13], tende a possibilitar a especificação dos processos de acordo com o domínio do projeto específico e das características de organização; a instanciação do processo de software para propriedades dos projetos; sua simulação a partir dos parâmetros de configuração (prazo, pressões, custo, recursos, etc.); uma execução mais próxima do que se espera para um processo organizacional; e uma avaliação a partir da coleta de métricas desta execução.

Além desta seção introdutória, o artigo apresenta outras cinco seções. A seção 2 aborda as propriedades que compõem o ambiente de implementação de processo de software proposto por [13]. Na seção 3 tem-se o detalhamento da fase de definição de processos de software neste ambiente. A seção 4 especifica a estrutura do processo adotada pelo ambiente. Na seção 5 encontramos a descrição das duas formas de publicação de processos de software definidas ao ambiente. Finalmente, a seção 6 apresenta as considerações finais deste artigo.

2. ImPProS: UM AMBIENTE DE IMPLEMENTAÇÃO PROGRESSIVA DO PROCESSO DE SOFTWARE

Uma evolução significativa nos ADSs foi detectada com a tecnologia de processos de software. A automação do processo de software foi incorporada aos ADSs mais recentes tornando-os ADSs centrados em processo (ou orientados a processo), também conhecido na literatura como PSEE - Process-Centered Software Engineering Environment [7]. Estes ambientes constituem uma nova geração de ADS que suportam além da função de desenvolvimento de software, também as funções associadas de gerência e garantia da qualidade durante o ciclo de vida do software.

No entanto, em alguns casos, percebe-se ao longo da execução do processo a partir destes ambientes de desenvolvimento que sua implementação nem sempre perfaz a realidade das características da organização ou do projeto desenvolvido por esta. Assim, para ajudar uma organização na implementação progressiva de um processo de software, é útil fornecer apoio automatizado por meio de um ambiente capaz de suportar as fases que a literatura

especializada propõe como necessárias. O termo “progressiva” decorre do fato de que a implementação do processo é aperfeiçoado com as experiências aprendidas na sua definição, simulação, execução e avaliação.

O ambiente proposto em [13], como projeto de Tese de Doutorado do CIn/UFPE e apoiado pela UNAMA, está sendo concebido (atualmente em desenvolvimento) com o objetivo principal de apoiar a implementação de um processo de software em uma organização. Dentro deste contexto podem ser caracterizados como seus objetivos específicos:

- Especificar um meta-modelo de processo de software a fim de definir uma terminologia única entre os vários modelos de qualidade de processo de software existentes, para uso do ambiente em seus serviços providos;
- Apoiar a definição de um processo de software para organização;
- Permitir a modelagem e instanciação deste processo;
- Permitir a simulação do processo a partir das características instanciadas para um projeto específico;
- Dar apoio à execução do processo de software tomando como base uma máquina de inferência;
- Possibilitar a avaliação dos critérios do processo de software;
- Apoiar a melhoria contínua do processo de software e o reuso através da realimentação e coleta das experiências aprendidas.

Vale ressaltar que todos os objetivos listados acima foram adaptados a partir da estrutura que compõe o meta-processo de software descrito em [15], das características propostas para a implementação de um processo de software [2] e do ciclo de vida para melhoria contínua de processo definido pelo Modelo IDEAL [12]. Para alcançar estes objetivos o ambiente foi concebido para adotar a arquitetura apresentada na Figura 1. Os componentes definidos na

arquitetura do ambiente encontram-se resumidamente descritos nas subseções a seguir e seu detalhamento pode ser encontrado em [13].

