

Os Padrões ISO/IEC 12207 e 15504 e a Modelagem de Processos da Qualidade de *Software*

Carlos Lahoz
Instituto de Aeronáutica e Espaço (IAE)
Centro Técnico Aeroespacial (CTA)
lahoz@iae.cta.br

Nilson Sant'Anna
Instituto Nacional de Pesquisas Espaciais
(INPE)
nilson@lac.inpe.br

Abstract

Nowadays, in the software engineering community, it is an general consensus that improving the software development processes, we improve the quality of the generated products. These processes must be clearly understood and well defined so that they can be implemented in an organization that design and develop software. This work presents a Support processes study, proposed by ISO/IEC12207 and 15504 standards, for the definition, modeling and implantation of the Software Quality processes in process centered environments. The modeling language SPEM - Software Process Engineering Metamodel - is used to express these processes models.

1. Introdução

Uma das áreas de pesquisa de destaque atual da Engenharia de *Software* é a do estudo e melhoria dos processos pelos quais o *software* é construído. De modo geral, processo de *software* é a denominação do conjunto de políticas, estruturas organizacionais, tecnologias, procedimentos e artefatos necessários para conceber, desenvolver, construir e manter um produto de *software* [2], tendo impacto direto na qualidade dos produtos gerados.

A gestão da qualidade, neste contexto, tem um papel atuante na definição de processos que possibilitem que os produtos gerados tenham um nível de qualidade aceitável e de forma a extrair informações em todo o desenvolvimento dos produtos. A qualidade deve buscar continuamente formas de melhoria de processo e de produto.

Diversos modelos de processo de desenvolvimento de *software* surgiram nos últimos anos como, por exemplo, os apresentados pelos padrões 12207 [5] e 15504 [6]. O padrão 12207 prescreve um processo para o

desenvolvimento e manutenção de *software*, através da determinação de um conjunto de atividades essenciais que devem ser completadas para se obter um produto de *software*. Já o padrão 15504 define uma estrutura para avaliação e melhoria de processos de engenharia de *software*, e prescreve práticas básicas que devem ser realizadas para que se atinjam certos níveis de maturidade.

Estes modelos representam o processo de *software* sob a ótica de seu funcionamento, mas não apresentam como estes processos devem ser definidos, ficando esta responsabilidade uma atribuição das organizações fabricantes de *software*.

Para a definição e modelagem dos processos da qualidade, como ponto de partida, foram utilizados as abordagens destes dois padrões, 12207 e 15504, e escolhida a linguagem de modelagem SPEM[11]. Segunda a visão proposta pelo trabalho, os processos Suporte são diretamente relacionados com a gestão da qualidade e deverão, futuramente, ser implementados no Ambiente Integrado para o desenvolvimento e gestão de projetos de *software* [9], projeto oriundo do INPE, que atualmente é apoiado pelo Programa de Capacitação de Recursos Humanos (RHAÉ) do CNPq.

O artigo foi dividido da seguinte forma: a seção 2 apresenta os padrões de processo de *software* utilizados como base do estudo. Na seção 3, uma definição para a gestão qualidade neste contexto. Na 4, modelagem de processo e, em particular, a linguagem de modelagem de processo SPEM, são apresentadas. Na seção 5, é ilustrado uma idéia mais prática sobre modelagem de processo. Finalmente as considerações sobre este estudo são apresentadas na seção 6.

2. Padrões de Processo de *Software*

A melhoria de processo de *software* tem sido praticada há aproximadamente vinte anos, e teve um grande impulso por volta de 1987, quando o *Software Engineering Institute* (SEI) em Pittsburgh, EUA, lançou seu *Capability Maturity Model*, que originou o *Capability*

Maturity Model para Software, ou SW-CMM. Desde então, não só o SW-CMM teve um desenvolvimento considerável, como também outros métodos de melhoria de processo de *software* surgiram, como o ISO/IEC 15504/SPICE e o ISO/IEC 12207.

