

## **Introdução ao TMM – Test Maturity Model**

Emerson Rios  
Ricardo Cristalli

### **Introdução ao processo de teste estruturado**

Durante as décadas de 60, 70 e 80, a atividade de teste era executada no fim do ciclo de desenvolvimento, normalmente pelos próprios analistas de sistemas ou desenvolvedores. O foco era avaliar se o sistema estava funcionando e o objetivo era procurar erros. Não haviam ainda técnicos especializados na atividade de testar aplicações e nem metodologias estruturadas de teste.

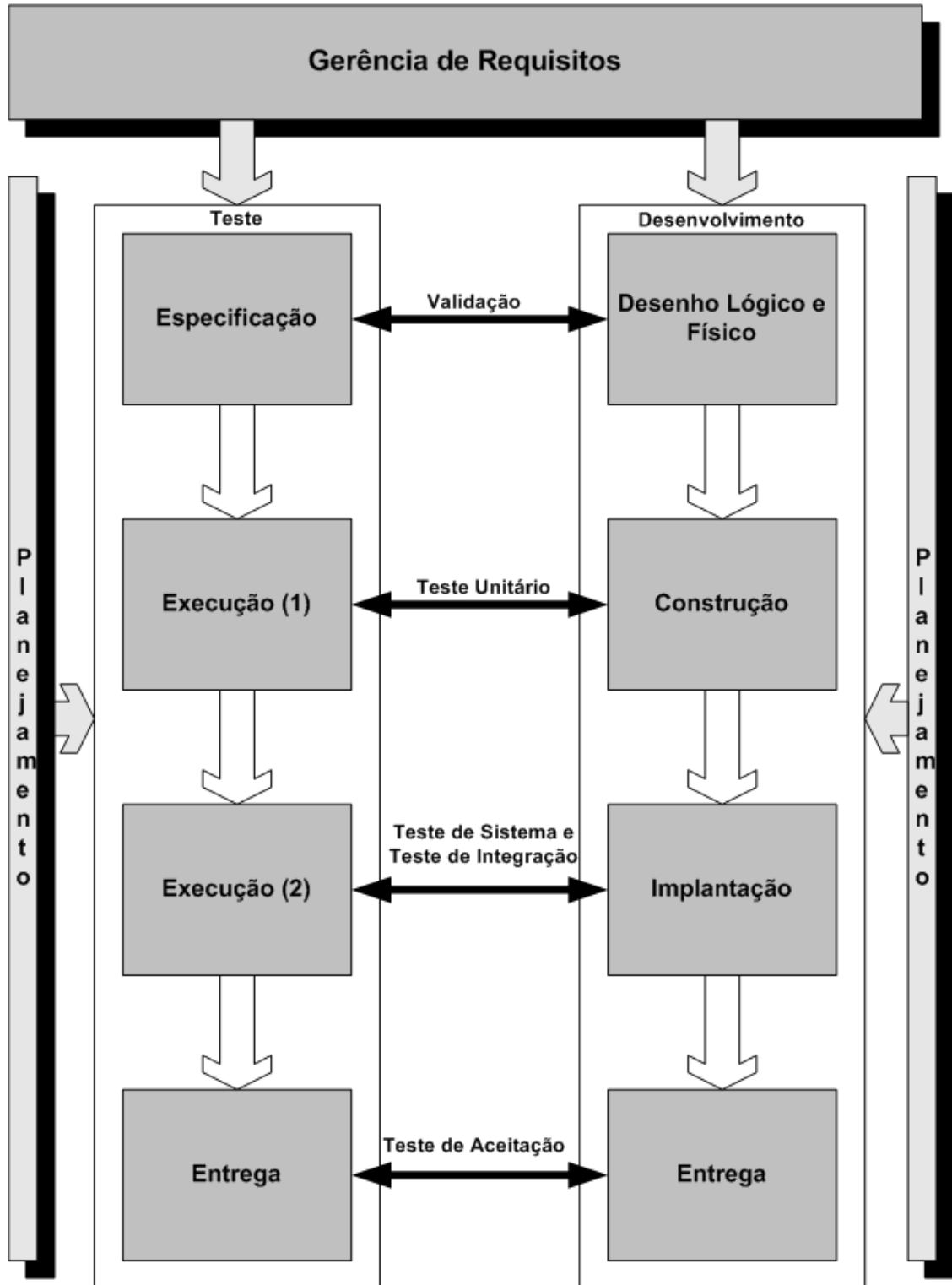
Apesar desta realidade ainda permanecer na maioria das empresas até os dias atuais, foi em 1979 que Glenford Myers publicou aquele que atualmente ainda é considerado um dos melhores livros de teste existentes no mercado com o título de “The Art of Software Testing”. Este livro continua sendo a bíblia de muitos dos testadores do século 21. Myers basicamente lembrava que testar era procurar defeitos e não provar que o software estava funcionando.

Um artigo publicado na revista Communications of the ACM com o título “The Growth of Software Testing”, escrito por David Gelperin e Bill Hetzel descrevia um processo de evolução dos testes e lançava um documento que chamou de Plano de Testes. Esse documento, que até hoje ainda é usado, deveria ser escrito a partir dos requisitos do sistema e por si só já ajudava a reduzir a quantidade de defeitos dos sistemas, dando aos testadores os objetivos a serem alcançados durante a atividade de teste.