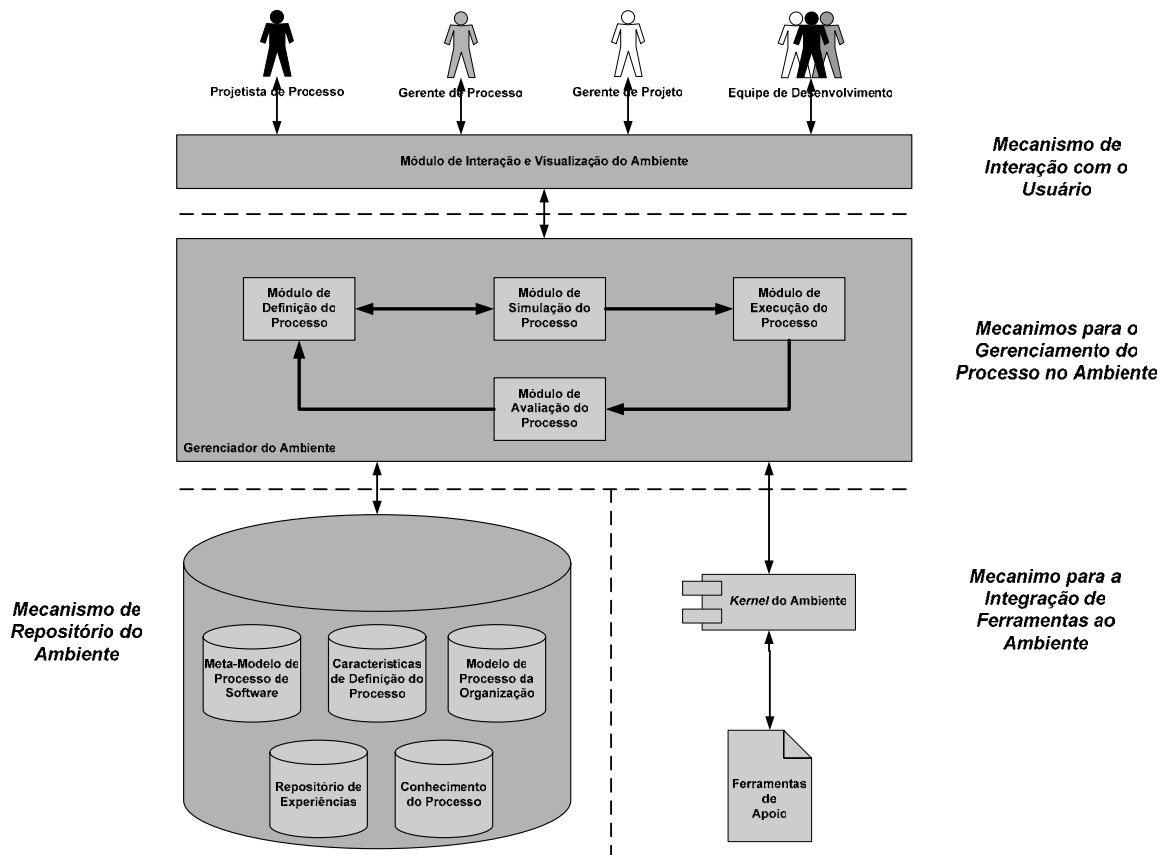


Figura 1. Arquitetura do ImPProS [13]

3. DEFINIÇÃO DO PROCESSO DE SOFTWARE NO ImPProS

Nesta seção é apresentado um detalhamento de todos os serviços providos pelo ImPProS no que tange a definição e melhoria de processos de software. Na definição de processos de software do ImPProS, visto na Figura 2, adaptada do modelo definido por [16], inicialmente encontra-se o Meta-modelo de processo de software, composto de componentes e dos relacionamentos entre esses que são oriundos do mapeamento de algumas normas e modelos de qualidade para processo de software (CMMI [4], SPICE – ISO 15504 [8], ISO 12207 [9], etc.).

O objetivo deste meta-modelo é determinar uma terminologia única para a definição de processos de software no ImPProS. Vale ressaltar que a estrutura do meta-modelo foi definida de forma a contemplar todos os componentes de um processo de software (processo,

atividades, artefatos, etc.), mas não restringir a sua composição para algumas normas/modelos de processo de software (comum nos trabalhos relacionados na seção anterior), ou seja, dependendo da norma/modelo a ser usada, o usuário pode fazer o mapeamento da mesma usando como base a terminologia da ISO/IEC 12207 e definir os seus processos a partir do uso deste novo meta-modelo.

