

Qualidade de Software

José Barreto Júnior

O que é qualidade? Existem diversas definições. Algumas pessoas que tentaram uma definição simples chegaram a frases como:

- Qualidade é estar em conformidade com os **requisitos** dos clientes
- Qualidade é **antecipar e satisfazer** os desejos dos clientes
- Qualidade é **escrever tudo** o que se deve fazer e fazer tudo o que foi escrito

Segundo a atual norma brasileira sobre o assunto (NBR ISO 8402), qualidade é:

*A totalidade das características de uma **entidade** que lhe confere a capacidade de satisfazer às **necessidades explícitas e implícitas***

Nota-se que esta definição formal exige alguns complementos, principalmente para definir o que são as entidades, as necessidades explícitas e as necessidades implícitas. A **entidade** é o produto do qual estamos falando, que pode ser um bem ou um serviço. As **necessidades explícitas** são as próprias condições e objetivos propostos pelo produtor. As **necessidades implícitas** incluem as diferenças entre os usuários, a evolução no tempo, as implicações éticas, as questões de segurança e outras visões subjetivas.

Por exemplo, a qualidade de um prato de comida (a entidade, o produto) está relacionada com a satisfação de necessidades (requisitos) tais como: sabor, aparência, temperatura, rapidez no serviço, preço, higiene, valor nutricional, etc... Para avaliar a qualidade de um produto, você deve fazer uma lista destas necessidades e analisar cada uma destas necessidades.

Certificação de Qualidade

Um aspecto interessante da qualidade é que não basta que ela exista. Ela deve ser **reconhecida** pelo cliente. Por causa disso, é necessário que exista algum tipo de **certificação** oficial, emitida com base em um **padrão**. Você provavelmente já conhece alguns certificados mais comuns:

- O selo do SIF de inspeção da carne
- O selo da ABIC nos pacotes de café
- O certificado da Secretaria de Saúde para restaurantes (classe "A" são os melhores)
- A classificação em estrelas dos hotéis (hotéis com cinco estrelas são ótimos)
- Os certificados de qualidade da série ISO-9000

Você já deve ter ouvido muitas propagandas de empresas falando de sua **certificação ISO-9000**. Isto nada mais é do que um padrão de qualidade (reconhecido mundialmente) pelo qual esta empresa foi avaliada e julgada. Para que seja possível realizar uma **avaliação** e um **julgamento**, é necessário haver um padrão ou norma. Existem alguns organismos normalizadores reconhecidos mundialmente:

- [ISO - International Organization for Standardization](#)
- [IEEE - Instituto de Engenharia Elétrica e Eletrônica](#)
- [ABNT - Associação Brasileira de Normas Técnicas](#)

A norma ISO-9000, por exemplo, foi criada pela ISO para permitir que todas as empresas do mundo possam avaliar e julgar sua qualidade. Existindo um **padrão único mundial**, uma empresa do Brasil, mesmo não tendo nenhum contato com uma outra empresa na Europa, pode garantir a ela a qualidade de seu trabalho.

A **Certificação** em uma norma ou padrão é a emissão de um documento oficial indicando a conformidade com esta determinada norma ou padrão. É claro que, antes da emissão do certificado, é preciso realizar todo um processo de avaliação e julgamento de acordo com uma determinada norma. Embora uma empresa possa auto-avaliar-se ou ser avaliada por seus próprios clientes, o termo **Certificação** costuma ser aplicado apenas quando efetuado por uma empresa independente e idônea, normalmente especializada neste tipo de trabalho. No Brasil, o INMETRO é o órgão do governo responsável pelo credenciamento destas instituições que realizam a certificação de sistemas de qualidade.

Qualidade do Produto x Qualidade do Processo

Uma das evoluções mais importantes no estudo da qualidade está em notar que a **qualidade do produto** é algo bom, mas que **qualidade do processo** de produção é ainda mais importante. No caso do prato de comida, por exemplo, você pode dizer mais sobre a qualidade observando como o prato foi preparado do que analisando o produto final. Afinal, você não consegue ter certeza da higiene ou o valor nutricional apenas comendo o prato.

Esta descoberta aconteceu durante a própria evolução dos conceitos de qualidade, ao longo dos anos. Observe na tabela abaixo como aconteceu esta evolução:

| | | |
|----------------------------------|--|------|
| Inspecção pós-produção | Avalia o produto final, depois de pronto | 1900 |
| Controle estatístico da produção | Avalia os subprodutos das etapas de produção | 1940 |
| Procedimento de produção | Avalia todo o procedimento de produção | 1950 |
| Educação das pessoas | Avalia as pessoas envolvidas no processo | 1960 |
| Otimização dos processos | Avalia e otimiza cada processo | 1970 |
| Projeto robusto | Avalia o projeto de produção | 1980 |
| Engenharia simultânea | Avalia a própria concepção do produto | 1990 |

Hoje em dia, você pode consultar normas e padrões tanto para produtos quanto para processos. Obviamente, os certificados mais valiosos são aqueles que certificam o processo de produção de um produto e não aqueles que simplesmente certificam o produto. Entretanto, é comum encontrar empresas que perseguem os dois tipos de padrão de qualidade.

Qualidade de Software

Agora que você já sabe o que é qualidade e como ela pode ser avaliada, vamos tentar aplicar estes conceitos aos produtos de software e ao processo de desenvolvimento de software. Inicialmente, vamos encontrar um grande problema: muitas pessoas acham que **criar programas é uma arte** que não pode seguir regras, normas ou padrões. Isto acontece principalmente porque:

- Produtos de software são complexos, até mais do que o hardware onde executam
- Software não têm produção em série. Seu custo está no projeto e desenvolvimento
- Software não se desgasta e nem se modifica com o uso
- O Software é invisível. Sua representação em grafos e diagramas não é precisa.
- A Engenharia de Software ainda não está madura, é uma tecnologia em evolução
- Não há um acordo entre os profissionais da área sobre o que é Qualidade de Software

Apesar de tudo isso, você precisa entender que **o problema não está no Software** em si, mas na forma como as pessoas tem desenvolvido software até os dias de hoje. Você provavelmente já ouviu dizer que "Se os engenheiros construíssem prédios como os analistas constroem software, um único pica-pau destruiria a humanidade". Exageros à parte, você precisa se conscientizar que nós **precisamos aplicar na indústria de software os conceitos de qualidade**, urgentemente.

Atualmente, muitas instituições se preocupam em criar normas para permitir a correta avaliação de qualidade tanto de produtos de software quanto de processos de desenvolvimento de software. Apenas para ter uma visão geral, observe o quadro abaixo com as principais normas nacionais e internacionais nesta área:

| Norma | Comentário |
|---------------------------|---|
| ISO 9126 | Características da qualidade de produtos de software. |
| NBR 13596 | Versão brasileira da ISO 9126 |
| ISO 14598 | Guias para a avaliação de produtos de software, baseados na utilização prática da norma ISO 9126 |
| ISO 12119 | Características de qualidade de pacotes de software (software de prateleira, vendido com um produto embalado) |
| IEEE P1061 | Standard for Software Quality Metrics Methodology (produto de software) |
| ISO 12207 | Software Life Cycle Process. Norma para a qualidade do processo de |

| | |
|------------------------------------|---|
| | desenvolvimento de software. |
| NBR ISO 9001 | Sistemas de qualidade - Modelo para garantia de qualidade em Projeto, Desenvolvimento, Instalação e Assistência Técnica (processo) |
| NBR ISO 9000-3 | Gestão de qualidade e garantia de qualidade. Aplicação da norma ISO 9000 para o processo de desenvolvimento de software. |
| NBR ISO 10011 | Auditoria de Sistemas de Qualidade (processo) |
| CMM | Capability Maturity Model. Modelo da SEI (Instituto de Engenharia de Software do Departamento de Defesa dos EEUU) para avaliação da qualidade do processo de desenvolvimento de software. Não é uma norma ISO, mas é muito bem aceita no mercado. |
| SPICE ISO 15504 | Projeto da ISO/IEC para avaliação de processo de desenvolvimento de software. Ainda não é uma norma oficial ISO, mas o processo está em andamento. |

No final desta página, você encontra alguns [links relacionados a Qualidade e Qualidade de Software](#).

Engenharia de Software

Como você já viu, a disciplina que vai nos ajudar a entender o processo de desenvolvimento de software é a Engenharia de Software. É através dela que poderemos chegar à qualidade. Existe, entretanto, um grande problema a ser resolvido: tecnicamente, **ela não existe**.

O problema é que, para que uma disciplina seja considerada realmente uma **Engenharia**, é necessário atender a alguns requisitos básicos que a Engenharia de Software, pelos menos até agora, não atende. Veja a definição de Engenharia:

"A Engenharia deve criar soluções com uma relação custo-benefício adequada para problemas práticos, pela aplicação de conhecimentos científicos, para construir coisas a serviço da humanidade."

