Medição de Tamanho para Sistemas de *Data Mart*

Angélica Toffano S. Calazans
Caixa Econômica Federal
angelica.calazans@caixa.gov.br

Káthia Marçal de Oliveira, Rildo Ribeiro dos Santos
Universidade Católica de Brasília
{kathia, rildo}@ucb.br

Resumo

Para melhor controlar tempo, custo e recursos em projetos de software as organizações necessitam uma forma adequada de estimar o tamanho dos projetos antes mesmo de eles realmente iniciarem. Nesse contexto, diferentes abordagens foram propostas para estimar o tamanho de um sistema. A maioria dessas abordagens tem como objetivo medir o tamanho de qualquer tipo de sistema, não importando a tecnologia. Contudo, alguns autores argumentam que cada tecnologia tem particularidades específicas e que essas devem ser levadas em consideração. Sistemas de *Data Mart*, por exemplo, têm particularidades diferentes dos sistemas tradicionais. É importante, portanto, ter uma abordagem de estimativa que considere estas particularidades quando na medição de *Data Mart*. Este trabalho apresenta uma adaptação da abordagem de Análise por Pontos de Função para estimativa de tamanho de *Data Mart*. São apresentados, também, resultados da aplicação desta proposta em projetos reais da indústria.

Palavras Chaves: Métricas funcionais, *Data Mart*, estimativa de tamanho, Análise por pontos de Função.

Abstract

To better control the time, cost and resources assigned to software projects, organizations need a proper estimate of their size even before the projects actually start. Accordingly, several approaches were proposed to estimate the size of a software project. Most of these approaches aim at measuring the size of any type of software system, whatever the technology. However some authors argue that each technology has specific particularities, which must be taken into account. *Data Mart* systems, for instance, have particularities in their development that are different from the traditional software systems. It is important, therefore, to have an estimation approach that considers those particularities while measuring the *Data Mart* size. This work presents an adaptation of the Function Points Analysis approach for *Data Mart* size estimation. It also presents and discusses results on *Data Mart* projects developed in the industry.

Key words: Functional measurement, *Data Mart*, size estimation, Function Point Analysis.

1 Introdução

A medição na engenharia de software é um dos fatores importantes para a geração de um produto com qualidade. Segundo Fenton e Pfleeger [5], mensura-se para compreender, controlar e aperfeiçoar o processo de produção de software. Medidas permitem um melhor controle dos projetos, proporcionam o aperfeiçoamento contínuo do processo e do produto de software e o estabelecimento de bases de comparação para o desenvolvimento futuro.

A mensuração do tamanho do software na gestão de projetos também está vinculada à necessidade de obter e/ou melhorar estimativas de prazo, custo e recursos gerando expectativas mais realistas para o usuário [12]. Dessa forma, é essencial que a mensuração de tamanho seja o mais aproximada possível da realidade, pois ela é um fator de impacto nas demais variáveis [3].

Várias abordagens[1] têm sido propostas e aperfeiçoadas, desde a década de 1960, com o objetivo de definir o tamanho de um software e a maior parte dessas propostas busca medir o tamanho de qualquer tipo de software, independente da tecnologia. Sistemas de *Data Warehouse/Data Mart*, no entanto, são um tipo especial de software, com características particulares como, por exemplo, o fato dos usuários utilizarem o software somente para consultas e não atualização dos dados; o desenvolvimento baseado em dados existentes em outros sistemas sem a geração de novos dados; a necessidade de um projeto de Extração, Transformação e Carga (ETL) e a utilização de uma modelagem multidimensional dos dados.

A estimativa em projetos de software envolve, na maioria das vezes, a previsão de quatro variáveis: tamanho (dimensão do software produzido ou a produzir), esforço (trabalho necessário para o desenvolvimento do software), prazo (tempo necessário para o desenvolvimento do software) e qualidade (envolvendo fatores como manutenabilidade, confiabilidade e outros)[1].

A primeira variável a ser determinada é sempre o tamanho do software, produzido ou a produzir, pois é a partir da definição da dimensão que é possível definir o esforço e o prazo necessários para o desenvolvimento do software.

Torna-se necessário adequar as abordagens de medição de tamanho definidas para sistemas tradicionais para que considerem as características específicas de *Data Warehouse/Data Mart* e com isso gerem estimativas mais reais.

2 Objetivos, metodologia e classificação da pesquisa

Esta dissertação tem por objetivo a definição de uma proposta de mensuração de tamanho para Projetos de *Data Mart*.

São objetivos específicos: (i) Estudar as características principais de sistemas de *Data Mart*/*Data Warehouse*, identificando aspectos diferenciados em relação aos sistemas transacionais; (ii) Estudar algumas abordagens de métricas de tamanho existentes, analisar sua aplicabilidade a esse contexto e identificar a melhor alternativa para adequação à tecnologia de *Data Mart*; (iii) Propor a adequação de uma das abordagens de métricas de tamanho para projetos de *Data Mart*; (iv) Utilizar e avaliar a nova adequação em projetos de *Data Mart*; e, (v) Comparar os resultados da aplicação dessa proposta de adequação com os resultados da abordagem original escolhida como base para adequação.

O tipo de pesquisa utilizada classifica-se como pesquisa aplicada, uma vez que busca a resolução de problema concreto, que é a mensuração de projetos de *Data Mart*.

Com relação aos meios de investigação, foram utilizados: a pesquisa bibliográfica baseada em material publicado em livros, revistas, jornais, redes eletrônicas, anais; estudo de caso circunscrito aos projetos possíveis de serem mensurados em três instituições investigadas; investigação documental, apoiada em pesquisa documental dos sistemas a serem pontuados; e, pesquisa de campo por meio da aplicação de entrevistas estruturadas com os responsáveis pelos projetos de *Data Mart* nas três instituições pesquisadas, a fim de obter informações sobre os projetos.

Foram analisadas, por meio de pesquisa bibliográfica, as características diferenciadas de um processo de desenvolvimento de um sistema de *Data Mart*, as deficiências que as métricas possuem quando aplicadas a sistemas de *Data Mart* e as críticas existentes a estas abordagens.

[1] Lines of code – LOC (1950/1960), a abordagem Halstead (1972), Function Point Analysis (FPA) ou Análise por Pontos de Função (APF) (1979), o MKII Function Point Analysis (1991) e o Full Function Points/COSMIC Full Functions Points (1997)

Foi construída uma proposta de adequação de uma dessas métricas ao contexto de *Data Mart*, com base na pesquisa bibliográfica e aplicada nos projetos de *Data Mart* das três instituições investigadas. Foram comparados os resultados dessa proposta e da métrica aplicada escolhida.

