

**UFPE - Universidade Federal de Pernambuco**  
**CIN - Centro de Informática**

**Gestão da tecnologia da informação usando análise por  
pontos de função**

Por:

Alexsandro José de Melo Farias  
Diogo Cunha  
Fabrício de Siqueira Teles  
Leonardo Pinto de Holanda Rodrigues  
Pedro Machado Manhães de Castro

Recife, 13 de agosto de 2004.

## Sumário

|  |           |
|--|-----------|
| <b>RESUMO.....</b>   | <b>4</b>  |
| <b>ABSTRACT .....</b>  | <b>5</b>  |
| <b>1. INTRODUÇÃO .....</b>   | <b>6</b>  |
| 1.1 MOTIVAÇÃO .....  | 6         |
| 1.2 OBJETIVOS.....   | 7         |
| <b>2. FUNDAMENTAÇÃO TEÓRICA – PRIMEIRAS TÉCNICAS.....</b>              | <b>7</b>  |
| 2.1 LINHAS DE CÓDIGO .....   | 7         |
| 2.2 SISTEMA HALSTEAD .....   | 7         |
| 2.3 ANÁLISE POR PONTOS DE FUNÇÃO .....                                 | 8         |
| <b>3. VISÃO GERAL DA ANÁLISE DE PONTOS DE FUNÇÃO .....</b>             | <b>8</b>  |
| 3.1 BENEFÍCIOS COM O USO DA APF .....                                  | 8         |
| 3.2 ETAPAS PARA AVALIAÇÃO - CINCO ETAPAS .....                         | 9         |
| 3.2.1 Etapa I - Identificação do Tipo de Contagem a ser utilizado..... | 9         |
| 3.2.2 Etapa II - Definição da Fronteira da Aplicação .....             | 9         |
| 3.2.3 Etapa III - Contagem de Pontos de Função não Ajustados.....      | 9         |
| 3.2.3.1 Regras para Contagem dos Pontos de Função Brutos .....         | 9         |
| 3.2.3.2 Cálculo de Pontos de Função não Ajustados ou Brutos .....      | 11        |
| 3.2.4 Etapa IV - Cálculo do Fator de Ajuste .....                      | 11        |
| 3.2.5 Etapa V - Contagem de Pontos de Função Ajustados .....           | 18        |
| <b>4. GESTÃO DA TECNOLOGIA DA INFORMAÇÃO.....</b>                      | <b>20</b> |
| 4.1. ESTIMATIVA DA PRODUTIVIDADE COM PONTOS DE FUNÇÃO .....            | 20        |
| 4.1.1. Produtividades no ambiente de desenvolvimento .....             | 20        |
| 4.1.2. Produtividade no ambiente de manutenção .....                   | 20        |
| 4.1.3. Produtividade no ambiente de suporte .....                      | 21        |
| 4.1.4. Escolha da taxa de produtividade apropriada .....               | 21        |
| 4.1.5. Reutilização e Produtividade .....                              | 21        |
| 4.1.6. Refinamento das estimativas.....                                | 22        |
| 4.1.7. Análise de Resultados .....                                     | 22        |
| 4.1.8. Atualização da Base de Informações .....                        | 23        |
| 4.2. CUSTO DE SERVIÇOS DE INFORMÁTICA .....                            | 23        |
| 4.3. QUANDO E ONDE FAZER REENGENHARIA.....                             | 24        |
| 4.3.1. Identificando aplicações para reengenharia.....                 | 24        |
| 4.3.2. Estimando benefícios (calculando o retorno) .....               | 25        |
| 4.4. ESTIMANDO CASOS DE TESTE .....                                    | 26        |
| 4.4.1. Entendendo o Potencial de Defeitos.....                         | 26        |
| 4.5. ENTENDENDO FAIXAS DE PRODUTIVIDADE AMPLAS.....                    | 27        |
| 4.6. ENTENDENDO O AUMENTO DO ESCOPO.....                               | 27        |

|   |           |
|---|-----------|
| 4.7. ESTIMANDO O CUSTO, CRONOGRAMA E ESFORÇO DO PROJETO ..... | 27        |
| 4.8. CALCULANDO O CUSTO REAL DO SOFTWARE .....                | 28        |
| 4.9. ENTENDENDO OS CUSTOS DE MANUTENÇÃO.....                  | 28        |
| 4.10. AJUDANDO EM NEGOCIAÇÕES CONTRATUAIS .....               | 29        |
| 4.11. DESENVOLVENDO UM CONJUNTO PADRÃO DE MÉTRICAS.....       | 29        |
| <b>5. CONSIDERAÇÕES FINAIS .....</b>                          | <b>30</b> |
| <b>6. REFERÊNCIAS BIBLIOGRÁFICAS.....</b>                     | <b>32</b> |

## Resumo

Este artigo aborda a técnica de Análise por Ponto de Função (APF) como ferramenta adequada para auxiliar no processo de gestão de Tecnologia da Informação (TI). Gerentes de TI podem medir a produtividade relacionada com o esforço realizado no processo de desenvolvimento, otimização e manutenção de sistemas de informação. Através desta técnica é possível calcular o esforço despendido por unidade, ou por atividade no departamento de TI, independente do tipo do sistema, da tecnologia utilizada para desenvolver o produto, e da habilidade dos desenvolvedores, e ainda, fornecer subsídios para melhor compreensão das correções, falhas e dos problemas de planejamento dos projetos já concluídos ou em andamento. Permite criar um ambiente padrão na área de TI com possibilidade de comparação de seu desempenho com o mercado.

## Abstract

This article approaches the technique of Function Point Analysis (PFA) as adjusted tool to assist in the management process of Information Technology (IT). IT managers can measure the productivity related with the effort carried through in the development, optimization and maintenance process. Through this technique it is possible to calculate the effort expended for unit, or activity in the TI department, independent of the system type, the used technology to develop the product, and of the ability of the developers, and still, to supply subsidies better understanding of the corrections, imperfections and of the problems of planning of the finished projects or in progress. It allows creating TI standard environment and the possibility of comparison of its performance with the market.

## 1. Introdução

A evolução do uso da informação no mundo dos negócios não pode mais ser ignorada pois tornou - se fator determinante de competitividade. Empresas em todos os ramos de atividades passaram a se preocupar em buscar respostas para perguntas como: estamos usando adequadamente a informação? Possuímos suficientes recursos tecnológicos? Qual deve ser a estratégia de investimentos em Tecnologia da Informação (TI) em nossa empresa?

Na gestão de TI o maior problema consiste em se definir indicadores para o gerenciamento dos recursos para o atendimento pleno da demanda dos usuários com presteza e qualidade.

A demanda por produtos e serviços de informática cresce continuamente em função da necessidade de novos serviços, suportes e manutenções legais, enquanto o orçamento da área sofre reduções, sistemáticas, a cada plano de governo, ou a cada política, de enxugamento e/ou contenção de despesas, implementada.

Selecionar métricas adequadas para o gerenciamento de TI não é uma tarefa trivial, principalmente devido à natureza dinâmica da área. A complexidade aumenta quando observamos a existência de considerável participação das despesas com recursos humanos na execução orçamentária. Como gerenciar os recursos que se reduzem a cada ano, para que se possa atender um número cada vez maior de usuários? Quais as métricas de desempenho de sistemas são mais usadas? Qual o grau de satisfação de nossos clientes? Estas e outras perguntas são freqüentes na área de tecnologia da informação, e a busca por respostas constitui-se numa tarefa árdua para os administradores que, na maioria das vezes, são obrigados a assumir compromissos com usuários sem a certeza de dispor de recursos necessários para o atendimento.

### 1.1 Motivação

A técnica de análise por Ponto de Função auxilia os gerentes TI a medir a produtividade relacionada com o esforço realizado no processo de desenvolvimento, otimização e manutenção de sistemas. Baseia-se em três taxas de produtividade de sistemas - distribuição, modificação e suporte - todas consideradas sob o ponto de vista do usuário. Através desta técnica é possível calcular o esforço despendido por unidade, ou por atividade no departamento de TI e ainda, fornecer subsídios para melhor compreensão das correções, falhas e dos problemas de planejamento dos projetos já concluídos ou em andamento, de forma a desenvolver projetos futuros de maneira mais eficiente.

A taxa de produtividade pode ser entendida como a razão entre a quantidade de produtos liberados e o esforço despendido no desenvolvimento, expressa em um período de tempo. Para identificar oportunidades de melhoria, há que se conhecer a taxa atual de produtividade das unidades produtoras de forma a comparar com a meta definida. Por exemplo, na indústria automobilística, a saída é facilmente mensurada a partir da contagem de carros que saem da linha de produção. Porque cada carro é o mesmo e os meios pelos quais eles são manufaturados, e as atividades, executadas por cada trabalhador na linha de produção, são padronizadas, desta forma o método de contagem é válido. No caso de

sistemas de informação, entretanto, cada aplicação é diferente, o processo através do qual cada aplicação é desenvolvida varia, e as atividades executadas são significativamente afetadas pelas habilidades dos trabalhadores. Devido a estas variáveis este processo torna-se muito complexo, invalidando a utilização do método de contagem abordado em uma indústria automobilística. Este enfoque de contagem (ex.considerar o número de sistemas implementados por mês) não é apropriado para medir a quantidade de saída ou produto. Para o departamento de informática a produtividade deve ser medida através de um método que seja independente do tipo do sistema, da tecnologia utilizada para desenvolver o produto, e da habilidade dos desenvolvedores. A técnica de pontos de função - desenvolvida por Allan Albrecht da IBM - fornece o método que mais se aproxima do atendimento aos requisitos mencionados.