Dentro destes conceitos, a Engenharia de Software falha principalmente no que diz respeito à adequação do custo-benefício e à aplicação, em toda a sua extensão, de conhecimentos científicos. Atualmente, estes requisitos são atendidos apenas em parte.

É necessário definir, portanto, o que é exatamente a Engenharia de Software. Veja algumas tentativas de definição:

"...é a disciplina que integra métodos, ferramentas e procedimentos para o desenvolvimento de software para computadores."

"...é uma coleção de processos de gerenciamento, ferramental de software e atividades de projeto para o desenvolvimento de software. "

"...é um termo usado para referir-se a modelos de ciclo de vida, metodologias de rotina, técnicas de estimativa de custo, estruturas de documentação, ferramentas de gerenciamento de configuração, técnicas de garantia de qualidade e outras técnicas de padronização da atividade de produção de software."

No final desta página você encontrará alguns [links relacionados a Engenharia de Software](#).

Qualidade de Produtos de Software - ISO 9126

Quando se pensa em qualidade de um "produto físico", é fácil imaginar padrões de comparação, provavelmente ligado às dimensões do produto ou alguma outra característica física. Quando se trata de software, como podemos definir exatamente o que é a qualidade? Parece difícil...

Felizmente, para nós, a [ISO \(Organização Internacional de Padrões\)](#) já pensou bastante sobre o assunto. O suficiente para publicar uma norma que representa a atual padronização mundial para a qualidade de produtos de software. Esta norma chama-se ISO/IEC 9126 e foi publicada em 1991. Ela é uma das mais antigas da área de qualidade de software e já possui sua tradução para o Brasil, publicada em agosto de 1996 como NBR 13596.

Mas, afinal de contas, o que está escrito nesta norma ISO/IEC 9126 ou na NBR 13596? Bem, estas normas listam o conjunto de características que devem ser verificadas em um software para que ele seja considerado um "software de qualidade". São seis grandes grupos de características, cada um dividido em algumas subcaracterísticas.

Os nomes dados pelo ISO/IEC para as características e subcaracterísticas são um pouco complexos (para dizer a verdade, acho até que os próprios termos "características" e "subcaracterísticas" são mais complexos que o necessário). Entretanto, uma pessoa que trabalha com software não terá dificuldade em entendê-las. Observe na tabela abaixo a lista completa:

| Característica | Subcaracterística | Pergunta chave para a subcaracterística |
|--|---------------------|--|
| Funcionalidade (satisfaz as necessidades?) | Adequação | Propõe-se a fazer o que é apropriado? |
| | Acurácia | Faz o que foi proposto de forma correta? |
| | Interoperabilidade | Interage com os sistemas especificados? |
| | Conformidade | Está de acordo com as normas, leis, etc.? |
| | Segurança de acesso | Evita acesso não autorizado aos dados? |
| Confiabilidade (é imune a falhas?) | Maturidade | Com que frequência apresenta falhas? |
| | Tolerância a falhas | Ocorrendo falhas, como ele reage? |
| | Recuperabilidade | É capaz de recuperar dados em caso de falha? |
| Usabilidade | Intelegibilidade | É fácil entender o conceito e a aplicação? |

| | | |
|--|---------------------------|---|
| (é fácil de usar?) | Apreensibilidade | É fácil aprender a usar? |
| | Operacionalidade | É fácil de operar e controlar? |
| Eficiência (é rápido e "enxuto"?) | Tempo | Qual é o tempo de resposta, a velocidade de execução? |
| | Recursos | Quanto recurso usa? Durante quanto tempo? |
| Manutenibilidade (é fácil de modificar?) | Analisabilidade | É fácil de encontrar uma falha, quando ocorre? |
| | Modificabilidade | É fácil modificar e adaptar? |
| | Estabilidade | Há grande risco quando se faz alterações? |
| | Testabilidade | É fácil testar quando se faz alterações? |
| Portabilidade (é fácil de usar em outro ambiente?) | Adaptabilidade | É fácil adaptar a outros ambientes? |
| | Capac. para ser instalado | É fácil instalar em outros ambientes? |
| | Conformidade | Está de acordo com padrões de portabilidade? |
| | Capac. para substituir | É fácil usar para substituir outro? |

Esta norma não é muito extensa (contém apenas 10 páginas). Entretanto, ela define minuciosamente (em uma linguagem bem detalhista, estilo contrato) o que se pretende avaliar em cada característica e subcaracterística. Se você desejar ter o texto integral da norma NBR 13596, pode pedir uma cópia à ABNT. Você terá de pagar por ela, mas o valor é pequeno (menos de R\$20). O site da ABNT na Internet oferece funções de pesquisa e orçamento para compra de normas. Para maiores informações, contacte diretamente a ABNT:

[ABNT - Associação Brasileira de Normas Técnicas](#)

Av. Treze de Maio, 13 - 28º andar
 CEP 20003-900 Caixa Postal 1680
 Fone: (021) 210-3122 - Fax: (021) 532-2143

Métricas de Software

Embora a atual norma ISO 9126/NBR 13596 enumere as características e subcaracterísticas um software, ela ainda não define como dar uma nota a um software em cada um destes itens. Se você não está familiarizado com o processo de avaliação de software, pode ter dificuldades em tentar utilizar a norma. Se você pretende avaliar um software segundo esta norma, deve tentar atribuir valores (como se fossem notas ou conceitos) a cada uma das subcaracterísticas.

Algumas características podem ser realmente medidas, como o tempo de execução de um programa, número de linhas de código, número de erros encontrados em uma sessão de teste ou o tempo médio entre falhas. Nestes casos, é possível utilizar uma técnica, uma ferramenta ou um software para realizar medições. Em outros casos, a característica é tão subjetiva que não existe nenhuma forma óbvia de medi-la.

Ficam, portanto, as questões: como dar uma nota, em valor numérico, a uma característica inteiramente subjetiva? O que representa, por exemplo, uma "nota 10" em termos de "Segurança de Acesso"? Quando se pode dizer que a "Intelegibilidade" de um software pode ser considerada "satisfatória"? Criou-se, então, uma área de estudo à parte dentro da Qualidade de Software conhecida como Métricas de Software. O que se pretende fazer é definir, de forma precisa, como medir numericamente uma determinada característica.

Para avaliar uma determinada subcaracterística subjetiva de forma simplificada, por exemplo, você pode criar uma série de perguntas do tipo "sim ou não". Crie as perguntas de forma tal que as respostas "sim" sejam aquelas que indicam uma melhor nota para a característica. Depois de prontas as perguntas, basta avaliar o software, respondendo a cada pergunta. Se você conseguir listar 10 perguntas e o software obtiver uma resposta "sim" em 8 delas, terá obtido um valor de 80% nesta característica.

Obviamente, a técnica acima não é muito eficiente. Para melhorá-la, entretanto, você pode garantir um número mínimo perguntas para cada característica. Além disso, algumas perguntas mais importantes podem ter pesos maiores. É possível, ainda, criar perguntas do tipo ABCDE, onde cada resposta indicaria um escore diferenciado. Alguns estudiosos sugerem formas diferentes de medir uma característica, baseada em conceitos do tipo "não satisfaz", "satisfaz parcialmente", "satisfaz totalmente" e "excede os padrões". Estes conceitos, embora parecem muito subjetivos, não deixam de ser uma forma eficiente de medir uma característica.

Em todos os casos, um fato fica claro: nada ajuda mais a avaliar características de um software do que um avaliador experiente, que já realizou esta tarefa diversas vezes e em diversas empresas diferentes. Afinal, medir é comparar com padrões e um avaliador experiente terá maior sensibilidade do que um profissional que acaba de ler uma norma pela primeira vez.

Atualmente, a norma ISO/IEC 9126 está sendo revisada. A revisão, que deverá estar pronta nos próximos anos, não deverá modificar nenhuma das características básicas da 9126. A maior modificação será a inclusão de dois documentos adicionais para descrever métricas externas (relativas ao uso do produto) e métricas internas (relativas à arquitetura do produto). Veja algumas das modificações previstas para esta revisão:

- Algumas novas subcaracterísticas. Conformidade fará parte de todas as características. Atratividade será uma subcaracterística de usabilidade. Capacidade de coexistir será uma subcaracterística de portabilidade.
- A norma será dividida em três partes. A primeira (9126-1) incluirá definições e características. As duas seguintes descreverão métricas externas (9126-2) e internas (9126-3).
- A versão brasileira da revisão desta norma deverá ser chamada de NBR 9126-1, 9126-2 e 9126-3, segundo a numeração original da ISO/IEC.