2.1 Coleta e análise de dados

Foram coletados dados em três instituições governamentais federais instaladas em Brasília, Distrito Federal. Duas delas são bancárias e uma delas é uma empresa de desenvolvimento de Sistemas.

Foram realizadas entrevistas estruturadas com os responsáveis pelos sistemas de *Data Mart*, visando obter informações sobre os sistemas a serem medidos, sobre o tempo e sobre a quantidade de recursos humanos utilizados para os projetos de *Data Mart* já implantados.

Nas seções seguintes apresentamos, inicialmente, uma breve descrição sobre *Data Warehouse/Data Mart*, no que se refere a sua definição e processo de construção (seção 3); e sobre medição e análise por pontos de função, no que se refere a sua forma de medição (seção 4). Na seção 5, será apresentada a adequação da APF para *Data Mart*. Na seção 6 mostramos os resultados da aplicação desta adequação em alguns projetos. Finalmente, na seção 7, apresentamos a conclusão e contribuições deste trabalho.

3 Características de *Data Warehouse/Data Mart*

3.1 Definição

Segundo Inmon [7], um "*Data Warehouse* é uma coleção de dados orientada por assuntos, integrada, variante no tempo, e não volátil, que tem por objetivo dar suporte aos processos de tomada de decisão".

A tecnologia utilizada tanto no *Data Warehouse* como no *Data Mart* é a mesma, sendo que as variações que ocorrem são mínimas, mais voltadas para o volume de dados, abrangência da arquitetura e o foco [2]. Os *Data Mart* são voltados somente para uma determinada área referenciando um escopo menor, a uma unidade de negócio, já o *Data Warehouse* é voltado para os assuntos de toda empresa. Este trabalho utilizará a nomenclatura *Data Warehouse/Data Mart* indistintamente para designar esses sistemas que utilizam depósitos dimensionais de dados, atentando para o fato que *Data Warehouse* possui uma arquitetura mais abrangente de foco mais amplo e que *Data Mart* representa um escopo menor tanto de arquitetura como de foco, mas plenamente e coerentemente acoplado no *Framework* maior do *Data Warehouse* corporativo.

Existem quatro componentes separados e distintos no ambiente de *Data Warehouse* (Figura 1) [9]: sistemas operacionais de origem (sistemas que capturam as transações da empresa), *data stanging área (*área de armazenamento de dados e de conjunto de processos que preparam os dados de origem para serem utilizados), área de apresentação de dados (local onde os dados ficam armazenados e disponíveis ao usuário final) e ferramentas de acesso a dados (ferramentas OLAP e de mineração de dados que permitem aos usuários utilizar os dados de uma maneira rápida e fácil para executar análises mensuráveis).

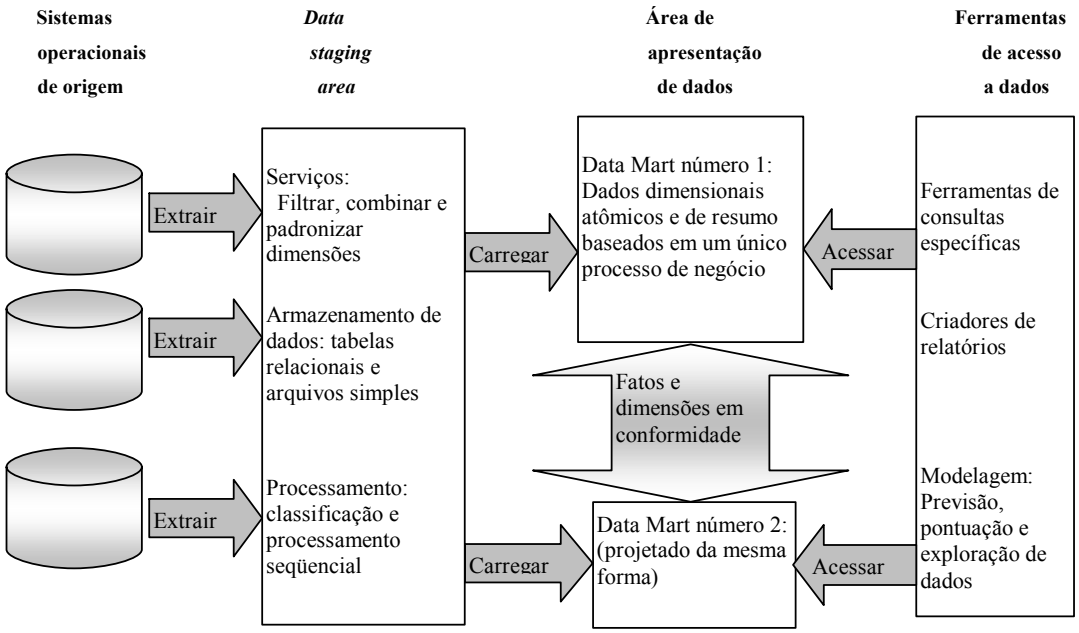

Figura 1 - Elementos básicos do *Data Warehouse*

Fonte: KIMBALL e ROSS [9]

3.2 Processo de Construção

As fases básicas para se criar e atualizar um *Data Warehouse* são [9]: (i) extração, (ii) transformação e (iii) carga dos dados (*ETL – Extraction, Transformation, Load*).

O processo de extração (i) envolve a leitura e compreensão dos dados de origem e cópia destes dados na *staging area* para serem manipulados posteriormente. Normalmente, cada sistema de origem é uma aplicação independente e que possui pouco compartilhamento de dados comuns como produto, cliente e geografia com outros sistemas transacionais da empresa. A integração destes dados é uma das tarefas que geram mais esforço no projeto de um *Data Warehouse*. A quantidade de sistemas transacionais envolvidos, suas estruturas de dados[2] e o nível de documentação (o *Data Mart* necessita apresentar todos os conceitos e as origens dos dados) interferem diretamente na dimensão do sistema de *Data Mart*. O processo de extração pode ser realizado de forma automatizada através de ferramenta de ETL. A existência ou não desta ferramenta também impacta o tamanho do produto seja na geração de um maior número de funcionalidades para a extração destes dados ou na exigência de conhecimento profundo, por parte dos desenvolvedores do *Data Mart*, das regras de negócio dos sistemas transacionais e definição de formas de extração.