## 1.2 Objetivos

Não considerando a tecnologia utilizada, para o desenvolvimento de um sistema de informação e o ambiente no qual ele será executado, cada sistema tem como objetivo comum - executar um conjunto definido de ações/funções para o usuário final. A técnica de pontos de função faz a medição de um sistema baseado nas características relativas ao seu funcionamento e objetivos identificadas a partir da visão do usuário. Assim, como o dimensionamento do sistema é considerado sob o ponto de vista do usuário, o processo envolve a capacidade funcional do sistema que está à disposição do usuário, não dependendo da tecnologia empregada para o desenvolvimento, ou do conhecimento, habilidade e experiência do grupo que desenvolveu ou fez a manutenção do sistema.

## 2. Fundamentação teórica – primeiras técnicas

Esta seção apresenta uma breve descrição três técnicas de dimensionamento de sistemas: linhas de código, sistema Halstead e pontos de função. Serão relatadas também as dificuldades / pontos fracos de cada uma dessas técnicas.

### 2.1 Linhas de código

Dimensionamento dos sistemas através da quantidade de linhas de códigos. Apresenta dificuldades para uso em comparações de sistemas desenvolvidos por linguagens diferentes; apresenta distorções, para efeito de comparação, por ser dependente do estilo dos programadores; apresenta impossibilidade de uso em estimativas, considerando que os sistemas somente poderiam ser medidos depois de codificados.

### 2.2 Sistema Halstead

Dimensionamento dos sistemas baseado na quantidade de operandos (itens de dados) e operados (comandos de linguagem de dados) Apresenta dificuldades por ser

dependente das características da linguagem de programação; dependente do estilo dos programadores; apresenta dificuldades para uso em estimativas, considerando que os programas somente poderão ser medidos depois de codificados.

## 2.3 Análise por Pontos de função

Mede a funcionalidade do sistema baseado na visão do usuário, apresentando as seguintes características:

- Independência de tecnologia utilizada
- Produção de resultados consistentes
- Baseada na visão do usuário
- Significância para o usuário final
- Utilização em estimativas
- Passível de automação.

Apresenta dificuldades por possuir relativa subjetividade, considerando que reflete a visão do usuário. A complexidade está relacionada com volumes de arquivos lógicos, registros lógicos e itens de dados, identificados. Considera 14 itens de influência subdivididos em subitens que abrangem todos os aspectos relacionados às necessidades de uso de ferramentas para atendimento das requisições dos usuários.

## 3. Visão Geral da Análise de Pontos de Função

Trata-se da técnica que permite medir a funcionalidade de um software ou aplicativo, sob a visão do usuário, e a partir da descrição dos requisitos do usuário.

Objetivos da técnica:

- Medir a funcionalidade dos sistemas;
- Independentemente da tecnologia que foi utilizada no seu desenvolvimento;
- Avaliar com base no que o sistema faz, ou seja, as funções que o sistema executa, independente da tecnologia em que foi implementado;
- Identificar um padrão de medida para a produtividade e qualidade;
- Fornecer ferramenta para auxiliar nas estimativas de desenvolvimento de software;
- Possibilitar o uso por diferentes projetos, em diferentes empresas e ambientes variados, demonstrando consistência na comparação;
- Ser comprehensível pelo pessoal não técnico;
- Ser utilizável em todo o ciclo de desenvolvimento do software;
- Ser um processo simples de mensuração;
- Ser conciso e consistente.

### 3.1 Benefícios com o Uso da APF

O uso da técnica de análise por pontos de função traz os seguintes benefícios:

- Dimensionamento dos sistemas, em produção e/ou em desenvolvimento bem como, solicitações de manutenção;

- Apoio para estimativa de custos e recursos requeridos para o desenvolvimento e manutenção de software;
- Apoio para gerenciamento da qualidade e produtividade no processo de desenvolvimento do software;
- Apoio para a tomada de decisão relativa à seleção para aquisição de pacotes, e contratação de serviços.

## 3.2 Etapas para Avaliação - Cinco Etapas

Esta seção relata as cinco etapas necessárias para se realizar a contagem da análise por pontos de função.

### 3.2.1 Etapa I - Identificação do Tipo de Contagem a ser utilizado

O quê vamos medir? Consiste na identificação do objeto a ser medido, como sendo um projeto de desenvolvimento, manutenção ou produção.

### 3.2.2 Etapa II - Definição da Fronteira da Aplicação

Quais os limites do que vamos medir? Esta é a etapa em que é estabelecido o escopo do sistema objeto da avaliação, sob a visão do usuário. Neste momento são identificados todos os relacionamentos do sistema, sob avaliação, com o seu exterior, e são identificados às pertinências dos dados e os processos suportados pelo sistema que está sendo contado. Em resumo a fronteira separa o sistema das aplicações externas.

### 3.2.3 Etapa III - Contagem de Pontos de Função não Ajustados

Reflete o conjunto de funções disponibilizadas ao usuário, e o resultado da contagem pode ser considerado como pontos de função brutos, face à necessidade de se observar outras variáveis que influenciam nos cuidados e esforços a serem despendidos durante o processo de desenvolvimento do sistema. Assim as funções são definidas como:

- *Arquivos Lógicos Internos;*
- *Arquivos de Interface Externa;*
- *Entradas Externas;*
- *Saídas Externas;*
- *Consultas Externas.*

#### 3.2.3.1 Regras para Contagem dos Pontos de Função Brutos

Após a definição sobre o tipo de contagem que será realizada, ou seja, se o sistema a ser dimensionado é um projeto de desenvolvimento, manutenção ou produção, e após a

identificação da fronteira da aplicação, segue-se à identificação dos grupos de funções tipo dados e transações, e a contagem de seus registros e itens de dados.

- *Arquivo Lógico Interno - ALI*: trata-se de grupos de dados ou informações de controle, inter-relacionados, requisitados pelo usuário como necessidades de informação, cuja manutenção, realizada por um processo elementar da aplicação (alteração, inclusão, exclusão), acontece dentro da fronteira da aplicação.
- *Arquivo de Interface Externa - AIE*: são grupos de dados, inter-relacionados, necessários para a aplicação, mantidos e armazenados fora do sistema que está sendo dimensionado. Baseia-se na Identificação dos dados armazenados fora da fronteira da aplicação, dados que não sofrem manutenções pela aplicação que está sendo avaliada, dados identificados como necessidades de informação do usuário e dos dados que são contados como ALI para outra aplicação. A complexidade é determinada em função da quantidade de registros lógicos e itens de dados referenciados.
- *Entradas Externas - EE*: são grupos de dados que entram no sistema, utilizados para a manutenção dos Arquivos Lógicos Internos, ou seja, provocam uma inclusão, exclusão e/ou alteração nos dados dos ALI. Baseia-se na Identificação dos processos que recebem dados externos e que atualizam arquivos lógicos internos. A complexidade funcional da EE é determinada em função da quantidade de arquivos lógicos internos e itens de dados referenciados.
- *Saídas Externas - SE*: representam as atividades do sistema que transformam dados dos arquivos lógicos internos e geram resultados que são exibidos ao usuário. A contagem das saídas externas é baseada na quantidade de arquivos lógicos referenciados e na quantidade de itens de dados referenciados. A Identificação da Saídas Externas – SE baseia-se nos processos que enviam dados para fora da fronteira da aplicação. Contando-se cada processo que envia dados ou informações para fora da fronteira da aplicação, bem como cada processo que necessitar de lógica de processamento diferente da utilizada em outras saídas externas que tenham o mesmo formato. A complexidade funcional das SE é determinada em função da quantidade de arquivos lógicos e itens de dados referenciados.
- *Consultas Externas - CE*: são requisições de informações que, para serem satisfeitas, precisam que sejam combinados parâmetros de entradas e saídas que permitem a recuperação da informação solicitada pelo usuário. Nenhum arquivo lógico interno é mantido durante o processo, e as informações que compõem a saída não são dados derivados, ou seja, correspondem, exatamente, aos dados solicitados, na forma em que estiverem armazenados nos arquivos lógicos internos. Baseia-se na Identificação dos Processos onde uma entrada está associada a uma recuperação e exibição de dados, sem que haja processamentos, ou seja, não acontece transformação dos dados para serem exibidos aos usuários. São contados os arquivos lógicos e itens de dados referenciados deverão ser contados, considerando as entradas e saídas, separadamente, bem como cada processo de recuperação de dados que seleciona dados com base em parâmetros de entrada fornecidos.

