

XIII CONGRESSO NACIONAL DE PROCESSAMENTO DE DADOS

OUTUBRO 1980



SUcesu

RIO DE JANEIRO

XIII CONGRESSO NACIONAL DE PROCESSAMENTO DE DADOS

ANAIS DO XIII CNPD

Hotel Nacional-Rio
Outubro de 1980

COMISSÃO ORGANIZADORA

Raulino Carvalho de Oliveira
Salvador Perrotti
Sylvio de Carvalho S. Mattos
Carlos Eduardo C. Fonseca
Raul Isiris
Fernando Vianna
Miguel Stábile

Eduardo Jorge Caldas Pereira
Gabriel de Almeida
Ronaldo Latsch
Carlos Alberto Pontes
Maria Elzira C. Isiris
Marlene Caldeira

COMISSÃO TÉCNICA

Fernando Vianna
Artur Edson Dias Pereira
Francisco Antônio Dantas Monteiro
José Eduardo Thyrso de Lara

Paulo Frota Simas de Oliveira
Jorge Peñaranda Coimbra
Carlos Jorge Zimmermann
Luiz Carlos Pinna

**PONDO SOB CONTROLE O
DESENVOLVIMENTO DE SISTEMAS AUTOMATIZADOS**

ARNDT VON STAA

PONDO SOB CONTROLE O DESENVOLVIMENTO DE SISTEMAS AUTOMATIZADOS

Arndt von Staa

Pontifícia Universidade Católica do Rio de Janeiro

Palavras-chave: Aprovação, Ciclo de Vida, Controle de Alterações, Desenvolvimento Controlado, Documentação, Especificação, Fases, Plano do Desenvolvimento, Qualidade, Sistemas Automatizados, Software, Verificação.

RESUMO:

São apresentados os princípios gerais a serem satisfeitos para que consigamos desenvolver sistemas automatizados úteis, duráveis, utilizáveis e eficientes, onde este desenvolvimento ocorre de forma econômica e controlada, dando certeza de que alcançaremos tal objetivo. Os princípios gerais são: O planejamento do desenvolvimento; a presença de especificações; a contínua verificação da qualidade dos componentes e do sistema em si; o controle de alterações; a construção a partir da documentação; o processo de construção por refinamentos sucessivos; e um ciclo de vida que descreve a sequência de refinamentos sucessivos.

O presente estudo foi conduzido tendo em mente a determinação dos objetivos a serem alcançados por um sistema de desenvolvimento de sistemas automatizados assistido por computador.

1 – INTRODUÇÃO

O desenvolvimento de sistemas automatizados é regido por diversos fatores que limitam a escolha de alternativas e os níveis de qualidade e utilidade que podem ser atingidos. Muitos desses fatores não tem sido devidamente observados, o que tem levado a frustrações, insatisfações, perdas elevadas, etc. Por outro lado, quando sistemas são pequenos ou muito simples, a não observância desses fatores causa somente contratempos pequenos. Foi constatado que as dificuldades encontradas ao desenvolver-se um sistema automatizado aumenta pelo menos quadraticamente com o crescimento em tamanho e/ou complexidade deste sistema [Boehm73, Putnam77a]. Assim ao passarmos de sistemas simples para complexos, ou ao desenvolvermos software, produto, repentinamente nos defrontamos com dificuldades cuja ordem de grandeza sequer imaginávamos.

O desejo de produzir sistemas automatizados de forma industrial, ao invés da sua criação artesanal, motivou estudos sobre o processo de desenvolvimento de tais sistemas. Esses estudos levaram a reconhecer-se fatores que afetam a produção de sistemas automatizados e que limitam o grau de qualidade e utilidade atingível. Um dos fatores identificados é a necessidade de um

controle eficaz sobre o desenvolvimento do sistema. Através deste controle, serão assegurados os níveis de qualidade e utilidade desejados, é assegurada a presença de documentação e, finalmente, é assegurado o emprego econômico de recursos (tempo, pessoal, material, equipamento, dinheiro, etc.) durante o desenvolvimento.

Neste texto esboçaremos o que se entende por produção de sistemas automatizados sob o ponto de vista de uma disciplina de engenharia, em contraste com o desenvolvimento artesanal destes sistemas. Os seguintes pontos devem ser observados ao produzir-se software de forma industrial:

- a) Otimização dos requisitos de utilidade e de qualidade dentro dos fatores limitantes do desenvolvimento, em especial os fatores custo e tempo;
- b) existência de documentação fidedigna e que sirva efetivamente para orientar o desenvolvimento do software e sua posterior manutenção;
- c) Especificação do produto, produção de acordo com a especificação e certificação contínua do produto;
- d) Planejamento do desenvolvimento, e posterior

acompanhamento e controle segundo o plano de desenvolvimento;

- e) Aprovação dos componentes e, por fim, do próprio sistema quanto à satisfação dos requisitos de utilidade, de qualidade, de proteção e segurança, onde esta aprovação é efetuada continuamente desde o início do projeto.

2 – PLANEJAMENTO DO DESENVOLVIMENTO

Sistemas automatizados tendem a levar bastante tempo para serem desenvolvidos. Além disso, o desenvolvimento requer a disponibilidade de diversos recursos (tempo, pessoas, produtos intermediários, equipamentos, suporte, dinheiro, etc.) que, se não disponíveis na época exata, causam sérios transtornos ao desenvolvimento. Por outro lado, a utilização de recursos não balanceada no tempo aumenta os custos operacionais da instituição que desenvolve o Sistema, aumentando destarte os custos a serem rateados pelos diversos sistemas desenvolvidos por esta instituição. Finalmente, pela própria natureza do processo de desenvolvimento de sistemas automatizados, existem restrições quanto ao grau de paralelismo com que pode ser levado a cabo o desenvolvimento.

Em consequência do dito acima, necessita-se de um plano de desenvolvimento que determine, entre outras coisas:

- i – As atividades de desenvolvimento;
- ii – As datas em que estas atividades deverão iniciar e terminar;
- iii – Os recursos e produtos necessários para iniciar-se cada atividade;
- iv – Os produtos resultantes de cada atividade;
- v – A data em que os diversos produtos resultantes estarão disponíveis, e, também, a data em que estarão aprovados (controle de qualidade completo);
- vi – A evolução no tempo do consumo de recursos discriminada por recurso (financeiro, pessoal, equipamentos, material, etc.) e por atividade.