Guias para a Avaliação da Qualidade - ISO 14598

Todos notaram a necessidade de mais detalhes sobre como avaliar a qualidade de um software. As características e subcaracterísticas da norma ISO/IEC 9126 apenas começaram o trabalho. Faltava definir, em detalhes, como atribuir um conceito para cada item. Afinal, sem uma padronização, que valor teria uma avaliação?

A ISO, consciente deste problema, está finalizando o trabalho em um conjunto de Guias para a Avaliação da Qualidade segundo a norma ISO/IEC 9126. Estes guias descrevem, detalhadamente, todos os passos para que se avalie um software. Embora o trabalho nesta norma ainda não esteja totalmente pronta, já existem informações detalhadas sobre o que será esta norma, quando for oficialmente publicada.

Esta nova norma trará muitos recursos interessantes aos avaliadores, já que trata o processo de avaliação em grande detalhe. Ela leva em conta a existência de três grupos interessados em avaliar um software, o que define os três tipos básicos de certificação:

| Certificação | Quem realiza | Finalidade |
|---------------------|-----------------------------------|---|
| de 1a. parte | Empresas que desenvolvem software | Melhorar a qualidade de seu próprio produto |
| de 2a. parte | Empresas que adquirem software | Determinar a qualidade do produto que irão adquirir |
| de 3a. parte | Empresas que fazem certificação | Emitir documento oficial sobre a qualidade de um software |

Esta norma se constituirá, na verdade, de seis documentos distintos, relacionados entre si. Veja:

| Norma | Nome | Finalidade |
|--------------|------------------------------|--|
| 14598-1 | Visão Geral | Ensina a utilizar as outras normas do grupo |
| 14598-2 | Planejamento e Gerenciamento | Sobre como fazer uma avaliação, de forma geral |
| 14598-3 | Guia para Desenvolvedores | Como avaliar sob o ponto de vista de quem desenvolve |
| 14598-4 | Guia para Aquisição | Como avaliar sob o ponto de vista de quem vai adquirir |
| 14598-5 | Guia para Avaliação | Como avaliar sob o ponto de vista de quem certifica |
| 14598-6 | Módulos de Avaliação | Detalhes sobre como avaliar cada característica |

Em resumo, esta nova norma complementar  a ISO/IEC 9126 e permitir  uma avalia o padronizada das caracter sticas de qualidade de um software.   importante notar que, ao contr rio da 9126, a 14598 vai a detalhes m nimos, incluindo modelos para relat rios de avalia o, t cnicas para medi o das caracter sticas, documentos necess rios para avalia o e fases da avalia o. Como um exemplo, observe um modelo de relat rio de avalia o, segundo um anexo da norma 14598-5:

| Se o | Itens |
|------------------|---|
| 1 - Pref cio | Identifica o do avaliador Identifica o do relat rio de avalia o Identifica o do contratante e fornecedor |
| 2 - Requisitos | Descri o geral do dom nio de aplica o do produto Descri o geral dos objetivos do produto Lista dos requisitos de qualidade, incluindo - Informa es do produto a serem avaliadas - Refer ncias  s caracter sticas de qualidade - N veis de avalia o |
| 3 - Especifica o | Abrang ncia da avalia o Refer ncia cruzada entre os requisitos de avalia o e os componentes do produto Especifica o das medi es e dos pontos de verifica o Mapeamento entre a especifica o das medi es com os requisitos de avalia o |
| 4 - M todos | M todos e componentes nos quais o m todo ser  aplicado |
| 5 - Resultado | Resultados da avalia o propriamente ditos Resultados intermedi rios e decis es de interpreta o Refer ncia  s ferramentas utilizadas |

As normas 14598-1, 14598-4 e 14598-5 j  foram publicadas. As demais est o em processo de finaliza o. Est  sendo feito pela ABNT um trabalho de tradu o desta norma (tanto dos itens j  publicados quanto das vers es preliminares dos itens restantes). Com isso, esta norma ter  sua vers o brasileira pouco tempo depois do final de sua publica o pela ISO.

Qualidade de Pacotes de Software - ISO 12119

Esta norma foi publicada em 1994 e trata da avalia o de pacotes de software, tamb m conhecidos como "software de prateleira". Al m de estabelecer os requisitos de qualidade para este tipo de software, ela tamb m destaca a necessidade de instru es para teste deste pacote, considerando estes requisitos. A norma divide-se em itens, da seguinte forma:

| Item | Descri o |
|--------------|----------|
| 1. Escopo | |
| 2. Defini es | |

| | |
|------------------------------|--|
| 3. Requisitos de qualidade | |
| 3.1. Descrição do Produto | Descreve o produto, de forma a ajudar o comprador em potencial, servindo como base para testes. Cada declaração deve ser correta e testável. Deve incluir declarações sobre funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade. |
| 3.2. Documentação do usuário | Deve ser completa, correta, consistente, fácil de entender e capaz de dar uma visão geral do produto. |
| 3.3. Programas e dados | Descreve em detalhes cada uma das funções do software, incluindo declarações sobre funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade. |
| 4. Instruções para teste | |
| 4.1. Pré-requisitos de teste | Lista de itens necessários ao teste, incluindo documentos incluídos no pacote, componentes do sistema e material de treinamento. |
| 4.2. Atividades de teste | Instruções detalhadas sobre os procedimentos de teste, inclusive instalação e execução de cada uma das funções descritas. |
| 4.3. Registro de teste | Informações sobre como os testes foram realizados, de tal forma a permitir uma reprodução destes testes. Deve incluir parâmetros utilizados, resultados associados, falhas ocorridas e até a identidade do pessoal envolvido. |
| 4.4. Relatório de teste | Relatório incluindo: identificação do produto, hardware e software utilizado, documentos utilizados, resultados dos testes, lista de não conformidade com os requisitos, lista de não conformidade com as recomendações, datas, etc. |

Um dos grandes méritos desta norma está na profundidade com que são descritas cada uma das características e subcaracterísticas mencionadas na norma 9126. A norma inclui detalhes que devem estar presentes no produto, tais como:

- Documentação do usuário de fácil compreensão
- Um sumário e um índice remissivo na documentação do usuário
- Presença de um Manual de instalação com instruções detalhadas
- Possibilidade de verificar se uma instalação foi bem sucedida
- Especificação de valores limites para todos os dados de entrada, que deverão ser testados
- Operação normal mesmo quando os dados informados estão fora dos limites especificados
- Consistência de vocabulário entre as mensagens e a documentação
- Função de auxílio (help) com recursos de hipertexto
- Mensagens de erro com informações necessárias para a solução da situação de erro
- Diferenciação dos tipos de mensagem: confirmação, consulta, advertência e erro
- Clareza nos formatos das telas de entrada e relatórios

- Capacidade de reverter funções de efeito drástico
- Alertas claros para as conseqüências de uma determinada confirmação
- Identificação dos arquivos utilizados pelo programa
- Identificação da função do programa que está sendo executada no momento
- Capacidade de interromper um processamento demorado

Outras características importante são a ênfase nos testes e os modelos de relatórios incluídos. Tudo isso facilita grandemente o trabalho do avaliador. Uma versão traduzida desta norma será publicada em breve ABNT.

Qualidade do Processo de Software

Os estudos sobre qualidade mais recentes são na sua maioria voltados para o melhoramento do processo de desenvolvimento de software. Não é que a qualidade do produto não seja importante, ela é. Mas o fato é que, ao garantir a qualidade do processo, já se está dando um grande passo para garantir também a qualidade do produto.

O estudo da Qualidade do Processo de Software é uma área ligada diretamente à Engenharia de Software. O estudo de um ajuda a entender e aprimorar o outro. Em ambas as disciplinas, estuda-se modelos do processo de desenvolvimento de software. Estes modelos são uma tentativa de explicar em detalhes **como** se desenvolve um software, quais são as etapas envolvidas. É necessário compreender cada pequena tarefa envolvida no desenvolvimento.

Entre os estudos nesta área de maior importância, podemos citar:

- ISO 9000-3 - Normas para aplicação da [série ISO 9000](#) em processos de software
- [ISO 12207 - Processos do Ciclo de Vida do Software](#)
- [CMM - Capability Maturity Model](#)
- [PSP - Personal Software Process](#)
- [ISO 15504 - SPICE - Software Process Improvement and Capability dEtermination](#)
- Modelo Trillium
- Metodologia Bootstrap
- Engenharia de Software Cleanroom

Dentre os trabalhos na área de Qualidade de Processo de Software, o único que realmente é norma oficial é o ISO 9000-3, que faz parte da série ISO 9000. Os demais modelos são normas não-oficiais criados por empresas e institutos ou então são normas em estágio de desenvolvimento. Muitos dos modelos estão disponíveis na Internet, em texto integral. No final desta página você encontrará alguns [links sobre Modelos de Qualidade de Processo de Software](#). A seguir serão analisados em maior detalhes alguns destes modelos.