Na fase de transformação (ii) modifica-se a estrutura do armazenamento de dados. Nesta fase ocorrem ”transformações em potencial, como filtragem dos dados (correções de erros de digitação, solução de conflitos de domínio, tratamento de elementos ausentes), combinação de dados de várias origens, cancelamento de dados duplicados e atribuições de chaves” [9]. Nesta fase também podem ser aplicados níveis de desnormalização e renormalização[3], combinação[4], auditoria no conteúdo de dados[5] e agregações para melhorar o

[2] Definição da estrutura em que estão os dados de origem: VSAM, Banco de Dados Relacional (DB2, Sybase, Oracle, etc), Banco de dados hierárquico (IDMS), etc.

[3] Reunificação das hierarquias de dados, separadas pela normalização dentro de uma tabela desnormalizada.

desempenho das consultas para o usuário (considerando a previsão de volume de dados). Toda esta transformação ocorre na *staging area* e também impacta no tamanho de um projeto de *Data Mart*.

A fase de carga (iii) é um processo interativo, pois o *Data Warehouse* tem que ser povoado continuadamente e refletir de forma incremental as mudanças dos sistemas operacionais. Manutenções que possam ocorrer nas fontes de dados interferem diretamente na dimensão do projeto, pois além das transformações precisarem ser re-definidas e aplicadas, a carga também é alterada a cada modificação das fontes de dados das origens. A carga é realizada no banco de dados do DW, na área de apresentação de dados.

Neste banco de dados (que pode ser desenvolvido em uma tecnologia de banco de dados multidimensional ou relacional) os dados são armazenados em cubos. Um modelo multidimensional possui três elementos básicos: fatos, que são definidos como a coleção de itens de dados, composta de dados de medidas e de contexto, representando um item de negócio, uma transação de negócio ou um evento de negócio; dimensões que são os elementos que participam de um fato e determinam o contexto de um assunto de negócios e medidas que são atributos numéricos que representam um fato.

4 Medição de Software

Medição é o processo através do qual números ou símbolos são atribuídos a entidades do mundo real de forma a tornar possível caracterizar cada entidade através de regras claramente definidas [5], ou seja, é o processo de obtenção de uma medida para uma entidade do mundo real. Uma medida fornece uma indicação de quantidade, dimensão, capacidade ou tamanho de algum produto de software ou de um processo.

Para se chegar a uma medida de software, existem técnicas de estimativas que avaliam as variáveis de tamanho, esforço e prazo. Estas técnicas podem ser classificadas basicamente em Analógicas (baseada na experiência de quem faz estimativas), Modelos Algoritmos (modelos matemáticos, por exemplo, LOC que pontua o número de instruções fontes) e Análise de Funcionalidade (baseada nas funcionalidades do software, por ex. APF) [12].

Algumas das principais abordagens utilizadas para análise de funcionalidade são a Análise por Pontos de Função, definida desde 1979 e que vem continuamente sendo utilizada e melhorada desde então. A seguir serão descritas as principais características da APF por ser essa a métrica adequada nesse trabalho.

4.1 Análise por Pontos de Função

A Análise por Pontos de Função (APF) mede o tamanho do *software* pela quantificação de suas funcionalidades, baseadas no projeto lógico ou a partir do modelo de dados segundo a visão e os requisitos do usuário final [4]. Suas principais características são: medir independente da tecnologia, ser aplicável desde o início do sistema, apoiar a análise de produtividade e qualidade e estimar o tamanho do *software* com uma unidade de medida padrão.

O Quadro 1 apresenta os passos que devem ser observados para mensuração de tamanho do *software* utilizando esta abordagem [4].

---

[4] Realizada quando fontes de dados possuem os mesmos valores de chaves representando registros iguais ou complementares ou atributos de chaves não iguais, incluindo equivalência textual de códigos de sistemas de legados distintos.

[5] O processo de transformação deve realizar constantes verificações de somas, contagens de linhas e testes.

Quadro 1 – Passos para aplicação da APF

Passos	Observações
i) Estabelecer o objeto da contagem	Se projetos de desenvolvimento, projetos de manutenção ou contagem de uma aplicação
ii)Determinar a fronteira de medição	A fronteira de medição deve ser sempre determinada sob o ponto de vista do usuário
iii) Contar as funções de dados e suas complexidades[6]	Arquivos Lógicos Internos (ALIs - que são grupos lógicos de dados mantidos dentro da fronteira da aplicação) e Arquivos de Interface Externa (AIEs – arquivos somente referenciados pela aplicação)
iv) Contar as funções transacionais e suas complexidades	Entradas Externas (EEs), Saídas Externas (SEs) e Consultas Externas (CEs)
v) Determinar o Fator de Ajuste	Conjunto de 14 características que influenciarão a complexidade do *software*. São elas: comunicação de dados, processamento distribuído, performance, utilização de equipamento, volume de transações, entrada de dados on-line, eficiência do usuário final, atualização on-line, processamento complexo, reutilização de código, facilidade de implantação, facilidade operacional, múltiplos locais, facilidade de mudanças)
vi) Determinar o tamanho do projeto	Considera as funções de dados e transacionais e suas complexidades, fator de ajuste e tipo de projeto

O resultado da contagem de funções de dados e transacionais e suas complexidades é uma medida chamada de contagem não ajustada (NoPFnão ajustado). O ajuste na mensuração é efetuado através do Fator de Ajuste determinado.

A determinação do Fator de Ajuste considera a avaliação de cada característica numa escala de 0 (nenhuma influência) a 5 (grande influência). A determinação deste fator de ajuste (FA) é baseada na equação: FA = 0,65 + (0,01 x Soma das características gerais do sistema).

Para determinar o tamanho do projeto consideram-se fórmulas específicas como por exemplo, para medir aplicação ou projetos de desenvolvimento, a seguinte: NoPFajustado = NoPFnão ajustado x FA.

5 Medição de Tamanho de *Data Mart*

Existem diferenças substanciais entre a construção de um *software* transacional e a construção de um produto de *Data Warehouse*/*Data Mart*. Além do processo ser bastante diferenciado, o resultado, o tratamento dos dados, a visão do usuário possuem características muito diferentes quando comparados a um sistema tradicional. Existem, no caso desta tecnologia, funcionalidades que não são visualizadas pelo usuário final e que impactam no tamanho do sistema que esta sendo mensurado, como por exemplo todas as funcionalidades geradas para tratar os dados na *staging area*.

