

## **Medidas de tamanho de desenvolvimento e de melhorias de software**

por Caio Juliano Paro ([cparo@dba.com.br](mailto:cparo@dba.com.br))

### **Introdução**

Vivenciamos no dia-a-dia das empresas de tecnologia (em particular, as que desenvolvem soluções de software), a disseminação, amadurecimento e aplicação de conceitos relacionados à gerência de projetos. Com o crescimento e fortalecimento da principal associação profissional nesse assunto, o PMI (Project Management Institute), essas empresas buscam no aprimoramento de suas técnicas de gerenciamento, obter um diferencial competitivo no mercado de desenvolvimento de software.

Fazendo parte do processo de gestão, surge como uma necessidade fundamental na área de gerenciamento de projetos de desenvolvimento de software, a adoção de um processo eficiente e simples para medição do tamanho dos projetos de desenvolvimento e melhoria de sistemas. Esse artigo visa apresentar algumas técnicas padrão para determinar esse tamanho.

### **Razões para Medir**

Existem diversos motivos que levam as empresas a adotarem um processo para medir o tamanho dos projetos de desenvolvimento de sistemas:

- Prover subsídio para determinar esforço, recursos, duração e custo do desenvolvimento;
- Gerenciar contratos de software;
- Indicar a qualidade do produto;
- Formar uma baseline para estimativas;
- Avaliar a produtividade do processo.

### **Tipos de Medidas de Tamanho de Desenvolvimento de Sistemas**

Abaixo, estão relacionadas algumas técnicas padrão para determinar o tamanho do desenvolvimento de sistemas.

#### ***Linhas de Código***

A técnica de mensuração por linhas de código (LOC – Lines of Code) é uma das medidas mais antigas para determinação do tamanho, esforço e, conseqüentemente, produtividade no desenvolvimento de software. Ela consiste basicamente na contagem da quantidade do número de linhas de código de um programa de software. É uma técnica de fácil automação, eliminando esforços manuais. Porém, é uma técnica que conta com muitas desvantagens. Podemos citar, entre elas, o fato de que é inexato ter que medir a produtividade de um projeto de desenvolvimento com a saída de somente uma das fases (fase de codificação); a experiência do desenvolvedor, pois o número de linhas de código pode variar de pessoa a pessoa; a diferença entre linguagens (o número de linhas de código

de uma aplicação desenvolvida em Cobol certamente é diferente da quantidade de linhas de código da mesma aplicação desenvolvida em linguagem C); a ausência de padrões de contagem, já que não há uma definição de certas características, como a contagem de comentários, declarações de dados, e a incapacidade de medição do tamanho de um sistema em fases iniciais de desenvolvimento, tornando o gerenciamento do projeto um tanto quanto complicado.

### ***Análise de Pontos de Função (APF)***

Podemos citar a técnica de Análise de Pontos de Função (APF) como a mais utilizada no mercado para mensuração do tamanho de projetos de desenvolvimento e melhoria de sistemas. Criada no início da década de 70 por Alan Albrecht na IBM, a Análise de Pontos de Função consiste na determinação do tamanho funcional do sistema através da visão do usuário do mesmo, independente da tecnologia utilizada. A unidade de medida utilizada na determinação do tamanho do sistema é o Ponto de Função. Essa técnica é respaldada pelo IFPUG (International Function Point Users Group), que é responsável, entre outros, pela elaboração, manutenção e divulgação de um manual de práticas de contagem (CPM – Counting Practices Manual), atualmente em sua versão 4.2.1, além de manter um programa de certificação para profissionais especializados em aplicar a técnica de APF (CFPS – Certified Function Point Specialist).

Essa técnica conta com vários benefícios e vantagens para utilização:

- É uma ferramenta para determinar o tamanho de um pacote de aplicação adquirido através da contagem de todas as funções incluídas no pacote;
- Um fator de normalização para comparação de software;
- Um meio para estimar custo e recursos requeridos para o desenvolvimento e manutenção de software;
- Uma ferramenta para medir as unidades de um produto de software para suportar a análise de qualidade e produtividade;
- Promove benchmarking entre várias organizações;
- Manutenção de dados históricos;
- Pode medir o tamanho em qualquer fase do ciclo de vida do desenvolvimento de sistema.