A Série ISO 9000

Esta série é um conjunto de normas da ISO que define padrões para garantia e gerenciamento da qualidade. Veja algumas destas normas abaixo:

| Norma | Trata de |
|--------------|--|
| ISO 9001 | Modelo para garantia da qualidade em projeto, desenvolvimento, produção, instalação e assistência técnica. |
| ISO 9002 | Modelo para garantia da qualidade em produção e instalação |
| ISO 9003 | Modelo para garantia da qualidade em inspeção e ensaios finais |
| ISO 9000-1 | Diretrizes para escolher entre as normas ISO 9001, 9002 e 9003 |
| ISO 9000-3 | Orientação para a aplicação da ISO 9001 em Software |

Entre as normas 9001, 9002 e 9003, a primeira é a que mais se adequa ao desenvolvimento e manutenção de software. Como toda norma deste grupo, ela é usada para garantir que um fornecedor atende aos requisitos especificados nos diversos estados do desenvolvimento. Estes estágios incluem projeto, desenvolvimento, produção, instalação e suporte.

A norma ISO 9000-3 (não confundir com a ISO 9003) traz os roteiros para aplicar a ISO 9001 especificamente na área de desenvolvimento, fornecimento e manutenção de software. Todas as orientações giram em torno de uma "situação contratual", onde uma outra empresa contrata a empresa em questão para desenvolver um produto de software. Veja na tabela abaixo os processos definidos na ISO 9000-3:

| Grupo | Atividade |
|-----------------------------------|--|
| Estrutura do Sistema de Qualidade | Responsabilidade do fornecedor Responsabilidade do comprador Análise crítica conjunta |
| Atividades do Ciclo de Vida | Análise crítica do contrato Especificação dos requisitos do comprador Planejamento do desenvolvimento Projeto e implementação Testes e validação Aceitação Cópia, entrega e instalação Manutenção |
| Atividades de Apoio | Gerenciamento de configuração Controle de documentos Registros da qualidade Medição Regras, convenções Aquisição Produto de software incluído Treinamento |

O processo de certificação de uma empresa de software segundo as normas ISO 9001 / 9000-3 segue um conjunto de passos bem definidos:

1. A empresa estabelece o seu sistema de qualidade
2. A empresa faz uma solicitação formal a um órgão certificador, incluindo detalhes do negócio da empresa, escopo da certificação solicitada e cópia do manual de qualidade
3. O órgão certificador faz uma visita à empresa, colhe mais dados e explica o processo de certificação
4. O órgão certificador verifica se a documentação do sistema de qualidade está de acordo com a norma ISO
5. O órgão certificador envia uma equipe à empresa com fins de auditoria. Nesta visita, será verificado se todos na empresa cumprem o que está documentado no manual de qualidade.
6. O órgão certificador emite o certificado de qualidade
7. O órgão certificador realiza visitas periódicas à empresa para assegurar que o sistema continua sendo efetivo

Se você desejar ter o texto integral das normas da série ISO 9000, pode pedir uma cópia à ABNT. Você terá de pagar por ela, mas o valor é pequeno (tanto a ISO 9001 quanto a ISO 9000-3 custam menos de R\$25, cada). O site da ABNT na Internet oferece funções de pesquisa e orçamento para compra de normas.

ISO 12207 - Processos do Ciclo de Vida do Software

Este padrão formaliza a arquitetura do ciclo de vida do software, que é um assunto básico em Engenharia de Software e também em qualquer estudo sobre Qualidade do Processo de Software. Esta norma possui mais de 60 páginas e detalha os diversos processos envolvidos no ciclo de vida do software. Estes processos estão divididos em três classes: Processos Fundamentais, Processos de Apoio e Processos Organizacionais.

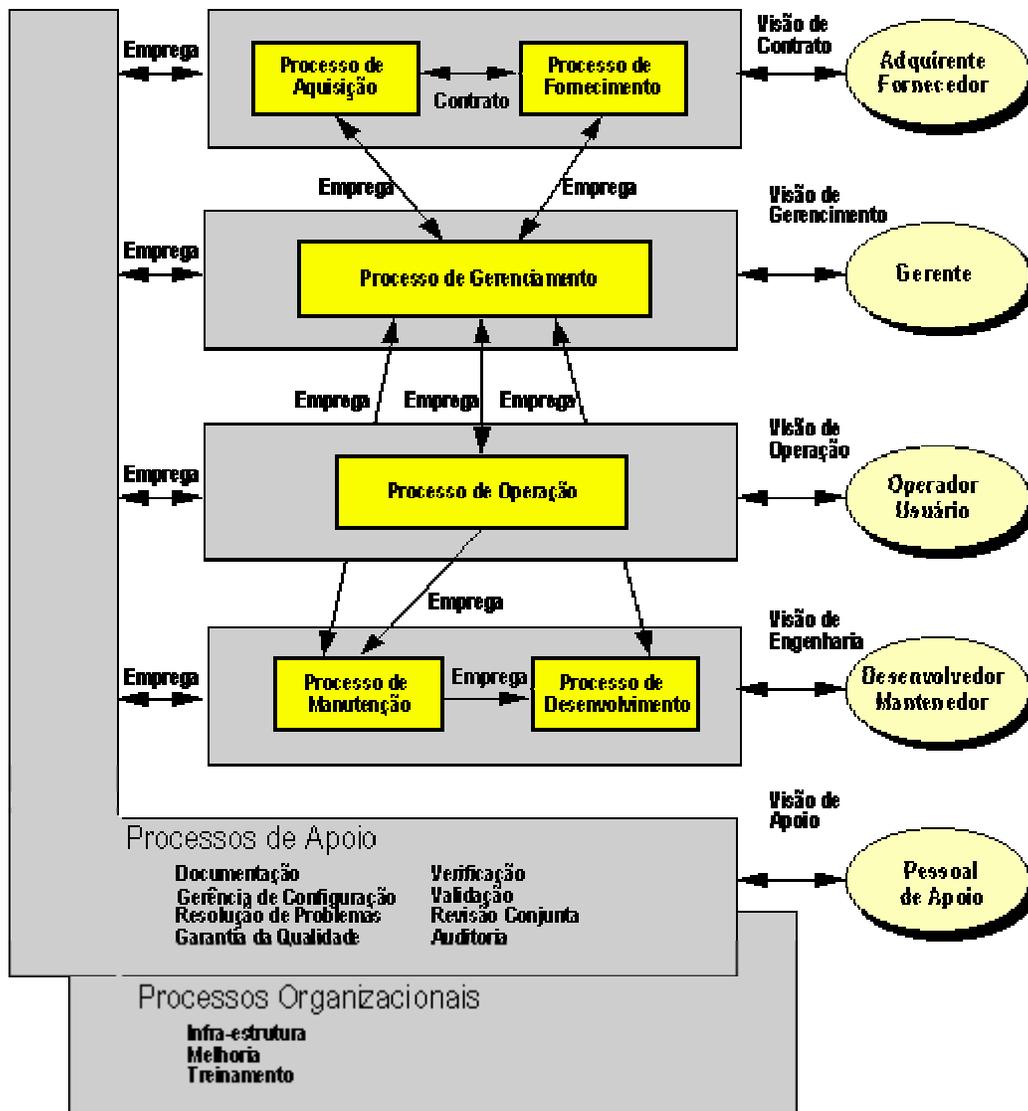
Veja a lista completa dos processos na tabela abaixo:

| Processos Fundamentais | Início e execução do desenvolvimento, operação ou manutenção do software durante o seu ciclo de vida. |
|-------------------------------|--|
| Aquisição | Atividades de quem um software. Inclui: definição da necessidade de adquirir um software (produto ou serviço), pedido de proposta, seleção de fornecedor, gerência da aquisição e aceitação do software. |
| Fornecimento | Atividades do fornecedor de software. Inclui preparar uma proposta, assinatura de contrato, determinação recursos necessários, planos de projeto e entrega do software. |
| Desenvolvimento | Atividades do desenvolvedor de software. Inclui: análise de requisitos, projeto, codificação, integração, testes, instalação e aceitação do software. |
| Operação | Atividades do operador do software. Inclui: operação do software e suporte operacional aos usuários. |
| Manutenção | Atividades de quem faz a manutenção do software. |

| | |
|----------------------------------|--|
| Processos de Apoio | Auxiliam um outro processo. |
| Documentação | Registro de informações produzidas por um processo ou atividade. Inclui planejamento, projeto, desenvolvimento, produção, edição, distribuição e manutenção dos documentos necessários a gerentes, engenheiros e usuários do software. |
| Gerência de Configuração | Identificação e controle dos itens do software. Inclui: controle de armazenamento, liberações, manipulação, distribuição e modificação de cada um dos itens que compõem o software. |
| Garantia da Qualidade | Garante que os processos e produtos de software estejam em conformidade com os requisitos e os planos estabelecidos. |
| Verificação | Determina se os produtos de software de uma atividade atendem completamente aos requisitos ou condições impostas a eles. |
| Validação | Determina se os requisitos e o produto final (sistema ou software) atendem ao uso específico proposto. |
| Revisão Conjunta | Define as atividades para avaliar a situação e produtos de uma atividade de um projeto, se apropriado. |
| Auditoria | Determina adequação aos requisitos, planos e contrato, quando apropriado. |
| Resolução de Problemas | Análise e resolução dos problemas de qualquer natureza ou fonte, descobertos durante a execução do desenvolvimento, operação, manutenção ou outros processos. . |
| Processos Organizacionais | Implementam uma estrutura constituída de processos de ciclo de vida e pessoal associados, melhorando continuamente a estrutura e os processos. |
| Gerência | Gerenciamento de processos. |
| Infra-estrutura | Fornecimento de recursos para outros processos. Inclui: hardware, software, ferramentas, técnicas, padrões de desenvolvimento, operação ou manutenção. |
| Melhoria | Atividades para estabelecer, avaliar, medir, controlar e melhorar um processo de ciclo de vida de software. |
| Treinamento | Atividades para prover e manter pessoal treinado. |