### 3.2.3.2 Cálculo de Pontos de Função não Ajustados ou Brutos

Após a definição da fronteira da aplicação, da escolha do tipo de contagem, da agregação das funções de acordo com as cinco categorias de funções e da classificação da complexidade, podemos calcular os pontos de função não ajustados, ou brutos, aplicando as regras contidas na seguinte tabela:

| DESCRÍÇÃO                                      | COMPLEXIDADE FUNCIONAL | TOTAL COMPLEXIDADE | TOTAL TIPO FUNÇÃO |
|--|------------------------|--------------------|-------------------|
| <b>ALI - Arquivo Lógico Interno</b>            | Simples X 7 =          |                    |                   |
|  | Média X 10 =           |                    |                   |
|  | Complexa X 15 =        |                    |                   |
| <b>AIE – Arquivo de Interface Externa</b>      | Simples X 5 =          |                    |                   |
|  | Média X 7 =            |                    |                   |
|  | Complexa X 10 =        |                    |                   |
| <b>EE - Entrada Externa</b>                    | Simples X 3 =          |                    |                   |
|  | Média X 4 =            |                    |                   |
|  | Complexa X 6 =         |                    |                   |
| <b>SE - Saída Externa</b>                      | Simples X 4 =          |                    |                   |
|  | Média X 5 =            |                    |                   |
|  | Complexa X 7 =         |                    |                   |
| <b>CE – Consulta Externa</b>                   | Simples X 3 =          |                    |                   |
|  | Média X 4 =            |                    |                   |
|  | Complexa X 6 =         |                    |                   |
| <b>Total de Pontos de Função Não Ajustados</b> |                        |                    |                   |

### 3.2.4 Etapa IV - Cálculo do Fator de Ajuste

A metodologia de pontos de função considera que outros fatores afetam o tamanho funcional de um sistema. Estes fatores estão relacionados com características da aplicação: Observações:

- o nível de influência de cada uma das 14 características, a serem observadas varia de 0 a 5, correspondendo a uma escala de influência que parte da hipótese de nenhuma influência(0) até o grau de influência máxima(5)
- as características gerais do sistema podem influenciar no seu tamanho variando no intervalo de **-35% a +35%**. Isto implica em um intervalo de variação para o fator de ajuste da ordem de **0,65 a 1,35**. O fator de ajuste é responsável pela correção das distorções da etapa anterior. Baseia-se nas características gerais do sistema, correlacionando-as com uma tabela de referência que possui 14 itens, e determina o valor do nível de influência de cada item no dimensionamento do sistema.

- processo de cálculo: avaliar o impacto de cada uma das 14 características em relação ao sistema que está sendo avaliado, atribuindo pontuação de 0 a 5 para cada característica. Calcular o nível de influência através da soma dos pontos obtidos em cada uma das 14 características. Aplicar a seguinte fórmula:  
**Fator de Ajuste = (NI \* 0,01) + 0,65 (1)** onde: NI = somatório da pontuação atribuída a cada uma das 14 características, refletindo o nível de Influência global no dimensionamento do sistema.

Abaixo segue uma breve descrição de cada uma das 14 características do sistema:

- 1. *Comunicação de dados* – Grau de influência variando de 0 a 5 : os aspectos relacionados aos recursos utilizados para a comunicação de dados do sistema deverão ser descritos de forma global. Descrever se a aplicação utiliza protocolos diferentes para recebimento/envio das informações do sistema.  
Pontuação:

- 0 - aplicação batch ou funciona stand-alone;
  - 1 - aplicação batch, mas utiliza entrada de dados ou impressão remota;
  - 2 - aplicação batch, mas utiliza entrada de dados e impressão remota;
  - 3 - aplicação com entrada de dados on-line para alimentar processamento batch ou sistema de consulta;
  - 4 - aplicação com entrada de dados on-line, mas suporta apenas um tipo de protocolo de comunicação;
  - 5 - aplicação com entrada de dados on-line e suporta mais de um tipo de protocolo de comunicação;

- 2. *Funções distribuídas* – Grau de influência variando de 0 a 5 : Esta característica refere-se a sistemas que utilizam dados ou processamento distribuído , valendo-se de diversas CPU.

Pontuação:

- 0 - aplicação não auxilia na transferência de dados ou funções entre os processadores da empresa;
  - 1 - aplicação prepara dados para o usuário final utilizar em outro processador (do usuário final), tal como planilhas em pc;
  - 2 - aplicação prepara dados para transferência, transfere – os para serem processados em outro equipamento da empresa (não pelo usuário final);
  - 3 - processamento é distribuído e a transferência de dados é online e apenas em uma direção;
  - 4 - processamento é distribuído e a transferência de dados é online e em ambas as direções;
  - 5 - as funções de processamento são dinamicamente executadas no equipamento (CPU) mais apropriado.

3. *Desempenho*: Trata-se de parâmetros estabelecidos pelo usuário como aceitáveis, relativos a tempo de resposta.

Pontuação:

- 0 - nenhum requerimento especial de desempenho foi solicitado pelo usuário;
  - 1 - requerimentos de desempenho foram estabelecidos e revistos, mas nenhuma ação especial foi requerida;

- 2 - tempo de resposta e volume de processamento são itens críticos durante horários de pico de processamento. Nenhuma determinação especial para a utilização do processador foi estabelecida. A data limite para a disponibilidade de processamento é sempre o próximo dia útil;
- 3 - tempo de resposta e volume de processamento são itens críticos durante todo o horário comercial. Nenhuma determinação especial para a utilização do processador foi estabelecida. A data-limite necessária para a comunicação com outros sistemas é limitante.
- 4 - os requerimentos de desempenho estabelecidos requerem tarefas de análise de desempenho na fase de planejamento e análise da aplicação.
- 5 - além do descrito no item anterior, ferramentas de análise de desempenho foram usadas nas fases de planejamento, desenvolvimento e/ou implementação para atingir os requerimentos de desempenho estabelecidos pelos usuários.

*4 - Utilização do equipamento:* Trata-se de observações quanto ao nível de utilização de equipamentos requerido para a execução do sistema. Este aspecto é observado com vista a planejamento de capacidades e custos.

Pontuação

- 0 - nenhuma restrição operacional explícita ou mesmo implícito foi incluída.
- 1 - existem restrições operacionais leves. Não é necessário esforço especial para atender às restrições.
- 2 - algumas considerações de ajuste de desempenho e segurança são necessárias.
- 3 - são necessárias especificações especiais de processador para um módulo específico da aplicação.
- 4 - restrições operacionais requerem cuidados especiais no processador central ou no processador dedicado para executar a aplicação.
- 5 - além das características do item anterior, há considerações especiais que exigem utilização de ferramentas de análise de desempenho, para a distribuição do sistema e seus componentes, nas unidades processadoras.

*5. Volume de transações:* Consiste na avaliação do nível de influência do volume de transações do projeto, desenvolvimento, implantação e manutenção do sistema.

Pontuação

- 0 - não estão previstos períodos de picos de volume de transação;
- 1 - estão previstos picos de transações mensalmente; mensalmente, trimestralmente, anualmente ou em certo período do ano.
- 2 - são previstos picos semanais;
- 3 - são previstos picos diários;
- 4 - alto volume de transações foi estabelecido pelo usuário, ou o tempo de resposta necessário atinge nível alto o suficiente para requerer análise de desempenho na fase de projeto.
- 5 - além do descrito no item anterior, é necessário utilizar ferramentas de análise de desempenho nas fases de projeto, desenvolvimento e/ou implantação.

**6. Entrada de dados on-line:** a análise desta característica permite quantificar o nível de influência exercida pela utilização de entrada de dados no modo on-line no sistema.

Pontuação

- 0 - todas as transações são processadas em modo batch.
- 1 - de 1% a 7% das transações são entradas de dados on-line
- 2 - de 8% a 15% das transações são entradas de dados on-line
- 3 - de 16% a 23% das transações são entradas de dados on-line
- 4 - de 24% a 30% das transações são entradas de dados on-line
- 5 - mais de 30% das transações são entradas de dados on-line

**7. Interface com o usuário:** a análise desta característica permite quantificar o grau de influência relativo aos recursos implementados com vista a tornar o sistema amigável permitindo incrementos na eficiência e satisfação do usuário final, tais como:

- Auxílio à navegação (teclas de função, acesso direto e menus dinâmicos);
- Documentação e help on-line;
- Movimento automático do cursor;
- Movimento horizontal e vertical de tela;
- Impressão remota (via transações on-line);
- Teclas de função preestabelecidas;
- Processos batch submetidos a partir de transações on-line, utilização intensa de campos com vídeo reverso, intensificados, sublinhados, coloridos e outros indicadores;
- Impressão da documentação das transações on-line através de Hard copy;
- Utilização de mouse;
- Menus pop-up;
- O menor número possível de telas para executar as funções de negócio;
- Suporte bilíngüe (contar como 4 itens);
- Suporte multilingüe. (contar como 6 itens).

Pontuação

- 0 - Nenhum dos itens descritos;
- 1 - de um a três itens descritos;
- 2 - de quatro a cinco dos itens descritos;
- 3 - mais de cinco dos itens descritos, mas não há requerimentos específicos do usuário quanto a amigabilidade do sistema;
- 4 - mais de cinco dos itens descritos, e foram estabelecidos requerimentos quanto a amigabilidade fortes o suficiente para gerarem atividades específicas envolvendo fatores, tais como minimização da digitação, para mostrar inicialmente os valores utilizados com mais freqüência;
- 5 - mais de cinco dos itens descritos, e foram estabelecidos requerimentos quanto a amigabilidade fortes o suficiente para requerer ferramentas e processos especiais para demonstrar antecipadamente que os objetivos foram alcançados.