Podemos citar também algumas desvantagens, como a necessidade de um bom nível de experiência no assunto para efetuar uma contagem acurada; a incapacidade de automação do processo de contagem, sendo necessária a contagem manual; a necessidade significativa de um nível de detalhe de informações do software para uma medição mais confiável (entradas, saídas, consultas, registros, bases, etc), entre outros.

Esta técnica conta com uma série de variações, que buscam aprimorar conceitos existentes e acrescentam outros na contagem. Podemos citar como as principais variantes da APF:

#### *1- Mark II (ou MK II)*

Essa técnica foi criada por Charles Symons como um método proprietário da empresa KPMG nos anos de 1985/86, sendo atualmente de domínio público e regida pela UKSMA

(United Kingdom Software Metrics Association). Visa melhorar a escala de tamanho funcional, contando mais apuradamente a complexidade de processamento interno de Sistemas de Informações Gerenciais. Uma das diferenças principais entre APF e MK II é que a primeira técnica conta “Arquivos Lógicos” uma vez para cada parte de software sendo mensurada, enquanto que a MK II conta “tipos de entidade” toda vez que elas são referenciadas em cada transação lógica. Segundo o manual do MK II, as duas métricas pontuam similarmente os projetos com até 400 pontos de função. Ultrapassando esse valor, a tendência é que a métrica MK II pontue valores maiores que a APF.

## *2- Feature Points*

Criado no final dos anos 80 por Capers Jones, este modelo visa acrescentar um novo item à contagem tradicional de Pontos de Função. Além das funções de dados (Arquivos Lógicos Internos e Arquivos de Interface Externa) e funções de transação (Entradas Externas, Saídas Externas e Consultas Externas), foi embutido o item de análise de algoritmos. Propõe resolver as dificuldades que a técnica de Análise de Pontos de Função apresenta quando conta sistemas em tempo real de controle de processos e outros que apresentam alta complexidade algorítmica. Além disso, o método também reduz os pesos empíricos das funções de dados e de transação. Este método não tem muita aceitação devido à dificuldade inerente de unificar maneiras padrão de definir e atribuir um peso para algoritmos de tamanho e complexidade aumentados.

## *3- COSMIC – FFP*

A técnica nomeada COSMIC-FFP (Common Software Measurement International Consortium – Full Function Points) foi lançada em 2000, através do consórcio chamado COSMIC, liderado por Alain Abran e Charles Symons. Baseada na técnica de FFP, que foi criada por um grupo da Universidade de Quebec em Montreal, Canadá, a COSMIC-FFP é intencionada a contar além de Sistemas de Informações Gerenciais, as aplicações em tempo real e multiplataforma. Ela não leva em conta a manipulação de dados complexos (algoritmos) e requerimentos técnicos e de qualidade. Existem muitos estudos acerca dessa métrica, e é possível o crescimento da utilização da mesma no mercado.

## *UCP – Use Case Points*

Técnica proposta em 1993 por Gustav Karner para medir o tamanho de projetos de software orientados a objeto, o Use Case Points (ou Pontos por Caso de Uso) explora o modelo e a descrição do caso de uso, substitui algumas características técnicas propostas pela APF, cria os fatores ambientais e propõe uma estimativa de produtividade. É um método simples e fácil de se utilizar. Porém, conta com algumas desvantagens, como: só pode ser utilizado por empresas que adotem os casos de uso como forma de expressão dos requisitos; não conta com um instituto que mantenha as regras de contagem do UCP; ainda está em fase de pesquisas, sendo pouco utilizado se comparado com a APF, entre outros. Assim como a técnica COSMIC-FFP, é possível o crescimento da utilização da mesma no mercado.