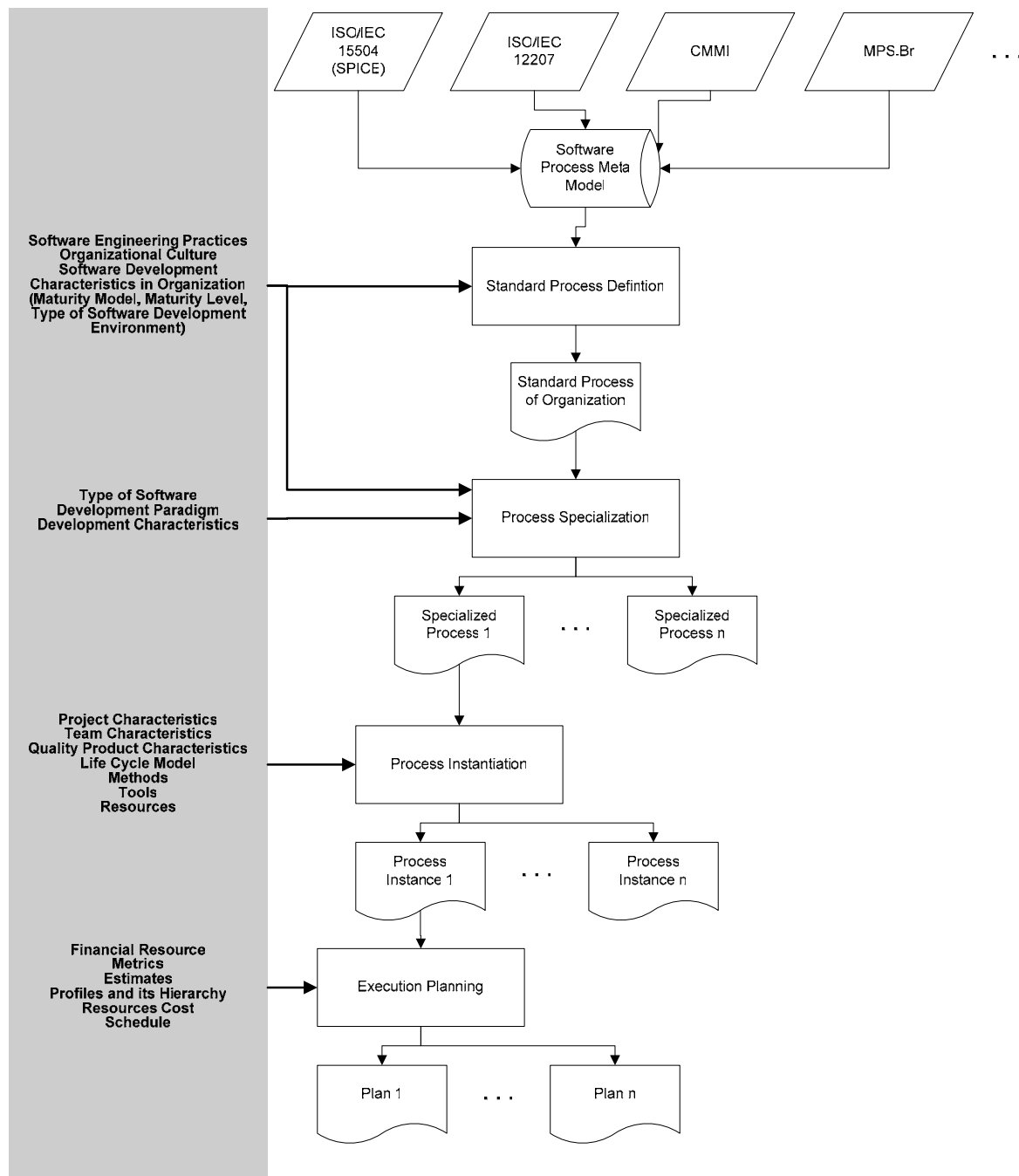


Figura 2. Estrutura de Definição do Processo de Software no ImPProS [13]

Por sua vez, a definição de um processo padrão estabelece uma estrutura comum a ser utilizada pela organização nos seus projetos de software e constitui a base para a definição de

todos os seus processos. Dessa forma, estabelece-se um processo básico que servirá como ponto de partida para a posterior definição dos processos de software adequados às diferentes características de cada projeto, permitindo economia de tempo e esforço na definição de novos processos.

Tendo em vista que tipos de software diferentes possuem características distintas e requerem diferentes abordagens de desenvolvimento, o processo de software padrão da organização deverá ser adaptado (especializado) considerando-se as características relacionadas ao tipo de software (por exemplo, sistemas de informação) e ao paradigma de desenvolvimento utilizado (por exemplo, orientação a objetos). Assim, durante a etapa de especialização do processo padrão, atividades poderão ser adicionadas ou modificadas, de acordo com o contexto para qual se está realizando a especialização.

A instanciação para projetos específicos consiste na adaptação de um processo especializado a um projeto, considerando-se as suas peculiaridades. Nesta etapa, são definidos o modelo de ciclo de vida, os métodos e as ferramentas que serão utilizadas no projeto, os recursos humanos e suas responsabilidades ao longo do processo e os artefatos (produtos) consumidos e gerados. A norma ISO/IEC 9126 [10] é utilizada para identificar os requisitos de qualidade do produto. Tais características irão influenciar o processo no que se refere a atividades, métodos e técnicas. As atividades do processo especializado deverão ser adaptadas ao modelo de ciclo de vida escolhido para o projeto e novas atividades poderão ser inseridas em um determinado ciclo de vida. Assim, como resultado desta fase é gerada uma instância do processo contendo os componentes necessários para o processo de software a fim de representar o projeto específico a ser desenvolvido, atendendo desta forma todas as características deste desenvolvimento.

O ImPProS propôs, a estrutura da Figura 2, três frentes de contribuição que, a partir de análises feitas em definições de processo cotidianamente e o que a literatura especializada propõe como uso, aperfeiçoaram a especificação do processo de software nos três níveis

definidos e possibilitaram com que os componentes definidos ao processo pudessem ser simulados a fim de antever problemas na execução do mesmo pela equipe do projeto, a saber:

- Inclusão de um conjunto de novas características organizacionais, de projetos de software e de produtos de software;
- Sugestão de componentes de processo de software a partir de definições de processos anteriormente feitas e conhecimentos aprendidos ao longo destas definições;
- Nível de Planejamento do Processo Instanciado para que este possa servir como base para a simulação deste processo a um projeto específico.
- A definição do plano de execução de um processo instanciado consta de uma especificação de algumas características do planejamento do processo que possibilitem a sua prévia execução.

Desta forma, o usuário pode modelar um ou mais planos para verificar como o processo instanciado se comporta a partir das características de execução de um projeto específico: recurso financeiro; métricas; estimativas; cargos e sua hierarquia; custo; e cronograma. Estes atributos serão parametrizados para a ferramenta de simulação analisar a execução do processo de software instanciado.

Estas três etapas de definição do processo (definição do processo padrão, especialização e instanciação do processo) consistem em adaptar o meta-modelo de processo para atender a objetivos específicos, através da análise das características organizacionais e de projetos de software.

4. ESTRUTURA PADRÃO DO PROCESSO DE SOFTWARE DO ImPProS

O ImPProS por possuir como uma de suas características a definição do processo de software sob a forma de um modelo de processo de software e sua representação diagramática, possui como estrutura geral de composição dos processos de software o modelo baseado nas definições de ontologias de processo de software de Falbo [6].