8. *Atualizações on-line*: Mede a influência no desenvolvimento do sistema face a utilização de recursos que visem a atualização dos Arquivos Lógicos Internos , no modo on-line.

Pontuação

- 0 - nenhuma.
- 1 - atualização on-line de um a três arquivos lógicos internos. O volume de atualização é baixo e a recuperação de dados é simples;
- 2 - atualização on-line de mais de três arquivos lógicos internos. O volume de atualização é baixo e a recuperação dos dados é simples;
- 3 - atualização on-line da maioria dos arquivos lógicos internos;
- 4 - em adição ao item anterior, é necessário proteção contra perdas de dados que foi projetada e programada no sistema;
- 5 - além do item anterior, altos volumes trazem considerações de custo no processo de recuperação. Processos para automatizar a recuperação foram incluídos minimizando a intervenção do operador.

9. *Processamento complexo*: a complexidade de processamento influencia no dimensionamento do sistema, e, portanto deve ser quantificado o seu grau de influência, com base nas seguintes categorias: processamento especial de auditoria e/ou processamento especial de segurança foram considerados na aplicação; Processamento lógico extensivo; Processamento matemático extensivo; Processamento gerando muitas exceções, resultando em transações incompletas que devem ser processadas novamente. Exemplo transações de auto-atendimento bancário interrompidas por problemas de comunicação ou com dados incompletos; Processamento complexo para manusear múltiplas possibilidades de entrada/saída. Exemplo: multimídia.

Pontuação

- 0 - nenhum dos itens descritos;
- 1 - apenas um dos itens descritos;
- 2 - dois dos itens descritos;
- 3 - três dos itens descritos;
- 4 - quatro dos itens descritos;
- 5 - todos os cinco itens descritos.

10 - *Reusabilidade*: a preocupação com o reaproveitamento de parte dos programas de uma aplicação em outras aplicações, implica em cuidados com padronização. O grau de influência no dimensionamento do sistema é quantificado observando-se os seguintes aspectos.

Pontuação

- 0 - nenhuma preocupação com reutilização de código;
- 1 - código reutilizado foi usado somente dentro da aplicação;
- 2 - menos de 10% da aplicação foi projetada prevendo utilização posterior do código por outra aplicação;
- 3 - 10% ou mais da aplicação foi projetada prevendo utilização posterior do código por outra aplicação;
- 4 - a aplicação foi especificamente projetada e/ou documentada para ter seu código reutilizado por outra aplicação e a aplicação é customizada pelo usuário em nível de código – fonte;

- 5 - a aplicação foi especificamente projetada e/ou documentada para Ter seu código facilmente reutilizado por outra aplicação e a aplicação é customizada para uso através de parâmetros que podem ser alterados pelo usuário.

*11. Facilidade de implantação:* a quantificação do grau de influência desta característica é medida, observando-se o plano de conversão e implantação e/ou ferramentas utilizadas durante a fase de testes do sistema.

Pontuação

- 0 - nenhuma consideração especial foi estabelecida pelo usuário e nenhum procedimento especial é requerido na implantação;
- 1 - nenhuma consideração especial foi estabelecida pelo usuário, mas procedimentos especiais são necessários na implantação;
- 2 - requerimentos de conversão e implantação foram estabelecidos pelo usuário e roteiro de conversão e implantação foram providos e testados. O impacto da conversão no projeto não é considerado importante;
- 3 - requerimentos de conversão e implantação foram estabelecidos pelo usuário e roteiro de conversão e implantação foram providos e testados. O impacto da conversão no projeto é considerado importante;
- 4 - além do item 2, conversão automática e ferramentas de implantação foram providas e testadas;
- 5 - além do item 3, conversão automática e ferramentas de implantação foram providas e testadas.

*12. Facilidade operacional :* a análise desta característica permite quantificar o nível de influência na aplicação, com relação à procedimentos operacionais automáticos que reduzem os procedimentos manuais, bem como, mecanismos de inicialização, salva e recuperação, verificados durante os testes do sistema.

Pontuação

- 0 - nenhuma consideração especial de operação, além do processo normal de salva foi estabelecido pelo usuário;
- 1-4 - verifique quais das seguintes afirmativas podem ser identificadas na aplicação. Selecione as que forem aplicadas. Cada item vale um ponto, exceto se definido explicitamente:
  - Foram desenvolvidos processos de inicialização, salva e recuperação, mas a intervenção do operador é necessária.
  - Foram estabelecidos processos de inicialização, salva e recuperação, e nenhuma intervenção do operador é necessária (conte como dois itens)
  - A aplicação minimiza a necessidade de montar fitas magnéticas.
  - A aplicação minimiza a necessidade de manuseio de papel.
- 5 - a aplicação foi desenhada para trabalhar sem operador, nenhuma intervenção do operador é necessária para operar o sistema além de executar e encerrar a aplicação. A aplicação possui rotinas automáticas para recuperação em caso de erro.

*13. Múltiplos locais:* esta característica consiste na observação da arquitetura do projeto, observando-se a necessidade de instalação do sistema em diversos lugares.

Pontuação

- 0 - os requerimentos do usuário não consideraram a necessidade de instalação em mais de um local;
- 1 - a necessidade de múltiplos locais foi considerada no projeto, e a aplicação foi desenhada para operar apenas em ambiente de software e hardware, idênticos;
- 2 - a necessidade de múltiplos locais foi considerada no projeto, e a aplicação está preparada para trabalhar apenas em ambientes similares de software e hardware;
- 3 - a necessidade de múltiplos locais foi considerada no projeto, e a aplicação está preparada para trabalhar sob diferentes ambientes de hardware e/ou software;
- 4 - plano de documentação e manutenção foram providos e testados para suportar a aplicação em múltiplos locais, além disso, os itens 1 ou 2 caracterizam a aplicação;
- 5 - plano de documentação e manutenção foram providos e testados para suportar a aplicação em múltiplos locais, além disso, o item 3 caracteriza a aplicação.

*14. Facilidade de mudanças (flexibilidade):* focaliza a preocupação com a influencia da manutenção no desenvolvimento do sistema. Esta influência deve ser quantificada baseando na observação de atributos, tais como: disponibilidade de facilidades como consultas e relatórios flexíveis para atender necessidades simples. (conte como 1 item); disponibilidade de facilidades como consultas e relatórios flexíveis para atender necessidades de complexidade média (conte como 2 itens); disponibilidade de facilidades como consultas e relatórios flexíveis para atender necessidades complexas (conte 3 itens); se os dados de controle são armazenados em tabelas que são mantidas pelo usuário através de processos on-line, mas mudanças têm efeitos somente no dia seguinte; se os dados de controle são armazenados em tabelas que são mantidas pelo usuário através de processos on-line, as mudanças têm efeito imediatamente (conte como 2 itens)

Pontuação

- 0 - nenhum dos itens descritos;
- 1 - um dos itens descritos;
- 2 - dois dos itens descritos;
- 3 - três dos itens descritos;
- 4 - quatro dos itens descritos;
- 5 - todos os cinco itens descritos.

Grau Descrição

- 0 - nenhuma influência
- 1 - influência mínima
- 2 - influência moderada
- 3 - influência média
- 4 - influência significante
- 5 - influência forte

| Características Gerais do Sistema   | Nível de Influência |
|---|---------------------|
| 1 Comunicação de Dados<br>2 Processamento de Dados Distribuído<br>3 Performance<br>4 Configuração do Equipamento<br>5 Volume de Transações<br>6 Entrada de Dados On-line<br>7 Interface com o Usuário<br>8 Atualização On-line<br>9 Processamento Complexo<br>10 Reusabilidade<br>11 Facilidade de Implantação<br>12 Facilidade Operacional<br>13 Múltiplos Locais<br>14 Facilidade de Mudanças |                     |
| <b>Nível de Influência Total (NI)</b>   |                     |
| <b>Fator de Ajuste = ( NI * 0,01 ) + 0,65 =</b>   |                     |

### 3.2.5 Etapa V - Contagem de Pontos de Função Ajustados

Trata-se do processo que realiza a correção das possíveis distorções acometidas durante o cálculo dos pontos de função não ajustados, aproximando as medidas à situação real.