Apesar de terem se originado de comunidades diferentes, os padrões 12207 e o 15504 foram desenvolvidos praticamente ao mesmo tempo e têm muito em comum. O padrão 15504, que começou com o chamado WG10 (estabelecido pela ISO/IEC/JTC1 SC7 do SPICE, *Software Process Improvement and Capability dEtermination*), publicado em julho de 1995 preocupa-se com as atividades de avaliação e melhoria de processos, enquanto que o ISO/IEC 12207, publicado em agosto de 1995, o enfoque principal está nos ciclo de vida de desenvolvimento de *software*, incorporando os processos de desenvolvimento, aquisição, fornecimento e gerenciamento de *software*. Ambos, no entanto, baseiam-se num modelo muito semelhante.

2.1. O ISO/IEC 12207

O padrão internacional ou modelo de referência ISO/IEC 12207 - Processos do Ciclo de Vida do *Software* tem como principal objetivo fornecer uma estrutura única para que o adquirente, fornecedor, desenvolvedor, mantenedor, operador, gerentes e técnicos envolvidos com o desenvolvimento de *software* utilizem uma linguagem comum que é estabelecida na forma de processos bem definidos.

A estrutura da norma foi concebida de maneira a ser flexível, modular e adaptável às necessidades de quem a utiliza. Para isto, ela está fundamentada em dois princípios básicos: modularidade e responsabilidade. Modularidade, no sentido de processos com mínimo acoplamento e máxima coesão. Responsabilidade, no sentido de estabelecer um responsável único por cada processo, facilitando a aplicação da norma em projetos onde várias pessoas podem estar legalmente envolvidas.

Agrupam as atividades que podem ser executadas durante o ciclo de vida em processos primários – ou fundamentais, de suporte, e organizacionais. Os processos fundamentais atendem ao contrato entre fornecedor e adquirente e à execução do desenvolvimento, da operação ou da manutenção de produtos de *software* durante o seu ciclo de desenvolvimento. Os processos de apoio, auxiliam e contribuem para o sucesso e a qualidade do projeto de *software*. Os processos organizacionais são empregados por uma organização para estabelecer e implementar uma estrutura constituída pelos processos do ciclo de vida e pelo pessoal envolvido no seu desenvolvimento. Cada processo é dividido em um conjunto de atividades, e cada atividade em um conjunto de tarefas. Os processos de apoio e organizacionais devem existir independentemente

da organização e do projeto que está sendo executado. Os processos fundamentais são instanciados de acordo com a situação. Estes processos, atividades e tarefas podem ser adaptados de acordo com as características de um projeto de *software*

2.2. O ISO/IEC 15504

O padrão, ou modelo de referência ISO/IEC 15504 foi desenvolvido como um *framework* para avaliação de processos de engenharia de *software* e da organização do projeto e do negócio. Ele organiza e classifica as melhores práticas em duas dimensões: categorias de processo e níveis de capacidade. Cada uma das categorias de processo é detalhada em processos mais específicos, ou subcategorias (Cliente-fornecedor, Engenharia, Projeto, Suporte e Organização). Este modelo visa também avaliar a capacidade da organização em cada processo, permitindo assim sua melhoria. Cada um dos processos deve ser classificado em níveis (incompleto, executado, gerenciado, estabelecido, previsível e otimizado).

Seu modelo de referência serve de base para o processo de avaliação como um conjunto padronizado de processos fundamentais, que orientam para uma boa engenharia de *software*. Cada uma das categorias de processo é detalhada em processos mais específicos, ou subcategorias. O resultado de uma avaliação é um perfil da instituição em forma de matriz, onde temos os processos nas linhas e os níveis de capacitação nas colunas.

O 15504 possui um conjunto de nove documentos que endereçam avaliação de processo, assessoria de treinamento e competência, determinação da capacidade e melhoria de processo e está se tornando um modelo de referência para outros padrões como o CMMI [2], patrocinado pelo Departamento de Defesa norte-americano, com colaborações da indústria, governo e pelo SEI. O CMMI tem como um de seus objetivos básicos, integrar os diversos modelos CMM existentes, bem como pretende garantir compatibilidade com a 15504, através da sua visão contínua de modelo, composto por áreas de processos universais e fundamentais. Possui ainda um outra dimensão composta por uma métrica para a avaliação da capacidade de cada processo em uma organização.

3. Gestão da Qualidade de Software

Qualidade é um conceito complexo, porque possui significados diversos para diferentes pessoas, em um contexto altamente dependente. Portanto, não é trivial haver medidas simples de qualidade aceitáveis para todos. Para estimar ou melhorar a qualidade de *software* numa organização, deve-se definir as características de

qualidade, nas quais se está interessado e, então, decidir como serão medidas [7].