O plano de desenvolvimento nada mais é do que uma descrição da sequência de atividades que, se executadas, nos levarão ao produto final desejado. Existe então a necessidade de contínuo controle da execução segundo o plano. Para que isto seja possível deverão ser incluídos pontos de controle no plano. Para

destes pontos de controle, estes devem ser definidos em termos de produtos acabados e aprovados, e não em termos de esforço, ou consumo de recursos.

Para permitir um acompanhamento efetivo, devemos poder determinar com frequência que pontos de controle foram alcançados. Isto requer a criação de uma hierarquia de atividades, ou seja, devemos definir atividades complexas em termos de conjuntos de atividades menos complexas. As atividades dos níveis mais baixos desta hierarquia deverão ter uma duração estimada de duas a três semanas. Cada uma das atividades, elementares ou compostas, deve levar a produtos acabados e aprovados. Agindo desta forma serão tornadas raras as crises, uma vez que poder-se-á determinar com antecedência a possibilidade da ocorrência delas. Em geral esta antecedência é suficiente para tomar medidas preventivas.

Para ser útil, o plano deve ser antes de mais nada realista. De nada adianta produzir-se um plano impecável quanto à forma e consistência interna, porém não exequível. O plano deve ser examinado, portanto, com o máximo de rigor e somente ser aprovado quando houver acordo entre todos que participaram de sua confecção. Além disso, a confecção do plano deve ser efetuada por todos aqueles que serão responsáveis por executá-lo. De outra forma não será possível garantir-se a exequibilidade do plano.

É óbvio que o excesso de zelo também é prejudicial. O objetivo de qualquer plano é essencialmente predizer o futuro, tendo por base estimativas, hipóteses e restrições. Porém, durante a confecção do plano, ou de seu detalhamento, não temos suficiente conhecimento de todos os condicionantes que encontraremos durante o desenvolvimento sendo planejado. Assim, é possível e até provável que, durante o desenvolvimento, descubramos que algumas estimativas e hipóteses não são verdadeiras, o que nos obriga a alterar o plano ajustando-o às novas condições.

O plano de desenvolvimento é produzido por refinamentos sucessivos. Ou seja, inicialmente é elaborado um plano geral e pouco detalhado e, à medida que o desenvolvimento prossegue, este plano é detalhado. Desta forma podemos manter sob controle erros de estimativas e de hipóteses inviáveis. Este procedimento tem o inconveniente de tornar difícil a geração de orçamentos pré-estabelecidos e não reajustáveis.

Quando a incerteza quanto ao plano global criado durante a fase de planejamento inicial for muito grande, a época em que os planos das fases posteriores forem detalhados, é também ocasião para uma revisão cuidadosa do cronograma, dos custos de desenvolvimento e da análise da rentabilidade do sistema sendo desenvolvido. Cabe observar que procedendo desta forma, teremos orçamentos flexíveis e reajustáveis em pontos previamente definidos. Além disso, o reajuste é efetuado somente se for assegurada a rentabilidade do sistema em desenvolvimento.

Para permitir o planejamento inicial a ser efetuado com suficiente credibilidade, necessitamos dispor de estatísticas relativas a projetos passados. Estas estatísticas permitirão a produção de estimativas mais precisas para o projeto como um todo. A coleção destas estatísticas requer a existência de uma sistemática de coleta de dados confiável, de baixo custo e produzindo dados úteis para a confecção de futuros planos [Leite79].

Como já foi dito, poderão ser encontrados erros no plano a executá-lo. Estes erros advêm principalmente de erros de estimativas devidos à falta de conhecimento de detalhes relativos ao sistema a ser desenvolvido quando da confecção ou do detalhamento do plano. A existência de tais erros não é nociva em geral. Nociva é a existência de erros conhecidos já antes da aprovação do plano, e nociva é principalmente a falta de um plano de desenvolvimento, uma vez que isto fatalmente levará a uma falta de objetividade.

3 – ESPECIFICAÇÃO E PRODUÇÃO

O desenvolvimento controlado de sistemas automatizados requer a presença de especificações precisas. As especificações devem definir:

- i – O que deverá ser feito pelo sistema, ou componente do sistema (critério de correção);
- ii – As restrições que deverão ser respeitadas para que a implementação seja considerada satisfatória ao instalar o sistema ou componente (critérios de sucesso);
- iii – As restrições que deverão ser respeitadas para que a implementação seja considerada satisfatória no decorrer da vida útil do sistema (critério de durabilidade);
- iv – As restrições que deverão ser respeitadas para que o sistema ou componente seja ameno ao uso e/ou integração (critério de interface).

Uma vez de posse da especificação, torna-se possível partir para a produção em conformidade com esta especificação. Cabe salientar que neste momento o documento “externo” do produto (componente, documento, ou sistema) já existe. Não somente isto, este documento permite o controle de qualidade do produto após o término de sua confecção. Assim, documentos passam a existir antes dos produtos que descrevem, sendo a produção dirigida e controlada pela própria documentação.

Durante a fase de definição do sistema é criada a especificação funcional. É claro que durante esta fase não possuímos uma especificação que nos permita verificar a correção da especificação funcional produzida.

Assim torna-se necessário que, além da verificação “sintática” da especificação funcional, também o usuário do sistema se pronuncie quanto à adequação desta especificação funcional as suas necessidades. Esta aprovação deve ser formal, realizada segundo um ritual bem definido, devendo ser efetuada antes da continuação do desenvolvimento do sistema.

A produção de componentes do sistema e, conseqüentemente, do sistema em si, é efetuada segundo um ciclo de produção [Staa79]. Este ciclo consta de:

- i – Um sub-ciclo de produção propriamente dito durante o qual é confeccionado, verificado e corrigido o produto a ser elaborado; e
- ii – Um sub-ciclo de correção da especificação. Este tem por objetivo efetuar as alterações que se tornem necessárias nas especificações. A necessidade da alteração pode ser determinada durante qualquer uma das atividades de produção, bem como externamente a este ciclo de produção particular.

