



Exacta

ISSN: 1678-5428

exacta@uninove.br

Universidade Nove de Julho

Brasil

Lara Baptista, Gabriel; Vanalle, Rosângela Maria; Arantes Salles, José Antônio  
Utilização de sistemas de medição de desempenho em projetos de desenvolvimento de software  
Exacta, vol. 10, núm. 2, 2012, pp. 181-191  
Universidade Nove de Julho  
São Paulo, Brasil

Disponível em: <http://www.redalyc.org/articulo.oa?id=81024617003>

- ▶ [Como citar este artigo](#)
- ▶ [Número completo](#)
- ▶ [Mais artigos](#)
- ▶ [Home da revista no Redalyc](#)

redalyc.org

Sistema de Informação Científica

Rede de Revistas Científicas da América Latina, Caribe, Espanha e Portugal  
Projeto acadêmico sem fins lucrativos desenvolvido no âmbito da iniciativa Acesso Aberto

# Utilização de sistemas de medição de desempenho em projetos de desenvolvimento de software

*The use of performance-measuring systems in software development projects*

Gabriel Lara Baptista

Mestrando em Engenharia de Produção, Programa de Mestrado de Engenharia em Produção da Universidade Nove de Julho – UNINOVE.  
São Paulo, SP – Brasil.  
gabriel.baptista@uninove.br

Rosângela Maria Vanalle

Pós-Doutorado pela Universidade Complutense de Madrid – UCM. Programa de Mestrado de Engenharia em Produção da Universidade Nove de Julho – UNINOVE.  
São Paulo, SP – Brasil.  
rwanalle@uninove.br

José Antônio Arantes Salles

Pós-Doutorado pela Universidade Complutense de Madrid – UCM. Programa de Mestrado de Engenharia em Produção da Universidade Nove de Julho – UNINOVE.  
São Paulo, SP – Brasil.  
salles@uninove.br

## Resumo

O uso de sistemas de medição de desempenho em projetos de *software* é ainda pouco explorado, apesar de ser considerado importante para redução de falhas nesses projetos. Este artigo apresenta um estudo sobre a percepção e conhecimento de um grupo de profissionais envolvidos com projetos de *software*, no que diz respeito ao uso de normas e modelos que suportem a medição do desempenho. O resultado da pesquisa sugere que existe limitação de conhecimento das normas e dos modelos apresentados, o que enfatiza a necessidade de novas abordagens para disseminação da medição de desempenho no processo de desenvolvimento de produtos de *software*. São apresentadas, no fim do trabalho, as diretrizes básicas para a construção de um modelo que atenda as premissas dessa nova abordagem.

**Palavras-chave:** CMMI-DEV. Sistemas de medição de desempenho. Métricas de software. MPS.BR.

## Abstract

The use of performance-measuring systems in software projects is still uncommon in spite of their being considered important to minimize errors in that class of projects. This article presents a study on the perceptions and knowledge of a group of professionals involved in software projects, in relation to the use of standards and models that support the measurement of performance. The results suggest there is limited knowledge about the standards and models presented, which emphasizes the need for new approaches for making more widespread the use of performance measuring in the software development process. The paper concludes by describing basic guidelines for building a model that is in line with the assumptions of this new approach.

**Key words:** CMMI-DEV. MPS.BR. Performance-measuring systems. Software metrics.

## 1 Introdução

A construção de *software* vem sendo estudada nas últimas quatro décadas e sabe-se que tal tarefa é crítica. Isso porque a engenharia de *software*, disciplina que se preocupa com todos os aspectos de sua produção, tem sofrido, ao longo dos anos, dificuldades para cumprir prazos, custos e funcionalidades esperados, o que torna tais fatores desafiadores para a área (DIJKSTRA, 1972; HUMPHREY, 1995; SOMMERVILLE, 2011). Um exemplo disso, pode ser visto nos resultados apresentados pelo *Chaos Report*, que apontam sucesso em apenas 32% dos projetos de *software* (STANDISH GROUP INTERNATIONAL, 2009).

A gestão dos projetos de *software* vem recebendo atenção tanto do meio acadêmico quanto do empresarial (NA, et al., 2007). Sommerville (2011) sugere que a diferença principal entre engenheiros de *software* e programadores é que a engenharia de *software* é um processo gerenciado. DeMarco (1986) afirma que não se pode controlar o que não se pode medir. Por tal razão, a medição de desempenho em projetos de *software* pode ser considerada uma ferramenta útil para gestores de projetos dessa natureza (ERALP, 2004; GANSSLE, 2009; KOSCIANSKI; SOARES, 2007; SOFTWARE ENGINEERING INSTITUTE, 2010; SOFTEX, 2011), já que um dos principais propósitos de um sistema de medição de desempenho é entregar um conjunto de informações confiáveis que auxilie na tomada de decisão (UKKO; TENHUNEN; RANTANEN, 2007).

Os sistemas de medição de desempenho vêm sendo estudados e aplicados ao longo do tempo por diversos autores (DÍAZ; GIL; MACHUCA, 2005; EL-BAZ, 2011; GAIARDELLI; SACCANI; SONGINI, 2007; UKKO; TENHUNEN; RANTANEN,

2007). Modelos como o *Balanced Scorecard* (KAPLAN; NORTON, 1997) e o *Prisma* (NEELY; GREGORY; PLATTS, 1995) incentivaram uma visão mais abrangente de desempenho, com o intuito de criar um sistema de medição balanceado e não apenas focado na perspectiva financeira, sempre se preocupando em alinhar a estratégia corporativa com o sistema de medição.