Segundo Sommerville [10] podemos identificar quatro fatores que afetam a qualidade dos produtos de software: tecnologia de desenvolvimento; qualidade dos processos; qualidade das pessoas; cronograma (recursos e tempo). No que tange a qualidade dos processos estamos nos referindo ao grau de satisfação e eficiência do processo ou um conjunto de processos, utilizados por uma organização ou projeto para: planejar, gerenciar, executar, monitorar, controlar e melhorar as atividades relacionadas à construção de software [12].

Dentro do aspecto qualidade de produto e de processo, a gestão da qualidade de *software* atua como um atividade “guarda-chuva”, que é aplicada ao longo de toda o processo de engenharia de *software*, podendo, segundo a visão de Pressman [8] abranger (1) a definição de métodos e ferramentas para o desenvolvimento do produto; (2) revisões e técnicas formais de avaliação que são aplicadas durante todo o ciclo de vida do produto; (3) controle dos produtos, sua documentação e suas mudanças; (4) procedimentos para a garantia e adequação aos padrões de desenvolvimento, quando aplicáveis; (5) mecanismos de medição e divulgação. Além destas atividades, podemos acrescentar a criação de uma abordagem cultural da qualidade na organização [13].

3.1. Os processos da gestão da qualidade

Segundo a visão da gestão da qualidade deste estudo, as atividades da qualidade estão presentes em toda a estrutura de uma organização produtora de *software*. Enquanto o 15504 define uma estrutura para avaliação e melhoria dos processos de engenharia de *software* o 12207, preocupa-se com os próprios processos. Ambos baseiam-se num modelo muito semelhante, de modo que uma abordagem baseada no 15504 está alinhada a 12207.

No padrão 15504 a categoria de Gerenciamento (MAN) consiste dos processos que contém práticas de natureza genérica que podem ser usadas por gerentes de projeto ou de processos de um ciclo de vida de *software*. Dentre as diversas categorias, identificamos a de Gerenciamento da Qualidade, MAN.3, que tem o propósito monitorar a qualidade dos produtos ou serviços e garantir a satisfação do cliente. Este processo propõe o estabelecimento de um foco para o monitoramento da qualidade de produto e processo tanto no projeto como na organização.

Podemos identificar também nos dois modelos que na categoria Suporte a maioria dos processos são voltadas para as atividades da gestão da qualidade. A categoria SUP consiste dos processos que podem ser aplicados por qualquer dos outros processos, inclusive do próprio Suporte, nos vários pontos do ciclo de vida do *software*

Os processos SUP incluem os processos de Documentação, de Garantia da Qualidade, de Gerenciamento da Configuração, de Verificação, de Validação, de Revisão Conjunta, de Auditoria e de Resolução de Problemas. Recentemente a 12207 sofreu uma revisão [4] que incorporou aos processos de Suporte os processos de Usabilidade e Avaliação de Produto.

4. Modelagem de Processos

Na última década surgiram numerosos formalismos e linguagens de modelagem, também chamadas de Linguagens de Modelagem de Processos (*Process Modeling Languages* – PMLs), que permitem representar, de maneira precisa e compreensível, várias das características ou elementos do processo de *software*.

Segundo Fuggeta [3], a modelagem de processos pode ser utilizada com diversos propósitos como por exemplo para (1) o entendimento do processo, (2) representar a estrutura e organização, (3) projeto de um novo processo, (4) treinamento e aprendizado de equipes sobre procedimentos e operações organizacionais, (5) avaliação de problemas e propostas de melhorias

Os elementos mais comuns que estão presentes em um modelo de processo, segundo Acuña [1] são agentes ou atores, papéis, atividades e artefatos ou produtos. Os agentes ou atores são entidades que executam um processo e podem ser pessoas, sistemas ou ferramentas. Um ator pode desempenhar vários papéis. Os papéis descrevem um conjunto de responsabilidades de atores ou grupos, os direitos e as habilidades necessárias para realizar uma atividade específica do processo de *software*. Uma atividade inclui e implementa os procedimentos, as regras, as políticas e os objetivos para gerar e modificar um conjunto de artefatos. Um artefato ou produto é o (sub)produto e a matéria prima de um processo. Os (sub)produtos podem ser criados, acessados ou modificados durante a atividade do processo.