- Cálculo de pontos de função ajustados de um projeto de desenvolvimento:
  - $PF_{Desenvolvimento} = [(PF_{Não-Ajustados} + \text{Pontos de função adicionados pelo processo de conversão}) \times (\text{fator de ajuste})]$
- Cálculo de pontos de função ajustados de um projeto de manutenção: a contagem dos pontos de função na manutenção de sistemas é levemente diferente da contagem de ponto de função no esforço de desenvolvimento de um novo sistema. Os cinco tipos de função: arquivo lógico interno e arquivo de interface externa e entrada externa, saída externa, consulta externa, são, também, considerados na manutenção. Além disso, o número de arquivos referenciados, de itens de dados, de registros lógicos e os relacionamentos com outros sistemas e arquivos são, também, contados e ajustados a partir da atribuição de pesos como nas matrizes de complexidades descritas anteriormente. Considerando que o sistema já esteja instalado, entretanto, para cada tipo de função, os desenvolvedores de sistemas devem determinar se uma nova função está sendo adicionada ao sistema e/ou se uma função existente está sendo modificada ou atualizada. A medida do tamanho do ponto de função de melhorias do sistema é diferente da medida do ponto de função de esforço de desenvolvimento de sistema. Os cinco tipos de função, listados acima, ainda são considerados. São contadas cada nova função que passa a existir e cada função que deixa de existir. O valor da melhoria, ou da correção (ex: o número de pontos de

função afetados) é determinado pela multiplicação do número de pontos de função não ajustado pelo ajuste de complexidade do processamento.

- Cálculo de pontos de função ajustados de uma aplicação (sistemas em produção): este tipo de contagem é utilizado tanto no caso de dimensionamento de sistemas já implantados, no início de implantação da técnica de análise de pontos de função, quanto na situação da ocorrência de um processo de manutenção. Após a realização da manutenção, a contagem dos pontos de função da aplicação deve ser atualizada. O dimensionamento de uma aplicação pode ser feito a partir do resultado da contagem de pontos de função de projetos de desenvolvimento, e/ou a partir do resultado da contagem de pontos de função de um projeto de manutenção. A fórmula para o dimensionamento de sistemas já implantados (em produção)
- $PF_{\text{aplicação}} = PF_{\text{Não ajustados}} \times \text{Fator de ajuste}$ .

Onde:

*PF da aplicação*: pontos de função da aplicação (tamanho funcional do sistema)  
*PF\_Não ajustados*: pontos de função não ajustados calculados para a aplicação.

**Fator de ajuste**: é o ajuste detectado face o cálculo dos graus de influência das 14 características que implicam no esforço e consumo de recursos para o desenvolvimento do sistema.

### **Fórmulas**

Fórmula para o dimensionamento de sistemas já implantados (em produção), a partir de resultado de contagem de pontos de função de projeto de desenvolvimento.

$$PF_{\text{aplicação}} = PF_{\text{do Proj. Desenvolvimento}} - (PF_{\text{da conversão}} \times \text{fator de ajuste})$$

Onde:

*PF da aplicação*: total de pontos de função da aplicação.

*PF do Proj. Desenvolvimento*: total de pontos de função do projeto de desenvolvimento.

*PF da conversão*: total de pontos de função não ajustados das funções de conversão de dados no projeto de desenvolvimento;

*Fator de ajuste*: é o valor do ajuste detectado face o cálculo dos graus de influência das 14 características que implicam no esforço e consumo de recursos para o desenvolvimento do sistema.

Fórmula para o dimensionamento de sistemas já implantados (em produção), a partir de resultado de contagem de pontos de função de projeto de manutenção:

$$PF_{\text{aplicação}} = (PF_{\text{não ajustado}} + PF_{\text{incluído}} + PF_{\text{alterado atual}} - PF_{\text{alterado anterior}} - PF_{\text{excluído}}) \times \text{Fator de ajuste atual}$$

Onde:

*PF da aplicação*: total de pontos de função da aplicação.

*PF não ajustado*: total de pontos de função não ajustados da aplicação antes do projeto de manutenção

*PF incluído*: total de pontos de função não ajustados das funções das funções que foram incluídas na aplicação.

*PF alterado atual:* total de pontos de função não ajustados das funções que foram modificadas na aplicação.(mostra as funções após a modificação)

*PF alterado anterior:* total de pontos de função não ajustados das funções que foram modificadas na aplicação (mostra as funções antes da modificação)

*PF excluído:* Total de pontos de função não ajustados das funções que foram removidas da aplicação.

*Fator de ajuste atual:* valor do ajuste detectado após o projeto de manutenção

## 4. Gestão da Tecnologia da Informação

Esta seção relata diferentes aspectos / visões gestão da tecnologia da informação usando análise por pontos de função.

### 4.1. Estimativa da Produtividade com Pontos de Função

Um dos maiores benefícios da utilização da técnica de ponto de função é a possibilidade de medir o tamanho dos projetos ou melhorias e suporte para a elaboração de estimativas mais coerentes e corretas. Os parágrafos a seguir descrevem como as três taxas de entrega de produtividade, modificação e suporte podem ser utilizados para previsão de esforços de trabalhos futuros. Pode-se fazer as primeiras estimativas do tamanho de todos os pontos de função do desenvolvimento de um projeto em particular, durante o planejamento do projeto, podendo, continuamente, refiná-lo até que o cálculo esteja completo no final do projeto detalhado. Para se estimar todo o esforço despendido em um projeto, na etapa de planejamento, o tamanho do ponto de função estimado para o desenvolvimento do projeto é dividido pela taxa de entrega atual.

#### 4.1.1. Produtividades no ambiente de desenvolvimento

A taxa de produtividade do desenvolvimento é chamada de taxa de entrega, e é calculada encontrando-se a razão entre *número de pontos de função entregues* e a *quantidade de unidades de tempo* utilizada para produzi-los. Fazendo um acompanhamento desta taxa na atividade de desenvolvimento de projetos, o sucesso dos sistemas pode ser medido e refletirem a aceitabilidade do programa de melhoria de produtividade adotado. É através da comparação da taxa de entrega observada em uma base regular e a taxa de modificações que poderemos identificar o sucesso do programa de melhoria adotado.

#### 4.1.2. Produtividade no ambiente de manutenção

No ambiente de manutenção, a produtividade pode ser expressa através da razão entre o esforço requerido (ex: entrada) e o tamanho do sistema em termos de pontos de função (ex: saída). A taxa de melhoria ou taxa de mudança: razão entre o tamanho, em pontos de função, da melhoria e o esforço de trabalho requerido para fazer a modificação. O esforço relativo ao trabalho executado pode ser medido tanto por meses trabalhados como

por horas ou dias trabalhados. Os desenvolvedores de sistemas devem usar uma medida consistente de forma que eles possam analisar as tendências das taxas de produtividade

#### **4.1.3. Produtividade no ambiente de suporte**

Ao estimar os requisitos de suporte, pode – se determinar o número de horas que serão necessárias para suportar o portfólio de sistemas e, baseado neste número calcular os recursos necessários e dimensionar a equipe de suporte. Deve-se refinar, continuamente, a estimativa conforme o tamanho do sistema, em termos de funcionalidades (ponto de função) e capacidades se aproximem daquelas que realmente irão ser implementadas. Para uma estimativa justa, é essencial que a alimentação da base de contagem e - o histórico das taxas de produtividade em outros projetos – estejam devidamente atualizadas e corretas.

#### **4.1.4. Escolha da taxa de produtividade apropriada**

A taxa de produtividade utilizada para calcular o esforço para novos trabalhos de desenvolvimento ou melhorias deveria satisfazer a entrega ou taxa de modificação que ocorre no nível do projeto, até que a taxa média para todo o departamento de sistemas, ou para a unidade apropriada dentro da área de sistemas, possa ser utilizada. Embora esta taxa não seja tão justa como taxa única por projeto, a estimativa global é, significativamente, mais correta do que outros métodos de estimativa. Para estimativas e comparações, é essencial que projetos e ambientes similares sejam utilizados. Por exemplo, a taxa de entrega para aplicação baseada em micro computador não deve ser usada para estimar o esforço de desenvolvimento para uma aplicação descentralizada baseada em grande porte.

#### **4.1.5. Reutilização e Produtividade**

A reutilização provoca um impacto significativo sobre a produtividade. A medição deste impacto requer uma técnica específica dentro da análise de pontos de função. Interessa-se em quão produtivo os projetos poderiam ser se possibilidades de reuso fossem utilizadas. Para medir o efeito de reusabilidade na produtividade, os desenvolvedores de sistemas podem comparar projetos similares analisar a utilização de reuso de código, da seguinte forma, por exemplo:

Projeto A: Taxa de entrega = 21,5 PF por mês de trabalho

Projeto B: Taxa de entrega = 29,0 PF por mês de trabalho

Baseado na análise, considerando que a única diferença entre estes projetos é a reutilização, o efeito na produtividade é estimado em cerca de 35%. Este percentual difere, dependendo do grau de reusabilidade alcançado (por exemplo, a indústria informa que em 90% de reuso, o crescimento do ganho em produtividade fica em torno de 600%). Os desenvolvedores e os especialistas em manutenção do repositório (exemplo: local de armazenamento central de dados, processo, código, diagramas e projeto) podem assumir este ganho de produtividade. Pode-se, também, contar os pontos de função da reutilização do repositório. Devido à natureza de algum componente de reuso, o repositório entregará

muitos pontos de função como um produto, os ganhos de produtividade são compreendidos pelos usuários destes recursos.