No que se trata da produção de *software* alinhada a sistemas de medição de desempenho, Pressman (2011) afirma que medir é fundamental em qualquer engenharia e que a Engenharia de *Software* não é exceção. Jones (2008) afirma que um dos pontos-chave para que as empresas de *software* alcancem o controle do desenvolvimento de *software* tem relação com a medição da produtividade e qualidade em seu desenvolvimento de forma acurada. Por outro lado, ele ressalta que, infelizmente, os desenvolvedores de *software* estão muito distantes de outros profissionais quanto ao estabelecimento de padrões de medição e objetivos relevantes.

Com base nas questões acima, neste artigo, apresenta-se um estudo sobre a percepção e conhecimento de um grupo de profissionais envolvidos com projetos para desenvolvimento de *software*, no que diz respeito ao uso de padrões, modelos e técnicas que suportem a medição do desempenho em seus projetos. Para tanto, um questionário foi aplicado a esse grupo e os resultados serão apresentados nas seções seguintes. Na segunda seção, apresentam-se os conceitos relativos aos sistemas de medição de desempenho de forma geral, o modo como a Engenharia de *Software* trata a questão de medição de desempenho e normas e modelos que apoiam o uso de métricas de *software*. Na terceira seção, é apresentada a pesquisa de campo, sendo mostrados os resultados obtidos. Já na quarta, são apresentadas as conclusões.

## 2 Embasamento teórico

## 2.1 Sistemas de medição de desempenho

A medição de desempenho é um processo de quantificar a ação, no qual a medição relaciona-se com o processo de quantificação, e o desempenho é presumido como derivado de ações tomadas ao longo da produção (SLACK; CHAMBERS; JOHNSTON, 2009).

Quanto à construção de um Sistema de Medição de Desempenho (SMD), El-Baz (2011) apresenta a adaptação dos nove passos típicos e necessários (NEELY; GREGORY; PLATTS, 1995), relacionados no Quadro 1.

Passo	Ação
1	Definir a missão da empresa.
2	Definir os objetivos estratégicos da empresa, utilizando-se da missão como um guia.
3	Desenvolver um entendimento de cada uma das áreas funcionais em relação aos objetivos estratégicos.
4	Para cada uma das áreas, definir medições de desempenho globais.
5	Comunicar os objetivos estratégicos para cada nível da empresa.
6	Garantir consistência entre os objetivos estratégicos e o critério de desempenho de cada nível.
7	Garantir a compatibilidade das medidas de desempenho de cada área.
8	Utilizar o SMD.
9	Periodicamente reavaliar o SMD em função das necessidades da empresa e do ambiente competitivo atual.

### Quadro 1: Nove passos para desenvolver um SMD

Fonte: El-Baz (2011)

Entretanto, mesmo com passos pré-definidos para construção de um SMD, o alinhamento entre a estratégia e as operações da empresa pode ser considerado um dos fatores críticos para a

sua aplicação (ATKINSON, 2006; BHAGWAT; SHARMA, 2007; MACEDO; BARBOSA; CAVALCANTE, 2009).

Neely, Gregory e Platts (1995) afirmam que se deve também desenvolver um entendimento de cada uma das áreas funcionais em relação aos objetivos estratégicos, determinando medições de desempenho globais para cada uma das áreas. A comunicação dos objetivos estratégicos deve ser executada para cada nível da empresa, e a garantia de consistência desses objetivos com o critério de desempenho de cada nível deve ser realizada.

Por fim, deve haver a garantia de compatibilidade das medidas de desempenho de cada área; e o SMD deve ser utilizado, sempre sendo reavaliado, em função das necessidades da empresa e do ambiente competitivo atual.

## 2.2 Medição de desempenho para a Engenharia de *Software*

No que se diz respeito à produção de *software*, Humphrey (1989) afirma que são quatro os principais papéis da medição, conforme apresentado no Quadro 2. Sabe-se, portanto, que um sistema de medição de desempenho adequado trará benefícios à corporação que, se alinhados com sua estratégia, auxiliarão no alcance das metas estratégicas (KAPLAN; NORTON, 1997).

Muitos estudos relacionados às métricas de *software* também se relacionam com a previsão. Estas métricas são utilizadas como estimativas-base para monitoramento e controle de projetos futuros, com características semelhantes a projetos já executados (SOFTWARE ENGINEERING INSTITUTE, 2010). Questões referentes ao esforço, tempo e custo para completar uma atividade são fatores amplamente estudados e ainda demandam novas pesquisas (SOMMERVILLE, 2011).

Quando se fala de controle de *software* baseado em métricas, a utilização de uma base histórica se faz necessária. Koscienski e Soares (2007)

Papel	Descrição
Controlar	Com métricas para controle, torna-se possível identificar processos, produtos e/ou serviços que precisam de interferência da gestão.
Avaliar	As métricas de avaliação permitem o entendimento se os processos, produtos e/ou serviços estão de acordo com os padrões, objetivos e critérios de aceitação propostos pela corporação.
Entender	Métricas de entendimento auxiliaram na pesquisa e entendimento sobre os processos, produtos e/ou serviços da empresa.
Prever	As métricas de previsão auxiliaram a empresa a estimar valores e prever situações que possam vir a ocorrer.