Especificações são produtos elaborados durante algum ciclo de produção anterior ao que agora está em curso. Uma vez pronta a especificação, ela servirá como partida para o desenvolvimento de diversos outros produtos, que por sua vez, poderão vir a ser especificações. Existe portanto, uma hierarquia e uma ordenação parcial de atividades de produção com relação à época em que ocorrem os ciclos de produção. Esta ordenação parcial é o principal fator limitante da capacidade de desenvolvimento em paralelo.

Em um determinado instante estarão ativos diversos ciclos de produção dependentes de uma mesma especificação. Qualquer alteração nesta especificação tem como consequência alterações nos ciclos de produção já concluídos ou em progresso e que dependem desta especificação. É evidente que o custo de efetuar a alteração da especificação é proporcional ao número de ciclos de produção completados ou ativos e dependentes desta especificação.

Como já foi mencionado, cada fase do ciclo de vida de um sistema é composto por um número variável de atividades. O conjunto de produtos desenvolvidos por estas diversas atividades é, então, o “produto” da fase.

Por sua vez, as atividades de uma fase podem ser decompostas em sub-atividades e estas, por sua vez, podem ser decompostas em sub-sub-atividades e assim por diante até chegarmos a tarefas indivisíveis. As decomposições são, a princípio, totalmente arbitrárias. Durante a decomposição somente deve ser observado que as sub-atividades cubram completamente a respectiva atividade. Observa-se que, para cada uma das atividades descritas, existe um ciclo de produção, possivelmente composto por diversos outros ciclos de

produção dispostos em série e/ou paralelo, conforme a decomposição de atividades em sub-atividades sequenciais ou paralelas.

Cabe agora conceituar melhor o que se entende por atividade. Durante o desenvolvimento de um produto, podem ser efetuadas diversas tarefas, que podem ser por exemplo: A produção de esboços de especificações, a redação de textos preliminares, a transcrição de textos de um meio para outro (digitação, datilografia), a condução de exames de requisitos de qualidade, etc. Do ponto de vista do controle e acompanhamento do desenvolvimento, os resultados destas tarefas não tem importância, tendo importância somente o resultado final a ser entregue. Uma atividade é, portanto, um conjunto de tarefas e/ou outras atividades, que leve a um produto aprovado e que seja relevante do ponto de vista do controle e acompanhamento do desenvolvimento (execução do plano), assim a cada atividade deve corresponder um ponto de controle.

4 – VERIFICAÇÃO CONTINUADA

Os produtos, quaisquer que sejam as suas naturezas, resultantes de uma atividade, ou conjunto de atividades, devem sempre passar por uma aprovação. Desta forma é assegurado que o volume de erros residuais e o custo de eliminação destes erros seja diminuído. O controle de qualidade continuado assegura, também, melhores práticas de desenvolvimento.

Estas consequências da aprovação continuada do sistema em desenvolvimento levam a uma redução do custo total do sistema (desenvolvimento, mais manutenção, mais produção), apesar de terem a tendência de aumentar os custos das fases de especificação. A redução do custo total advém principalmente de uma substancial redução do esforço necessário para testes, reajuste ("retrofit"), atividades de adaptação e correção durante o desenvolvimento, bem como de uma substancial redução do esforço de manutenção.

Como já foi dito, a cada atividade deve corresponder um ponto de controle. Já foi dito também, que atividades podem ser decompostas em sub-atividades, formando uma hierarquia de atividades. Como, para cada atividade, temos um ponto de controle, existe, também, uma hierarquia de pontos de controle.

Pontos de controle sucedem as atividades a que dizem respeito, correspondendo ao término das tarefas de aprovação de todos os produtos destas atividades e de suas sub-atividades caso existam.

Devido ao seu grau de importância maior, o ponto de controle ao final de uma fase é chamado de marco.

do desenvolvimento de sub-sistemas, cada qual com seu ciclo de vida próprio. Portanto, pode ser definida uma hierarquia de marcos para o controle do desenvolvimento destes sub-sistemas e, conseqüentemente, do sistema envolvente.

A aprovação de um produto (documento, componente, sub-sistema, sistema) é por sua vez uma tarefa que requer tempo para ser completada. Em alguns casos é desejável utilizar-se o produto mesmo que ainda não formalmente aprovado. Nestes casos devemos definir dois pontos de controle, um ao dar-se por acabado o produto, o outro após a sua aprovação. Cabe salientar que um produto acabado é um produto testado (verificado).

A tarefa de aprovação é composta por tarefas de exame da:

- i – Obediência às normas e padrões estabelecidos;
- ii – Completeza da documentação;
- iii – Satisfação de todos os requisitos dos métodos de teste e/ou verificação especificados;
- iv – Verificação da concordância do produto com a sua especificação;
- v – Verificação da eficácia do produto (adequação do produto a outros componentes do sistema ou ao usuário).

Durante a aprovação poderão ser constatadas inadequações do produto. A revisão e/ou correção destas inadequações se dá no intervalo de tempo destinado à tarefa de aprovação. Portanto, deve-se prever suficiente folga nos prazos destinados a tarefas de aprovação para que se possa acomodar o tempo necessário para correções.

Nem todas as aprovações requerem a participação de membros de todas as entidades envolvidas, em particular o usuário. No caso de marcos, porém, esta participação mais ampla é importante, uma vez que somente desta forma será possível assegurar-se a qualidade, utilidade e rentabilidade do sistema, bem como da adequação do sistema ao usuário.

Mesmo quando os testes são efetuados com o intuito de encontrar o maior número de erros possível, a contínua aprovação de produtos e conjuntos de produtos à medida que o desenvolvimento prossegue, transforma as atividades de teste e aprovação em comprovações da qualidade do sistema desenvolvido. Assim, apesar de nossos esforços em contrário, testes e aprovações deixam de ser constatações da existência de inúmeros erros e/ou inadequações.

Isto decorre de uma redução significativa dos erros e inadequações acidentalmente introduzidas nos produtos, e não de uma possível superficialidade dos testes. Desta forma é reduzido o número de iterações de correção

sendo também antecipadas as dificuldades que seriam descobertas somente quando o esforço de recuperação já tiver se tornado excessivo.