#### **4.1.6. Refinamento das estimativas**

A taxa de produtividade usada para calcular este refinamento é baseada no conhecimento, em informações de relatório, e é incluída, certamente, nos fatores de produtividade. Por exemplo, se todos os projetos usam técnicas estruturadas o ganho de produtividade destas técnicas já deverá estar incluído na taxa. Novas técnicas e ferramentas, que são parte do programa de melhoria de produtividade, no entanto, não estão incluídas nas taxas. O efeito global pode ser medido através da comparação entre as taxas para um projeto utilizando-se estas novas facilidades e as taxas de mercado obtidas a partir de estudos de ‘benchmarking’.

A estimativa para projetos futuros utilizando técnicas ou ferramentas pode ser refinada através do ajuste do ganho de produtividade esperado, assim como as novas facilidades se tornam maduras e alcançam total penetração na organização, assim, a taxa média deve ser ajustada de acordo com as metas da organização.

#### **4.1.7. Análise de Resultados**

Ganho de produtividade é a diferença entre as taxas correntes de entrega, modificação, ou taxa de suporte. Depois que o ganho é calculado, analisando-se o ambiente, pode-se compreender a razão para o ganho a partir da execução de análises de pontos de função.

Neste momento, torna-se importante a análise do porquê das mudanças nas taxas ou das causas que provocam diferenças entre as taxas de projetos similares. Esta análise fornece os seguintes benefícios:

- Permite medir a dimensão das iniciativas de melhoria de produtividade incluídas nos planos de desenvolvimento. (Ex. a criação de testes de regressão);
- Permite identificar oportunidades para melhorias de produtividade em outras áreas comparando as melhores práticas, ou seja, comparando as técnicas de melhorias utilizadas que obtiveram mais sucesso, quando utilizadas em outros projetos (ex: utilização de código comum);
- Permite subsidiar tomadas de decisões de suporte, tais como gerência de quais iniciativas como a reengenharia em sistemas existentes que exigem mais tempo que a média.

Este tipo de análise comparativa de projetos similares ajuda a responder questões como as seguintes:

- A introdução de um gerador de código torna o projeto mais produtivo? Caso afirmativo, em que grau?
- É mais produtivo comprar um pacote de software durante o desenvolvimento ou realizar a compra durante a manutenção?
- Qual é o efeito da reutilização de código nas taxas de produtividade globais?

O perfil de aplicação do sistema fornece a informação necessária para a execução deste tipo de análise. Este perfil, que é usado para descrever as características do processo de desenvolvimento, é critico para a análise do impacto comprehensível de ferramentas específicas, técnicas, e processos que formam o programa de melhoria de produtividade. O perfil deve incluir:

- Linguagem
- Hardware
- Técnicas e ferramentas de produtividade utilizadas
- O ambiente alvo
- O ambiente de desenvolvimento
- A idade do sistema que está sendo suportado
- A experiência do grupo de desenvolvimento

Comparando as taxas de produtividade dos projetos, o exame dos perfis dos projetos permite a análise das causas das diferenças de produtividades entre projetos. Fazendo o acompanhamento das taxas de modificação, pode-se medir o sucesso de tais iniciativas de melhoria de produtividade assim como a introdução de novas ferramentas.

#### **4.1.8. Atualização da Base de Informações**

O tamanho de um sistema em pontos de função (ou base de contagem) deve ser mantido atualizado, pois é a base para toda medição que envolva produtividade. Assim que novas capacidades forem adicionadas ao sistema, ou funções forem removidas, todo o tamanho do sistema em pontos de função deve ser modificado. Por exemplo, para um sistema entregue a usuários no final de determinado ano, por exemplo, em 1995, outras versões serão liberadas várias vezes durante o ano, ou por motivo de melhorias, correções, ou manutenções legais. Para cada versão, uma nova capacidade funcional é adicionada, funções são eliminadas e deverão ser feitas modificações no tamanho das funções existentes na base da contagem anterior às modificações. Os seguintes indicadores podem ser calculados conforme segue:

- A produtividade do desenvolvimento ou a taxa de entrega em determinada unidade de tempo (quantidade de pontos de função entregues por unidade de tempo);
- A melhoria de produtividade ou taxa de mudança face às alterações realizadas na aplicação para fins de melhoria a partir de determinado período (cálculo comparativo entre as taxas de produtividade observadas nos dois períodos de tempo);
- A produtividade da atividade de suporte ou a taxa de suporte em relação à manutenção do sistema (comparação entre as taxas de atendimento de suporte e de manutenção).

#### **4.2. Custo de Serviços de Informática**

Os serviços de processamento de dados podem ser segmentados em três tipos básicos: desenvolvimento, manutenção/ suporte e produção. A literatura menciona cinco principais modelos para a apuração de custos com serviços de processamento de dados, que

em geral são adotados pelos bureaux de serviços. Como sendo: critério tipo Test Unit, critério tipo Pontos de função, critérios tipo soma de volumes, critérios tipo volume de registros processados, e critérios tipo block time ou hora sala. Esses modelos são utilizados, em geral, não para a apuração do custo propriamente dito, mas para a determinação de preços. O resultado da aplicação de um modelo de agregação de custo deve apresentar as seguintes características:

- Estimativa dos custos de um serviço de informática deve ser independente de características muito específicas de cada sistema.
- Não se deve levar em consideração a estrutura interna dos prestadores de serviços, no caso de serviços terceirizados.
- O modelo deve ser adaptável a qualquer mudança que possa ocorrer no ambiente.
- O modelo ao ser implementado, deve contemplar um módulo para simulações de custos, com a finalidade de análise de cenários, para planejamento estratégico.

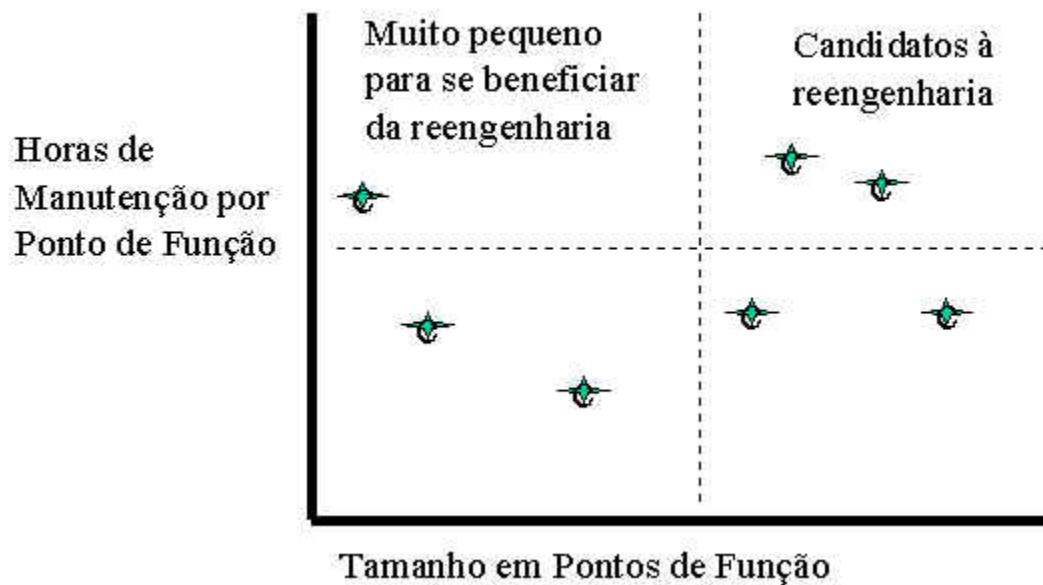
## 4.3. Quando e onde fazer reengenharia

Muitos projetos de reengenharia são conduzidos sem nenhuma análise de custo / benefício. A análise de custo / benefício procura a melhor razão entre os benefícios e os custos; isto significa, por exemplo, encontrar os aplicativos que mais se beneficiarão dos esforços de reengenharia. Para determinar quais aplicativos mais se beneficiarão da reengenharia, duas perguntas devem ser feitas:

1. Como identificamos aplicativos que devem ser objeto de reengenharia?
2. Como calculamos os benefícios potenciais do esforço de reengenharia?

### 4.3.1. Identificando aplicações para reengenharia

Uma vez que uma organização tenha completado uma linha de base (estabelecido o número de pontos de função) de todos os aplicativos em produção, as horas de manutenção por ponto de função podem ser calculadas para cada aplicativo. Horas de manutenção por ponto de função podem ser representadas em um gráfico semelhante ao mostrado abaixo:



Ao longo do eixo dos  $x$  está o tamanho (medido em pontos de função) e, sobre o eixo dos  $y$ , as horas de manutenção por ponto de função. Os pontos representam cada aplicativo medido.

No canto superior direito, estão aqueles aplicativos que são ao mesmo tempo grandes e caros de manter por ponto de função. Esses são os aplicativos que deveriam ser objeto de projetos de reengenharia. Isto é, esses aplicativos são os que fornecerão o maior retorno por dólar investido. Após a realização da reengenharia, a taxa de Horas de Manutenção / Pontos de Função deveria ser reavaliada, e o gráfico redesenhado. As Horas de Manutenção por ponto de função desses aplicativos que passaram pela reengenharia deverão cair, passando-os para o quadrante inferior direito do gráfico.