Processos são coleções de atividades relacionadas que têm lugar durante o desenvolvimento de um produto. Basicamente, um processo consiste de um conjunto estruturado de atividades e, por conseguinte, toda a infra-estrutura envolvida na realização destas (artefatos, procedimentos e recursos). Por sua vez **Atividades** são as tarefas ou trabalhos a serem realizados. Uma atividade requer recursos e pode consumir ou produzir artefatos. Para sua realização, uma atividade pode adotar um procedimento. Uma atividade pode ser decomposta em outras atividades. Além disso, atividades, em qualquer nível, podem depender da finalização de outras atividades, denominadas pré-atividades. O conceito de atividade está presente em todos os modelos de processo de software e são consideradas primitivas que geram artefatos a partir de artefatos de entrada auxiliados por recursos. Atividades podem corresponder a diferentes níveis, seja uma tarefa ou uma etapa do processo de desenvolvimento.

Para descrever as etapas do processo e as atividades a serem realizadas em cada etapa, surge o conceito de **Modelo de Ciclo de Vida**, que estrutura atividades e define abordagem de como organizar um projeto em fases. O ciclo de vida é iniciado quando um software é concebido até quando entra em desuso, ou seja, contém um conjunto de atividades de desenvolvimento, operação e manutenção. Aliado a este conceito temos a **Combinação**, a qual define a forma como um conjunto de fases de um modelo de ciclo de vida deve ser realizado e especifica o tipo de ordenação em que a estrutura pode ser: seqüencial ou iterativa.

Os **Artefatos** são produtos de software produzidos ou consumidos por atividades durante a sua realização. São exemplos de artefatos: manuais de qualidade, manuais de revisão, diagramas de fluxos de dados, diagramas de objetos, código fonte, etc. Um artefato pode ser decomposto em outros artefatos (composição de artefatos). É a entrada ou produto de uma atividade, podendo ser artefatos de código, documentos ou componentes de software.

Já os **Procedimentos** são condutas bem estabelecidas e ordenadas para a realização de atividades. Alguns procedimentos podem ser parcialmente automatizados por ferramentas de software. São utilizados para auxiliar a realização das atividades, podendo ser direcionados a

um tipo específico de atividade, devendo ser adequados a uma tecnologia de desenvolvimento e a um paradigma. Aliado a este conceito, temos o **Padrão de Atividades** que um procedimento deve sugerir para a execução de uma atividade. Representa um comportamento em que decomposições de uma atividade têm em comum.

As pessoas, as ferramentas de software, os equipamentos, ou quaisquer outras infra-estruturas necessárias à execução de uma atividade, recebem o nome de **Recurso**. Um recurso humano, especificamente, desempenha um papel na execução das atividades do processo. São elementos necessários para a realização de uma atividade, tais como agentes humanos, equipamentos de hardware e ferramentas de software. Apóiam ou atuam na realização da atividade, mas não podem ser consideradas “matérias-primas” para a atividade, ou seja, apenas auxiliam o processo, mas não são incorporados ao produto de software sendo considerados recursos para a atividade.

Podemos encontrar, ainda, os conceitos de Paradigma de Desenvolvimento, que são princípios e conceitos que orientam o desenvolvimento (por exemplo: o estruturado e o orientado a objetos), e a Tecnologia de Desenvolvimento que representa a tecnologia a ser empregada no desenvolvimento do software (é o caso das tecnologias convencional de processamento de dados e de sistemas baseados em conhecimento). Por fim, para limitar e/ou restringir a execução das atividades definidas no processo, tem-se as Restrições, que especificam as regras de definição do processo de software.

5. FORMAS DE PUBLICAÇÃO DO PROCESSO DE SOFTWARE NO ImPProS

Com o objetivo de solucionar os problemas da indústria de software mundial, várias tecnologias vêm sendo experimentadas no sentido de apoiar o ciclo de vida do software. Um dos esforços mais significativos corresponde à definição de metodologias voltadas a disciplinar o processo de desenvolvimento através do estabelecimento de etapas bem definidas, proporcionando, desta forma, um mecanismo de controle para o processo [15]. Como consequência, muitas organizações de desenvolvimento de software buscam a

maturidade no processo de software, usando medidas como o SEI-Capability Maturity Model Integration (CMMI) para estruturar as iniciativas de melhoria de processo.

O modelo CMMI propõe cinco estágios, sendo que uma organização de desenvolvimento de software pode se encaixar desde o nível inicial, onde o processo é caótico, até o nível otimizado, onde o gerenciamento de software é ideal. Nesse modelo, algumas áreas-chave são identificadas para que uma organização passe de um nível ao outro do modelo, assumindo um papel importante na verificação da qualidade do software produzido. Uma das áreas-chave determinadas pelo CMMI, para que uma organização possa obter aumento na maturidade dos seus processos, consiste na definição rigorosa de processos que determinem as estratégias gerenciais adotadas durante o desenvolvimento de qualquer produto de software pela organização.