5 – CONTROLE DE ALTERAÇÕES

Por mais que se procure acertar de saída, ocorrerão sempre eventos imprevistos que nos obrigam a modificar durante o desenvolvimento: O plano, as especificações, os documentos e/ou os componentes já aprovados. Estas necessidades de alteração podem ser devidas à percepção de necessidades do usuário não previstas antes, mudanças do meio ambiente em que o sistema operará (por exemplo mudanças de legislação), mudanças de disponibilidade de recursos, etc. Em outras palavras, o processo de desenvolvimento tende a ser um processo iterativo, em que, algumas vezes, precisamos optar por técnicas do gênero tentativa e erro. Contudo, devemos estar sempre preocupados com a minimização do número de iterações. Para conseguir isto, será necessário, em alguns casos, desenvolver protótipos, modelos de simulação, ou conduzir experimentos simples, para que possamos reduzir o grau de incerteza. Novamente, o investimento com a condução de tais experimentos deve ser analisado com cuidado, investindo-se mais nesta análise quanto maior for a incerteza.

Erros, ou inadequações, de um produto podem advir de falhas de produção, ou de falhas de especificação. Ao deparar-se com um erro de especificação, é normal ter-se que recompor o ambiente no qual esta especificação foi desenvolvida. Isto pode ser caro, e o custo aumenta com o aumento do intervalo de tempo decorrido desde a aprovação da especificação até a observação da sua inadequação. O aumento do custo é devido à necessidade de corrigir-se outros produtos que foram desenvolvidos a partir da especificação em erro. O desejo de evitar este incremento em cascata do custo, é a principal razão para que se efetuem aprovações frequentes, iniciando já nas primeiras atividades do desenvolvimento do sistema automatizado. Estas aprovações são particularmente importantes quando o produto a ser aprovado é uma especificação.

Como já foi dito, apesar de todos os nossos esforços para evitá-las, ocorrerão alterações tanto em especificações quanto em produtos acabados. Deve existir, então, um controle sobre pedidos de alterações. Neste controle deve ser feita uma análise dos efeitos (custos-benefício) da alteração, e deve ser garantida uma disseminação rápida e eficaz das alterações aceitas.

Alterações durante o desenvolvimento são efetuadas, em geral, da mesma forma como o são correções, adaptações e extensões durante a fase de operações. Para que o custo destas alterações seja mantido pequeno, desde cedo os requisitos de manutenibilidade devem

ser observados. Mais uma vez isto torna necessário a existência de documentação, de critérios de aprovação, bem como de dados para efetuar-la, já desde as primeiras atividades do projeto de desenvolvimento.

A atividade de controle de alterações deve ser cuidadosamente definida e planejada para evitar oscilações durante o desenvolvimento. Oscilações desta natureza resultam em incrementos substanciais nos custos do desenvolvimento [Metzger73, Rocha78]. As oscilações podem decorrer de correções em cascata, ou de correções repetidas em um mesmo produto. Estas oscilações são eliminadas, desde que as alterações sejam sempre examinadas em todos os seus aspectos, em particular quanto aos efeitos colaterais.

Para reduzir o número de alterações em transito, é conveniente estipular uma ordenação das alterações propostas segundo prioridades. As alterações são então, acumuladas em cada nível de prioridade. Para cada nível de prioridade é definida uma regra de trabalho descrevendo os procedimentos de exame e disseminação das alterações aceitas, bem como da urgência com que isto deverá ser feito.

6 – DOCUMENTAÇÃO

O controle de alterações e o constante exame da qualidade dos produtos, exigem a presença de documentação. Ou seja, a documentação passa a existir antes do desenvolvimento do componente. Desta forma a documentação passa a ser o guia do desenvolvimento e não somente algo a ser entregue no final.

Várias vantagens podem ser enumeradas. A primeira certamente é a própria existência da documentação, que, devido, a ter sido utilizada durante o desenvolvimento, possui alto grau de qualidade.

Outra vantagem da existência de documentação é a possibilidade de poder-se trocar e/ou contratar novo pessoal sem afetar excessivamente o desenvolvimento. Finalmente, a existência de documentação leva a uma maior objetividade ao medir-se a qualidade, a utilidade e os benefícios de sistemas de computação.

Segundo alguns, a preocupação com documentação e exame da qualidade, da utilidade e dos benefícios, aumenta o tempo necessário para implantar-se o sistema. Na prática, porém, o tempo usualmente não é aumentado, mesmo em se tratando de projetos pequenos. A razão principal para a redução do esforço e do tempo necessário para desenvolver sistemas em ambiente controlado é a sensível redução de tentativas e erros, e a maior objetividade quanto aos caminhos a tomar para conseguir-se chegar ao produto desejado [Yohe74].

Os produtos do desenvolvimento de sistemas automatizados são equipamentos, por exemplo terminais de propósito

particular, e documentos. Estes documentos poderão existir sob forma impressa, ou seja, legível por seres humanos, ou estarão registrados em algum meio legível mecanicamente. Em outras palavras, constituem documentos não só os textos descritivos, como também os programas e arquivos resultantes do desenvolvimento.

Quando desenvolvemos um sistema por refinamentos sucessivos, partimos do geral para o detalhe. Isto reflete-se na produção, também por refinamentos sucessivos, dos documentos. Para evitar tornar-se repetitiva a redação ou transcrição de documentos é necessário um planejamento cuidadoso da sequência com que serão adicionados detalhes aos documentos. Idealmente dever-se-ia contar com suporte automatizado para a produção de documentos.

Documentos devem ser concisos, objetivos, corretos, completos, certificáveis, legíveis, etc. É claro que é difícil atingir-se a todos estes objetivos. Documentos devem ser sobretudo compreensíveis por todos aqueles que os deverão ler. Existe, portanto uma barreira de comunicação. Para vencermos esta barreira, é necessário utilizar-se uma linguagem adequada para transmitir as informações que desejamos transmitir por intermédio dos documentos. Cabe observar aqui que desejamos transmitir informação com precisão, portanto os documentos não podem conter ambiguidades, tampouco podem existir itens que deem margem a interpretações diferentes por diferentes leitores. Isto requer um estilo de redação apropriado a textos técnicos. Este estilo, aliado a linguagem utilizada, deve levar a documentos fáceis de ler e entender, e suficientemente formais para que possamos examinar e aprovar com maior rigor e menor esforço estes documentos [Wolverton77].