### 4.3.2. Estimando benefícios (calculando o retorno)

Poucas informações são necessárias para estimar os benefícios de um projeto de reengenharia. Tais informações incluem as horas de manutenção atuais por ponto de função, as horas de manutenção por ponto de função esperadas após o projeto de reengenharia e o período de retorno desejado. Por exemplo, seja um aplicativo com 1.000 pontos de função e 10 horas de manutenção por ponto de função. O valor desejado, para que o aplicativo passe para o quadrante inferior direito, é 8 horas de manutenção por ponto de função. Isto é, o projeto de reengenharia deveria reduzir o custo de manutenção em 2 horas por ponto de função. Se o período de retorno desejado for um ano, então o projeto de reengenharia não deveria levar mais de 2.000 horas. De forma mais simples, 2.000 horas é a diferença entre as horas de manutenção por ponto de função antes e depois (2 neste caso), multiplicadas pelo número de pontos de função (1.000 neste caso).

A mesma análise acima efetuada pode ser executada para comparar os custos das melhorias aos de desenvolvimento. É um truismo dizer que os dólares de melhoria por ponto de função deveriam ser menores que os dólares de desenvolvimento por ponto de função. Se

os dólares de desenvolvimento por ponto de função fossem inferiores aos de melhoria, isso significaria que seria mais barato desenvolver um novo aplicativo do que melhorar o existente.

## 4.4. Estimando Casos de Teste

Muitas tentativas têm sido feitas para estabelecer uma relação entre pontos de função e o esforço associado ao desenvolvimento de software. Grande parte da dificuldade no estabelecimento desta relação é devida ao fato de ser examinado apenas o relacionamento entre os pontos de função e o ciclo de vida inteiro do software. Esta seção aborda o relacionamento entre os pontos de função e a atividade de teste - em particular, a relação entre casos de teste e pontos de função.

A relação entre pontos de função e o número de casos de teste é muito forte. Capers Jones estima que o número de casos de teste é aproximadamente igual ao número de Pontos de Função elevado a 1,2 ( $PF^{1,2}$ ). Isto é, os casos de teste crescem a uma taxa mais rápida do que os pontos de função. Este é um fato intuitivo, porque conforme um aplicativo cresce, os inter-relacionamentos nele contidos tornam-se mais complexos.

Por exemplo, se um aplicativo em desenvolvimento tiver 1.000 pontos de função, deverão existir aproximadamente 4.000 casos de teste. Obviamente, a equipe de desenvolvimento deveria começar a criar os casos de teste tão rapidamente quanto possível. Se a equipe esperar até que a codificação tenha terminado, provavelmente serão criados muito menos do que 4.000 casos de teste.

### 4.4.1. Entendendo o Potencial de Defeitos

O número de defeitos está relacionado ao número de resultados possíveis. Este número está relacionado ao número de casos de teste. Há um grande benefício em estimar o número de casos de teste. Entender o número de casos de teste leva à análise lógica de comparar os casos reais com os esperados. Se os casos reais forem em quantidade inferior à esperada, a cobertura de testes será inadequada. A cobertura de testes é uma indicação direta dos defeitos potenciais e custos futuros com manutenção. Quão maior for a distância entre a quantidade esperada e a real de casos de teste, maior o potencial de defeitos não detectados durante a respectiva fase de testes.

O que se segue é um princípio básico da qualidade de software: o único caminho para elevar a qualidade do software em produção é se e somente se todo o software migrado para a produção for de qualidade superior ao software atualmente em produção. É importante entender tanto os defeitos potenciais do software de produção, quanto os defeitos potenciais do novo software. Pode ser impossível estimar o número de defeitos embutidos no software atualmente em produção, mas é possível estimar o potencial de defeitos para cada nova versão do software. A diferença nas quantidades real e estimada dos casos de teste é um bom indicador do potencial de defeitos.

## 4.5. Entendendo faixas de Produtividade Amplas

Taxas de produtividade inconsistentes entre diferentes projetos podem ser uma indicação de que não está sendo seguido um processo padrão. A produtividade é definida como a razão entradas / saídas. No caso de software, a produtividade é definida como a quantidade de esforço requerida para entregar um certo conjunto de funcionalidades (medidas em pontos de função).

Muitas organizações sofrem de taxas de produtividade variando em mais ou menos 100 por cento. Isto é verdade mesmo quando essas organizações contam os pontos de função corretamente e registram consistentemente os tempos gastos. As organizações reclamam freqüentemente que a análise de produtividade não oferece retorno. Na maioria das organizações, o software é criado de muitas formas diferentes, com a utilização de diversos processos (e, em muitos casos, nenhum processo). Se o software for desenvolvido de maneira inconsistente, as taxas de produtividade também serão inconsistentes. O mesmo serve para qualquer processo produtivo. Grandes variações de produtividade entre diferentes projetos indicam que não está sendo seguido um processo padrão. À medida que as equipes de desenvolvimento se alinham com um processo padronizado, as faixas de produtividade deverão se estabilizar e tornarem-se mais consistentes.

## 4.6. Entendendo o Aumento do Escopo

A Análise de Pontos de Função provê um mecanismo para acompanhar e monitorar o aumento do escopo ("scope creep") de um projeto. As contagens de Pontos de Função ao fim do levantamento de requisitos, análise, projeto e implementação podem ser comparadas. A contagem de pontos de função ao final do levantamento de requisitos e / ou do projeto pode ser comparada com os pontos de função realmente entregues. Se o número de pontos de função tiver aumentado, pode-se assumir que o projeto tornou-se mais bem definido, ou realmente cresceu em tamanho. A quantidade acrescida é um indicador da qualidade do levantamento dos requisitos e / ou de sua comunicação à equipe do projeto. O aumento do tamanho dos projetos deve ser acompanhado em todos os casos. Se o grau de crescimento dos projetos decrescer ao longo do tempo, é natural assumir que houve melhoria na comunicação com o usuário.

## 4.7. Estimando o Custo, Cronograma e Esforço do Projeto

O sucesso nas estimativas com a utilização de pontos de função baseia-se em diversas técnicas de estimativa: De Cima Para Baixo ("Top-Down"), Analogia e Opinião de Especialista ("Expert Advice"). A estimativa De Cima Para Baixo é uma técnica que estima o cronograma inteiro, assim como o custo e o esforço, utilizando parâmetros de uso genérico. Os parâmetros de uso genérico e as comparações efetuadas são baseados em dados históricos de benchmark, com a utilização de técnicas de Analogia. A Opinião de Especialistas pode ser obtida através de especialistas com experiência em projetos semelhantes ou na utilização de pontos de função.

A comparação de projetos semelhantes é crítica para o sucesso nas estimativas. Ao avaliar projetos semelhantes, devem ser efetuadas as seguintes considerações:

- Tipo de Plataforma de Hardware - Mainframe, Client Server, PC Isolado
- Tipo de Linguagem - COBOL, C, C++
- Tipo de Projeto - Software Básico, Middleware, Software Aplicativo
- Tipo de Sistema Operacional - MVS, Windows, Unix

Uma vez que tenham sido identificados os projetos semelhantes, os seguintes dados devem ser coletados:

- Taxa de Entrega Histórica (horas por ponto de função) dos projetos semelhantes
- Cronogramas Históricos (duração do cronograma por ponto de função) dos projetos semelhantes
- Custo Histórico (dólares por ponto de função) dos projetos semelhantes

Uma vez que o tamanho do projeto tenha sido determinado em pontos de função, as estimativas de horas, custo e cronograma podem ser calculados. Os cálculos devem ser efetuados a partir de dados de projetos semelhantes, conforme acima explanado. Por exemplo, se for determinado que o tamanho do projeto atual é 500 pontos de função e que o custo histórico para um projeto semelhante foi \$10 por ponto de função, então o custo total esperado para o projeto seria  $\$10 (\$/\text{Ponto de Função}) \times 500 \text{ PF} = \$ 5.000$  dólares. Cálculos semelhantes poderiam ser efetuados para o cronograma, a duração e as horas.

## 4.8. Calculando o Custo Real do Software

A maioria das organizações calcula o custo do software a menor, por larga margem. O verdadeiro custo do software é a soma de todos os custos durante a vida do projeto, incluindo-se aí todos os custos esperados de melhoria e manutenção. De fato, o cálculo real seria o valor presente de todos os desenvolvimentos, melhorias e manutenções no decorrer da vida do projeto. Este tipo de análise demonstra a recompensa de se investir na fase inicial da análise e projeto. Quanto mais se investir nas fases iniciais, mais será economizado na redução dos custos de manutenção e melhoria. É importante ser capaz de calcular o custo unitário, a fim de avaliar o investimento inicial e compará-lo aos gastos futuros. O custo unitário pode ser em horas / PF ou \$ / PF. Os aumentos no investimento inicial deveriam reduzir proporcionalmente os custos unitários das futuras melhorias e manutenções.