As décadas de 80 e 90 marcaram o surgimento dos grupos de garantia de qualidade na estrutura da área de TI, escritórios de projetos, e mais recentemente os grupos de teste, quando a atividade de testar software passou a fazer parte de um processo independente do processo de desenvolver software, embora continuassem integradas. Os modelos de melhoria do processo de software são basicamente voltados para a melhoria do desenvolvimento do software, como o CMMI e o mps.BR, e o seu objetivo é criar produtos de software melhores, mas não garantem que esses softwares serão testados adequadamente.

A criação de um processo independente de teste demandou algumas necessidades de metodologias, métricas e melhorias, que já existiam no processo de desenvolvimento de software original, mas que precisavam ser adaptadas a este novo processo.

## Modelo de Integração entre os processos de desenvolvimento e teste

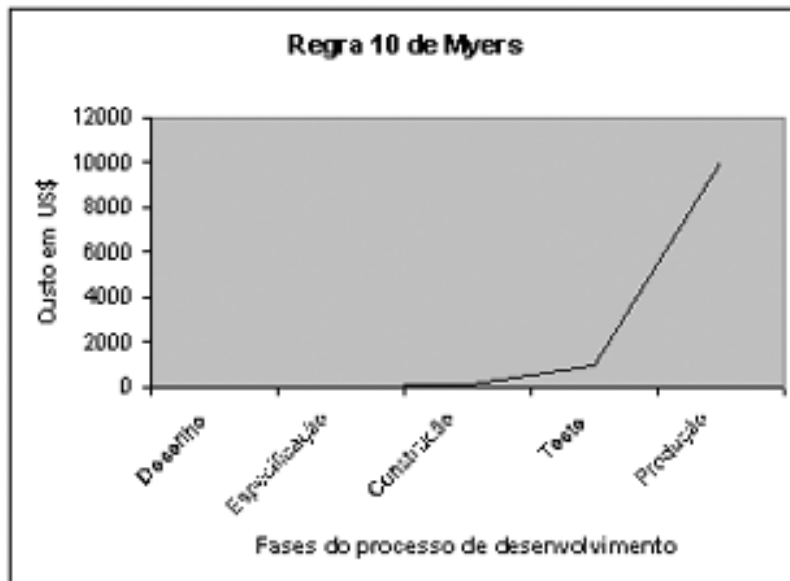


Em 1996 foi desenvolvido por técnicos do Illinois Institute of Technology (Illinois State University) um modelo para permitir fazer um diagnóstico do processo de teste dentro das empresas, definindo o seu nível de maturidade e a partir desta avaliação definir um roteiro gradativo de melhorias.

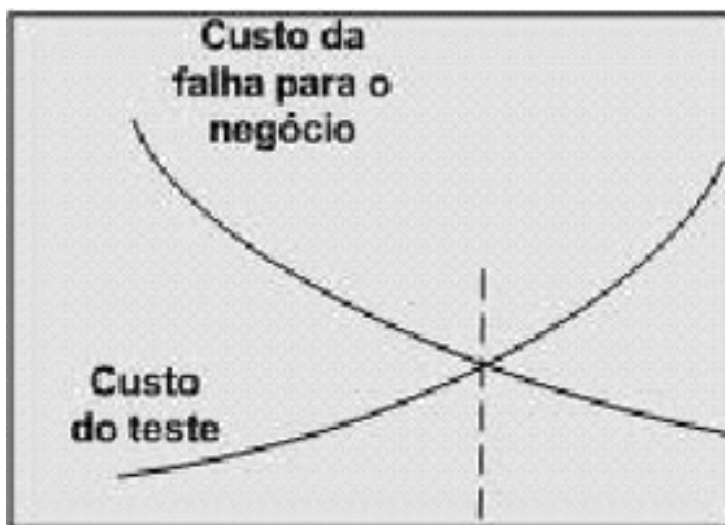
### **Modelos de avaliação da maturidade do processo de teste**

Existem vários modelos de avaliação da maturidade do processo de testes. Alguns como o TPI (Test Process Improvement), são bastante usados na Europa e outros como o TMM (Teste Maturity Model) e TCMM (Test Capability Maturity Model) que são mais populares nos Estados Unidos. Isso também ocorre com relação ao CMMI (Estados Unidos) e, o que foi durante um período, o Bootstrap (Europa). Não pretendemos neste artigo entrar no mérito de qual dos modelos é o melhor ou o mais popular, apenas escolhemos um deles, no caso o TMM, que é o mais popular nos EUA, e vamos procurar mostrar o seu relacionamento com o CMMI. Para aqueles que não estão acostumados com os modelos de melhoria de processos de teste, eu gostaria também de lembrar que o TPI, bastante usado na Europa, é também um dos mais consistentes modelos nesta área. Um dos autores, Emerson Rios, modestamente, já criou um modelo junto com o Trayahu Moreira, publicado no nosso livro Teste de Software, que foi editado pela AltaBooks, e que agora se encontra na sua segunda edição. Na verdade o nosso modelo foi apenas uma tentativa de dar uma visão mais didática ao processo de melhoria, de modo que os leitores iniciantes pudessem entender a finalidade desses modelos.

A atividade de testar componentes de software, dentro de um processo de desenvolvimento, é muito importante pois é através dela que sai o aval para a garantia dos produtos que são entregues aos usuários ou clientes. A falta de maturidade do processo de testes muitas vezes leva a organização a tratá-lo como uma atividade menor e sem importância. Nós sabemos que milhares de empresas amargam sérios prejuízos em todo o mundo por causa de defeitos de softwares. Essas organizações, apesar de saberem que o custo da correção de um defeito em ambiente de produção chega a ser milhares de vezes maior do que a sua correção no início do processo de desenvolvimento, ainda insistem em negligenciar o processo de teste. Os sinais do mundo globalizado estão mostrando outros caminhos, como, por exemplo, aquele trilhado pela Índia, país extremamente pobre, que é uma potência mundial quando se trata de tecnologia da informação. Este país exporta 10 bilhões de dólares de produtos ligados a esta área do conhecimento e possui cerca de 80 empresas certificadas entre os diversos níveis do CMMI, enquanto o Brasil, segundo o MCT, 21 (vinte e uma) empresas foram certificadas CMMI até agosto de 2006, sendo que 6 (seis) são nível 5 e 4 (quatro) nível 3, ficando as demais no nível 2 (onze empresas).