As linguagens utilizadas devem ser sucintas, ou seja, devem ser adequadas em número de detalhes ao nível de detalhe em que está sendo produzido o documento. Assim, no início do desenvolvimento, ou seja, nos níveis de menor detalhe, desejamos ver o sistema como um todo, enquanto que nos níveis de detalhe elevado desejamos ver as ações a serem efetuadas ao operar-se o sistema. Estas últimas devem ser apresentadas sob a forma de programas devidamente comentados [Lucena78].

Os níveis de linguagem deveriam permitir uma transcrição fácil de um nível para outro. As propostas mais radicais neste sentido, propõem uma única linguagem de especificação possuidora da propriedade de ser hierárquica [Ross77, Gane79, Demarco78].

Nos primeiros níveis, as linguagens mais adequadas são as gráficas, por permitirem, em pouco espaço,

conduzir um volume considerável de informações e, também, permitirem um exame mais aprimorado da sua qualidade. São exemplos de linguagens gráficas: Fluxos de dados; relações entre entrada, processamento e saída; diagramas de transição; diagramas estruturados; diagramas hierárquicos, etc.

Nos níveis de maior detalhe, as linguagens, mais adequadas são os próprios programas com comentários adequados [Lucena78]. Os comentários incluídos devem descrever, entre outras coisas, as condições que devem ser satisfeitas antes e após grupos de comandos contidos em blocos estruturados.

Além dos documentos formais (especificações), é requerido, em muitas ocasiões, texto explanatório que facilite a compreensão dos documentos. Também este texto deve ser sujeito a estilo e linguagem próprios.

Em cada instituição devem existir normas técnicas e glossários, tornando homogêneos os documentos produzidos pelas diversas equipes ativas nos diversos projetos de desenvolvimento da instituição. A existência de normas e glossários facilita o entendimento dos documentos, uma vez que existirá somente um jargão utilizado para a comunicação. Desta forma é reduzido o custo de produção, aprovação e compreensão dos diversos documentos.

Documentos devem ser completos. A verificação da completeza tende a ser uma operação demorada. São sugeridos, então, o desenvolvimento e a normalização de roteiros de documentação, que, quando devidamente preenchidos, levem a documentos completos (roteiros "pret-a-porter"). Estes roteiros possivelmente exigirão mais dados do que os necessários para determinados produtos. Estes requisitos excessivos deve ser indicados com "não se aplica", ao invés de serem simplesmente omitidos. Desta forma estará assegurada a completeza do documento, enquanto o roteiro levar a documentos completos.

Sistemas automatizados podem ser visualizados como sendo compostos de processos (procedimentos e programas) que recebem, transformam e geram dados, sendo comum dados produzidos em um processo servirem de entrada para outro. É evidente que tais dados precisam ser especificados cuidadosamente, antes mesmo de iniciar-se o desenvolvimento dos processos entre os quais fluem.

Para poderem operar corretamente, os processos necessitam receber dados que atendam às especificações de entrada. Mais uma vez implica na existência de uma especificação rigorosa dos dados de entrada antes de iniciar-se o desenvolvimento do processo que recebe estes dados.

Assim, um dos documentos a serem produzidos durante

o desenvolvimento é o diretório de dados [Furtado78]. Este documento contém informações a respeito do fluxo de dados, dos arquivos, das áreas de comunicação entre programas e módulos, dos protocolos de comunicação, das especificações dos dados, das normas relativas à representação dos dados, das restrições de utilização dos dados, etc. O diretório de dados é, portanto, o documento central em torno do qual gravita a restante documentação do sistema automatizado.

O diretório de dados, é, também, um instrumento de padronização, o que traz consigo uma maior facilidade de integração e comunicação entre os diversos sistemas e grupos de desenvolvimento de sistemas automatizados.

7 – CICLO DE VIDA

Qualquer sistema passa por um ciclo de vida. Em particular sistemas automatizados passam pelas fases de definição, construção, implantação e operação. A tomada de consciência do ciclo de vida de sistemas automatizados, trouxe consigo todo um arsenal de técnicas mais ou menos efetivas para o planejamento, controle de produção, desenvolvimento, controle de qualidade etc., vinculadas ao ciclo de vida. O objetivo destas ferramentas é procurar maior fidelidade nas estimativas de tempo e esforço de desenvolvimento, garantir a visibilidade do desenvolvimento, permitir a antecipação da aprovação, enfim, permitir ter-se confiança de estar desenvolvendo o que se deseja dentro dos limites de tempo, esforço, utilidade e qualidade previstos.

A definição de um ciclo de vida para o sistema a ser desenvolvido não é somente uma tarefa de planejamento. É também uma tarefa que assegura que o sistema alcance a qualidade e utilidade desejada dentro das restrições de recursos disponíveis.

A idéia de um ciclo de vida é aplicável tanto ao sistema em si quanto aos componentes deste sistema. O ciclo de vida descreve, portanto, a evolução no tempo de cada componente. Desta forma, não se deve entender o ciclo de vida como uma sequência de atividades, mas sim como uma malha de atividades, onde cada uma destas atividades pode estar ativa simultaneamente com relação a produtos diferentes.

7.1 Fase de Definição:

Durante a fase de definição do sistema são especificados os atributos de qualidade e utilidade, bem como os benefícios esperados, sendo produzidos o plano de desenvolvimento do sistema automatizado, a especificação funcional e o diretório de dados lógico. A fase de definição possui as seguintes sub-fases:

Concepção

Objetivo

- Identificação de necessidades ou de possíveis melhorias, que poderiam ser atendidas através de um novo sistema ou através da extensão de um sistema já existente. No caso de software produto, é nesta fase que ocorre a identificação das possibilidades de mercado.

Produtos

- Descrição das dificuldades observadas. No caso de software produto, teremos uma descrição das necessidades do mercado;
- Descrição das principais condições a serem observadas pela solução;
- Versão preliminar da proposta de desenvolvimento.

Especificação de Requisitos

Objetivo

- Produção da versão final da proposta de desenvolvimento
- Determinação dos atributos de qualidade e utilidade e a determinação dos pesos e dos níveis a serem atingidos pelos atributos de qualidade e utilidade;
- Estimativa da rentabilidade esperada;
- Esboço de uma possível solução.