## 4.9. Entendendo os Custos de Manutenção

Muitos orçamentos de manutenção são estabelecidos com base na performance de anos anteriores. Muitas organizações tentam reduzir os custos de manutenção e não planejam aumentar seus custos de manutenção de ano a ano, para nenhum aplicativo em particular. Se a quantidade de nova funcionalidade for determinada e adicionada ao produto básico, o custo unitário de manutenção pode de fato cair, enquanto o gasto real permanece constante ou aumenta. Se os custos de manutenção estiverem crescendo a uma taxa inferior à taxa de crescimento dos pontos de função, então os custos de manutenção estarão de fato

caindo. Por exemplo, se uma organização gasta \$ 100.000 para manter 10.000 pontos de função (\$ 10 por ponto de função), e o número de pontos de função cresce 10 por cento e passa a ser 11.000, permanecendo constantes os dólares gastos em manutenção, teremos o custo unitário de manutenção caindo para \$ 9 por ponto de função. Muitos executivos de Sistemas de Informações deixam de compreender este conceito.

## 4.10. Ajudando em Negociações Contratuais

Os gerentes de contratos podem utilizar seu conhecimento de pontos de função para criar e gerenciar projetos com base no preço por ponto de função, bem como para comparar os preços dos fornecedores. Tais indivíduos estabelecem a utilização eficaz de terceiros no desenvolvimento, validam propostas com base no tamanho em pontos de função e podem avaliar o impacto de projetos cancelados.

Os pontos de função podem ser utilizados para ajudar a especificar os principais produtos a serem recebidos de um fornecedor, para assegurar a entrega dos níveis adequados de funcionalidade e para desenvolver medidas objetivas de eficácia em relação ao custo e qualidade. São mais eficazmente utilizados em contratos de preço fixo, como forma de especificar exatamente o que deve ser entregue. De uma perspectiva interna, o gerenciamento bem sucedido de contratos de preço fixo depende completamente da existência de representações acuradas do esforço. A estimativa deste esforço (ao longo de todo o ciclo de vida) pode acontecer somente quando há a aplicação de uma métrica normalizada, como é o caso na utilização de pontos de função.

Resumindo, a análise de pontos de função provê o melhor método objetivo para a avaliação do tamanho de um projeto de software e para o gerenciamento desse tamanho durante o desenvolvimento. É, por dois motivos, o melhor método para gerenciar riscos. Primeiro, o cliente (usuário / especificador) pode aceitar mais facilmente o risco para um dado tamanho de projeto de software (em pontos de função). Segundo, o desenvolvedor pode aceitar mais facilmente o risco para o custo de produção (o custo por ponto de função). A adesão a uma forma consistente de contagem de pontos de função otimiza este relacionamento e facilita o desenvolvimento dentro do prazo e orçamento estabelecidos.

O estabelecimento de preço para "software externo" (i.e., projeto para utilização fora da organização) pode ser mais fortemente ligado ao esforço de produção quando são exigidas métricas funcionais. Se um desenvolvedor de software sabe exatamente qual o seu custo interno de desenvolvimento de um ponto de função, ele pode facilmente incorporar a esse custo os algoritmos utilizados para fixar os preços externos. Sem um claro entendimento do tempo e esforço gastos por ponto de função, o estabelecimento de preços para pacotes de software vai continuar a ser difícil.

## 4.11. Desenvolvendo um conjunto Padrão de Métricas

É claro, os pontos de função precisam ser coletados em associação com outras medidas. De fato, somente os pontos de função, por si só, oferecem pouco ou nenhum benefício. Muitas métricas precisam ser reportadas ao nível organizacional. Por exemplo,

tanto métricas de produtividade / custo quanto de qualidade precisam ser reportadas ao nível da organização. As métricas de Custo / Produtividade são utilizadas para demonstrar a taxa e o custo da funcionalidade entregue. As métricas de Qualidade são utilizadas para demonstrar os níveis existentes de qualidade e para acompanhar os esforços de melhoria no processo de desenvolvimento de software. Tais métricas devem ser acompanhadas e mapeadas ao longo do tempo.

### Métricas de Custo / Produtividade

1. *Custo por Ponto de Função*: mede o custo médio para entregar ou manter um ponto de função. Este dado pode ser utilizado para desenvolver um banco de dados histórico com a finalidade de estimar projetos futuros.
2. *Pontos de Função por Mês ou Ano do Calendário*: mede a quantidade média de tempo para entregar um ponto de função à produção com um dado quantitativo de pessoal.
3. *Pontos de Função por Pessoa Mês*: mede o número médio de pontos de função entregue pela aplicação de um mês de esforço.

### Métricas de Qualidade

1. *Defeitos por Pontos de Função Instalados*: correlaciona a qualidade do software ao tamanho do aplicativo.
2. *Horas de Manutenção por Pontos de Função Instalados*: correlaciona o esforço de suporte ao tamanho do aplicativo, para o software atualmente instalado, incluindo o legado. Aplicativos com taxas elevadas podem ser alvo de reengenharia ou substituição.
3. *Defeitos por Novos Pontos de Função Instalados*: avalia a qualidade do novo software, para garantir que as melhorias no processo de entrega estão tendo efeito.

As métricas devem ser indicadores e não medidas exatas de performance. Elas devem prover granularidade suficiente para mostrar as tendências gerais, identificar as áreas com problemas e demonstrar progresso. Se a tentativa de tornar as métricas perfeitas fizer com que elas sejam reportadas dois a três meses depois de medidas, muito tempo estará sendo gasto com precisão e pouco com ação. Deve-se evitar tornar as métricas precisas demais.

## 5. Considerações Finais

O sucesso de um programa de métricas depende de uma série de fatores. Estes variam desde a conscientização dos profissionais envolvidos de que medir horas de trabalho em termos de produtividade até a existência de recursos técnicos e humanos necessários para a manutenção e monitoramento do programa. É importante ressaltar que um programa de medição traz benefícios para o empregado à medida que possibilita dimensionar melhor a carga de trabalho de forma a garantir a qualidade. Com esta informação o desenvolvedor poderá requerer mais recursos bem como dimensionar o valor pecuniário de seu trabalho além de estimar com mais precisão o tempo necessário para o desenvolvimento das suas tarefas. Para o empresário, este terá uma base de informações imprescindível para estimar

seus custos e apreciar seus produtos, bem como estabelecer cronogramas com promessas de entrega de produtos em prazos mais factíveis.

Há muitos usos para pontos de função além de estimar o cronograma, esforço e custo. Muitos gerentes de projeto não acreditam que os pontos de função possuem qualquer utilidade. De certa forma, eles estão certos. Muitas organizações estão utilizando pontos de função e métricas de software para reportar tendências a nível organizacional. Muitas equipes de projeto enviam dados a um grupo central de métricas e nunca mais tornam a ver seus dados. Isto é análogo a enviar os dados a um buraco negro. Se os gerentes de projetos começarem a entender como os pontos de função podem ser utilizados para estimar casos de teste, calcular custos de manutenção e assim por diante, eles provavelmente investirão na contagem de pontos de função.

Desta forma, para se fazer uma boa gestão da tecnologia da informação, é necessário ter alguns cuidados na hora de se estabelecer um programa de métricas. Deve-se buscar um programa de medição que:

- Tenha um “patrocinador” executivo que forneça suporte top-down;
- Forneça medidas que sejam inteligíveis e úteis para todos os níveis de gerenciamento dentro da organização;
- Seja compreendido por todos no departamento de sistemas como seu propósito e tipo de medição que ele executa;
- Realizar um estudo de benchmarking na área de sistemas, para identificar volume de recursos e métricas existentes;
- Desenvolver um repositório de métricas que devem ser alcançadas para prover as medições definidas nos passos anteriores e identificar as fontes das informações;
- Preparar uma apresentação dos propósitos e benefícios do programa de métricas e difundir o conceito para os gerentes da área de sistemas;
- Garantir, através de treinamentos na metodologia de pontos de função, que um especialista nesta matéria esteja disponível para consultoria interna;
- Formar especialistas em análise de pontos de função que se tornem disseminadores da técnica, dentro da empresa;
- Determinar qual a estratégia a ser adotada para contabilizar os pontos de função dos sistemas , criando um inventário de sistemas;
- Garantir que um treinamento adequado fique disponível para a equipe de desenvolvimento, tanto através de fontes externas como treinamentos com pessoal interno;
- Começar a contabilização, e mais importante, publicar os resultados regularmente de forma que eles fiquem disponíveis para toda a equipe de sistemas. Isto estabelece a credibilidade do programa de métricas.

## 6. Referências Bibliográficas

1. IFPUG “ Function Point Practices Manual Release 4.0. Jan/94
2. Brown, Darlene P. Et alii. Using function points to measure productivity from the user's point of view. Information management. Warren, 1990
3. Pontos de Função e Medidas - O que é um Ponto de Função?, por Carol Dekkers, traduzido por Mauricio Aguiar (15.06.99)
4. Orçando com APF (Word), por NESMA, traduzido por Altair L. Ribeiro (29.12.1999)
5. Como Funciona a APF (Word), por NESMA, traduzido por Altair L. Ribeiro (29.12.1999)
6. Pontos de Função Aplicados a Novas Tecnologias, por David Longstreet, traduzido por Mauricio Aguiar (30.12.99)
7. Dicas Para a Contagem de Web Sites, por IFPUG, traduzido por Mauricio Aguiar (11.01.00)
8. Dicas Para a Contagem de Aplicativos Cliente/Servidor (Word), por IFPUG, traduzido por Mauricio Aguiar (20.02.00)