A mensuração de um projeto de *Data Warehouse* com a abordagem APF fica prejudicada considerando que este processo, na visão do usuário, recebe dados já armazenados por outros sistemas e disponibiliza-os de forma que ferramentas adquiridas possam ser utilizadas para minerar ou consultar historicamente estes dados. A métrica APF examinada não trata exemplos específicos para contagem de tamanho de sistemas de *Data Warehouse/Data Mart.*

Dessa forma é necessário analisar cada um dos itens que são considerados na APF e adequá-los às características de *Data Mart* apresentadas na seção 3. Apresentaremos, a seguir como adaptamos a APF considerando os seis (6) passos de medição apresentados na seção anterior.

[6] Cada função de dado ou transacional terá um peso diferente dependente de sua complexidade. Diversas tabelas baseadas na quantidade de elementos de dados, de registros e de arquivos referenciados são utilizadas para determinar a complexidade de cada função em Baixa, Média ou Alta. Ex. uma EE que possui 3 elementos de dados é considerada de baixa complexidade e corresponde a 3 PF.

No que se refere a **(i) estabelecer o objeto da contagem** não é necessário nenhuma adequação, pois identificar o objeto da contagem segue os mesmos padrões (desenvolvimento, manutenção, etc) de um sistema transacional.

Com relação a **(ii) Determinar a fronteira de medição** do aplicativo é necessário verificar se os dados de origem são fornecidos pelos sistemas de origem e neste caso a fronteira de medição fica restrita as tabelas internas do projeto de *Data Mart*. Se os dados de origem não são fornecidos pelos sistemas de origem e cabe a equipe responsável pelo sistema de *Data Mart* gerar a extração, os arquivos dos sistemas de origens serão pontuados como AIE, tornando a fronteira de medição do *Data Mart* mais ampla.

No que se refere a **(iii) contar as funções de dados** deve ser considerado que o usuário possui a visão das dimensões e fatos que necessitará para suas pesquisas. Todos os fatos e dimensões devem ser pontuados como ALIs. Como AIE serão pontuados os dados corporativos utilizados no projeto.

O usuário também possui a visão de que estes dados deverão ser tratados, limpos, agregados, sumarizados em uma área antes de serem disponibilizados para consulta. Com base nesta visão e considerando também a necessidade de se computar os dados da *staging área*, sugerimos que para todos os ALI computados inicialmente sejam também computados ALI para a *staging área*. Os dados da *staging área* são dados que permanecem e que são utilizados constantemente para as novas cargas e atualizações.

A definição da complexidade para cada função de dado será aplicada conforme a proposta da APF.

Para **(iv) contar as funções transacionais** deve-se para cada ALI considerar uma EE, pois eles são atualizados a partir dos dados de sistemas transacionais e funcionam como uma EE. Na realidade o processo de carga é muito mais complexo e gera muito mais processos do que apenas um como está sendo sugerido, mas considerando a visão do usuário sugerimos a definição de uma EE para cada ALI. Com relação a SE ou CE sugere-se que sejam computadas qualquer solicitação de relatórios/consultas para facilitar a consulta do usuário final, respeitando-se a distinção entre SE e CE da proposta APF. A definição da complexidade para cada função transacional será aplicada conforme a proposta da APF.

O passo referente a **(v) Determinar o Fator de ajuste** implicou numa análise cuidadosa das características gerais propostas na APF no que se refere à *Data Mart*. Como resultado dessa análise percebemos que das 14 características gerais:

- 4 características gerais são aplicáveis a este tipo de software: Processamento distribuído de dados, Desempenho, Reusabilidade de Código e Facilidade Operacional.
- 2 características gerais poderiam ser adaptadas para *Data Warehouse/Data Mart*, são elas:
    - Eficiência do usuário final[7] que poderia ser adequado para considerando a Quantidade de agregação[8] onde a definição dos níveis de agregação necessários tem como objetivo proporcionar eficiência para o usuário final.
    - Processamento complexo[9] que poderia ser adequado para Qualidade dos dados[10] onde a quantidade de tratamento (de dados e das exceções) necessário ao projeto pode ser comparada como um nível de complexidade do processo.

[7] Aspectos relacionados com a eficiência do aplicativo na interação com o usuário**.**

[8] Definição dos níveis de agregação necessários de forma a melhorar o desempenho das consultas do usuário final.

[9] Aspectos relacionados com a complexidade do processamento.

[10] Descreve o grau previsto para tratamento de exceções identificadas inicialmente com relação à qualidade de dados, dados rejeitados, erro de conteúdo, etc.

As demais características gerais (no total de 8) estão intrinsecamente relacionadas a sistemas transacionais o que implica em quando analisados no contexto da *Data Warehouse/ Data Mart* sempre receberiam o valor 0 (nenhuma influência).

Baseados nessa análise resolvemos considerar as 4 características realmente aplicáveis, substituir as 2 características possíveis de adequar por nomes mais pertinentes, e propor novas características que representem o escopo de *Data Mart*.

Segundo Lokan e Abran [10], as características gerais do *software,* propostas pela APF, identificam vários aspectos funcionais e não-funcionais do *software* que são utilizados na mensuração de tamanho funcional.

Para embasar a substituição e a definição de novas características foi realizada pesquisa bibliográfica e identificados os principais aspectos funcionais e não-funcionais que influenciam na construção de sistemas de *Data Warehouse/Data Mart.* Para cada uma das características, foram definidos os níveis de influência numa escala de 0-5 conforme proposto na APF. A proposta final de características gerais para *Data Mart* pode ser visualizada na Quadro 2.