**Quadro 2: Papéis da medição**

Fonte: Humphrey (1989).

afirmam que o gerenciamento de qualidade estará incompleto enquanto não houver um registro desses dados para todos os projetos executados. Humphrey (1995) também apresenta em seu livro dedicado ao desenvolvimento pessoal de *software* a necessidade de armazenamento de um histórico, o que sugere tal prática como característica dos sistemas de medição de desempenho para projetos de *software*.

Pressman (2011) enfatiza também a importância de utilização métricas de *software* para entendimento e gestão de processos e produtos de *software*. Ele afirma que a medição resulta em uma mudança cultural, em que coleta de dados, computação das métricas e avaliação dos dados são os três passos que devem ser aplicados para o início de um programa de métricas.

De forma contraditória ao apresentado até agora pela bibliografia selecionada – que destaca o uso das métricas para gerenciamento de projetos de *software* –, em uma pesquisa levantada pelo Ministério da Ciência e Tecnologia, afirma-se que somente 39,6% das empresas pesquisadas medem o desempenho do processo de *software* de for-

ma sistemática (MINISTÉRIO DA CIÊNCIA E TECNOLOGIA, 2010). O mesmo estudo enfatiza ainda que a avaliação do desempenho das pessoas e dos processos da organização também está abaixo do mínimo aceitável, o que sugere que essa seja ainda uma área para muitas pesquisas.

### 2.3 Normas e modelos de processos que apoiam a medição de desempenho

A Engenharia de *Software* possui um conjunto de normas e processo que apoiam a medição de desempenho. Dentre elas, pode-se destacar a norma ISO 12207 (ABNT, 2009), a norma ISO 15504 (ABNT, 2008), o modelo internacional CMMI-DEV – *Capability Maturity Model Integration for Development* (SOFTWARE ENGINEERING INSTITUTE, 2010) e o modelo nacional MPS. BR – Melhoria de Processo do *Software* Brasileiro (SOFTEX, 2011).

A 12207 (ABNT, 2009) é a norma que trata da questão de processos de ciclo de vida de *software*. Koscienski e Soares (2007) afirmam que essa norma oferece uma estrutura para que uma organização defina os seus processos de produção de *software*, cobrindo todo seu ciclo de vida, de requisitos até a manutenção e retirada do produto. A norma requer que a empresa, em suas atividades, se preocupe em medir os processos de desenvolvimento de *software*, armazenando dados históricos dos projetos, de modo a entender os processos que está executando.

A norma ISO 15504 (ABNT, 2008) trata da questão de avaliação do processo de desenvolvimento de *software*. Atualmente, a norma possui sete partes, já traduzidas para o português. Salviano (2009) afirma que as partes oito e nove estão em desenvolvimento, e que existe um projeto atual para realizar a evolução da atual norma para uma série ISO/IEC 33000.

A norma ISO 15504 presta-se a realização de avaliações dos processos com dois objetivos, a saber: melhorar e determinar a capacidade dos processos (ANACLETO; VON WANGENHEIM; SALVIANO, 2005). Para tanto, ela define o conceito de nível de capacidade, que aponta o quanto capaz é a empresa em relação à execução do processo avaliado. A norma apresenta seis níveis de capacidade, que vão de incompleto a otimizado, e seu quarto nível refere-se diretamente a empresas que têm a capacidade de prever resultados a partir de medições detalhadas e analisadas. A norma ainda conta com uma área de processo dedicada à medição, considerada base para os modelos CMMI-DEV e MPS.BR (SOFTWARE ENGINEERING INSTITUTE, 2010; SOFTEX, 2011).

O CMMI-DEV é atualmente o *framework* para melhoria de processo de desenvolvimento de *software* mais conhecido e aceito mundialmente. Tal reconhecimento está ligado às suas origens, já que toda a sua história parte da necessidade do mercado norte-americano em relação a avaliar a maturidade das empresas fornecedoras de *software*, com a criação do CMM (HUMPHREY, 1995).

O modelo, atualmente, encontra-se em sua versão 1.3 e é composto por 22 áreas de processo, que estão posicionadas em cinco níveis de maturidade ou seis níveis de capacidade, dependendo do tipo de representação utilizado. Ele está embasado nas propostas das normas ISO 9000, ISO 12207, ISO 15504, entre outras (SOFTWARE ENGINEERING INSTITUTE, 2010).

Já se faz obrigatório, a partir do Nível 2 de maturidade do CMMI-DEV, a aplicação de medição e análise nos projetos de *software*. Segundo o seu documento guia, a escolha das métricas de um projeto deve estar alinhada com as necessidades da organização e as do projeto. O Quadro 3 apresenta os objetivos e práticas específicas dessa área de processo do modelo. O modo de coleta das medições também deve ser definido,

<b>Objetivos e práticas 1</b>	
SG1. Alinhar Atividades de Medição e Análise	
SP1.1. Estabelecer Objetivos de Medição	
SP1.2. Especificar Medida	
SP1.3. Especificar Procedimentos de Coleta e Armazenamento de Dados	
SP1.4. Especificar Procedimento de Análise	
<b>Objetivos e práticas 2</b>	
SG2. Fornecer Resultados de Medição	
SP2.1. Coletar Dados Resultantes de Medição	
SP2.2. Analisar Dados Resultantes de Medição	
SP2.3. Armazenar Dados e Resultados	
SP2.4. Comunicar Resultados	

### Quadro 3: Área de processo de medição e análise do CMMI-DEV

Fonte: Software Engineering Institute (2010).

bem como a maneira como as métricas serão armazenadas. Após a coleta e análise dos dados, as medidas devem ser comunicadas a todos os envolvidos relevantes do projeto.