A regra 10 de Myers indica que o custo da correção dos defeitos tende a ser cada vez menor tanto mais tarde ele for descoberto. Defeitos encontrados nas fases iniciais da etapa de desenvolvimento do software são mais baratos de serem corrigidos do que aqueles encontrados na produção.



Por outro lado existe também um momento onde o custo de prosseguir o teste será mais caro do que a falha causada por um defeito encontrado tardiamente. Os modelos de melhoria de processos têm por objetivo permitir que através de um processo eficiente os defeitos venham a ser encontrados no momento em que custará mais barato corrigi-los.

### Descrevendo o TMM

Os modelos de avaliação da maturidade do processo de desenvolvimento de software, tais como CMMI (Capability Maturity Model Integrated), ISO-15504-SPICE e mps.BR não tratam adequadamente o processo de teste, razão pela qual o Illinois Institute of Technology criou o TMM-Test Maturity Model. O objetivo do TMM é dar suporte às organizações na melhoria do processo de testes e tomou como base os seguintes paradigmas:

- É um modelo complementar ao CMMI com o qual mantém compatibilidade;
- É baseado na avaliação da situação atual do processo de testes através de regras claras e objetivas;
- É uma linha para a melhoria contínua do processo de testes;
- É um modelo baseado nas melhores práticas de teste existentes no mercado.

Evidentemente que um modelo de melhoria de processo, como os citados anteriormente, poderia com pouca mudança ser adequado a um processo de teste, porém neste artigo iremos apenas nos ater ao uso do TMM e em alguns momentos faremos uma comparação com o CMMI.

O propósito do TMM segundo os seus autores está calcado nas seguintes afirmativas:

- Deve ser uma forma de avaliação interna da equipe para identificar o seu correto estado de capacitação;
- Dever ser claro para que as gerências superiores possam claramente aceita-lo como um programa de melhoria de teste;
- Deve permitir que a equipe de garantia de qualidade possa desenvolver e implementar planos de melhoria de processo;
- Deve permitir que as equipes de desenvolvimento possam também melhorar os seus testes;
- Os usuários e clientes devem também poder ocupar o seu papel no processo de melhoria.

Embora exista um paralelismo entre o CMMI e o TMM os seus níveis não necessariamente se correspondem, como veremos adiante neste artigo. Os 5 (cinco) níveis do TMM são os seguintes:

Tabela 1

<b>Níveis</b>	<b>Descrição dos níveis</b>
<b>1</b>	Inicial
<b>2</b>	Fase de definição
<b>3</b>	Integração
<b>4</b>	Gestão e medições
<b>5</b>	Otimização, prevenção de defeitos e controle de qualidade

Da mesma forma que o CMMI onde o nível 1 não possui nenhuma Área Chave de Processo associada, o mesmo ocorre com o TMM, cujas metas de maturidade são as seguintes:

Tabela 2

<b>Níveis</b>	<b>Obj</b>	<b>Descrição do objetivo</b>	<b>Explicações</b>
1	-	Teste normalmente feito pela equipe de desenvolvimento de forma rudimentar	A atividade de testar é um processo caótico e não existe uma diferença entre testar e buscar erros. O teste é executado sem ferramentas, equipes treinadas e outros recursos.
2	1	Desenvolver os objetivos do teste	O propósito do teste é mostrar que o software funciona.
	2	Iniciar um processo de planejamento de teste	
	3	Institucionalizar técnicas e métodos básicos de teste	
3	1	Estabelecer uma organização de teste de software	O propósito do teste é mostrar que o software não funciona
	2	Integrar o teste no ciclo de vida do software	
	3	Controlar e monitorar o processo de teste	
	4	Estabelecer um programa de treinamento	
4	1	Estabelecer um programa amplo de revisão	O propósito do teste não é provar nada mas reduzir os riscos causados pelos defeitos a níveis aceitáveis.
	2	Estabelecer um programa de medições de teste	
	3	Evoluir a qualidade do software	
5	1	Aplicar processos de prevenção de defeitos	A atividade de testar não é um ato, mas uma disciplina mental que resulta em baixos riscos para o software com pouco esforço de teste.
	2	Controlar a qualidade	
	3	Otimizar o processo de teste	

O TMM, a exemplo do CMMI, também possui um modelo de validação, baseado num questionário, onde é verificado o nível de maturidade da área de TI em relação ao teste. Definido o nível, a equipe de testes deverá se submeter a um processo de migração para um nível superior de maturidade, o que vai significar uma melhoria no processo de testes. Para isso algumas práticas devem ser adotadas baseadas nas metas estabelecidas para o nível a ser alcançado.