Em face da crescente utilização de modelos como o CMMI para avaliar a qualidade do software produzido e da inerente complexidade em gerenciar o seu processo, a comunidade internacional investiu na construção de ferramentas e ambientes que fornecem apoio automatizado à condução desses projetos [15]. Essa iniciativa evoluiu para o estabelecimento de uma nova área de pesquisa e aplicação da Ciência da Computação denominada Tecnologia de Processo de Software, a qual também é influenciada pela crescente utilização de ferramentas CASE (Computer-Aided Software Engineering) nas diferentes etapas do processo.

Todo software é produzido através de algum processo. Entretanto, na maioria das vezes este processo é incoerente e está implícito, levando à falta de previsibilidade, falta de capacidade de repetição dos passos e falta de uma base que permita aperfeiçoamento. Para resolver tais problemas são necessários processos explícitos e coerentes que possam ser seguidos consistentemente, monitorados, compreendidos e evoluídos.

A produção de software envolve atividades complexas desempenhadas por pessoas com capacidades, experiências e expectativas diferenciadas. Uma forma de analisar e amadurecer

tal processo é através da sua descrição, a qual consiste de um modelo de processo de software. A descrição formal de um processo de software é uma atividade que permite que o mesmo seja analisado, compreendido, automatizado (executado) e melhorado.

Um **modelo de processo de software** é uma descrição abstrata do processo de software. Vários tipos de informação devem ser integradas em um modelo de processo de software para indicar quem, quando, onde, como e por que os passos são realizados. Para representar um modelo de processo de software, a literatura especializada define que podem ser usadas diferentes formas de publicação acerca de sua composição:

- Uso de uma **linguagem de modelagem do processo de software**, a qual deve oferecer recursos para descrever e manipular os passos do processo;
- **Arquivos em forma de documentos** que descrevam detalhadamente como o processo foi definido e quais os componentes que o especificam;
- Manipulação de uma **linguagem de marcação**, como o XML (*eXtensible Markup Language*) [11], que permita com que os componentes do processo possam ser padronizadas em *tags* possibilitando a integração de dados em fontes diferentes, múltiplas formas de visualizar os dados, definição de um padrão aberto e atualizações granulares dos documentos.

O ImPProS faz uso de uma linguagem de modelagem que permite com que seus processos de software possam ser representados diagramaticamente, através das notações propostas pelo SPEM – Software Process Engineering Metamodel [14]. O SPEM é um meta-modelo que pode ser usado para descrever um processo concreto de desenvolvimento de software ou uma família de processos relacionados. O SPEM adota uma abordagem orientada a objetos para modelar processos e usa UML – *Unified Modeling Language*, como notação. A escolha do SPEM, em face a outros modelos gráficos de representação de processos, se dá pela importância que o mesmo vem tomando no mercado da engenharia de software atual. Um estudo e aplicação de uma instância do SPEM no ImPProS pode ser melhor detalhada em [1].


Este trabalho detalhará a apresentação das duas outras formas de publicação de processos de software no ImPProS: o uso de arquivos em forma de documentos e a manipulação de uma linguagem de marcação. As seções a seguir apresentam a automação destas duas formas no ImPProS.

5.1. PUBLICAÇÃO DE PROCESSO NO ImPProS A PARTIR DE DOCUMENTOS

O processo de software é um conjunto de trabalhos e resultados associados. Uma forma de tornar o processo visível é através de documentos. Esses documentos são produzidos e/ou fornecidos pelos diferentes agentes envolvidos no processo de desenvolvimento. Se os agentes desenvolvedores documentarem o processo, então há necessidade de consistência e ordenamento entre os documentos produzidos.

Da mesma forma, podemos destacar a importância em documentar todos os componentes que perfazem um processo de software, conforme definidos na seção 4, pelo fato de auxiliar a comunicação e o entendimento de todas as partes envolvidas. No ImPProS, por possuir uma estrutura de definição do processo de software baseada em 3 (três) níveis (definição do processo padrão, especialização do processo e instanciação do processo), foi definido três *templates* (documento formatado que o usuário vai preencher com as suas informações, orientando a redação de um trabalho e caracterizando a estrutura do mesmo) que abrigassem todas as informações especificadas ao longo de sua definição, o que permite uma melhor organização e entendimento dos processos, possibilitando a adaptação dos mesmos às características específicas (vistas na Figura 2), por exemplo, de cada projeto e produto de software ou organização.

Estes documentos, de uma forma geral, vistos na Figura 3, estão organizados em três partes principais, como sempre detalhados em documentos que descrevem um processo de software organizacional. Após a Figura 3 cada uma destas partes que compõem o documento encontra-se detalhada.