O MPS.BR foi fruto de uma iniciativa brasileira ocorrida no fim do ano de 2003, coordenada pela Associação para Produção da Excelência do *Software* Brasileiro (SOFTEX) e apoiada pelo Ministério de Ciência e Tecnologia (MCT), além de outras entidades governamentais e privadas (SOFTEX, 2011).

A iniciativa tem como objetivo principal a melhoria de processo do *software* brasileiro, seguindo como referências as normas internacionais ISO 15504, ISO 12207 e o modelo CMMI-DEV. O grande fator que contribuiu para a criação do modelo foi compatibilizar os custos de implantação com o cenário brasileiro, ponto discutido em um estudo do MIT de 2003, indicando que poucas empresas nacionais buscavam o modelo CMMI-DEV (SOCIEDADE BRASILEIRA DA COMPUTAÇÃO, 2010). Conceitualmente, o modelo se assemelha muito aos já apresentados até aqui, com o mesmo conceito de nível de maturidade e áreas de processo.

Nesse modelo, a área de processo de medição tem como propósito a coleta, o armazenamento, a análise e o relato dos dados relativos a produtos e processos da organização, também apoiando os objetivos da empresa (SOFTEX, 2011). O modelo deixa claro ainda que o foco principal da medição seja o de apoiar a tomada de decisão em relação aos projetos e processos, sendo as métricas criadas a partir de necessidades estratégicas de informação da organização. Ele sugere ainda a utilização de um método de medição, como, por exemplo, o *Goal-Question-Metric* (GQM), idealizado por Solingen e Berghout (1999).

Apesar de todos os esforços apresentados até aqui, existe grande discussão se os modelos são realmente aplicados na prática (CORDEIRO; FREITAS, 2008). A questão é mais discutível ainda quando se observa o uso de tais metodologias em pequenos grupos de desenvolvimento de *software*, fato mundialmente comum (ANACLETO, 2005; PINO et. al., 2010; AL-TARAWNEH et al., 2011). Em um estudo divulgado pela Pesquisa de Qualidade no Setor de Software Brasileiro (MINISTÉRIO DA CIÊNCIA E TECNOLOGIA, 2010), existe a confirmação desse cenário, já que menos de 30% das empresas pesquisadas tinham processos certificados ISO 9001, CMMI-DEV e/ou MPS.BR

va, optou-se pela aplicação de conceitos oriundos da estatística descritiva.

Tal questionário foi formulado com questões fechadas, algumas delas tendo como respostas as informações em escala Likert de cinco pontos, e optou-se pela sua aplicação eletrônica. Os resultados apresentados são oriundos de 64 respondentes, sendo 30% deles atualmente gerentes, coordenadores ou líderes de equipe. Os 70% restantes são analistas, programadores e testadores de *software*.

Dentre os participantes havia pessoas que atuavam em empresas cujo desenvolvimento de *software* é fonte principal de capital (34%), fonte secundária de capital (17%), base para gerir os negócios (36%) e empresas que somente utilizam *software*, sem necessariamente desenvolvê-lo (13%). A maior parte dos respondentes trabalha em empresas localizadas no estado de São Paulo (87%). Nesta pesquisa, não houve preocupação em definir a localização exata dos demais respondentes, mas sabe-se que 25% deles também atuam como gerentes, coordenadores ou líderes de equipe no Brasil. Sabe-se também que alguns dos respondentes trabalham em empresas que possuem certificação nos modelos CMMI-DEV e/ou MPS.BR discutidos anteriormente, cujo foco principal é a melhoria do processo de desenvolvimento de *software*.

Com base no objetivo do artigo de analisar a percepção e conhecimento de um grupo de profissionais envolvidos com projetos para desenvolvimento de *software* em relação à medição do desempenho, cinco fatores foram avaliados, quais sejam: o nível de conhecimento em relação às normas e aos modelos discutidos no artigo; o conhecimento dos respondentes em relação às métricas de *software*; o modo que as empresas dos voluntários avaliam o desempenho do processo de desenvolvimento de *software*; a utilização da avaliação do desempenho dos projetos de *software* para reali-

### 3 A pesquisa de campo

Com o objetivo de observar a perspectiva de profissionais em relação à utilização de sistemas de medição de desempenho em projetos de *software*, uma pesquisa de campo qualitativa foi conduzida. Como técnica para condução de tal estudo, optou-se pela aplicação de um questionário, de forma a descobrir a opinião dos envolvidos sobre o assunto estudado (SEVERINO, 2007). Para análise dos dados, por se tratar de uma abordagem qualitati-

zação de ações preventivas; e a utilização do desempenho dos projetos de *software* para avaliar os resultados financeiros das empresas.