### Relação entre os modelos TMM e CMMI

Tabela 3

Níveis do TMM	Níveis do CMMI correlatos	Áreas chaves de processo correspondentes
1	1	
2	2	Gerência de Requisitos Gerência de Configuração Planejamento do Projeto de Software
3	2	Garantia de Qualidade de Software e Supervisão e Acompanhamento e Monitoração de Projeto de Software
	3	Foco no Processo da Organização Definição do Processo da Organização e Treinamento Validação
4	3	Revisão por parceiros Verificação
	4	Gerência Quantitativa de Processo
5	5	Análise e Decisão

O quadro mostra que existe uma correlação aproximada mas não precisa entre os níveis do TMM e o que seria o nível correspondente no CMMI. Como podemos observar não existe uma correspondência direta, pois a evolução no TMM tende a ser mais rápida, embora exista uma ligação muito forte com a evolução do CMMI. Por exemplo, alcança-se o nível 3 do TMM enquanto a área de desenvolvimento ainda está no nível 2. No entanto é impossível migrar-se para o TMM nível 4 sem que a organização esteja no nível 3, para que seja mantido o equilíbrio entre os processos.

### Estrutura do modelo de maturidade TMM

Cada nível deverá ter a seguinte divisão:

Objetivos de maturidade;

Sub-objetivos de maturidade;

Atividades, tarefas e responsabilidades.

Os objetivos de maturidade são equivalentes às áreas de processo do CMMI. O TMM possui 13 (treze) objetivos de maturidade nos seus 5 (cinco) níveis.

Para cada objetivo de maturidade, sub-objetivos mais concretos são definidos de tal forma que o objetivo possa ser sustentado adequadamente.

Atividades e tarefas são ações específicas que precisam ser tomadas para melhorar a capacidade de teste e as responsabilidades pela sua execução são passadas para gerentes, desenvolvedores/testadores e usuários/clientes.

O TMM foi baseado nas seguintes fontes de conhecimento:

- CMMI de onde o conceito de níveis de maturidade, com todas as suas divisões e sub-divisões, foi tirado;
- O modelo de teste de Gelperin e Hetzel;
- Práticas industriais correntes;
- As fases progressivas de Beizer de um modelo mental de testadores, que mostra que a maturidade de cada organização é conseguida através de habilidades, conhecimentos e atitudes das pessoas que trabalham nela.

### **Avaliação dos níveis de maturidade**

Em nenhum material do TMM que tivemos acesso encontramos em algum lugar uma descrição do que seriam as possíveis evidências que deveríamos procurar para avaliar se a organização atingiu o nível de maturidade desejado. A experiência em trabalhos de avaliação em empresas em CMMI e mps.BR nos levou a sugerir no decorrer desse trabalho possíveis evidências que deveriam ser buscadas pelo avaliador. Cabe portanto esclarecer, que, sempre que citarmos evidências, essas foram sugeridas por esses autores e não fazem parte do documento original criado pelo Illinois Institute of Technology.

#### **Nível 1 - Inicial**

Neste nível os testes são executados de forma caótica. Possivelmente não existe uma equipe responsável pela execução dos testes que via de regra são executados pelos próprios desenvolvedores. O nível de qualidade é bastante baixo e a incidência de defeitos encontrados em produção é bastante alta.

#### **Nível 2 – Fase de definição**

Para que a área de testes consiga cumprir as metas para atingir o nível 2 precisa implantar os seguintes objetivos de maturidade:

- Desenvolver os objetivos de teste;
- Iniciar um processo de planejamento de teste;
- Institucionalizar técnicas e métodos básicos de teste.

Neste nível o teste já tem um ciclo de vida e é planejado, além de ser suportado por técnicas e métodos. Desta forma pode ser repetido para cada um projeto de software. Existe uma separação clara nos níveis de teste.

### **Desenvolver os objetivos de teste**

A organização deve separar claramente os níveis de teste, ou seja, quem executa os testes unitários, de integração, de sistema e de aceitação.

Os sub-objetivos de maturidade são os seguintes:

- A organização deve formar uma equipe de teste para cada um dos seus níveis de teste.  
Evidências possíveis: Deve haver um documento definindo quem irá executar cada nível de teste (unitário, integração, sistema e aceitação).
- A equipe de teste deve desenvolver e registrar os seus objetivos;  
Evidências possíveis: A equipe encarregada pelos testes deve definir claramente os seus objetivos e registrá-los num documento.
- Todos os documentos criados pela equipe de teste, e especificamente os seus objetivos, devem ser distribuídos por todos os gerentes de projetos e desenvolvedores.  
Evidências possíveis: Os documentos criados pela equipe de teste devem ser distribuídos para todos os envolvidos. O comum seria isso estar registrado no plano do projeto do software ou no plano de teste identificado como plano de comunicação, conforme definido no próximo sub-objetivo.. Uma lista de objetivos claros a serem alcançados pela equipe deve ser suficiente para atender a esse sub-objetivo.
- Os objetivos de teste devem estar nos planos de teste. Neste caso a empresa poderá já trabalhar com um documento definido para este fim como o chamado Plano de Teste, ou ter esses objetivos embutidos no plano do projeto de desenvolvimento.  
Evidências possíveis: Deve haver um plano de teste ou um plano do projeto de software que contemple esses objetivos. Lembre-se que neste nível o teste pode ainda estar sendo feito por uma equipe montada exclusivamente para este fim.

### **Iniciar um processo de planejamento de teste**

O processo de planejamento espera-se deva seguir os preceitos mínimos para a elaboração de um plano de teste bem definido. Um caminho consistente é aquele definido pelo PMI. O Plano de Teste pode ser um documento separado e preparado exclusivamente pela equipe de teste ou pode estar embutido no plano do projeto de desenvolvimento do software. Entendemos que um plano específico deve ser muito mais fácil de entender e por isso o melhor caminho é a adoção de um Plano de Teste específico para o projeto de teste.