Quadro 2 – Características gerais propostas para sistemas de *Data Mart*

Características gerais	Níveis de influência
1 Processamento distribuído de dados Aspectos relacionados com processamento e funções distribuídas. Comentários: Leitura via *client* ou via *Internet* ou *Intranet* pode receber o valor 2 a 4.	0. O aplicativo não auxilia na transferência de dados ou funções entre os processadores envolvidos; 1. O aplicativo prepara dados para que o usuário final os utilize em outro processador (planilhas de cálculo, por exemplo); 2. O aplicativo prepara dados e os transfere para que outros equipamentos os utilizem; 3. O processamento é distribuído e a transferência de dados acontece de forma on-line apenas em uma direção; 4. O processamento é distribuído e a transferência de dados acontece de forma on-line em ambas as direções; 5. As funções de processamento são dinamicamente executadas no equipamento mais apropriado.
2 Desempenho/*Performance* Aspectos relacionados a parâmetros estabelecidos pelo usuário quanto a tempos de resposta.	0. Nenhum requerimento especial de *performance* foi solicitado pelo usuário; 1. Requerimentos de *performance* 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. Porém, nenhuma determinação especial foi estabelecida quanto à utilização do processador. A data limite para a disponibilidade do processamento é sempre o próximo dia útil; 3. Tempo de resposta e volume de processamento são itens críticos durante todo o horário comercial. Não há determinação especial para a utilização do processador. A data limite para a comunicação com outros aplicativos é um item importante e deve ser considerado. 4. quando, além do descrito no item 3, os requisitos de *performance* estabelecidos requerem tarefas de análise de *performance* na fase de análise e desenho do aplicativo. 5. quando, além do descrito no item 4, ferramentas de análise de *performance* precisam ser usadas nas fases de desenho, desenvolvimento ou mesmo na fase de implementação para que os requisitos do usuário sejam atendidos plenamente.
3 Utilização de ferramenta apropriada para extração e carga Indica o nível de automatização e complexidade do processo de *Data Mart*	0. quando é utilizada ferramenta para 100% do processo de extração/transformação/carga; 1. quando é utilizada ferramenta para até 80% do processo de extração/transformação/carga 2. quando é utilizada ferramenta para até 60% do processo de extração/transformação/carga; 3. quando é utilizada ferramenta para até 40% do processo de extração/transformação/carga; 4. quando é utilizada ferramenta para até 20% do processo de extração/transformação/carga; 5. quando nenhuma ferramenta é utilizada para extração/transformação/carga dos dados transacionais

Quadro 2 (cont.) – Características gerais propostas para sistemas de *Data Mart*

Características gerais	Níveis de influência
4 Quantidade de sistemas transacionais envolvidos no projeto Descreve o grau em que a quantidade de interfaces com outros sistemas influenciará o desenvolvimento da aplicação. Quando a quantidade de sistemas transacionais é alto, influencia o projeto, desenvolvimento, implantação e suporte da aplicação.	0. Não se aplica 1. quando o projeto envolve 1 sistema transacional; 2. quando o projeto envolve de 2 a 3 sistemas transacionais; 3. quando o projeto envolve de 4 a 5 sistemas transacionais 4. quando o projeto envolve de 6 a 7 sistemas transacionais; 5. quando o projeto envolve mais de 8 sistemas transacionais.
5 Documentação dos sistemas transacionais de origem (Existência de Metadados dos sistemas de origem) Nível de documentação dos sistemas de origem, de forma a identificar a existência de metadados dos dados de origem.	0. quando todos os sistemas de origem possuem metadados; 1. quando acima de 90% dos sistemas de origem possuem meta dados; 2. quando acima de 70% dos sistemas de origem possuem metadados; 3. quando acima de 50% dos sistemas de origem possuem metadados; 4. quando acima de 30% dos sistemas de origem possuem metadados; 5. quando nenhum dos sistemas de origem possui metadados;
6 Quantidade de agregação *substituindo* **Eficiência do usuário final** Definição dos níveis de agregação necessários de forma a melhorar o desempenho das consultas do usuário final	0. nenhum nível de agregação identificado; 1. Um nível de agregação identificado; 2. De dois a três quantidades de agregação identificadas; 3. De quatro a cinco quantidades de agregação identificadas; 4. Seis ou sete quantidades de agregação identificadas; 5. Oito ou mais quantidades de agregação identificadas.
7 Freqüência de atualização das fontes de dados Descreve o grau em que os sistemas transacionais são alterados implicando em constantes alterações nas aplicações de extração e carga.	0. quando houver previsão de 0% a 10% de atualizações dos arq. de extração /carga; 1. quando houver atualizações de 10% a 20% dos arq. de extração /carga; 2. quando houver atualizações de 20% a 30% dos arq. de extração /carga; 3. quando houver atualizações de 30% a 40% arq. de extração /carga; 4. quando houver atualizações de 40% a 50% dos arq. de extração /carga; 5. quando houver atualizações de mais de 50% arq. de extração /carga.
8 Qualidade dos dados em substituição *ao* **Processamento complexo** Descreve o grau previsto para tratamento de exceções identificadas inicialmente com relação à qualidade de dados, dados rejeitados, erro de conteúdo, etc.	Ao considerar as características da aplicação verificar a necessidade de aplicabilidade dos seguintes itens: - Integração – envolve a geração de chaves substitutas para cada registro, de modo a evitar a dependência de chaves definidas no sistema legado; - Limpeza – correção de códigos e caracteres especiais, resolvendo problemas de domínios, tratando dados perdidos e corrigindo valores duplicados ou errados; - Eliminação – eliminar campos e dados provenientes dos sistemas legados que não serão úteis ao *Data Mart.* - Combinação – realizada quando fontes de dados possuem os mesmos valores de chaves representando registros iguais ou complementares ou atributos de chaves não iguais, incluindo equivalência textual de códigos de sistemas legados distintos; - Verificação de integridade referencial – significa verificar se os dados de uma tabela são iguais aos dados correspondentes em outra tabela - Desnormalização e renormalização – consiste em reunificar as hierarquias de dados, separadas pela normalização dentro de uma tabela desnormalizada; - Conversão de tipo de dados – envolve transformação de baixo nível de forma a converter um tipo de dado em outro formato - Cálculos, derivação e alocação - são transformações a serem aplicadas sobre as regras de negócio identificadas durante o processo de levantamento de requisitos; - Auditoria no conteúdo dos dados – o processo de transformação deve realizar constantes verificações de somas, contagem de linhas e testes. Tem-se: 0. quando não ocorrer nenhuma das características acima; 1. quando ocorrer de uma a duas das características acima; 2. quando ocorrer de três a quatro das características acima; 3. quando ocorrer cinco a seis das características acima; 4. quando ocorrer sete a oito das características acima; 5. quando ocorrer todas as características acima;