### 3.1 Resultados obtidos

A Figura 1 apresenta os resultados obtidos no que diz respeito ao conhecimento dos entrevistados em relação às normas e modelos de processo que apoiam os sistemas de medição de desempenho para projetos de *software*. O resultado mostrou um baixo conhecimento para as duas normas ISO perguntadas (ISO 12207 e ISO 15504), com valores superiores a 70% dos respondentes. A falta de conhecimento dos modelos foi mais moderada, mas também contou com mais de 50% afirmando não conhecerem ou terem um conhecimento limitado a respeito dos modelos.

O número é semelhante quando se observa somente os coordenadores respondentes da pesquisa, como mostra a Figura 2. Esse resultado é negativo, pois demonstra que não existe grande

conhecimento dessas normas por parte das pessoas que atualmente gerem projetos de *software* nos modelos estudados. Tais resultados vão ao encontro dos obtidos na pesquisa realizada pelo MCT (MINISTÉRIO DA CIÊNCIA E TECNOLOGIA, 2010), que mostra que a avaliação do desempenho, tanto do processo como do produto, estão aquém das expectativas. Isso pode ter relação com o baixo conhecimento dos envolvidos diretos no processo de medição, quando se trata das normas e modelos que apoiam essa prática.

Um ponto de discussão que pode ser visto nos resultados apresentados acima é o maior conhecimento dos profissionais em relação ao modelo CMMI-DEV, em comparação com as normas ISO e o modelo MPS.BR. Persse (2007) sugere que o fato do conceito empregado no modelo já possuir mais de 20 anos e já ter sido provado o seu valor faz com que ele seja realmente um destaque tanto na academia quanto na indústria.

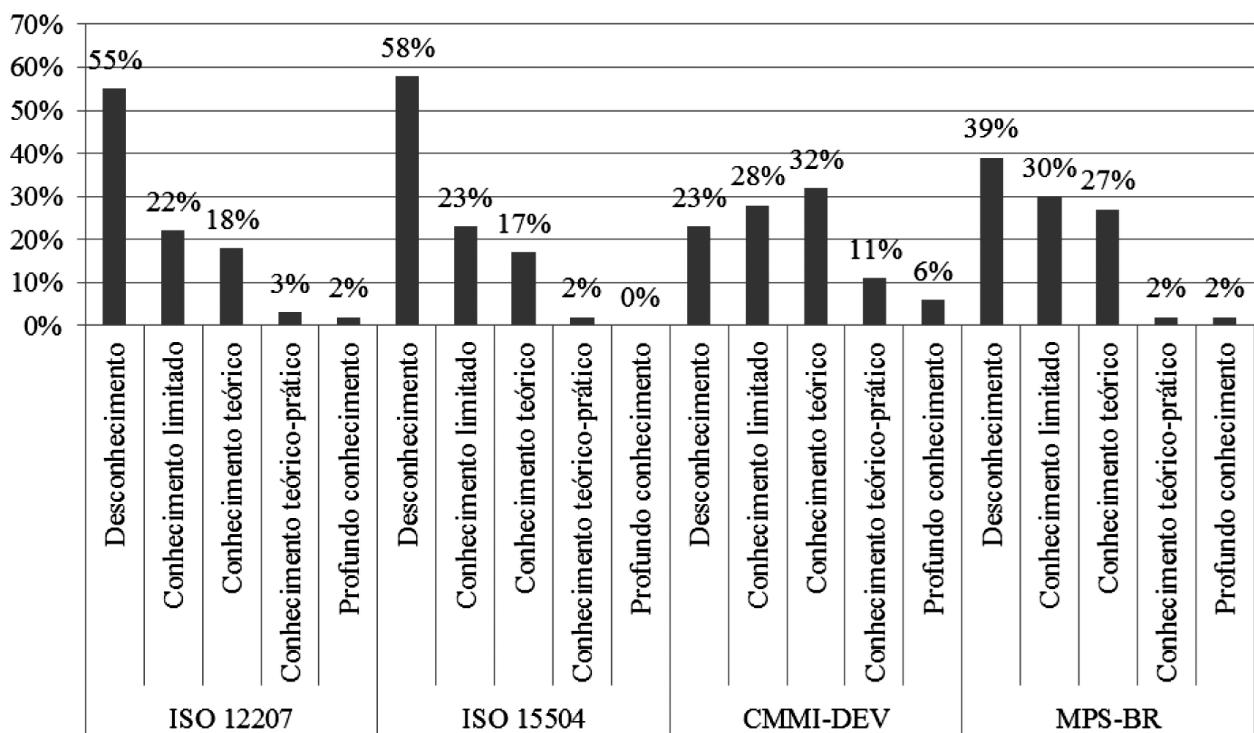
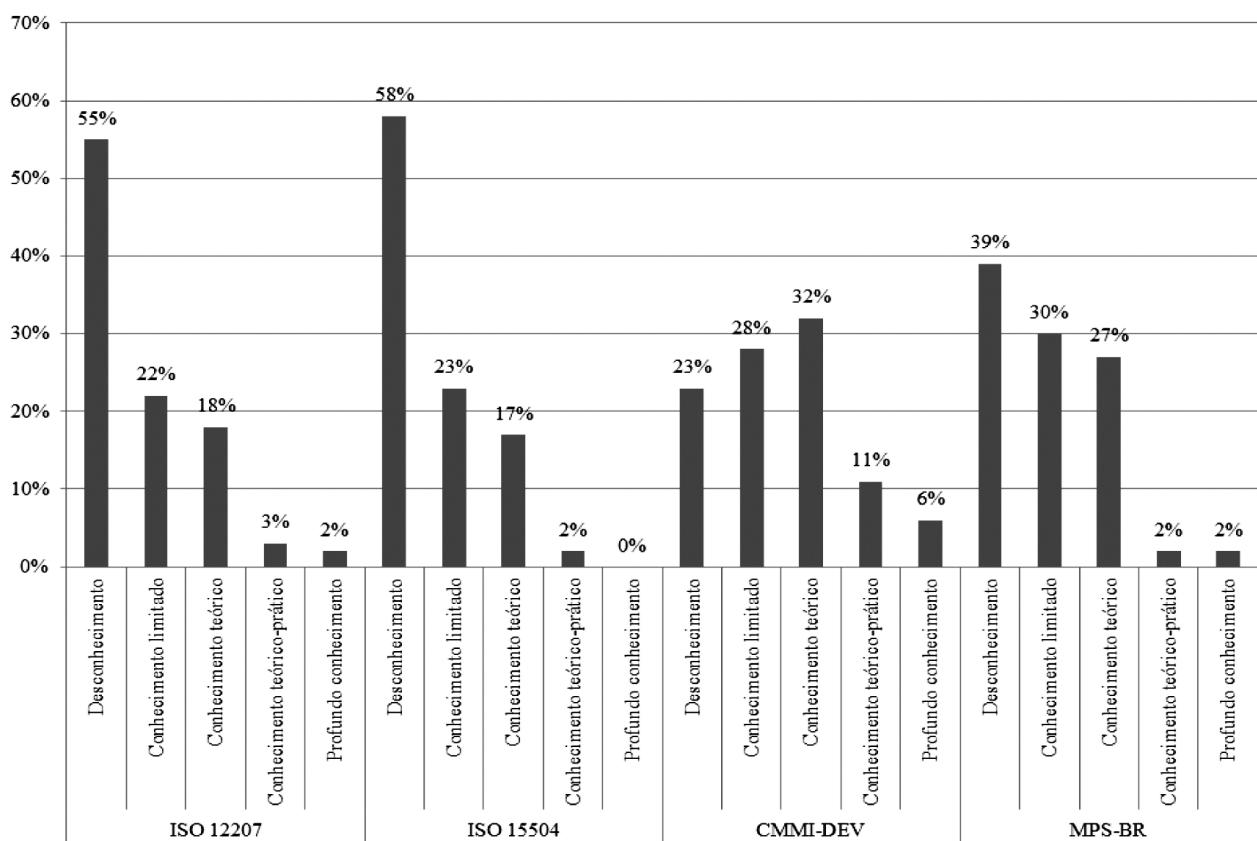