Os sub-objetivos de maturidade são os seguintes:

- Deve ser definido um documento de políticas de planejamento de teste que abranja toda a organização e esta política deve ser apoiada pelos diversos níveis de administração.  
Evidências possíveis: A organização ou empresa deve ter um documento de políticas onde o planejamento de teste esteja claramente definido.
- Deve haver um modelo de plano de teste institucionalizado na organização e seguido pela equipe de teste e pela equipe de desenvolvimento;  
Evidências possíveis: A organização ou empresa deve ter um local único que seja de livre acesso aos grupos envolvidos para armazenar o template previamente definido e institucionalizados a ser usado na elaboração do Plano de Teste.
- Os líderes de projeto e outros técnicos devem ser treinados para elaborar e usar o plano de teste.  
Evidências possíveis: A organização ou empresa deve planejar e executar treinamentos de institucionalização dos processos e as suas listas de presença devem ser arquivadas. Os templates definidos devem conter instruções de preenchimento para facilitar as auditorias.
- Os procedimentos para a elaboração do plano de teste devem contemplar como entrada para este documento os requisitos do software elaborados pelos usuários;  
Evidências possíveis: A organização ou empresa deve manter de forma controlada as versões das especificações / requisitos elaborados. Os procedimentos para elaborar o plano de teste devem claramente definir como entrada os requisitos do software.
- O plano deve ser controlado, alterado, avaliado e monitorado, inclusive, com o apoio de todos os gerentes.  
Evidências possíveis: Devem existir registros que claramente mostrem que o plano de teste está sendo monitorado. Uma boa evidência seriam registros de reuniões onde claramente este assunto tenha sido tratado. A organização ou empresa deve ter o controle das alterações das versões e estas devem sofrer aprovações pelos grupos envolvidos. Essas aprovações devem ser armazenadas de forma a propiciar o rastreamento nas auditorias.

### **Institucionalizar técnicas e métodos básicos de teste**

Os sub-objetivos de maturidade são os seguintes:

- Os gerentes devem institucionalizar um grupo de políticas que garantam que as técnicas e os métodos ou metodologias definidos sejam aplicados por toda a organização;  
Evidências possíveis: As listas de presenças dos treinamentos de institucionalização devem ser armazenadas e a cada novo profissional deve ser informado da existência das políticas e o mesmo deve tomar ciência.
- Uma equipe de teste deve ser formada para estudar, avaliar e recomendar um grupo de técnicas básicas de teste, assim como ferramentas, que suportem a execução dos testes na organização.  
Evidências possíveis: A organização ou empresa deve manter uma equipe com conhecimento em teste, com perfil definido para as atividades a serem desempenhadas. Esta equipe deve comprovar experiência e/ou treinamentos para desempenharem as funções definidas para cada cargo.

Alguns autores apresentam cada nível de maturidade através da visão de cada um dos seus atores ou participantes, neste nível temos as seguintes visões:

- Visão dos gerentes;
- Visão dos desenvolvedores e testadores;
- Visão dos usuários ou clientes.

Esta abordagem será usada apenas como exemplo para este nível 2.

### **Visão dos gerentes**

Os seguintes compromissos devem ser assumidos ao nível gerencial:

- As gerências superiores devem fornecer todo o apoio, recursos e treinamento necessários para a implantação de uma política de planejamento de testes;  
Evidências possíveis: Armazenar as listas de presença dos treinamentos realizados e ou os certificados que comprovem a experiência necessária as funções exercidas.
- O gerente do projeto de testes deve ser o responsável pela negociação dos compromissos e responsabilidades junto a equipe de desenvolvimento para a elaboração do Plano de Testes e de todos os artefatos criados pelo projeto de testes;  
Evidências possíveis: As reuniões para as definições com os grupos envolvidos devem gerar atas e estas devem ser armazenadas assim com todos os artefatos gerados.

- O gerente do projeto de testes deve ter a garantia de que os usuários aceitam e apóiam o Plano de Testes, incluindo a forma de encaminhamento dos requisitos.  
Evidências possíveis: A aprovação do documento Plano de Teste pelo grupo envolvido deve ser armazenada em local de fácil acesso para as auditorias.

### **Visão dos desenvolvedores e testadores**

Os seguintes compromissos devem ser assumidos pelas equipes de teste e de desenvolvimento:

- Fornecer subsídios para elaboração do Plano de Testes, ajudando no estabelecimento de metas de teste, métodos, procedimentos, ferramentas e cronogramas;  
Evidências possíveis: As reuniões para as definições com os grupos envolvidos devem gerar atas e estas devem ser armazenadas assim com todos os artefatos gerados.
- Desenvolver especificações de teste, casos de teste, critérios para identificação ou não de defeitos para cada um dos níveis do Plano de Testes;  
Evidências possíveis: Todos os artefatos listados no Plano de Teste devem ser elaborados e armazenados com um controle de versão para as suas possíveis alterações. Para cada artefato elaborado deve ser preparado um template e este deve conter as referidas instruções de preenchimento.
- Apoiar na definição dos equipamentos, softwares e outros recursos necessários para a execução dos testes.  
Evidências possíveis: Os recursos a serem utilizados no processo de teste devem ser devidamente acordados entre a equipe de teste no início de cada projeto e com a concordância dos envolvidos de forma a garantir a melhor utilização dos recursos computacionais.