Produtos

- Exame geral do sistema atual. O nível desta observação deve assegurar constantemente a visão geral do sistema e não a observação de detalhes, devendo indicar os principais defeitos deste, defeitos estes que deverão ser sanados pelo novo sistema ou pela extensão ao sistema. No caso de tratar-se de software produto, faremos um levantamento das necessidades e do potencial do mercado;
- Normas técnicas: Documentação, programação, aceitação por fase, controle de alterações, diretório de termos técnicos, segurança operacional, segurança de desenvolvimento, etc. Estas normas podem requerer o desenvolvimento de ferramentas de suporte ao desenvolvimento. É claro que as normas são desenvolvidas uma única vez, aplicando-se aos diversos projetos da instituição;
- Esboço de solução evidenciando os requisitos de qualidade, de utilidade e os benefícios desejados;
- Condições de relacionamento com outros sistemas;

- Condições de operação e desenvolvimento;
- Condições contratuais gerais;
- O plano e orçamento para o restante da fase de definição. Devido à elevada incerteza quanto a estimativas de custo e benefício, é recomendável que o financiamento do projeto seja particionado de forma a ser renegociável em pontos previamente definidos;
- Avaliação da proposta.

Aprovação da Proposta de Desenvolvimento

Objetivo

- Assinatura do contrato, ou do memorando de início de desenvolvimento.

Produtos

- Análise da rentabilidade do sistema a ser desenvolvido;
- Documento de aceitação assinado por representantes dos diversos grupos da entidade "usuário";
- Documento de aceitação assinado por representantes da Entidade "Cliente";
- Documento de aceitação assinado por representantes da Entidade "Coordenador";
- Contrato de desenvolvimento, ou memorando, assegurando a disponibilidade de recursos para o desenvolvimento.

Especificação Funcional

Objetivo

- Determinação das alterações de organização e métodos necessários para permitir a operação do novo sistema;
- Especificação funcional completa da solução proposta;
- Criação da parte lógica do diretório de dados.

Produtos

- Levantamento detalhado do sistema atual;
- Seleção de software;
- Determinação do fluxo de dados no sistema como um todo. Os dados de que se trata aqui, são na

de dados e os arquivos que transitam e são mantidos pelo sistema. Estes documentos, agregados e arquivos devem ser descritos quanto ao seu conteúdo, sem especificar detalhes de organização física, uma vez que o objetivo do fluxo de dados é definir o que o sistema faz e não como faz;

- Descrição dos elementos componentes do fluxo de dados;
- Especificação preliminar dos resultados a serem produzidos pelos diversos testes;
- Dicionário de termos técnicos;
- Parte lógica do diretório de dados. Compreende os nomes dos elementos e agregados de dados, seus limites de validade, sua interpretação, responsabilidades pela coleta dos dados, critérios de segurança e previsão dos custos e benefícios destes dados. Não compreende a organização dos dados, tamanhos, representações internas, codificações, etc. [Furtado78];
- Início da criação do manual do programador. O Manual do programador é dirigido aos diversos programadores ativos no desenvolvimento do sistema automatizado. Cada programador possuirá o seu manual próprio. O manual contém informações gerais (resumo do sistema, resumo do plano, normas técnicas, etc.) e informações dirigidas a cada programador (especificações, formatos, normas técnicas especiais, nomes de arquivos para os programas fonte, objeto e dados teste, etc). O manual do programador é criado à medida que o sistema vai sendo desenvolvido, através da incorporação de mais detalhes. A sua criação finda na fase de codificação;
- Organização e métodos necessários para operar o sistema sendo desenvolvido. Sistemas automatizados, para serem eficazes, devem ser simples, coesos e devem intercambiar o mínimo indispensável de dados entre seus componentes e o ambiente externo [Myers75]. O sistema antigo, principalmente se tiver sido criado por evolução histórica, frequentemente possui vícios e ineficiências internas que, se emulados no sistema automatizado, comprometerão seriamente a eficácia deste. Torna-se necessário, então, criticar e modificar os procedimentos do sistema antigo, afim de que a nova forma de proceder permita a introdução do sistema automatizado ou de uma extensão ao sistema antigo;
- Versão inicial, ainda incompleta, do manual do usuário. O manual do usuário contém informações sobre a utilidade do sistema, sobre dados a serem fornecidos ao sistema, resultados que poderão ser obtidos, critérios de qualidade, utilidade e rentabilidade, etc.

auditores;

- Especificação preliminar dos diálogos homem/máquina. Aqui também interessa somente a parte lógica, ou seja, o conteúdo destes diálogos e não sua disposição física;
- Especificação dos relatórios de monitoramento e auditoria.

Aprovação da Especificação Funcional

Objetivo

- Aprovação dos produtos da fase de especificação funcional.

Produtos

- Revisão dos requisitos de qualidade e utilidade;
- Revisão da análise de rentabilidade (análise de mercado);
- Revisão, correção e aprovação dos produtos gerados até agora.

Planejamento Inicial

Objetivo

- Determinar as atividades necessárias para desenvolver o sistema;
- Estabelecer objetivos e cronogramas de desenvolvimento, de consumo de recursos e de entrega de produtos.

Produtos

- Plano de desenvolvimento [Metzger73, Rocha78].

Aprovação do Plano de Desenvolvimento

objetivo

- Revisão da análise de rentabilidade;
- Explicitação dos requisitos de qualidade e utilidade e dos benefícios prioritários;
- Acordo de “congelamento” do Plano de desenvolvimento e da especificação funcional.

Produtos

- Plano de desenvolvimento aprovado;
- Normas técnicas aceitas e divulgadas;

- Especificação funcional revista e aprovada;

– Documento dando o acordo do “congelamento” das especificações, dos requisitos de qualidade e utilidade, dos benefícios esperados, e do plano de desenvolvimento. Este documento deve ser assinado por representantes das diversas entidades envolvidas. Cabe observar que continuam sendo possíveis alterações nos produtos gerados até agora, somente que o controle será formal, requerendo a aprovação de todos os signatários do acordo.