ImPProS

<Nome do Processo>

<Descrição do Processo>

Versão <Número>

<Nome da Organização>

<Nome do Processo>

Processo Especializado

Versão <Número>

Data da Versão: <Data de Publicação>

ImPProS

Confidencial

<Nome da Organização>

Página 2 de 6

<Nome da Organização>

<Nome do Processo>

Processo Especializado

Versão <Número>

Data da Versão: <Data de Publicação>

Histórico de Revisões

Data	Versão	De origem	Autor
<Data da Publicação>	<Número>	<De origem>	<Autor>

ImPProS

Confidencial

<Nome da Organização>

Página 2 de 6

<Nome da Organização>

<Nome do Processo>

Processo Especializado

Versão <Número>

Data da Versão: <Data de Publicação>

Conteúdo

I - Introdução

II - Dados do Processo

III - Características da Especialização do Processo

IV - Visão Geral do Processo

<No.>

<No.>

<No.>

<No.>

ImPProS

Confidencial

<Nome da Organização>

Página 3 de 6

<Nome da Organização>

<Nome do Processo>

Processo Especializado

Versão <Número>

Data da Versão: <Data de Publicação>

I - Introdução

Este documento compoende o procedimento especializado utilizado para o desenvolvimento de produtos de software na <Nome da Organização>, o <Descrição do Processo> (<Nome do Processo>), o qual adotado como linha-mestra por todos os membros envolvidos no desenvolvimento do projeto de software <Nome do Projeto de Software> da organização.

1.1 - Histórico

A qualidade sempre foi um item desejável para a garantia do sucesso de uma organização no mercado. Com a globalização e consequente crescimento da competitividade, surgiu uma necessidade muito mais forte de equiparação das empresas nacionais com padrões internacionais, já que os produtos passaram a ser produzidos para consumo mundial.

Alinhado as transformações pelas quais o mercado mundial vem passando, o corpo estratégico da <Nome da Organização> identificou que um projeto de melhoria organizacional era primordial para o crescimento e fortalecimento da empresa quanto ao cenário de desenvolvimento.

O <Descrição do Processo> representa uma das iniciativas deste projeto estratégico, que por sua vez compoende ações de melhoria em todos os processos da organização.

1.2 - Visão Geral deste Documento

Este documento está dividido em 4 seções. Além desta seção introdutória, a Seção 2 aborda algumas informações de projeto e implementação do processo; a Seção 3 detalha as características analisadas para a implementação do processo definido; a Seção 4 compoende uma visão geral do <Nome do Processo>.

II - Dados do Processo

Organização: <Nome da Organização>

Nome do Processo Padrão: <Nome do Processo Padrão>

Projeto de Software: <Nome do Projeto de Software>

Descrição do Projeto de Software: <Descrição do Projeto de Software>

Nome do Processo Especializado: <Nome do Processo Especializado>

Descrição do Processo: <Descrição do Processo Especializado>

Nível de Definição: Especialização do Processo Padrão

Situação de Definição: <Situação de Definição>

ImPProS

Confidencial

<Nome da Organização>

Página 4 de 6

Figura 3. Template do arquivo usado para documentar o processo de software no ImPProS

- **Apresentação do Documento:** parte do documento que abriga uma prévia especificação do nome do processo, sua descrição sucinta e a versão definida ao documento;

- **Histórico de Revisões:** parte que contempla a relação de revisões que o documento foi posto em prática, contendo a data da revisão, a versão do documento daquela revisão, uma breve descrição da revisão gerada e o(s) autor(es) desta revisão;
- **Conteúdo:** contempla o índice das seções que descrevem o processo de software, os quais estão no item a seguir;
- **Descrição do Processo:** parte formada por quatro seções que permite com que os responsáveis possam fazer uma breve introdução acerca da necessidade do processo de software; dos dados de geração do processo (Nome da Organização, Nome do Processo, Data da Definição, etc.); descrição detalhada das características que serviram como base para a definição do processo de software no ImPProS (um melhor detalhamento destas características pode se vista em [3][5]); e a visão geral do processo que contempla todos os componentes do processo de software, de acordo com o nível de sua definição. Esta parte apresenta dados diferentes de acordo com o nível de definição do processo de software, conforme pode ser visto na Tabela 1.

Tabela 1. Conteúdo da Descrição do Processo a partir do Nível de Definição do Processo

Seção do Documento / Nível de Definição	Definição do Processo Padrão	Especialização do Processo	Instanciação do Processo
I – Introdução	I.1 – Histórico I.2 – Visão Geral deste Documento		
II – Dados do Processo	Organização; Nome do Processo Padrão; Descrição do Processo; Nível de Definição; Situação da Definição; Data do Início da Definição; Responsável pela Definição; Equipe de Definição.		
		Projeto de Software; Descrição do Projeto de Software; Nome do Processo Especializado.	Nome do Processo Especializado; Projeto de Software; Descrição do Projeto de Software; Nome do Processo Instanciado.
III – Características do Processo	Tipo de Organização; Norma/Modelo de Maturidade ou de Referência; Nível de Maturidade; Características da Organização.		
		Tipo(s) de Projeto de Software; Paradigma(s) de	Características Relacionadas aos Usuários; Características

		Desenvolvimento; Tecnologia(s) de Desenvolvimento.	Relacionadas ao Problema; Características Relacionadas ao Produto; Características Relacionadas aos Recursos; Características Relacionadas à Equipe de Desenvolvimento; Características Relacionadas ao Desenvolvimento; Nível de Garantia de Qualidade do Produto; Características da Qualidade do Produto.
IV – Visão Geral do Processo	Nome do Processo; Descrição; Origem; Informações de Implementação; Restrição; Situação da Definição; Nome da Atividade; Descrição da Atividade; Tipo da Atividade; Origem da Atividade; Restrição da Atividade; Artefatos de Entrada da Atividade; Artefatos de Saída da Atividade; Procedimentos da Atividade; Recursos de Hardware da Atividade; Recursos de Software da Atividade; Recursos de Humano da Atividade.		
			Super-Atividade(s) da Atividade; Sub-Atividade(s) da Atividade; Pré-Atividade(s) da Atividade; Pós-Atividade(s) da Atividade.