A norma detalha cada um dos processos acima. Ela define ainda como eles podem ser usados de diferentes maneiras por diferentes organizações (ou parte destas), representando diversos pontos de vista para esta utilização. Cada uma destas visões representa a forma como uma organização emprega estes processos, agrupando-os de acordo com suas necessidades e objetivos.

As Visões têm o objetivo de organizar melhor a estrutura de uma empresa, para definir suas gerências e atividades alocadas às suas equipes. Existem cinco visões diferentes: contrato, gerenciamento, operação, engenharia e apoio. Veja na figura abaixo como estas visões se relacionam aos processos.



A ISO/IEC 12207 é a primeira norma internacional que descreve em detalhes os processos, atividades e tarefas que envolvem o fornecimento, desenvolvimento, operação e manutenção de produtos de software. A principal finalidade desta norma é servir de referência para os demais padrões que venham a surgir. Lançada em agosto de 1995, ela é citada em quase todos os trabalhos relacionados à Engenharia de Software desde então, inclusive aqueles relativos à qualidade. A futura norma ISO 15504 (SPICE), por exemplo, organiza seu trabalho segundo o que está descrito na 12207.

A versão brasileira da norma foi encaminhada para votação na ABNT em junho de 1997 e a expectativa da comissão encarregada da tradução é que ela se transforme em norma brasileira ainda em 1997.

CMM - Capability Maturity Model

Este "Modelo de Maturidade da Capacidade" é uma iniciativa do SEI (Software Engineering Institute) para avaliar e melhorar a capacitação de empresas que produzem software. O projeto CMM foi apoiado pelo Departamento de Defesa do Governo dos Estados Unidos, que é um grande consumidor de software e precisava de um modelo formal que permitisse selecionar os seus fornecedores de software de forma adequada. Embora não seja uma norma emitida por uma instituição internacional (como a ISO ou o IEEE), esta norma tem tido uma grande aceitação mundial, até mesmo fora do mercado americano. O modelo, publicado em 1992, não é extenso e pode ser obtido na própria Internet com facilidade. O CMM também é chamado de SW-CMM (Software CMM).

Maturidade

O CMM é um modelo para medição da maturidade de uma organização no que diz respeito ao processo de desenvolvimento de software. A definição do que é "Maturidade" pode ser melhor compreendida através da análise do quadro abaixo:

| Organizações maduras | Organizações imaturas |
|---|--|
| Papéis e responsabilidades bem definidos | Processo improvisado |
| Existe base histórica | Não existe base histórica |
| É possível julgar a qualidade do produto | Não há maneira objetiva de julgar a qualidade do produto |
| A qualidade dos produtos e processos é monitorada | Qualidade e funcionalidade do produto sacrificadas |
| O processo pode ser atualizado | Não há rigor no processo a ser seguido |
| Existe comunicação entre o gerente e seu grupo | Resolução de crises imediatas |

Níveis

O CMM classifica as organizações em cinco níveis distintos, cada um com suas características próprias. No nível 1, o das organizações mais imaturas, não há nenhuma metodologia implementada e tudo ocorre de forma desorganizada. No nível 5, o das organizações mais maduras, cada detalhe do processo de desenvolvimento está definido, quantificado e acompanhado e a organização consegue até absorver mudanças no processo sem prejudicar o desenvolvimento. Veja a tabela abaixo:

| Nível CMM | Descrição |
|--------------|---|
| 1) Inicial | O processo de desenvolvimento é desorganizado e até caótico. Poucos processos são definidos e o sucesso depende de esforços individuais e heróicos. |
| 2) Repetível | Os processos básicos de gerenciamento de projeto estão estabelecidos e permitem acompanhar custo, cronograma e funcionalidade. É possível |

| | |
|---------------|--|
| | repetir o sucesso de um processo utilizado anteriormente em outros projetos similares. |
| 3) Definido | Tanto as atividades de gerenciamento quanto de engenharia do processo de desenvolvimento de software estão documentadas, padronizadas e integradas em um padrão de desenvolvimento da organização. Todos os projetos utilizam uma versão aprovada e adaptada do processo padrão de desenvolvimento de software da organização. |
| 4) Gerenciado | São coletadas medidas detalhadas da qualidade do produto e processo de desenvolvimento de software. Tanto o produto quanto o processo de desenvolvimento de software são entendidos e controlados quantitativamente. |
| 5) Otimizado | O melhoramento contínuo do processo é conseguido através de um "feedback" quantitativo dos processos e pelo uso pioneiro de idéias e tecnologias inovadoras. |

Uma empresa no nível 1 não dá garantia de prazo, custo ou funcionalidade. No nível 2, a empresa já consegue produzir bons softwares, no prazo e a um custo previsível. O nível 3 garante um excelente nível de qualidade, tanto no produto quanto no processo de desenvolvimento como um todo. Não há, no mundo, muitas empresas que tenham chegado aos níveis 4 e 5...

Áreas-chave de processo (Key Process Areas ou KPAs)

Exceto no nível 1, todos os níveis são detalhados em áreas-chave de processo. Estas áreas são exatamente aquilo no que a organização deve focar para melhorar o seu processo de desenvolvimento de software. Para que uma empresa possa se qualificar em um determinado nível de maturidade CMM, deve estar realizando os processos relacionados às áreas-chave daquele determinado nível. Todas as áreas-chave estão citadas na tabela abaixo:

| Nível CMM | Foco | Áreas-chave de processo |
|--------------|--|--|
| 1) Inicial | Pessoas competentes e heróis | |
| 2) Repetível | Processos de gerenciamento de projetos | <ul style="list-style-type: none"> • Gerenciamento de requisitos • Planejamento do projeto • Visão geral e acompanhamento do projeto • Gerenciamento de subcontratados • Garantia da qualidade do software • Gerenciamento de configuração |
| 3) Definido | Processos de engenharia e apoio | <ul style="list-style-type: none"> • Foco do processo organizacional • Definição do processo organizacional • Programa de treinamento |

| | | |
|---------------|------------------------------------|---|
| | | <ul style="list-style-type: none"> • Gerenciamento de software integrado • Engenharia de produto de software • Coordenação intergrupos • Revisão conjunta |
| 4) Gerenciado | Qualidade do produto e do processo | <ul style="list-style-type: none"> • Gerenciamento quantitativo dos processos • Gerenciamento da qualidade de software |
| 5) Otimizado | Melhoramento contínuo do processo | <ul style="list-style-type: none"> • Prevenção de defeitos • Gerenciamento de mudanças tecnológicas • Gerenciamento de mudanças no processo |

Objetivos das áreas-chave de processo

O modelo CMM define um conjunto de dois a quatro objetivos para cada área-chave. Estes objetivos definem aquilo que deve ser alcançado no caso dos processos desta área-chave serem realmente realizados. Veja na tabela abaixo a lista destes objetivos.