### **Visão dos usuários ou clientes**

Os seguintes compromissos devem ser assumidos pelos usuários ou clientes:

- Fornecer apoio para a elaboração e aceitação do Plano de Testes;  
Evidências possíveis: As reuniões para as definições e posteriores aprovações, com os grupos envolvidos devem gerar atas e estas devem ser armazenadas assim com todos os artefatos gerados.
- Apoiar na descrição dos requisitos funcionais, de performance e outros atributos de qualidade tais como, segurança, portabilidade, usabilidade e interoperabilidade.

Evidências possíveis: As reuniões para as definições com os grupos envolvidos devem gerar atas e estas devem ser armazenadas assim com todos os artefatos gerados.

### **Nível 3 - Integração**

O objetivo a ser alcançado neste nível é um nível de integração entre o teste e o desenvolvimento do software, através dos ciclos de vida de ambos projetos. Neste nível deve estar constituído formalmente uma organização para a execução dos testes.

Para que a área de testes consiga cumprir as metas para atingir o nível 3 precisa implantar os seguintes objetivos de maturidade:

- Estabelecer uma organização de teste de software;
- Estabelecer um programa de treinamento técnico;
- Integrar o teste no ciclo de vida do software;
- Controlar e monitorar o processo de teste.

#### **Estabelecer uma organização de teste de software**

No nível 3 deve haver um grupo dentro da organização formalmente encarregado da execução dos testes. No nível anterior a equipe poderia ser temporária e alocada eventualmente, mas não estava integralmente voltada para esta atividade.

Os sub-objetivos de maturidade são os seguintes:

- Deve ser criada formalmente uma organização de teste de software dentro da estrutura organizacional da empresa.  
Evidências possíveis: A empresa deve possuir uma equipe fixa encarregada da execução dos testes dentro do seu organograma e separada da equipe de desenvolvimento e subordinados a uma gerência específica.
- Regras e responsabilidades devem ser definidas para o grupo de teste;  
Evidências possíveis: A área de teste deve possuir uma equipe apartada das demais equipes com as respectivas definições de cargos e funções para os seus membros devidamente descritas. A evidência a ser buscada deve ser um documento que comprove essa estrutura organizacional.
- Um grupo de técnicos treinados e motivados deverá compor o grupo de teste.  
Evidências possíveis: A organização ou empresa deve manter uma equipe especializada em teste, com perfil definido para as atividades a serem desempenhadas. Essa equipe deve comprovar experiência e/ou treinamentos para desempenharem as funções definidas para cada cargo. Regularmente (periodicidade estabelecida pela equipe de

qualidade) a equipe de qualidade deve analisar o índice de satisfação motivacional da equipe. Na busca dessa evidência devem ser avaliados também listas de presença em treinamentos e questionários de avaliação de satisfação.

- Deve haver uma ligação formal entre as atividades de teste e os usuários de forma a facilitar o trabalho do grupo de teste principalmente no que se refere ao entendimento dos requisitos.

Evidências possíveis: Os requisitos gerados devem ser analisados e aprovados pela equipe de teste de forma a criar um comprometimento entre as equipes (desenvolvimento, teste e usuários). A ata da reunião de kick off do projeto pode ser um desses registros de que os usuários têm pleno conhecimento da atividade de teste desde que conste claramente esse objetivo.

### **Estabelecer um programa de treinamento técnico**

Os sub-objetivos de maturidade são os seguintes:

- A gerência deve estabelecer um programa de treinamento com recursos financeiros e apoio administrativo.

Evidências possíveis: Programa de treinamento da empresa contemplando teste de software.

- Um comitê deverá desenvolver e divulgar uma política de treinamento;

Evidências possíveis: Algum registro que mostre a divulgação da política de treinamento da empresa e que contemple teste de software.

- Os planos e objetivos de treinamento devem ser elaborados a partir de informações coletadas junto dos gerentes.

Evidências possíveis: Documento que mostre que a equipe de teste esteve envolvida no processo de elaboração do plano de treinamento. Ex.: E-mail enviando as necessidades de treinamento.

- Um grupo de treinamento na empresa deve ser estabelecido com ferramentas, equipamentos e todo apoio necessário.

Evidências possíveis: A empresa deve ter uma equipe ou área de treinamento.

### **Integrar o teste no ciclo de vida do software**

Este objetivo visa a garantir que o processo de teste seguirá um ciclo de vida claramente definido e que este será totalmente integrado ao ciclo de desenvolvimento do software.

Os sub-objetivos de maturidade são os seguintes:

- O processo de teste deve ter um ciclo de vida separado em fases e sub-fases de tal forma que possa se integrar com o ciclo de vida do processo

de desenvolvimento, baseado em políticas organizacionais freqüentemente revistas junto aos gerentes.

Evidências possíveis: Procedimentos de teste, com ciclo de vida aderente ao processo de desenvolvimento, com registros de atualização. Deve haver uma política organizacional que sirva de base para escrever esses procedimentos. O registro dos documentos de teste sob uma gerência de configuração muitas vezes é suficiente para provar que existem diferentes versões.

- Um modelo de teste (por exemplo um modelo V) deve ser formalmente definido com fases e sub-fases claramente definidas.

Evidências possíveis: Procedimentos de teste distribuídos nas fases do ciclo de vida do projeto.

- Devem existir recursos que permitam a integração entre as atividades de teste e aquelas correspondentes no ciclo de vida de desenvolvimento do software;

Evidências possíveis: Plano de teste definindo claramente a participação da equipe de teste no projeto.

- Definições padronizadas devem ser desenvolvidas para os produtos de teste e a sua qualidade deve ser medida.

Evidências possíveis: Devem haver métricas que permitam medir a qualidade dos produtos de teste. Os produtos de teste devem estar claramente definidos nos procedimentos de teste.