5.2. PUBLICAÇÃO DE PROCESSO NO ImPProS A PARTIR DE XML

A outra solução para a publicação de processos de software apresentada pelo ImPProS é através do uso do XML. O *eXtensible Markup Language* (XML) é uma linguagem de marcação de dados (meta-markup language) que provê um formato para descrever dados estruturados. Isso facilita declarações mais precisas do conteúdo e resultados mais significativos de busca através de múltiplas plataformas. O XML também permite o surgimento de uma nova geração de aplicações de manipulação e visualização de dados via internet. O XML permite a definição de um número infinito de *tags* para dados estruturados. A mais importante característica do XML se resume em separar a interface com o usuário

(apresentação) dos dados estruturados, o que permite visualizar e processar o dado como quiser, utilizando diferentes folhas de estilo e aplicações.

No XML as regras que definem um documento são ditadas por DTDs (Document Type Definitions), as quais ajudam a validar os dados quando a aplicação que os recebe não possui internamente uma descrição do dado que está recebendo. Mas os DTDs são opcionais e os dados enviados com um DTD são conhecidos como dados XML válidos. Um analisador de documentos pode checar os dados que chegam analisando as regras contidas no DTD para ter certeza de que o dado foi estruturado corretamente. Os dados enviados sem DTD são conhecidos como dados bem formatados. Nesse caso, o documento pode ser usado para implicitamente se auto-descrever.

Com os dados XML válidos e com os bem-formatados, o documento XML se torna auto-descritivo porque as *tags* dão idéia de conteúdo e estão misturadas com os dados. Devido ao formato do documento ser aberto e flexível, ele pode ser usado em qualquer lugar onde a troca ou transferência de informação é necessária.

Assim sendo, como a composição da estrutura de um processo de software é dinâmica e possui diferentes tipos de componentes (atividades, recursos, procedimentos, etc.), como vistos na seção 4, descrevemos, com base no conteúdo apresentado por cada nível de definição do processo de software, como definido na Tabela 1, 3 (três) regras de validação dos componentes gerados ao processo de software no ImPProS. A Figura 4 apresenta o arquivo DTD para os níveis de definição do processo (definição do processo padrão e especialização do processo, e instanciação do processo), de forma agrupada, que possuem os elementos que devem estar presentes no XML e bem como os atributos, os relacionamentos e a multiplicidade existente entre estes elementos.

```

<!-- DTD para representar a Estrutura do Processo Padrão, Processo Especializado e Processo Instanciado no
ImPProS -->

<?xml version="1.0" encoding="UTF-8" ?>

<!DOCTYPE ProcessoInstanciadoImpproS [

<!ELEMENT processoinstanciado (processo*, modelociclovida)>
<!ATTLIST processoinstanciado
    nome      CDATA #REQUIRED
    descricao PCDATA #REQUIRED
    nivel     CDATA #FIXED "Processo Instanciado"
    situacao  CDATA #REQUIRED
    organizacao PCDATA #REQUIRED
    version   CDATA #REQUIRED
    data      CDATA #REQUIRED>

<!ELEMENT processo (atividade*)>
<!ATTLIST processo
    nome      CDATA #REQUIRED
    descricao PCDATA #REQUIRED
    origem    CDATA #REQUIRED
    infoimpl  PCDATA #REQUIRED
    restricao  PCDATA #REQUIRED
    situacao  CDATA #REQUIRED>

<!ELEMENT modelociclovida (fase*)>
<!ATTLIST modelociclovida
    nome      CDATA #REQUIRED
    tipo      CDATA #REQUIRED>

<!ELEMENT fase (atividade*)>
<!ATTLIST fase
    nome      CDATA #REQUIRED
    ordem     CDATA #REQUIRED
    niteracoes CDATA #REQUIRED>

<!ELEMENT atividade (superatividade, subatividade*, preatividade*, posatividade*, artefatoentrada*,
artefatosaida*, procedimento*, recursohardware*, recursossoftware*, recursohumano*)>
<!ATTLIST atividade
    nome      CDATA #REQUIRED
    descricao PCDATA #REQUIRED
    tipo      CDATA #REQUIRED
    origem    CDATA #REQUIRED
    restricao  PCDATA #REQUIRED>

<!ELEMENT superatividade CDATA>
<!ELEMENT subatividade CDATA>
<!ELEMENT preatividade CDATA>
<!ELEMENT posatividade CDATA>
<!ELEMENT artefatoentrada CDATA>
<!ELEMENT artefatosaida CDATA>
<!ELEMENT procedimento CDATA>
<!ELEMENT recursohardware CDATA>
<!ELEMENT recursossoftware CDATA>
<!ELEMENT recursohumano CDATA>

]>

```

Figura 5. DTD do Conteúdo proveniente da Definição do Processo Padrão, Especialização do Processo e Instanciação do Processo

6. CONSIDERAÇÕES FINAIS

Publicação do processo de software pode ser definida como uma documentação dos trabalhos de desenvolvimento de um projeto de software cuja finalidade seja comunicar a informação sobre o sistema de software ao qual ele pertence [17]. A documentação tem fundamental importância para a Engenharia de Software. Vários estudos têm sido realizados para minimizar os problemas em torno da documentação de software: documentação desatualizada e de baixa qualidade, processos de documentação dispendiosos e caros, documentação em abundância e sem propósito, dificuldade de acesso, entre outros.