As PMLs podem ser classificadas em executáveis, quando permitem que o modelo seja executado, ou não executáveis, quando provêm construções que permitem somente a especificação de um processo

4.1. A Linguagem de Modelagem SPEM

O SPEM, *Software Process Engineering Metamodel*, é um metamodelo, ou seja, um conjunto de construtores e regras para a criação de modelos. que pode ser usado para descrever um processo concreto ou uma família de processos de desenvolvimento de *software*. Ela permite somente a especificação de um processo de *software*, não sua execução. A execução do processo (*enactment*) não está no escopo deste modelo, o que o coloca, portanto, na

classe das linguagens de modelagem de processo não executáveis.

A linguagem foi desenvolvida pelo OMG (*Object Management Group*), organização internacional que promove a teoria e prática da tecnologia da orientação a objetos no desenvolvimento de *software*. Foi liberada ao uso em dezembro de 2001, para tentar suprir a necessidade de um padrão para as técnicas de modelagem de processo surgidas nos últimos anos. Define o conjunto mínimo de elementos de modelagem necessários para descrever qualquer processo de desenvolvimento de *software*, utilizando uma abordagem orientada a objetos e a UML (*Unified Modeling Language*) como notação.

Um fator que favorece a escolha do SPEM para a definição de processos é que ele se beneficia da expressividade da UML. Assim, desenvolvedores de *software* que estejam familiarizados com a UML podem reutilizar seus conhecimentos de modelagem de *software* no domínio da modelagem de processos de *software*.

5. Um exemplo prático

De acordo com o padrão 12207, a atividade de Garantia do Produto implica em garantir que os planos e produtos definidos no contrato estejam sendo produzidos, sejam consistentes conforme o especificado e fiquem prontos em tempo hábil. Ela também estabelece que estes produtos de *software* satisfaçam totalmente os requisitos contratuais especificados e sejam aceitos pelo adquirente (cliente). Na 15504, a prática básica 5 do processo Suporte, denominado Garantia da Qualidade estabelece que devem existir atividades para prover confiança de que os produtos do trabalho estejam sendo feitos de acordo com padrões e requisitos da qualidade. O padrão sugere que deve ser conduzida uma estratégia para a garantia da qualidade de forma a identificar a aderência dos produtos aos padrões, procedimentos e requisitos estabelecidos, identificando problemas ou não conformidades, como também relatar os resultados das atividades realizadas aos envolvidos. Nos dois padrões a garantia da qualidade se utiliza de outros processos, como o de Verificação, Validação, Revisão Conjunta, Auditoria e Resolução de Problemas como parte de suas funções, isto é, como resultado da atividade da gestão da qualidade.

5.1. A Garantia de Produto

Uma das atividades principais da qualidade, durante o desenvolvimento dos produtos de um projeto de *software* é a garantia de que os produtos gerados estão de acordo com o especificado no contrato e no projeto, no que tange aos padrões de desenvolvimento (documentos, conteúdo, padrão de código, etc), funcionalidade, atendimento a requisitos, configuração de desenvolvimento e entrega ao

cliente. Na proposta deste estudo, é estabelecido que ao se gerar um produto de *software* (qualquer documento preliminar, ou final, ou código unitário, por exemplo) pela primeira vez, deve ser feito seu encaminhamento a Gerência da Qualidade, para sua devida aprovação, e em caso de parecer conforme o produto ficará liberado automaticamente para o controle de configuração. Para isto fica-se estabelecido que primeiramente se instanciará um processo de Verificação, observando-se se os métodos e processos de desenvolvimento estão sendo aplicados corretamente. Posteriormente, caso o produto seja código, se fará sua Validação, através de testes previamente definidos. Ao se alcançar um marco específico, por exemplo, o marco de entrega do produto ao cliente, será iniciado o processo de Auditoria. Os produtos oriundos de modificações devem passar também pelo mesmo encaminhamento. Caso o produto obtenha parecer conforme, outro processo será disparado, que será o de Controle de Configuração. Caso obtenha parecer não conforme, o processo a ser instanciado é o de Resolução de Problema que tratará de emissão de investigação da não conformidade para solução e futuro encaminhamento.