- Deve haver procedimentos formais que permitam aos testadores trabalhar junto aos desenvolvedores com facilidade.

Evidências possíveis: Procedimentos de teste devidamente definidos e institucionalizados.

Observação geral: Empresas que possuem um processo de teste claramente definido normalmente conseguem atender a todos os sub-objetivos definidos.

## **Controlar e monitorar o processo de teste**

Os sub-objetivos de maturidade são os seguintes:

- A organização deve criar mecanismos e políticas para controlar e monitorar o processo de teste.

Evidências possíveis: Métricas de controle do processo de teste envolvendo a sua definição e o seu uso.

- Um grupo de medidas do processo de teste deve ser definida e coletada regularmente e os seus resultados distribuídos.

Evidências possíveis: Métricas de controle do processo de teste e evidências da sua distribuição pelas partes envolvidas no teste.

- Medidas corretivas devem ser tomadas sempre que o processo de teste se desviar significativamente do que estava planejado.

Evidências possíveis: Reuniões de acompanhamento do projeto de teste mostrando o progresso do projeto. Documento provando que os desvios foram registrados e que estão sendo acompanhados.

#### **Nível 4 - Gestão e medições**

Para que a área de testes consiga cumprir as metas para atingir o nível 4 precisa implantar os seguintes objetivos de maturidade:

- Estabelecer um programa de revisões;
- Estabelecer um programa de medições;
- Avaliação da qualidade do software.

Neste nível o foco é ampliar a atividade de teste e melhorar as suas medições. Devem ser incluídas atividades de revisões em todos os documentos criados pelo processo de teste. Desta forma as áreas de processo de validação e verificação devem estar cobertas neste nível por esta expansão da atividade de teste.

#### **Estabelecer um programa de revisões**

Os sub-objetivos de maturidade são os seguintes:

- Os diversos níveis de gerência devem instituir um programa formal de revisão e garantir a sua integração dentro da cultura da organização.  
Evidências possíveis: Documento definindo o funcionamento do programa de revisão. Exemplo: Plano de Revisão.
- O grupo de teste deve definir formalmente objetivos de revisão que deverão ser aplicados durante todo o ciclo de vida de teste de software e do ciclo de vida do desenvolvimento do software.  
Evidências possíveis: Plano de revisão dos documentos de teste definido e divulgado.
- Os itens de revisão devem estar claramente definidos por projeto;  
Evidências possíveis: Plano de revisão dos documentos de teste.
- As equipes devem ser treinadas para poderem cumprir essas práticas de revisão de artefatos  
Evidências possíveis: Registro de treinamentos em revisão.

#### **Estabelecer um programa de medições**

Os sub-objetivos de maturidade são os seguintes:

- Deve ser definida uma política de medição cujos objetivos contemplem o processo de teste de software.  
Evidências possíveis: Documento descrevendo a política de medição da empresa onde devem estar listadas as métricas de teste.

- Um plano de medições deve ser desenvolvido com mecanismos para coleta, análise e compilação de dados.  
Evidências possíveis: Plano de medições para o projeto de teste onde deve estar claramente definido quem irá coletar, analisar e compilar os dados. Isso significa buscar os dados, fazer uma revisão e enviar para os tomadores de decisão.
- Deve haver plano de ação cujo objetivo seja usar os resultados das medições para melhorar o processo de teste.  
Evidências possíveis: Plano de medição definindo as ações a serem tomadas a partir dos resultados coletados. Por exemplo, ações devem ser tomadas caso os valores atinjam determinado limite. Este plano deve estar relacionado com a lista de risco do projeto de teste.

Observação geral: Evidentemente que caso a empresa tenha um processo de medição instalado e funcionando corretamente, o que ocorre nas empresas com nível de maturidade CMMI 2 ou mps.BR F essas evidências serão encontradas com facilidade.

## **Avaliação da qualidade do software**

Os sub-objetivos de maturidade são os seguintes:

- Os grupos de garantia de qualidade devem definir políticas, objetivos de qualidade e atributos de qualidade para cada produto de software.  
Evidências possíveis: Processo de garantia de qualidade institucionalizado com os seus responsáveis claramente definidos.
- A organização deverá ter procedimentos formais para avaliar os resultados do processo de avaliação.  
Evidências possíveis: Procedimentos a serem tomados com os resultados das medições onde os responsáveis pela ação devem ser claramente definidos.
- O processo de teste deve ser estruturado, medido e avaliado para garantir que os objetivos de qualidade sejam atingidos.  
Evidências possíveis: Processo de teste contemplando os objetivos definidos para qualidade. Exemplo: Medições efetuadas em um projeto e resultados analisados.

Neste nível espera-se que algumas características de qualidade do software possam ser atingidas e avaliadas (ver ISO-9126).

## **Nível 5 - Otimização, prevenção de defeitos e controle de qualidade**

Para que a área de testes consiga cumprir as metas para atingir o nível 5 precisa implantar os seguintes objetivos de maturidade:

- Aplicação de processo para prevenção de defeitos;

- Controle de qualidade;
- Otimização do processo de teste

### **Aplicação de processo para prevenção de defeitos**

Os sub-objetivos de maturidade são os seguintes:

- A organização deve desenvolver e manter uma política que permita a prevenção de defeitos.  
Evidências possíveis: Política de prevenção definida e institucionalizada.
- A prevenção de defeitos deve ser uma política suportada pelos diversos níveis gerenciais.  
Evidências possíveis: Política de prevenção institucionalizada.
- Os defeitos encontrados e removidos devem ser identificados e registrados durante cada fase do ciclo de vida do software.  
Evidências possíveis: Os defeitos encontrados devem ser rastreados até a sua conclusão. Deve haver uma base de registros de defeitos.
- Um mecanismo deve ser definido para buscar a causa dos defeitos.  
Evidências possíveis: Os defeitos encontrados devem ser analisados e as suas causas identificadas e registradas. Deve haver um registro da resolução dos defeitos.
- Deve haver um plano integrado envolvendo testadores e desenvolvedores visando à prevenção de defeitos e este plano deve ser rastreado e monitorado.  
Evidências possíveis: Documento que comprove a integração entre as equipes na busca da melhoria contínua. O plano de prevenção deve estar definido e precisa ser monitorado.