Pode-se notar, ainda, que na literatura especializada os estudos sobre documentação de software visam solucionar problemas de falta de atualização, dificuldade de acesso, falta de qualidade, desorganização.

No tocante à tecnologia de processo de software, ponto-chave deste trabalho, percebe-se que as etapas para concepção e especificação de um processo de software são de extrema relevância para que o mesmo possa inferir a qualidade organizacional e dos produtos e/ou serviços por ela oferecidos. Mas estes procedimentos se não estiverem aliados a uma boa forma de publicar/documentar o processo de software acabam por não ajudar na compreensão das informações constantes e a sua disseminação.

Os dados desta pesquisa são úteis para demonstrar duas prováveis formas de publicar um processo de software, sem a necessidade da definição do mesmo ser automatizada. No contexto do ImPProS especificou-se o uso de duas maneiras: arquivos em forma de documentos; linguagem de marcação. A partir das concepções das estruturas de XML feitas e *templates* de documentos gerados, automatizou-se o seu uso a partir do ambiente e toda a publicação proveniente da definição do processo de software pode ser feita por ambas as formas.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] ARAÚJO, R. M., **Construção Gráfica de Processos de Desenvolvimento e Geração de uma Ontologia de Processo de Software**, Orientador Prof. Alexandre Vasconcelos,

- Trabalho de Graduação, CIn/UFPE, 2005.
- [2] BALDUIO, R., **Implementação de um processo de desenvolvimento de software: uma abordagem passo-a-passo**, Rational Software White Paper, 2002.
 - [3] BARBOSA, I. M., **Análise de Características de Projetos de Software para a Definição de Processo de Software**, Trabalho de Graduação apresentado ao CIn/UFPE, orientador Prof. Alexandre Vasconcelos, Recife-PE, 2005.
 - [4] CHRISSIS, M. B., KONRAD, M. and SHRUM, S., **CMMI Guidelines for Process Integration and Product Improvement**, Addison-Wesley, 2003.
 - [5] CUNHA, M. B. F. L., **Análise das Características Organizacionais para a Definição de Processo de Software**, Trabalho de Graduação apresentado ao CIn/UFPE, orientador Prof. Alexandre Vasconcelos, Recife-PE, 2005.
 - [6] FALBO, R. A., **Integração de Conhecimento em um Ambiente de Desenvolvimento de Software**, Orientadora Ana Regina Cavalcanti da Rocha, Tese de Doutorado. COPPE/UFRJ, 1998.
 - [7] HUMPHREY, W. S., **Managing the Software Process**, New York: Addison-Wesley, 1989.
 - [8] ISO/IEC TR 15504, Parts 1-9, **Information Technology – Software Process Assessment**, International Organization for Standardization, 1998.
 - [9] ISO/IEC TR 12207, **Amendment: Information Technology – Amendment to ISO/IEC 12207**, PDAM 3 version, 2000.
 - [10] ISO/IEC TR 9126, **Software engineering – product quality**, International Organization for Standardization, 2002.
 - [11] MARCHAL, Bernoit, **XML by Example**, Que, 2001.
 - [12] MCFEELEY, B., **IDEALSM: A User's Guide for Software Process Improvement**, Software Engineering Institute Handbook. Carnegie Mellon University. CMU/SEI-96-HB-001, 1996.
 - [13] OLIVEIRA, S., Vasconcelos, A., Rouiller, A. C., **Uma Proposta de um Ambiente de Implementação de Processo de Software**, Artigo publicado na Revista InfoComp – Revista de Ciência da Computação da UFLA – vol. 4, n. 1, Lavras-MG, 2005
 - [14] OMG – Object Management Group, **SPEM – Software Process Engineering Metamodel Specification**, version 1.1, formal/05-01-06, 2005
 - [15] REIS, C. A. L., **Ambientes de Desenvolvimento de Software e seus Mecanismos de Execução de Processos de Software**, Orientador Daltro Nunes. Exame de Qualificação do Doutorado. PPGC–UFRGS, 2000.
 - [16] ROCHA, A. R. C., MALDONADO, J. C. and WEBER, K. C., **Qualidade de software: teoria e prática**, São Paulo: Prentice-Hall, 2001.
 - [17] SOUZA, S. C. B. ; NEVES, W. C. G. ; ANQUETIL, N. ; OLIVEIRA, K. M., **Documentação essencial para manutenção de software II**, In: 1o Workshop de Manutenção de Software Moderna, 2004, Brasília. Anais do 1o Workshop de Manutenção de Software Moderna, 2004.