5.2. O Processo de Garantia de Produto

Os envolvidos e as atividades do processo de garantia de produtos consiste do Autor do Produto, que encaminha o produto para ser controlado, o Gerente da Qualidade que abre um Formulário de Encaminhamento de Produto, e determina um código que identificará tanto a categoria do produto a ser revisado como o tipo de processo a ser executado (Verificação, Validação, Auditoria). A partir daí são notificados todos os envolvidos com aquele Produto (Gerente Geral, Revisor, Autor, Gerente da Configuração, e se for o caso o Cliente). É alocado um Revisor para a Verificação e Validação, que estará encarregado da revisão do produto, definindo os critérios para a revisão, realizando a revisão, tirando possíveis dúvidas com o Autor do Produto, bem como emitindo o parecer do Revisor. Em seguida, o Gerente da Qualidade deve emitir o parecer final. O processo estará encerrado com a notificação dos envolvidos sobre o parecer.

6. Considerações Finais

Processos de desenvolvimento bem definidos são essenciais na engenharia de software. Neste trabalho, modelos de processo de Suporte do ISO/IEC 12207 e do 15504 são apresentados como ponto de partida para a definição dos processos da qualidade que devem ser implantados em um Ambiente Integrado.

Estes padrões prescrevem práticas e atividades que devem ser realizadas pelo processo de desenvolvimento específico de um projeto ou organização. No entanto,

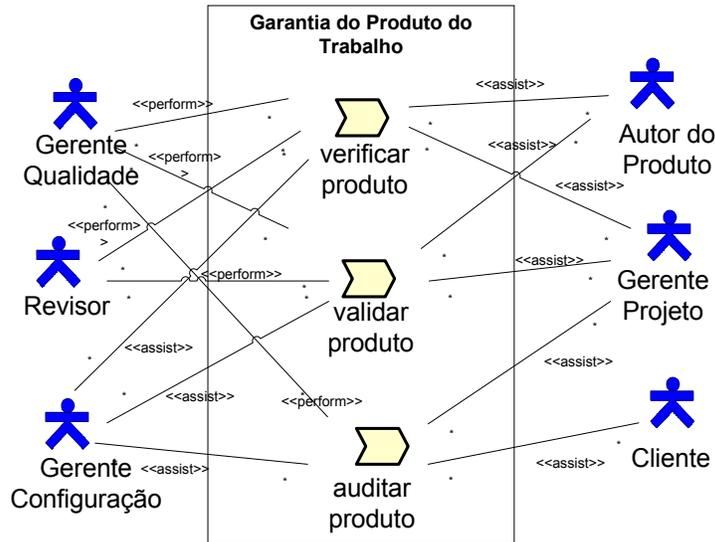


Figura 1. Use case Garantia do Produto do Trabalho.

não definem realmente “como” deve ser o processo seguido por uma organização que desenvolve software.

Para exemplificar um dos vários aspectos de um modelo de processo de *software*, utilizou-se o diagrama de caso de uso e de atividade da linguagem SPEM. Através do diagrama de casos de uso, Figura 1, pode-se mostrar as relações entre os papéis e as principais definições de trabalho da Garantia do Produto. Já no diagrama de atividades, Figura 2, permite-se à apresentação da seqüência de atividades com seus produtos de trabalho de entrada e saída, bem como os estados do fluxo de objetos do processo de Verificação. “Raias” são utilizadas para separar as responsabilidades dos diferentes envolvidos no processo.

Modelos, expressos em uma linguagem como o SPEM, são utilizados para a definição clara e precisa dos processos da qualidade, visando a posterior implementação destes em um ambiente de engenharia de software voltado a processo, os chamados Ambientes de Engenharia de Software Centrados em Processo (Process-centered Software Engineering Environment –PSEEs) A notação padrão UML utilizada pelo SPEM facilita a atividade de modelagem de processos, uma vez que se podem reutilizar os conhecimentos desta linguagem obtidos na prática da modelagem de *software*.