Figura 1: Conhecimento dos respondentes em relação às normas e modelos de processo de *software*

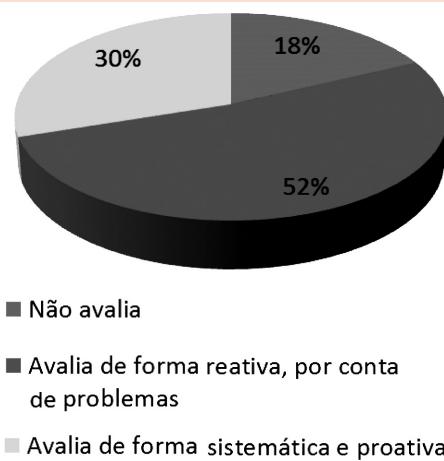


**Figura 2: Conhecimento dos coordenadores em relação às normas e modelos de processo de software**

Quanto à questão de conhecimento das métricas, apenas 9% dos entrevistados afirmaram não ter conhecimento sobre métricas de *software*. Entretanto, o nível de utilização formal das métricas também é baixo (19%), se comparado com o percentual de não utilização, apesar do conhecimento teórico (45%) ser ponto de destaque. Isso reforça a afirmação de Jones (2008) quando ele diz que profissionais da área de *software* estão distantes de outros profissionais quanto ao uso prático de medição de desempenho.

Outro resultado que confirma a pesquisa mais ampla realizada pelo MCT está relacionado ao modo como as empresas dos respondentes avaliam o desempenho do processo de desenvolvimento de *software*. A Figura 3 apresenta os resultados que mostram que apenas 30% das empresas avaliam de forma sistemática o desempenho do processo, número muito próximo ao obtido pelo MCT (2010).

Esse cenário é negativo e sugere uma grande distância entre a teoria e a prática. Fica a questão do motivo da não utilização de sistemas de medições com o objetivo de prevenção de problemas.



**Figura 3: Modo de avaliação do desempenho do processo de desenvolvimento de *software* por parte das empresas dos respondentes**

Apesar desse cenário negativo, 67% dos respondentes afirmam que a avaliação de desempenho permite que sejam realizadas ações preventivas para solução de problemas. Isso sugere que as pessoas entrevistadas têm ciência da importância das métricas. Somado a isso, 78% delas referem que o conhecimento sobre métricas de *software* varia de médio a muito importante para seu aprimoramento pessoal.