Neste instante existe uma especificação funcional e um plano relativos ao sistema a ser desenvolvido. A especificação determina os resultados que o sistema produzirá, os requisitos de qualidade e utilidade mínimos, e os benefícios esperados. O plano determina como se pretende chegar ao produto final e quantos recursos serão necessários para tal.

Alterações em produtos gerados até este instante tendem a ter repercussão fortemente negativa sobre o desempenho do desenvolvimento. Tais alterações devem, portanto, ser restritas aquelas que sejam amplamente justificadas. Se forem feitas alterações nestes produtos, deve ser revisto o plano, devendo ser dada atenção particular às parcelas de consumo de recursos. Por esta razão é imprescindível um acordo de “congelamento”, mesmo quando as diversas entidades pertencerem a mesma instituição.

O esforço consumido até este ponto é da ordem de 30% do esforço total do desenvolvimento [Brooks75, Boehm73]. Em alguns casos poderá ser mais. Isto é particularmente o caso quando existe pouco conhecimento a respeito do sistema a ser desenvolvido seja por representar uma inovação tecnológica, seja por inexistir suficiente conhecimento do sistema em si. Nestes casos é sugerido incluir-se uma fase de desenvolvimento de protótipos, de modelos de simulação e/ou de condução de experimentos anterior à fase de especificação funcional.

7.2 Fase de Construção

Nesta fase o sistema será construído em conformidade com a especificação funcional. Ou seja, durante esta fase, a especificação funcional é gradativamente transformada, passando pelas especificações lógica e física, culminando com o sistema desenvolvido.

A fase de construção é constituída pelas seguintes sub-fases:

Especificação Lógica

Objetivo

- Identificação dos programas e procedimentos;

- Especificação rigorosa das interfaces entre os diversos processos (procedimentos e programas);
- Diretório de dados, parte física completa;
- Diálogos homem/máquina completamente especificados;
- Manual do usuário completo.

Produtos

- Especificação funcional dos programas e procedimentos elementares. Durante a especificação funcional do sistema, são desenvolvidas especificações para processos do sistema. Estes frequentemente são processos compostos, ou seja, são compostos por diversos programas e procedimentos. Por exemplo, um processo de atualização é, usualmente, composto por vários processos elementares, tais como:

Crítica de dados, correção de dados errados, ordenação, atualização propriamente dita etc. Durante a especificação lógica do sistema, cada um destes processos elementares será especificado. De forma semelhante, alguns processos identificados na especificação funcional teriam solução melhor se reagrupados e depois decompostos segundo princípios diferentes. Isto poderá ser feito, por exemplo, para reduzir o volume de comunicação e/ou de transporte de dados. A tônica da especificação lógica é, então, a identificação e especificação funcional dos programas que compõem o sistema;

– Parte física do diretório de dados. A parte física do diretório de dados descreve a organização física dos dados, sejam estes itens, agregados, registros, arquivos ou conjuntos de arquivos (banco de dados). A parte física descreve critérios de validação, recuperação, manutenção, etc., não descreve no entanto os processos para verificar tais critérios. A parte física descreve detalhadamente a interface entre cada processo elementar. Desta forma é facilitado o desenvolvimento em paralelo dos componentes do sistema automatizado;

- Critério de aceitação por processo;
- Programa de treinamento;
- Primeira versão completa do manual do usuário;

– versão inicial e incompleta do manual do operador. O manual do operador contém informações sobre como operar os diversos equipamentos e como instalar o sistema nestes equipamentos, Fazem parte deste manual as descrições dos diversos procedimentos, requisitos de

previstas, etc. O manual do operador é dirigido ao usuário produtor;

- Plano detalhado da fase de especificação física;
- Diálogos homem/máquina, cotendo as descrições das interações ente o homem e a máquina. Estes diálogos podem ser efetuados via terminais, ou através de formulários e relatórios utilizados e/ou produzidos pelo sistema. Quando for prevista uma linguagem para facilitar a produção de relatórios ad hoc, esta linguagem faz parte da interface homem/máquina (linguagem de consulta);
- Especificação dos controles de segurança operacional;
- Especificação dos controles de monitoramento e auditoria;
- Definição (requisitos, plano, especificação, funcional) do sistema de conversão completa.

Aprovação da Especificação Lógica

Objetivo

- Revisão da análise de rentabilidade;
- Aprovação da definição do Sistema de conversão;
- Aprovação dos produtos da fase de especificação lógica.

Produtos

- Documento de aprovação dos produtos da fase de especificação lógica.

Especificação Física

Objetivo

- Definição dos procedimentos completa;
- Modularização dos programas.

Produtos

- Especificação funcional dos módulos de programa;
- Especificação das mensagens de erro e das operações de contingência;
- Definição dos procedimentos completa;
- Critérios de aceitação por módulo e agregado de módulos;
- Primeira versão completa do manual do operador;

- Especificações lógica e física do sistema de conversão.

Aprovação da Especificação Física

Objetivo

- Revisão da análise de rentabilidade;
- Aprovação dos produtos da fase de especificação física.

Produtos

- Documento de aprovação dos produtos da fase de especificação física.

Codificação

Objetivo

- Produção de programas;
- Produção de dados para testes;
- Execução do plano de teste de integração dos programas;
- Aceitação do sistema de conversão.

Produtos

- Listagens dos módulos de programas;
- Biblioteca de módulos;
- Procedimentos e programas de prevenção e recuperação em caso de acidente durante o desenvolvimento documentados (documentos distribuídos), implantados e em operação (norma de segurança de desenvolvimento);
- Listagens de resultados de testes;
- Biblioteca de dados para testes;
- Mensagens de erro revistas;
- Manual do programador completo, o manual do programador é, na realidade, um conjunto de manuais, contendo os documentos relativos a todos os procedimentos e programas criados durante o desenvolvimento. O manual do programador de desenvolvimento e ao usuário mantenedor;
- Parte de código do diretório de dados completa. O diretório de dados, além de normalizar nomes e propriedades físicas, do dado, normaliza, também, o código para efetuar o acesso, a validação, a atualização, o controle de uso, a proteção, etc.

[Furtado78]. A utilização deste código pelos programas simplifica a programação do sistema e a posterior produção de relatórios e respostas não previstas durante o desenvolvimento;

- Sistema de conversão desenvolvido e aprovado, ou seja, codificação teste de sistema e teste de aceitação do sistema de conversão completos.