| Nível CMM | Áreas-chave de processo | Objetivos |
|--------------|-----------------------------|--|
| 1) Inicial | | |
| 2) Repetível | Gerenciamento de requisitos | Os requisitos do sistema definidos para o software são controlados de forma a estabelecer um perfil mínimo a ser utilizado pela engenharia de software e pela administração Os planos, produtos e atividades do software são sempre consistentes com os requisitos de sistema definidos para o software |
| | Planejamento do projeto | Estimativas relativas ao software são documentadas para uso no planejamento e acompanhamento do projeto do software. As atividades de projeto de software e compromissos assumidos são planejados e documentados. Grupos e pessoas afetadas concordam com seus compromissos relacionados ao projeto do software. |

| | | |
|-------------|---|---|
| | Visão geral e acompanhamento do projeto | Resultados reais são acompanhados de acordo do com o planejamento do software Quando os resultados apresentam um significativo desvio do planejamento do software, são tomadas ações corretivas que são acompanhadas até o final do projeto Mudanças nos compromissos assumidos são feitas em comum acordo com os grupos e indivíduos afetados |
| | Gerenciamento de subcontratados | O contratante seleciona subcontratos qualificados O contratante e os subcontratados estão de acordo no que diz respeito aos compromissos assumidos um com o outro. O contratante e os subcontratados mantêm uma comunicação constante O contratante acompanha os resultados reais do subcontratado de acordo com os compromissos assumidos |
| | Garantia da qualidade do software | As atividades de garantia de qualidade de software são planejadas A conformidade dos produtos de software e atividades com os padrões, procedimentos e requisitos é verificada objetivamente. Os grupos e indivíduos afetados são informados das atividades de garantia de qualidade de software e de seus resultados. Questões relacionadas à não conformidade que não são resolvidas dentro do projeto de software são encaminhadas à gerência geral |
| | Gerenciamento de configuração | As atividades de gerenciamento de configuração são planejadas. Os produtos de trabalho de software são identificados, controlados e estão disponíveis. Mudanças nos produtos de trabalho identificados são controladas. Os grupos e pessoas afetadas são informados da situação atual e projetada dos produtos de trabalho de software. |
| 3) Definido | Foco do processo organizacional | São coordenadas atividades de desenvolvimento e melhoramento do processo de software em toda a organização Os pontos fortes e fracos do processo de desenvolvimento de software utilizado são identificados, de acordo com um padrão de processo. São planejadas atividades de desenvolvimento e melhoramento do processo, a nível de organização. |
| | Definição do processo organizacional | O processo padrão de desenvolvimento de software da organização é desenvolvido e mantido. A informação relacionada ao uso do processo padrão de desenvolvimento de software é coletada, revisada e disponibilizada. |

| | | |
|------------------|--|---|
| | Programa de treinamento | As atividades de treinamento são planejadas É fornecido treinamento para o desenvolvimento de habilidades e conhecimentos necessários para realizar o gerenciamento do software e as funções técnicas. As pessoas no grupo de engenharia de software e outros grupos relacionados a software recebem o treinamento necessário para realizar as suas funções |
| | Gerenciamento de software integrado | O processo de software definido para o projeto é uma versão adaptada do processo padrão de desenvolvimento de software da organização O projeto é planejado e gerenciado de acordo com o processo de desenvolvimento de software definido para o projeto |
| | Engenharia de produto de software | As atividades de engenharia de software são definidas, integradas e consistentemente realizadas para produzir o software. Os produtos de trabalho do software são mantidos consistentes entre si. |
| | Coordenação intergrupos | Todos os grupos de trabalho afetados concordam com os requisitos dos cliente. Todos os grupos de trabalho afetados concordam com os acordos entre os grupos de engenharia Os grupos de engenharia identificam, acompanham e resolvem todas as questões intergrupos. |
| | Revisão conjunta | Atividades de revisão conjunta são planejadas Defeitos nos produtos de trabalho são identificados e removidos. |
| 4) Gerenciado | Gerenciamento quantitativo dos processos | As atividades de gerenciamento quantitativo dos processos são planejadas A performance do processo de desenvolvimento de software definido para o projeto é controlada quantitativamente A capacidade do processo desenvolvimento de software padrão da organização é conhecida em termos quantitativos. |
| | Gerenciamento da qualidade de software | As atividades de gerenciamento da qualidade de software do projeto são planejadas. Objetivos mensuráveis da qualidade do produto de software e suas prioridades são definidos. O progresso real em direção à realização dos objetivos de qualidade para os produtos de software é quantificado e gerenciado. |
| 5) Otimizado | Prevenção de defeitos | As atividades de prevenção de defeitos são planejadas As causas comuns de defeitos são procuradas e identificadas As causas comuns de defeitos são priorizadas e sistematicamente eliminadas. |

| | | |
|--|--|---|
| | Gerenciamento de mudanças tecnológicas | A incorporação de mudanças tecnológicas é planejada Novas tecnologias são avaliadas para determinar seu efeito na qualidade e na produtividade Novas tecnologias adequadas são incorporadas na prática normal de toda a organização. |
| | Gerenciamento de mudanças no processo | O melhoramento contínuo do processo é planejado Toda a organização participa das atividades de melhoramento do processo de software O padrão de processo de software da organização e os processos de software de cada projeto definido são melhorados continuamente. |

Características comuns e práticas-base

As características comuns são itens a serem observados para que se possa verificar a implementação e institucionalização de cada área-chave de processo. Elas podem indicar se a área-chave de processo é eficiente, repetível e duradoura. São cinco as características comuns no modelo CMM e cada uma possui suas práticas-base a serem realizadas.

| Característica Comum | Descrição | Práticas-base relacionadas a |
|-------------------------------|--|--|
| Compromisso de realizar | Atitudes a serem tomadas pela organização para garantir que o processo se estabeleça e seja duradouro. | Estabelecimento de políticas e apadrinhamento de um gerente experiente. |
| Capacidade de realizar | Pre-requisitos que devem existir no projeto ou na organização para implementar o processo de forma competente. | Alocação de recursos, definição da estrutura organizacional e de treinamento. |
| Atividades realizadas | Papéis e os procedimentos necessários para implementar uma área-chave de processo. | Estabelecimento de planos e procedimentos, realização do trabalho, acompanhamento do trabalho e tomada de ações corretivas, se necessário. |
| Medições e análise | Necessidade de medir o processo e analisar as medições. | Realização de medições para determinar o estado e a efetividade das atividades realizadas. |
| Implementação com Verificação | Passos para garantir que as atividades são realizadas de acordo com o processo estabelecido. | Revisão, auditoria e garantia de qualidade. |

As práticas-chave descrevem as atividades que contribuem para atingir os objetivos de cada área-chave do processo. Em geral são descritas com frases simples, seguidas de descrições

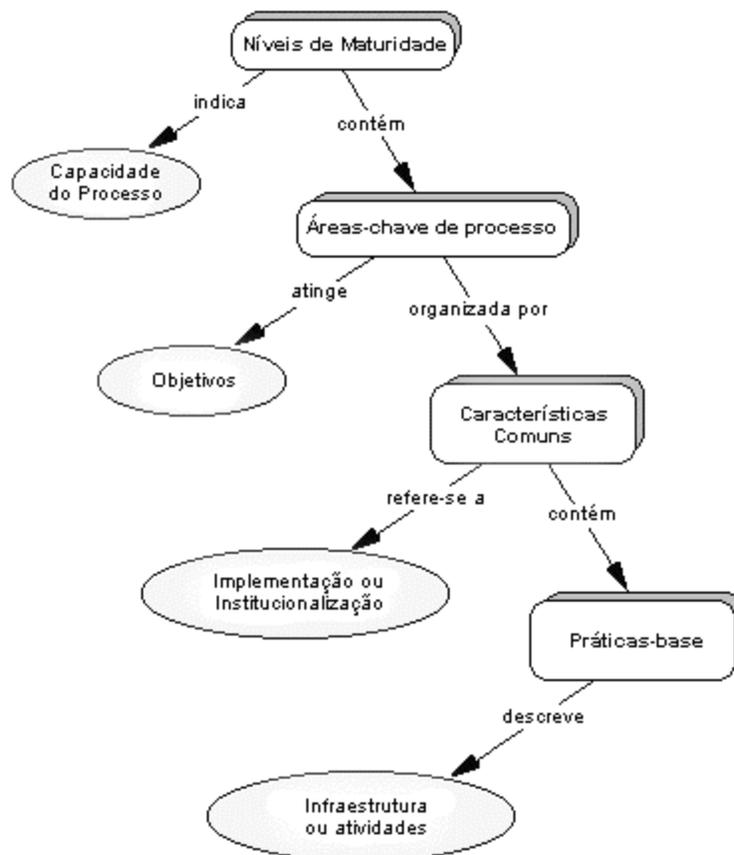
detalhadas (chamadas de subpráticas) que podem até incluir exemplos. As práticas-base devem descrever "o que" deve ser feito e não "como" os objetivos devem ser atingidos. O modelo CMM inclui um extenso documento em separado, chamado "Práticas-base para o CMM", que lista todas as práticas-chave e subpráticas para cada uma das áreas-chave de processo.