Por fim, a pesquisa mostrou que ainda existe uma grande dificuldade de alinhamento entre a estratégia da empresa e a avaliação de desempenho dos projetos de *software*, já que 55% dos respondentes mencionam que tal avaliação não faz parte da avaliação dos resultados financeiros de suas empresas. Isso confirma levantamentos feitos anteriormente, que consideram fator crítico para sucesso da implementação do SMD esse alinhamento entre estratégia e ação (ATKINSON, 2006; BHAGWAT; SHARMA, 2007; MACEDO; BARBOSA; CAVALCANTE, 2009). Essa não é uma dificuldade única da área de desenvolvimento de *software*, mas o fato da área estar defasada no que diz respeito à utilização de sistemas de medição de desempenho faz com que exista tamanha discrepância entre o que está sendo medido no operacional e o que deveria ser medido se observando a estratégia da organização.

Para tentar superar essas dificuldades, propõe-se um modelo de desenvolvimento de *software* que leve em consideração as métricas necessárias para a avaliação de seu desenvolvimento. Dos modelos existentes, diversas práticas foram apresentadas e aprimoradas, e um conjunto delas pode ser considerado como melhores práticas (BOOCH, 1998). Entretanto, não se pode dizer que somente tais práticas resolverão todos os problemas da Engenharia de *Software*, uma vez que apesar de elas existirem, os números mostram que os projetos de *software* ainda falham (STANDISH GROUP INTERNATIONAL, 2009). Os mesmos

números indicam que a utilização de conceitos ágeis tem aumentado o número de sucessos em projetos desse tipo.

Partindo dessa ideia, um modelo deveria unir alguns princípios, sem a preocupação em seguir uma determinada tendência de mercado, mas sim garantindo que as práticas atualmente conhecidas sejam discutidas em sua forma.

Nesse modelo, deve-se considerar que o projeto de *software* poderá ter um ou mais ciclos de gestão, organizados por meio da priorização de necessidades a ser implementadas. Com base em conceitos ágeis, espera-se que cada ciclo seja planejado de tal maneira que sejam entregues partes do *software* ao final de um curto espaço de tempo (SCHWABER; SUTHERLAND, 2011).

## 4 Considerações finais

A deficiência quanto ao conhecimento dos profissionais em relação às normas e aos modelos de processo de *software* que apoiam o uso de sistemas de medição de desempenho em projetos dificulta a aplicação de tais sistemas para auxiliar no controle, avaliação, entendimento e prevenção dos processos e produtos de *software*. Isso significa dizer que a dificuldade encontrada em atingir prazos, custos e funcionalidades esperadas nesse tipo de projeto poderia ser minimizada se o uso da medição fosse mais efetivo nessa indústria.

Os resultados obtidos neste estudo são semelhantes aos da pesquisa realizada pelo Ministério de Tecnologia Brasileiro (MCT, 2010), o que confirma que existe uma deficiência na utilização de sistemas de medição de desempenho para projetos de desenvolvimento de *software* no grupo estudado.

O fato de a pesquisa possuir uma abordagem qualitativa e ter um número concentrado de respostas no estado de São Paulo limita a afirmação de

que os dados obtidos são válidos para todo o Brasil. Entretanto, o método aplicado no artigo pode ser reutilizado para outros grupos com o objetivo de detectar se existe semelhança de comportamento.

Como estudo futuro, sugere-se o desenvolvimento do modelo de processo de desenvolvimento de *software* centrado na utilização de um sistema de medição de desempenho, que possa ressaltar a aplicação de tal técnica para uma gestão mais alinhada com as expectativas estratégicas, já que essa pareceu uma das maiores dificuldades enfrentadas pela comunidade em relação à utilização dos sistemas de medição de desempenho.

## Referências

ABNT – ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR ISO/IEC 15504-5:2008. Tecnologia da informação – Avaliação de processo – Parte 5: Um exemplo de modelo de avaliação de processo.* [S.I.], 2008.

ABNT – ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR ISO/IEC 12207:2009. Tecnologia da informação – Processos de ciclo de vida de software.* [S.I.], 2009.

AL-TARAWNEH, M. Y.; ABDULLAH, M. S.; ALI, A. B. M. A proposed methodology for establishing software process development improvement for small software development firms. *Procedia Computer Science*, v. 3, 2011.

ANACLETO, A.; VON WANGENHEIM, C. G.; SALVIANO, C. F. Um método de avaliação de processos de software em micro e pequenas Empresas. In: SBQS – SIMPÓSIO BRASILEIRO DE QUALIDADE DE SOFTWARE, 2005, Porto Alegre. *Anais...* Porto Alegre: [s.n.]. 2005.

ATKINSON, H. Strategy implementation: a role or the balanced scorecard? *Management Decision*, v. 44, n. 10, p. 1441-1460, 2006.

BHAGWAT, R.; SHARMA, M. K. Performance measurement of supply chain management: A balanced scorecard approach. *Computers & Industrial Engineering*, v. 53, p. 43-62, 2007.

BOOCH, G. Leaving Kansas. *IEEE Software*, v. 15, n.1, p. 32-35, Jan.-Feb. 1998.

CORDEIRO, A.; FREITAS, A. O cenário atual da qualidade de software. In: SIMPÓSIO DE ENGENHARIA DE PRODUÇÃO, 2008, Bauru/SP. *Anais...* Bauru: SIMPEP, 2008.