## Conclusão

Através desse levantamento básico das técnicas de medição de tamanho de software, fica evidente que a técnica de Análise de Pontos de Função mudou o paradigma de contagem, pois além da forte difusão no mercado, outras técnicas derivaram de seus conceitos básicos visando o aperfeiçoamento. Podemos destacar também os estudos acerca das técnicas COSMIC-FFP e Use Case Points, que tendem a se intensificar e difundir no mercado.

## Bibliografia

PMI – SP - <http://www.pmis.org.br/exe/pmi/instituto.asp>

Gomes, Alvaro Eduardo. Métricas e Estimativas de Software - O início de um rally de regularidade. [http://www.linhadecodigo.com.br/artigos.asp?id\\_ac=102&sub=0](http://www.linhadecodigo.com.br/artigos.asp?id_ac=102&sub=0)

Symons, Charles. Come Back Function Point Analysis (Modernized) – All is Forgiven! - <http://www.lrgl.uqam.ca/publications/pdf/647.pdf>

Andrade, Edméia Leonor Pereira. Pontos de Caso de Uso e Pontos de Função na gestão de estimativa de tamanho de projetos de software orientados a objetos <http://www.bfpug.com.br/Artigos/UCP/Tese%20Edmeia.zip>

Aguiar, Mauricio. Pontos de Função ou pontos por caso de uso? [http://www.bfpug.com.br/Artigos/UCP/Aguiar-Pontos\\_de\\_Funcao\\_ou\\_Pontos\\_por\\_Caso\\_de\\_Uso.pdf](http://www.bfpug.com.br/Artigos/UCP/Aguiar-Pontos_de_Funcao_ou_Pontos_por_Caso_de_Uso.pdf)

Hazan, Claudia. Implantação de um processo de medições de software. <http://www.bfpug.com.br/Artigos/Palestra%20Medicoes%20Claudia%20Hazan.pdf>

AZEVEDO, DOUGLAS JOSÉ PEIXOTO. Análise de Pontos por Função para Aplicações Orientadas a Documentos - [http://celepar7cta.pr.gov.br/Celepar/SiteCel.nsf/0/eeeadf84dc199adc83256a320015bf56/\\$FILE/DissertacaoDouglas.htm](http://celepar7cta.pr.gov.br/Celepar/SiteCel.nsf/0/eeeadf84dc199adc83256a320015bf56/$FILE/DissertacaoDouglas.htm). Março de 1999.

Gollapudi, Kurmanadham V.V.G.B., Global Microsoft Business Unit, Wipro Technologies. Function Points or Lines of Code? - An Insight - <http://www.gotdotnet.com/Community/UserSamples/Download.aspx?SampleGuid=791516BD-B0EF-491A-BE1E-0D622776197B>

Vasquez, Carlos Eduardo. Análise de pontos de função: medição, estimativas e gerenciamento de projetos de software / Carlos Eduardo Vasquez, Guilherme Siqueira Simões, Renato Machado Albert. – 1. ed. – São Paulo: Érica, 2003.

IFPUG - International Function Point Users Group. Function Point Counting Practices Manual (CPM), Release 4.2.1. Janeiro, 2005.

COSMIC – Common Software Measurement International Consortium. COSMIC-FFP Measurement Manual, Version 2.2. Janeiro, 2003.

UKSMA – United Kingdom Software Metrics Association. MK II Function Point Analysis Counting Practices Manual, Version 1.3.1. Setembro , 1998.

**Sobre o autor:**

Caio Juliano Paro tem 24 anos, é formado em Tecnologia de Processamento de Dados pela FATEC – SP, e atualmente estuda o curso de Especialização em Análise e Projeto de Sistemas, também na FATEC-SP. Possui a certificação CFPS – Certified Function Point Specialist, e trabalha com a técnica de Análise de Pontos de Função há cerca de dois anos, tanto na elaboração e validação de contagens quanto na divulgação da técnica através de apresentações e treinamentos.