### **Controle de qualidade**

Os sub-objetivos de maturidade são os seguintes:

- A organização deve desenvolver e manter procedimentos de controle de qualidade.  
Evidências possíveis: A empresa ou organização deve demonstrar que realiza auditoria nos processos e procedimentos visando buscar a melhoria contínua.
- Os grupos de teste e de controle de qualidade devem estabelecer metas claras de qualidade visando a redução de defeitos nos produtos de softwares.  
Evidências possíveis: Possuir objetivos e metas definidas no plano de medição da empresa. Não pode haver nenhuma medição que não se saiba o que fazer com os seus resultados.
- Os gerentes de teste devem incluir as metas de qualidade nos seus planos.

Evidências possíveis: Possuir objetivos e metas de qualidade definidas no plano de testes.

- O grupo de teste deve ser treinado em métodos estatísticos;  
Evidências possíveis: Lista de presença dos treinamentos realizados.
- Informações dos usuários devem ser coletadas para serem integradas ao modelo de qualidade.  
Evidências possíveis: Relatório de satisfação do cliente

### **Otimização do processo de teste**

Os sub-objetivos de maturidade são os seguintes:

- A organização deve desenvolver e manter procedimentos de melhoria do processo de teste.  
Evidências possíveis: Processo para viabilizar as buscas de melhorias. Exemplo: Processo de Melhoria definido e institucionalizado, o que normalmente ocorre com empresas CMMI nível 5 ou mps.Br nível A.
- O grupo de melhoria de processo de teste deve ser estabelecido para monitorar o processo de teste e identificar possibilidades de melhoria.  
Evidências possíveis: Definição do grupo de melhoria de teste com os seus responsáveis.
- Um mecanismo deve ser criado para avaliar novas ferramentas e tecnologias que possam melhorar a maturidade e capacidade do processo de teste.  
Evidências possíveis: Procedimentos para solução técnica com o estudo de alternativas. Demonstrar que existem estudos e que decisões foram tomadas.
- A efetividade do processo de teste deve ser constantemente avaliada e as decisões de quando interromper o processo de teste devem estar relacionadas aos objetivos de qualidade de uma forma mensurável.  
Evidências possíveis: As medições devem mostrar que um determinado nível de qualidade claramente definido já foi alcançado que o processo pode ser interrompido. Medições precisam ser feitas para avaliar o atendimento da meta.

### **Evidências diretas para todos os sub-objetivos**

Os avaliadores de maturidade de empresas nos modelos CMMI e mps.BR costumam comprovar as evidências diretas através de documentos de projetos mostrando que as práticas foram implementadas e que estão sendo usadas. No caso do TMM a idéia deve ser a mesma. O documento organizacional só serve para mostrar que o objetivo ou sub-objetivo está claramente definido, mas caso seja necessário demonstrar o seu uso o avaliador deverá buscar as tais evidências diretas. Por exemplo, o projeto de teste de um projeto é uma evidência direta de que o projeto está sendo planejado.

## Conclusão

Embora muitas melhorias tenham ocorrido na área de teste de software, a maioria das empresas ainda não trata a atividade de testar como um processo prioritário dentro da organização. O mais comum ainda é administrar os defeitos e tentar corrigi-los quando o sistema já está em produção, quando o custo de correção chega a ser mil vezes mais caro. Se não há equipes de teste formalmente constituídas, certamente não há o que melhorar o que não existe. Muitas empresas, caminham para o nível 2 do CMMI sem tratar o processo de teste com seriedade e vão continuar a gerar produtos sem qualidade.

Atualmente já existe um esforço grande no sentido de integrar os modelos de maturidade de teste, como o TMM que mostramos neste artigo, com os modelos de maturidade da capacitação para software como o CMMI e o mps.BR. Enquanto isso não ocorre precisamos tratar o teste como um processo separado e com características próprias, o que justifica a existência de modelos como o TMM.

## Bibliografia

Test Process Improvement – A practical step-by-step guide to structured testing” – Tim Koomen e Martin Pol – ACM Press Books – 1999.

Manging The Testing Process – Rex Black – Microsoft Press – 1999.

Testing Computer Software – Cem Kaner, Jack Falk e Hung Quoc Nguyen – John Wiley and Sons Inc. – 1999

Teste de Software – Emerson Rios, Trayahu Moreira Filho, Editora AltaBooks, 2002.

Engenharia de Software com CMM – Soeli T. Fiorini e outros, Editora Brasport, 1998.

CMMI – Guidelines for Process Integration and Product Improvement – Mary Beth Chrissis e outros – Ed. Adisson Wesley – 2004.

[www.stickyminds.com](http://www.stickyminds.com) - Site com vários artigos sobre teste de software

[www.cs.umn.edu](http://www.cs.umn.edu) - Site do Computer Science Insitute da Universidade de Minesota.

[www.iit.edu](http://www.iit.edu) - Site do Illinois Institute of Technology onde foi desenvolvido o TMM.