Quadro 2(cont.) – Características gerais propostas para sistemas de *Data Mart*

Características gerais	Níveis de influência
9 Reusabilidade de código Aspectos relacionados à reutilização do código do aplicativo.	0. Nenhuma preocupação com reutilização de código. 1. Reutilização de código apenas no aplicativo. 2. Menos de 10% do código do aplicativo foi projetado para ser utilizado em outros aplicativos. 3. 10% ou mais do código do aplicativo foi escrito para ser utilizado em outros aplicativos. 4. O código do aplicativo foi projetado para ser utilizado em outros aplicativos. A customização deve ser realizada em nível de código-fonte. 5. O código do aplicativo pode ser reutilizado em outros aplicativos com alto grau de parametrização. É apenas necessário que o usuário altere determinados parâmetros.
10 Estrutura dos dados de origem Definição da estrutura em que estão os dados de origem, VSAM, Relacional(DB2, *Sybase, Oracle*), Hierárquico – IDMS).	0. Não se aplica; 1. quando existir uma única estrutura dos dados de origem; 2. quando existir duas estruturas dos dados de origem; 3. quando existir três estruturas dos dados de origem; 4. quando existir quatro estruturas dos dados de origem; 5. quando existir mais de quatro estruturas.
11 Facilidade operacional Aspectos relacionados com a facilidade de operação do aplicativo. Avalia procedimentos operacionais automáticos e mecanismos de iniciação, salvamento e recuperação de dados.	0 Nenhuma consideração especial de operação além do processo normal de salvamento de dados; **1 a 4:** quando um ou todos os itens seguintes se aplicarem (selecionar todos os que se aplicam; cada item soma um ponto): ▪ Foram desenvolvidos procedimentos de iniciação, salvamento e recuperação, mas a intervenção do operador é necessária; ▪ Foram desenvolvidos procedimentos de iniciação, salvamento e recuperação, sem a necessidade de intervenção do operador (vale dois itens); ▪ O aplicativo minimiza a necessidade de montagem de fita magnética; ▪ O aplicativo minimiza a necessidade de manuseio de papel; 5 O aplicativo foi desenhado para trabalhar sem operador. Nenhuma intervenção do operador é necessária além de iniciar e encerrar o aplicativo porque este já contém rotinas automáticas de recuperação de erros.
12 Volume de dados Previsão do volume de dados do projeto O volume de dados interfere no tamanho e deve ser previsto visando garantir *performance*	1 Baixo (até 500 *gigabytes*) 3. Médio (de 500 *gigabyte*s a 1 *terabyte*) 5. Alto (acima de 1 *terabyte*)
13. Nível de conhecimento exigido pela equipe de *Data Mart* da base de dados/regras de negócio dos sistemas transacionais de origem (Vinculada à existência de ferramenta ETL, pois a existência obriga a equipe de *Data Mart* a conhecer todas as regras de negócio transacional e definir outras formas de extração).	1 Pouco conhecimento da equipe de *Data Mart* das regras de negócio dos sistemas transacionais 3 Médio conhecimento da equipe de *Data Mart* das regras de negócio dos sistemas transacionais 5 Alto conhecimento da equipe de *Data Mart* das regras de negócio dos sistemas transacionais

Finalmente para **(vi) determinar o tamanho do projeto** adequamos a fórmula proposta na APF.

As características gerais do software definidas na proposta inicial da APF eram 10 e possuíam uma variação de mais ou menos 25% ([10], [14]). Posteriormente foram modificadas para 14 e passaram a incrementar ou diminuir em até 35% do valor de pontos brutos computados. Analisando a abordagem inicial e atual das características gerais da APF, identifica-se que para cada característica geral de sistema foi atribuído o valor de 2,5.

Com a proposta voltada para o contexto de *Data Mart*, as características gerais de sistema foram reduzidas de 14 para 13. Foi necessário então adequar o valor da fórmula para um ajuste de mais ou menos 32,5 %. Para isto foi utilizada uma regra de três simples com relação à proposta atual. A fórmula de cálculo foi adequada para:

FA = 0,63 + (0,01 x Soma das características gerais de sistema).

6 Medição de Tamanho em Projetos Reais

Nesta seção será apresentada a aplicação dessa proposta em projetos reais da indústria, iniciando com o planejamento do estudo, a efetiva aplicação e concluindo com a análise dos resultados.

6.1 Planejamento

Visando utilizar e validar a proposta de adequação da APF para a tecnologia de *Data Mart*, esta foi aplicada em projetos de três instituições federais. A proposta foi aplicada em 6 projetos da primeira instituição, em 1 projeto da segunda e em 3 projetos da terceira instituição. Dessa forma foram avaliados ao todo 10 projetos de *Data Mart* objetivando efetuar uma comparação entre as duas abordagens (a APF tradicional e a proposta de adequação). Para isso foi considerado, o tempo estimado obtido de cada uma das propostas e o tempo real gasto para o desenvolvimento, verificando qual das duas abordagens se aproximava mais do tempo real.

Para cada instituição foram definidos os seguintes critérios para permitir a análise dos dados: quantidade de dias por mês (22 dias), carga horária diária (8 horas), quantidade de recursos alocados para construção do sistema (obtidos através de entrevistas dirigidas) e um fator de produtividade por pontos de função a ser utilizado. O fator de produtividade adotado para a primeira instituição (que já trabalhava com APF) foi de 16 horas por PF (qtd de horas que a empresa utiliza para programação na linguagem utilizada) e para as demais instituições em 8 horas e 6,8 horas por PF, respectivamente (baseado em dados de mercado (base ISBSG [8][11])) pois, estas empresas não possuíam base histórica de produtividade).

6.2 Aplicação

Foram aplicadas a APF e a Proposta nos 10 projetos de *Data Mart* que estão em fase de produção. Após a aplicação da métrica APF e da proposta foi necessário efetuar as comparações entre os resultados obtidos com relação ao tempo real dos projetos. Para isto foram utilizados os fatores de produtividade definidos na seção 6.1. Foi utilizada a quantidade de recursos efetivamente alocada nas equipes durante o tempo real de desenvolvimento.

A Figura 2 demonstra, de forma gráfica, a comparação entre a quantidade de pontos de função obtidos com a aplicação das duas abordagens: APF e proposta. Conforme pode ser observado, a proposta de medição voltada para esse contexto está definindo tamanhos maiores que a APF em todos os sistemas de *Data Mart* mensurados. Na Figura 3, pode ser analisada a comparação entre o tempo real de desenvolvimento, o tempo estimado da APF e da proposta elaborada.

[11] A produtividade adotada foi baseada na análise dos dados do banco de dados do *International Software Benchmarking Standards Group* (ISBG, 2002). O foco desse grupo é coletar, validar e publicar um repositório histórico de produtividade em projetos de *software*s.