DEMARCO, T. *Controlling software projects: management, measurement, and estimates.* USA: Prentice Hall, 1986.

DÍAZ, M. S.; GIL, M. J. Á.; MACHUCA, J. A. D. Performance measurement systems, competitive priorities, and advanced manufacturing technology. *International Journal of Operations & Production Management*, v. 25, n. 8, p. 781-799, 2005.

DIJKSTRA, E. W. The humble programmer. *Communications of the ACM*, v. 15, n. 10, p. 859-866, Oct. 1972.

EL-BAZ, M. A. Fuzzy performance measurement of a supply chain in manufacturing companies. *Expert Systems with Applications*, v. 38, n. 6, p. 6681-6688, Jun. 2011.

ERALP, Ö. *Design and implementation of a software development process measurement system.* Theses (Master of Science)– Middle East Technical University: Turksh, 2004.

GAIARDELLI, P.; SACCANI, N.; SONGINI, L. Performance measurement of the after-sales service network–Evidence from the automotive industry. *Computers in Industry*, v. 58, p. 698-708, 2007.

GANSSLE, J. Developing a good bedside manner. *Embedded Systems Design*, 2009. Disponível em: <<http://www.ganssle.com/articles/developingagoodbedsidemanner.htm>>. Acesso em: nov. 2011.

HUMPHREY, W. *A discipline for software engineering.* SEI series in software engineering. Boston: Addison-Wesley, 1995.

HUMPHREY, W. S. *Managing the software process.* Boston: Addison-Wesley Professional, 1989.

JONES, C. *Applied software measurement: global analysis of productivity and quality.* 3<sup>rd</sup> ed. Columbus: McGraw-Hill, 2008.

KAPLAN, S.; NORTON, P. *A estratégia em ação – balanced scorecard.* Rio de Janeiro: Campus, 1997.

KOSCIANSKI, A.; SOARES, M. *Qualidade de software: aprenda as metodologias e técnicas mais modernas para o desenvolvimento de software.* 2<sup>a</sup> ed. São Paulo: Novatec, 2007.

MACEDO, M. A. S.; BARBOSA, A. C. T. A. M.; CAVALCANTE, G. T. Desempenho de agências bancárias no Brasil: aplicando análise envoltória de dados (DEA) a indicadores relacionados às perspectivas do BSC. *Revista Economia e Gestão*, v. 9, n. 19, 2009.

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA. Pesquisa de qualidade no setor de software brasileiro 2009. Brasília, DF: Ministério da Ciência e Tecnologia, 2010.

- NA, K. et al. Software development risk and project performance measurement: Evidence in Korea. *Journal of Systems and Software*, v. 80, p. 596-605, 2007.

NEELY, A.; GREGORY, M.; PLATTS, K. Performance measurement system design: A literature review and research agenda. *International Journal of Operations and Production Management*, v. 15, n. 4, p. 80-116, 1995.

PERSSE, J. *Project management success with CMMI: seven CMMI process areas*. Upper Saddle river, NJ: Pearson Education, 2007.

PINO, F. J., et. al. Assessment methodology for software process improvement in small organizations. *Information and Software Technology*, v. 52, p. 1044-1061, 2010.

PRESSMAN, R. S. *Engenharia de software: uma abordagem profissional*. 7. ed. Porto Alegre: AMGH, 2011.

SALVIANO, C. F. Projetos da CE-21-007-10 e novidades da ISO/IEC 15504 (SPICE). São Paulo, 2009.

SCHWABER, K.; SUTHERLAND, J. *The Scrum Guide: The Definitive Guide to Scrum*. [S.l.]: Scrum.org, 2011.

SEVERINO, A. J. *Metodologia do trabalho científico*. 23. ed. São Paulo: Cortez, 2007.

SLACK, N.; CHAMBERS, S.; JOHNSTON, R. *Administração da produção*. São Paulo: Atlas, 2009.

SOCIEDADE BRASILEIRA DA COMPUTAÇÃO. Melhorias nos processos de software. *Computação Brasil*, Porto Alegre, n. 14, p. 14-16, 2010.

SOFTEX. MPS.BR – *Melhoria de processo do software brasileiro*: Guia geral. SOFTEX. Campinas, 2011.

SOFTWARE ENGINEERING INSTITUTE. CMMI® for development V1.3. Software Engineering Institute. Pittsburgh. 2010.

SOLINGEN, R.; BERGHOUT, E. *The goal/question/metric method: a practical guide for quality improvement of software development*. London: McGraw-Hill, 1999.

SOMMERVILLE, I. *Engenharia de software*. 9. ed. São Paulo: Pearson Prentice Hall, 2011.

STANDISH GROUP. *CHAOS Summary 2009*. The Standish Group International. Boston, 2009. Relatório.

UKKO, J.; TENHUNEN, J.; RANTANEN, H. Performance measurement impacts on management and leadership: Perspectives of management and employees. *International Journal of Production Economics*, v. 100, p. 39-51, 2007.

Recebido em 29 ago. 2012 / aprovado em 26 set. 2012

Para referenciar este texto

BAPTISTA, G. L.; VANALLE, R. M.; SALLES, J. A. A. Utilização de sistemas de medição de desempenho em projetos de desenvolvimento de *software*. *Exacta*, São Paulo, v. 10, n. 2, p. 181-191, 2012.