Estrutura

Em resumo, o CMM é definido em função de um conjunto de

- Níveis de maturidade
- Áreas-chave de processo
- Características comuns
- Práticas-base

Veja no gráfico abaixo como estes elementos se interligam na estrutura do CMM:



Até este ponto tínhamos falado da o CMM versão 1.1, que é a versão atual. Entretanto, o modelo CMM está sendo revisado. Foi publicado em 20/ago/97 uma segunda versão preliminar (draft B) do novo CMM v2. O SEI (Software Engineering Institute) promete a versão definitiva do novo modelo ainda para 1997. Esta versão promete corrigir e atualizar o modelo atual, além de compatibilizá-lo com padrões (ou propostas de padrões) que surgiram após o lançamento do CMM 1.1, como ISO 9000-3, ISO 12207 e ISO 15504.

PSP - Personal Software Process

O Modelo CMM é muito interessante, mas aplica-se mais a grandes empresas de software. O pessoal do Software Engineering Institute (SEI) acabou percebendo que havia a necessidade de definir um modelo mais simples, voltado para pequenas empresas ou até para um único indivíduo. Foi daí que surgiu o PSP, que significa "Processo Pessoal de Software".

Assim como o CMM, no modelo PSP, existem diversos níveis com características próprias. O modelo PSP possui os seguintes níveis:

| Nível | Nome | Atividades |
|----------------|--------------------------|--|
| PSP0 PSP0.1 | Medição Pessoal | Registro de tempo Registro de defeitos Padrão de tipos de defeitos Padrão de codificação Medida de tamanho Proposta de melhoramento do processo |
| PSP1 PSP1.1 | Planejamento Pessoal | Estimativa de tamanho Relatório de testes Planejamento de tarefas Cronogramas |
| PSP2 PSP2.1 | Qualidade Pessoal | Revisões de código Revisões de projeto Padrões de Projeto |
| PSP3 | Processo Cíclico Pessoal | Desenvolvimento cíclico |

No nível de **Medição Pessoal**, você aprende a registrar o tempo gasto em cada etapa do ciclo do desenvolvimento, registrando ainda os defeitos encontrados. Isto é conseguido através do uso de formulários adequados. O nível PSP0.1 inclui o uso de um padrão de codificação, de medidas padronizadas e do formulário de proposta de melhoramento do processo.

No nível de **Planejamento Pessoal**, você aprende a planejar. A idéia geral é obter a capacidade de estimar quanto tempo levará para realizar uma tarefa baseado nas medições feitas em tarefas semelhantes anteriormente. Neste nível aprende-se a assumir compromissos que podem realmente ser cumpridos. O nível PSP1.1 inclui o planejamento de tarefas e a elaboração de cronogramas.

No nível de **Qualidade Pessoal** você aprende a lidar com seus erros. Deve-se ter uma idéia precisa de quantos erros são cometidos (em média) em cada fase do ciclo de desenvolvimento. O modelo PSP mostra que a forma mais adequada para tratar erros é evitá-los desde a sua origem. Você deve utilizar os dados sobre defeitos já coletados para criar uma lista de verificação (checklist) a ser utilizada em suas revisões de projeto e de código. O nível PSP2.1 inclui a criação de padrões de projeto, bem como métodos de análise e prevenção de defeitos.

O nível de **Processo Cíclico Pessoal** é a última etapa do PSP. Neste nível, o PSP sai do desenvolvimento de pequenos programas para tratar do desenvolvimento de projetos maiores, embora ainda em nível pessoal. A idéia é dividir os grandes projetos em pequenos projetos que possam ser tratados no PSP2. Neste caso, o desenvolvimento acontece em passos incrementais.

O treinamento do PSP é realizado através de 10 exercícios de desenvolvimento de programas. Além servirem como exemplos de desenvolvimento, os exercícios propostos pelo treinamento do PSP são pequenos utilitários que ajudam você a aplicar o PSP, pois permitem medir o número de linhas e objetos nos seus programas, calcular desvio padrão, prever intervalos etc.

Uma descrição completa deste modelo e do treinamento proposto pode ser encontrada no livro "Introduction to the Personal Software Process", publicado em 1996 por Watts Humphery, o pai do PSP.

SPICE - Software Process Improvement and Capability dEtermination - ISO 15504

Introdução

O SPICE é uma norma em elaboração conjunta pela ISO e pelo IEC. Ela constitui-se de um padrão para a avaliação do processo de software, visando determinar a capacitação de uma organização. A norma visa ainda orientar a organização para uma melhoria contínua do processo. Ela cobre todos os aspectos da Qualidade do Processo de Software e está sendo elaborada num esforço conjunto de cinco centros técnicos espalhados pelo mundo (EUA, Canadá/América Latina, Europa, Pacífico Norte e Pacífico Sul).

Um grupo de estudos da ABNT está participando do processo de desenvolvimento, além de trabalhar na tradução das versões preliminares da norma para o português. Tenho a honra de participar como membro colaborador da comissão SPICE da ABNT.

O SPICE inclui um modelo de referência, que serve de base para o processo de avaliação. Este modelo é um conjunto padronizado de processos fundamentais, que orientam para uma boa engenharia de software. Este modelo é dividido em cinco grandes categorias de processo: Cliente-Fornecedor, Engenharia, Suporte, Gerência e Organização. Cada uma destas categorias é detalhada em processos mais específicos. Tudo isso é descrito em detalhes pela norma.

Além dos processos, o SPICE define também os 6 níveis de capacitação de cada processo, que pode ser incompleto, executado, gerenciado, estabelecido, previsível e otimizado. O resultado de uma avaliação, portanto, um perfil da instituição em forma de matriz, onde temos os processos nas linhas e os níveis nas colunas.

Categorias e Processos

Uma das contribuições do modelo SPICE é definir em seu modelo de referência todos os processos envolvidos no desenvolvimento de software, agrupados em categorias. Observe no quadro abaixo a estrutura completa das categorias, dos processos de cada categoria:

| Processo | Descrição |
|---|---|
| CUS - Cliente-Fornecedor | |
| Processos que impactam diretamente os produtos e serviços de software na fornecedor para o cliente. | |
| CUS.1 | Adquirir Software |
| CUS.2 | Gerenciar necessidades do Cliente |
| CUS.3 | Fornecer Software |
| CUS.4 | Operar Software |
| CUS.5 | Prover Serviço ao Cliente |
| ENG - Engenharia | |
| Processos que especificam, implementam ou mantêm um sistema ou produto de software e sua documentação | |
| ENG.1 | Desenvolver requisitos e o projeto do sistema |
| ENG.2 | Desenvolver requisitos de software |
| ENG.3 | Desenvolver o projeto do software |
| ENG.4 | Implementar o projeto do software |
| ENG.5 | Integrar e testar o software |
| ENG.6 | Integrar e testar o sistema |
| ENG.7 | Manter o sistema e o software |
| SUP - Suporte | |
| Processos que podem ser empregados por qualquer um dos outros processos | |
| SUP.1 | Desenvolver a documentação |
| SUP.2 | Desempenhar a gerência de configuração |
| SUP.3 | Executar a garantia da qualidade |
| SUP.4 | Executar a verificação dos produtos de trabalho |
| SUP.5 | Executar a validação dos produtos de trabalho |

| | |
|--|---------------------------------------|
| SUP.6 | Executar revisões conjuntas |
| SUP.7 | Executar auditorias |
| SUP.8 | Executar resolução de problemas |
| MAN - Gerência | |
| Processos que contém práticas de natureza genérica que podem ser usadas por quem gerencia projetos ou processos dentro de um ciclo de vida de software | |
| MAN.1 | Gerenciar o projeto |
| MAN.2 | Gerenciar a qualidade |
| MAN.3 | Gerenciar riscos |
| MAN.4 | Gerenciar subcontratantes |
| ORG - Organização | |
| Processos que estabelecem os objetivos de negócios da organização | |
| ORG.1 | Construir o negócio |
| ORG.2 | Definir o processo |
| ORG.3 | Melhorar o processo |
| ORG.4 | Prover recursos de treinamento |
| ORG.5 | Prover infra-estrutura organizacional |

A norma define detalhes de cada um dos processos mencionados acima. Para cada um deles existe uma definição mais detalhada, uma lista dos resultados da sua implementação bem sucedida e uma descrição detalhada de cada uma das práticas básicas.