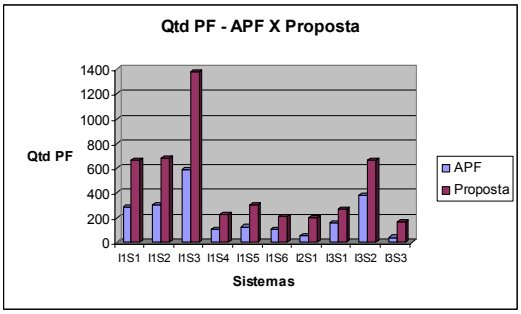

Figura 2
Comparação PF da APF x PF da Proposta

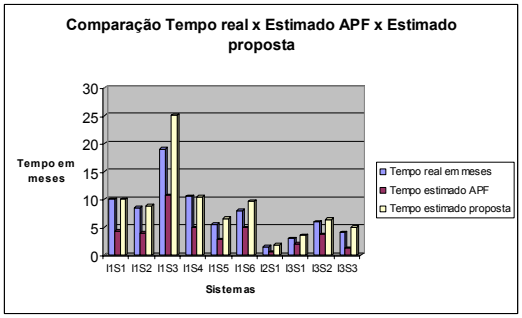

Figura 3
Comparação tempo real, estimado APF e Proposta

Conforme pode ser observado na Figura 3 a estimativa de tempo entre a abordagem proposta e o tempo real de desenvolvimento dos projetos de *Data Mart* ficou bem aproximada. Em contraste todas as estimativas efetuadas pela métrica APF para *Data Mart* ficaram abaixo do tempo real do sistema, o que demonstra, nesse escopo, uma mensuração não totalmente adequada de tamanho.

6.3 Análise dos resultados

Para realizar a análise dos resultados foi necessário verificar os tempos estimados (com a APF e com a Proposta) com o propósito de verificar o tempo mais aproximado ao tempo real no contexto de Sistemas de *Data Mart*.

A análise estatística foi realizada considerando:

- o tempo real utilizado para construção de *Data Mart*;
- o tempo estimado após a aplicação da APF; e,
- o tempo estimado após a aplicação da Proposta.

Foram definidos e aplicados os seguintes testes estatísticos: ANOVA e teste de Tukey. Para realizar a análise estatística foi utilizado o pacote estatístico SPSS[12] .

6.3.1 Análise de Variância - ANOVA (Analysis Of Variance)

Como são considerados três tratamentos (tempo real, tempo estimado APF e tempo estimado proposta) com o mesmo objetivo, mas obtidos de maneira diferente, pode se aplicar a ANOVA - Análise de Variância[13] para verificar se as amostras, que têm variâncias homogêneas, têm médias iguais ou diferentes, claro que, observadas as premissas estatísticas da normalidade e independência entre populações que participarão do teste.

ANOVA é conhecida como a técnica estatística mais empregada para interpretação de dados experimentais. Em geral, a finalidade da ANOVA é testar diferentes significâncias entre médias [11]. Wohlin et al. [15] também sugerem o uso de ANOVA quando existe três tratamentos a serem analisados para verificar uma hipótese.

Dessa forma, foram formuladas as hipóteses descritas no Quadro 3 :

[12] Software comercial que possui um conjunto de ferramentas estatísticas

13 Variância é uma média aritmética calculada a partir dos quadrados dos desvios obtidos entre os elementos da série e a sua média.

Quadro 3 – Hipóteses inicial e alternativa

Hipótese inicial	Hipótese alternativa
Hipótese nula[14]: **H0**: M1=M2=M3[15]	**H1**: Mi ≠ Mj para algum i≠j[16]

A hipótese inicial prevê a igualdade das médias dos três tratamentos (tempo real, tempo estimado APF e tempo estimado proposta), enquanto que a hipótese alternativa afirma que pelo menos uma das médias dos três tratamentos é diferente.

Primeiro foi definido o nível de significância[17] a ser aplicado. Segundo Vieira [13] a escolha do nível de significância é arbitrária e quando se escolhe o nível de significância 5%, é usual afirmar que o resultado é significante. Foi definido, então, nesta análise, o nível de significância de 5%.

Para se comparar mais de duas médias é necessário aplicar o p-valor [18]e nesta análise foi utilizado o cálculo do p-valor para rejeitar a Hipótese inicial em favor da Hipótese alternativa.

O teste p-valor é fornecido por programas estatísticos de computador e toda a vez que o p-valor for menor que o nível de significância estabelecido (neste estudo 0,05), rejeita-se a hipótese de que as médias são iguais.

O p-valor obtido foi de 0,017 , logo, existe pelo menos uma média estatisticamente diferente.

Uma ANOVA permite estabelecer se as médias das populações em estudo são, ou não são, estatisticamente iguais. No entanto, esse tipo de análise não permite detectar quais são as médias estatisticamente diferentes das demais. A ANOVA mostrou que as médias das populações não são iguais, mas não permite concluir qual é, ou quais são as médias diferentes das demais.

6.3.2 Teste Tukey HSD (Honestly Significant difference)

O teste de Tukey permite estabelecer a diferença mínima significante, ou seja, a menor diferença de médias de amostras que deve ser tomada como estatisticamente significante, em determinado nível.

O teste de Tukey é realizado considerando: o quadrado médio do resíduo da análise da variância, o número de repetições de cada tratamento e um valor dado em tabela. O valor de Tukey foi calculado pelo programa estatístico utilizado e foi obtido o valor da menor diferença significante(Tukey) de 3,2325.

De acordo com o teste de Tukey, duas médias são estatisticamente diferentes toda a vez que o valor absoluto da diferença entre elas for igual ou maior ao valor da menor diferença significante (Tukey).

A diferença absoluta entre o Tempo real e Tempo APF (4,3664) e o Tempo Proposta e Tempo APF (3,8715) são superiores ao valor de Tukey (3,2325), ou seja, são estatisticamente diferentes, considerando o nível de significância de 0,05.

---

[14] A hipótese nula é a negação da hipótese alternativa.

[15] Onde M1 = média do tempo real; M2 = média do tempo estimado após a aplicação da APF;e, M3 = média do tempo estimado após a aplicação da proposta.

[16] Onde M = média e i e j podendo assumir os valores de 1 a 3 (inclusive)

[17] Nível de significância é definido como a probabilidade de cometer o erro de tipo I, ou seja, rejeitar a hipótese nula (**H0**), quando ela é verdadeira.

[18] A estatística de *p-valor* avalia dados com relação a média de cada grupo, variância de cada grupo e variância ponderada, isto tudo implementado com uma distribuição teórica.

Em contrapartida, a média da estimativa de Tempo proposta é igual estatisticamente ao Tempo real, o que leva a concluir que a melhor ferramenta de medição para Sistemas de *Data Mart* é o Tempo Proposta calculado com base na proposta elaborada neste trabalho.

7 Conclusão e contribuições do trabalho

Uma das maiores dificuldades encontradas pela gestão de projetos é estimar o porte do que está sendo construído. Existem muitas abordagens para mensurar o tamanho de um software e não existe uma abordagem que seja melhor que outra, sob todos os aspectos, em qualquer situação. A abordagem de mensuração de tamanho deve ser escolhida e/ou adequada dependendo das características particulares do sistema que se pretenda desenvolver [3].