No exercício de modelagem da seção 5 apresentou-se apenas a visão de processo, através do diagrama de caso de uso e de atividades. Outras visões do processo podem ser modeladas, por exemplo, a do diagrama de estados, do modelo, suas transições e seus estados. que é usado para ilustrar o comportamento dos seus elementos

Referências

- [1] Acuña, S.; Ferré, X.; “*Software Process Modelling, Proceedings of the 5th World Multiconference on Systemics, Cybernetics and Informatics (SCI 2001)*, Orlando, Florida, USA.
- [2] CMMI Product Development Team, *CMMI for Systems Engineering/ Software Engineering/ Integrated Product and Process Development*, Software Engineering Institute, Carnegie Mellon University, Version 12, (CMU/SEI-2000-TR-031), 2000.
- [3] Fuggetta, A., “*Software Process: A Roadmap*”, *Future of Software Engineering*, Limerick Ireland, 2000.
- [4] *International Standard for Information Technology, Software life cycle processes*, ISO/IEC JTC1/SC7 N2529r – Amendment 1, 2001.
- [5] *International Standard for Information Technology, Software life cycle processes*, ISO/IEC 12207 Standard, 1995.
- [6] *Information Technology – Software Process Assessment Part 5: An assessment model an indicator guidance*, ISO/IEC, TR15504-1, 1998
- [7] “Kitchenham B.; Pfleeger, S. L., “*Software Quality: The Elusive Target*”, IEEE Software, January, 1996.
- [8] Pressman, R. S., “*Engenharia de Software*”, Makron Books, São Paulo, 1995
- [9] Sant’Anna, N., “Um ambiente integrado para apoio ao desenvolvimento e gestão de projetos de software para sistemas de controle de satélites”, São José dos Campos: INPE, 2000 (INPE - 8306 - TDI/765).

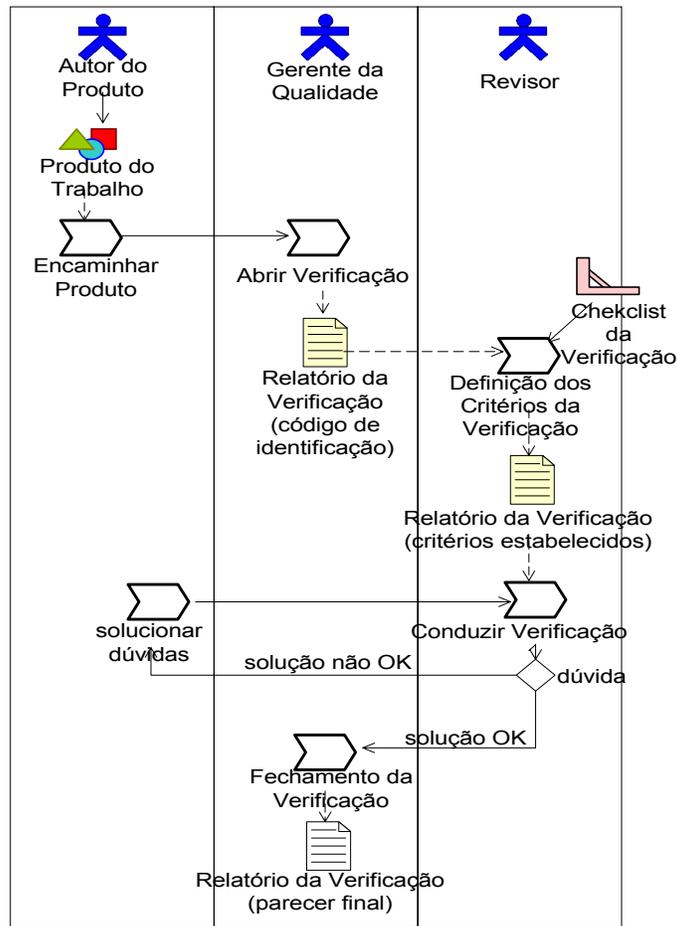


Figura 2. Diagrama de atividades Verificar Produto

[10] Sommerville, I., “*Software Engineering*”, 6th edition, Reading: Person Education Limited, Essex, 2001.

[11] SPEM - *Software Process Engineering Metamodel Specification, Object management Group -Software Version 1.0, formal, 02-11-14, November 2002.*

[12] Terry R, “*Software Process Assesment, Part1 Concepts and Introductory Guide, version 1*”, SPICE project, ISO/IEC, 1995.

[13] Zulmer, Richard E., “*Software Quality Engineering: The Deming Way*”, American Programmer, June 1989.