Níveis de Capacitação

O SPICE, entretanto, não se limita a listar categorias e processos. Seu principal objetivo, na realidade, é avaliar a capacitação da organização em cada processo e permitir a sua melhoria. O modelo de referência do SPICE inclui seis níveis de capacitação. Cada um dos processos mencionados acima deve ser classificado nestes níveis. Os níveis são descritos a seguir:

| Nível | Nome | Descrição |
|-------|------------|---|
| 0 | Incompleto | Há uma falha geral em realizar o objetivo do processo. Não existem produtos de trabalho nem saídas do processo facilmente identificáveis. |
| 1 | Realizado | O objetivo do processo em geral é atingido, embora não necessariamente de forma planejada e controlada. Há um consenso na organização de que as ações devem ser realizadas e quando são necessárias. Existem produtos de trabalho para o processo e eles são utilizados para atestar o atendimento dos objetivos. |
| 2 | Gerenciado | O processo produz os produtos de trabalho com qualidade aceitável e dentro do prazo. Isto é feito de forma planejada e controlada. Os produtos de trabalho estão de acordo com padrões e requisitos. |

| | | |
|---|--------------|---|
| 3 | Estabelecido | O processo é realizado e gerenciado usando um processo definido, baseado em princípios de Engenharia de Software. As pessoas que implementam o processo usam processos aprovados, que são versões adaptadas do processo padrão documentado. |
| 4 | Predizível | O processo é realizado de forma consistente, dentro dos limites de controle, para atingir os objetivos. Medidas da realização do processo são coletadas e analisadas. Isto leva a um entendimento quantitativo da capacitação do processo a uma habilidade de prever a realização. |
| 5 | Otimizado | A realização do processo é otimizada para atender às necessidades atuais e futuras do negócio. O processo atinge seus objetivos de negócio e consegue ser repetido. São estabelecidos objetivos quantitativos de eficácia e eficiência para o processo, segundo os objetivos da organização. A monitoração consistente do processo segundo estes objetivos é conseguida obtendo feedback quantitativo e o melhoramento é conseguido pela análise dos resultados. A otimização do processo envolve o uso piloto de idéias e tecnologias inovadoras, além da mudança de processos ineficientes para atingir os objetivos definidos. |

Os 9 manuais do SPICE

Esta norma se constituirá de um conjunto de 9 manuais, totalizando quase 400 páginas, conforme o detalhamento a seguir, baseado na atual versão preliminar (draft).

| Parte | Páginas | Descrição |
|-------|---------|---|
| 1 | 16 | Guia de Introdução e Conceitos |
| 2 | 38 | Modelo de referência para processos e capacidade de processos |
| 3 | 7 | Realizando uma avaliação |
| 4 | 36 | Guia para realização de uma avaliação |
| 5 | 145 | Um modelo de avaliação e guia de indicadores |
| 6 | 31 | Guia para qualificação de avaliadores |
| 7 | 47 | Guia para uso no melhoramento de processos |
| 8 | 25 | Guia para uso na determinação da capacidade do processo de fornecedor |
| 9 | 9 | Vocabulário |

Recentemente, os trabalhos do SPICE evoluíram bastante. Os otimistas acreditam que a norma sairá dentro de mais um ou dois anos. Tecnicamente, o trabalho pode alongar-se até o ano de 2001...

Sites sobre Qualidade de Software

Páginas Nacionais sobre Qualidade de Software

- [INSOFT - Qualidade de Software](#)
- [CTI-TAQS - Tecnologia para Avaliação de Qualidade de Software](#)
- [CITS - Centro Internacional de Tecnologia de Software](#)
- [CNPQ - Tecnologia para Avaliação de Qualidade de Software](#)
- [SGQ - Empresa de Consultoria em Qualidade](#)
- [ErgoList - Qualidade ergonômica da interface com o usuário](#)
- [ABNT - Associação Brasileira de Normas Técnicas](#)
- [ABNT - Subcomitê de Software](#)
- [ABNT - Comissão de Estudos SPICE](#)
- [MCT - Qualidade no Setor de Software Brasileiro](#)
- [SSQP/SW do PBQP - Subcomitê Setorial da Qualidade e Produtividade em Software do Programa Brasileiro de Qualidade e Produtividade](#)

Qualidade, Qualidade de Software

- [Quality Resources Online](#)
- [Griffith University, Australian Software Quality Institute](#)
- [Deming Electronic Network](#)
- [Quality Function Deployment](#)
- [US Air Force Software Tech Support Center](#)
- [The Software Quality Page](#)
- [Software Process Newsletter](#)
- [Software Productivity Centre](#)
- [R.S. Pressman](#)
- [The Dilbert Perspective](#)
- [Quality Week - Congresso sobre Qualidade](#)
- [CITS - Conferência Internacional de Tecnologia de Software](#)

Organizações Relacionadas à Qualidade de Software

- [IEEE Home Page](#)
- [IEEE Standards](#)
- [IEEE Computer Society](#)
- [ISO Online](#)
- [American Society for Quality Control](#)
- [Association for Computing Machinery](#)
- [Australian Software Quality Research Inst](#)
- [European Software Institute](#)
- [Inst for the Cert of Computing Prof](#)
- [Software Engineering Institute](#)
- [National Research Center of Canada](#)
- [Software Assurance Tech Center](#)

- [Software Inspections and Review Org](#)
- [Software Quality Institute](#)

Engenharia de Software

- [Software Engineering Online Service](#)
- [IEEE Technical Council on Software Engineering](#)
- [Software Engineering FAQ](#)
- [IIT Software Engineering](#)
- [Software Engineering Process Maturity Resources](#)
- [Software Productivity Consortium](#)
- [WWW Virtual Library - Software Engineering](#)
- [Tero Ahtee's software engineering reference list](#)
- [Software Productivity Research - Articles, Books, and White Papers](#)
- [Software Engineering Home Page](#)
- [Software Engineering Main Page](#)
- [Software Engineering Internet Resource List](#)
- [Software Engineering](#)
- [Williamson's Software Engineering Pages](#)
- [SPI WWW Sources](#)
- [Software Reengineering Web Home Page](#)

Métricas de Software

- [Software Metrics - Forum](#)
- [US Army Software Metrics System](#)
- [Object Oriented Software Metrics](#)
- [Software Metrics & Static Analysis](#)
- [Software Research Laboratory](#)

Teste de Software

- [Software Testing Article Archives](#)
- [Software Testing - Usenet](#)
- [Test Automation Guidance](#)
- [Testing Techniques Newsletter](#)
- [Technical Review Archive](#)

Qualidade do Processo de Software

- [SEI - Software Engineering Institute](#)
- [Defense Info Systems's Software Process Improvement Program](#)
- [Expert System Program and System Development Group](#)
- [Software Development: People, Process, Technology](#)
- [European Software Institute \(ESI\)](#)
- [Software Technology Support Center](#)

- [Software Design & Engineering: Process Management Issues](#)
- [Useful Internet Sites related to Software Process Improvement](#)
- [Software Productivity Center's Process Improvement Resources](#)
- [Managing Software Development](#)
- [Project Management Software FAQ](#)
- [Software Process, Quality and ISO 9000](#)

Modelo CMM

- [SEI - CMM - Capability Maturity Model](#)
- [SEI - CMM 1.1 - Texto completo para download](#)
- [SEI - CMM v2 - A nova versão do CMM](#)
- [SEI - P-CMM - People Capability Maturity Model](#)
- [Capability Maturity Model](#)
- [CMM Level 2 Focus Group](#)
- [CMM Questions](#)
- [Rochester Software Quality Association](#)
- [Integrated Product Development - CMM](#)
- [Software Engineering - CMM](#)
- [Software Factory Information Database](#)
- [ITG Software Process Improvement \(SPI\)](#)

Modelo SPICE

- [SPICE - European Software Institute](#)
- [SPICE - Texto da versão 1.00 draft para download](#)
- [SPICE - Australian Software Quality Research Institute](#)
- [SPICE - Artigo de Arnoldo Díaz Olavarrieta](#)

Outros Modelos

- [PSP - Personal Software Process](#)
- [The Trillium Model](#)
- [The Bootstrap Methodology](#)
- [Cleanroom Software Engineering](#)
- [Cleanroom Software Engineering Tutorial](#)

Artigos sobre Qualidade em Revistas On-Line

- [The Quality Factor](#) (Byte Internacional, Abril/97). Visão geral sobre ISO-9000-3 e CMM
- [How Software Doesn't Work](#) (Byte Internacional, Dezembro/95) Reportagem de Capa sobre Qualidade.
- [Qualidade de Software](#) (Developers Magazine, Junho/97) Reportagem de Capa sobre Qualidade de SW

Congressos de Engenharia e Qualidade de Software

- [CITS - Congresso Internacional de Tecnologia de Software](#)
- [SBES - Simpósio Brasileiro de Engenharia de Software](#)
- [Metrics' 97](#)
- [Automated Software Engineering - ASE'97](#)
- [IASTED International Conference Software Engineering](#)
- [Simposio Latinoamericano de Calidad y Productividad en Desarrollo de Software](#)
- [10h Software Engineering Process Group Conference: SEPG 98](#)
- [QWE97 - Quality Week Europe](#)
- [Software Development Seminars](#)
- [Systems Testing & Quality Assurance Techniques](#)
- [ASM Applications of Software Measurement](#)
- [IEEE International High Level Design Validation and Test Workshop](#)
- [STAR International Conference on Software Testing Analysis & Review](#)