Sistemas de *Data Warehouse/Data Mart* propõem uma nova visão no processo de desenvolvimento. Diferentemente dos sistemas transacionais, o processo de construção é mais complexo, especializado e com características específicas desse tipo de software. Isso requer que outros processos, como os processos de medição (atualmente voltados para sistemas transacionais), que interagem com o processo de construção de forma a viabilizar um melhor gerenciamento do processo de construção de um software, sejam adaptados de forma a garantir o gerenciamento adequado de recursos, custos e tempo.

Com essa motivação em mente, esta dissertação teve por objetivo a definição de uma proposta de mensuração de tamanho para Projetos de *Data Mart* e os seguintes objetivos específicos foram elaborados: (i) Estudar as características principais de sistemas de *Data Mart/Data Warehouse*, identificando aspectos diferenciados em relação aos sistemas transacionais; (ii) Estudar algumas abordagens de métricas de tamanho existentes, analisar sua aplicabilidade a este contexto e identificar a melhor alternativa para adequação à tecnologia de *Data Mart*; (iii) Propor a adequação de uma das abordagens de métricas de tamanho para projetos de *Data Mart*; (iv) Utilizar e avaliar a nova adequação em projetos de *Data Mart*; e, (v) Comparar os resultados da aplicação desta proposta de adequação com os resultados da abordagem escolhida.

Foram estudadas as principais características da tecnologia *Data Mart/Data Warehouse* e identificadas as principais diferenças com relação aos sistemas transacionais. Foram estudadas algumas abordagens de métricas de tamanho existentes, seus pontos fortes e as críticas existentes. Dessa forma atende-se aos objetivos (i) e (ii).

Para o objetivo (iii), foi definida a APF como a mais indicada a ser adequada para o contexto de *Data Mart*. A abordagem APF possui maior maturidade tanto no meio acadêmico como na indústria, pois é uma das abordagens mais utilizadas e estudadas atualmente [6]. Apesar das críticas quanto à sua adequabilidade a diversos tipos de software, existem bases de produtividade histórica de mercado, o que facilita a sua aplicabilidade em organizações que não possuem uma base histórica de produtividade. Na proposta de adequação, os mesmos passos de contagem foram mantidos mas, foram indicadas novas formas de pontuar as funções de dados e transações e foram criadas novas características gerais de sistemas.

Para atender aos objetivos (iv) e (v), a proposta de adequação foi aplicada em alguns projetos de três instituições. Foram efetuadas análises para verificar a adequabilidade da proposta a esse contexto. Ao longo do estudo de casos reais, foi demonstrado que nossa proposta aplicada a projetos de *Data Mart* garantiu uma melhor representação do tamanho do sistema, o que dá indícios de sua melhor adequação para este contexto.

Além desse benefício, foram identificadas outras contribuições desta dissertação:

- a adequação de uma métrica existente para o contexto de *Data Mart*;
- a aplicação e avaliação de uma métrica existente (APF) considerando o contexto de *Data Mart* em projetos reais em três instituições estudadas.

Contudo, a utilização prática no contexto dessas três instituições em que os projetos de *Data Mart* foram avaliados, permitiu identificar alguns pontos que poderão ser alvos de trabalhos futuros:

- Investigar outras tecnologias e domínios que necessitam ser mensurados e verificar a adequabilidade das métricas existentes para esses domínios e tecnologias;
- Aplicar esta abordagem em um número maior de projetos de *Data Mart*; e,
- Expandir a aplicação das métricas para as demais categorias de projetos envolvendo a área de BI (Bussiness Intelligence) como um todo, ou seja, abordar os aspectos da construção de *Data Warehouse* corporativo e as variações no conceito do Data Warehouse, como *Active Data Warehouse*, *Real-time Data Warehouse*, entre outros.

Bibliografia

[1] AGUIAR, Maurício. Estimando os projetos com Cocomo II no RUP. **Developers Magazine**. Set/2002.

[2] BARBIERI, C. **BI-Business Intelligence – Modelagem & tecnologia**. Rio de Janeiro: Axcel Books do Brasil Editora, 2001.

[3] CALAZANS, A., OLIVEIRA, K., SANTOS, R. Dimensionando Data Marts :Uma adequação de uma métrica funcional. In: **II Simpósio Brasileiro de Qualidade de Software**, 2003. Fortaleza. Anais SBQS 2003. Fortaleza, Unifor, 2003.

[4] IFPUG. International Function Point Users Group. **Function Point Counting Practices Manual**: Release 4.1. Ohio: IFPUG. 2000. 1 v.

[5] FENTON, N., PFLEEGER, S. **Software metrics a rigorous & practical approach**. 2nd. Ed., PWS Publishing Company, 1997.

[6] GARMUS, D., HERRON, D. **Function point analysis – measurement practices for successful software projects.** Addison-Wesley Information Technology Series, 2000.

[7] INMON, W.H. , **Definition of a Data Warehouse**. 1999. Disponível em: <www.billinmon.com/library/articles/dwedef.asp. Acesso em 05 Mai 2003.

[8] ISBSG. Benchmarking Repository, Release 6. ISBSG. Abr, 2002.

[9] KIMBALL, R., ROSS, M. **Data warehouse toolkit: o guia completo para modelagem multidimensional**.Rio de Janeiro: Campus, 2002. 494 p.

[10] LOKAN, C., ABRAN, A. Multiple viewpoints in functional size measurement. In: **International Workshop on Software measurement - IWSM'99.** Canada. p. 121-132, 1999.

[11] PHADKE, M.S., **Quality Engineering Using Robust Design**. Prentice Hall, Englewood Cliffs New Jersey, 1989.

[12] SIMÕES, C. Sistemática de Métricas, qualidade e produtividade. **Developers' Magazine**, Brasil, 1999.

[13] VIEIRA, S. **Estatistica Experimental**. 2.ed, São Paulo: Atlas. 1999.

[14] WITTIG, G., MORRIS, P., FINNIE, G., RUDOLPH, E. **Formal methodology to establish function points coefficientes.** School of Information Technology. Australia, [1997?].

[15] WOHLIN,C., RUNESON,P., HOST,M.,OHLSSON, M.,REGNELL, B., WESSLEN, A. **Experimentation in Software Engineering An Introduction**. Kluwer Academic Publishers. Londres, 2000.