Aprovação da Codificação

Objetivo

- Revisão da análise de rentabilidade;
- Aprovação dos resultados da fase de codificação;
- Aprovação do sistema de conversão;
- Acordo de disponibilidade de recursos para a instalação do sistema.

Produtos

- Comprovação da suficiência dos testes de módulos;
- Comprovação da suficiência dos testes de integração dos programas;
- Documento de aprovação dos produtos da fase de codificação;
- Proteção de bibliotecas de programas e dados para teste;
- Documento assegurando a disponibilidade dos recursos necessários para a instalação do sistema.

No final da fase de construção o sistema já existe, não tendo sido aceito ainda para entrar em operação. Além do sistema objeto, já existe e está aprovado o sistema de conversão. O esforço consumido até este ponto no ciclo de vida é da ordem de 60% [Brooks75, Boehm73].

Ao final da fase de construção, o cliente (responsável pelo financiamento) e o usuário se comprometem a tornar disponíveis os recursos necessários para as fases seguintes. Estes recursos são tipicamente pessoas a serem treinadas e recursos computacionais. Estes recursos podem tornar mais dispendiosa as atividades do usuário, uma vez que poderão ser necessárias tanto pessoas para operar o sistema antigo, quanto para receber treinamento e operar o sistema novo.

Caso estas atividades não sejam cuidadosamente planejadas, o caos poderá resultar.

7.3 Fase de Implantação

Nesta fase são conduzidos diversos testes, com o

objetivo de verificar se o sistema efetua o que foi especificado, se estas especificações são satisfatórias e se ele efetivamente atende às necessidades dos diversos usuários. Os testes são conduzidos em diversos níveis. Primeiro é verificado se o sistema efetua o desejado utilizando-se para isto dados criados exclusivamente para teste. Após, o sistema é exercitado com dados reais, porém ainda executa em ambiente controlado, ou seja, instrumentado. Finalmente o sistema passa a ser operado em condições totalmente reais. No final desta última sub-fase o sistema antigo é descontinuado. São seguintes as sub-fases da implantação:

Teste de Sistema

Objetivo

- Confrontar o sistema produto com as especificações;
- Preparação dos arquivos operacionais;
- Revisão da documentação;
- Treinamento dos usuários.

Produtos

- Manual do usuário revisto;
- Manual do operador revisto;
- Manual do programador revisto;
- Listagens de testes e comprovação da suficiência destes;
- Diretório de dados revisto;
- Arquivos contendo dados para teste de sistema para utilização futura;
- Arquivos operacionais criados e corretos (resultados da operação do sistema de conversão);
- Sub-sistema de segurança operacional aprovado e em operação normal;
- Treinamento dos usuários concluído.

Aprovação do Teste de Sistema

Objetivo

- Revisão da análise de rentabilidade;
- Aprovação dos produtos da fase de teste de

Produtos

- Documento de aprovação do teste de sistema;
- Arquivos operacionais aprovados e em manutenção normal.

Teste de Aceitação

Objetivo

- Operação experimental pelo usuário utilizando o sistema ainda instrumentado. O sistema ainda reside na instalação utilizada para o desenvolvimento. Esta não necessariamente é a que será utilizada durante a operação;
- Manutenção dos arquivos operacionais.

Produtos

- Avaliação do sistema pelo usuário;
- Manuais do operador, do programador e do usuário em forma final;
- Porção do diretório de dados relativo a este sistema em forma final. O diretório de dados é um documento que aplica a todos os sistemas da instituição. Quando da produção do diretório, algumas inconsistências com relação a outros sistemas podem ter sido introduzidas. Neste ponto, no entanto, estes defeitos já foram todos sanados e o diretório obedece a todas as normas estabelecidas;
- Documentação de desenvolvimento e manutenção em forma final. Os manuais de sistema, manutenção etc., estão embutidos na documentação produzida durante o desenvolvimento;
- Banco de dados (arquivos) operacional mantido e em atividades;
- Usuário capacitado a utilizar o sistema independentemente da presença de membros das outras entidades.

Aprovação do Teste de Aceitação

Objetivo

- Revisão da análise de rentabilidade;
- Aprovação final da documentação;
- Aprovação do sistema produto.

Produtos

Documento de aprovação da documentação;

- Documento de aprovação do sistema produto;
- Aprovação do conteúdo do banco de dados operacionais.

Teste no Local

Objetivo

- Instalação do sistema;
- Avaliação do sistema e do desenvolvimento do sistema.

Produtos

- Operação experimental do sistema novo em paralelo com o sistema antigo utilizando equipamentos e instalações em que residirá o sistema durante a fase de operação. Tratando-se de software produto esta fase corresponde a instalação e teste do software no sistema do cliente. Esta sub-fase pode portanto ocorrer repetidas vezes, mesmo não existindo alteração no sistema original;
- resultados de operação normais;
- Avaliação do sistema pelo usuário;
- Resumo da história do desenvolvimento do sistema;
- Arquivamento dos produtos intermediários do desenvolvimento.

Aprovação da Operação no Local

- Descontinuação do sistema antigo, ou início do uso do software produto;

Produtos

- Documento autorizando a descontinuação do sistema antigo, ou o uso do software produto;
- Documento de aprovação da avaliação do sistema;
- Documento de aprovação da história do desenvolvimento do sistema.

Neste instante o sistema encontra-se em operação normal. Possivelmente é previsto um período de garantia durante o qual o desenvolvedor deverá estar disponível para corrigir quaisquer erros e dificuldades porventura encontrados.

7.4 Fase de Operação

Durante esta fase o sistema é operado e mantido

normalmente. A manutenção e a operação correm por conta dos grupos produtor e mantenedor da entidade usuário. Possivelmente existirá um período inicial durante o qual o desenvolvedor prestará auxílios em caso de falhas, erros ou dificuldades de operação.

Durante a fase de operação, ou seja, durante a vida útil do sistema, serão percebidas também novas necessidades que poderiam ser satisfeitas através de extensões ao sistema. Estas extensões devem ser encaradas como sendo projetos especiais, devendo ser examinadas da mesma forma como se um novo sistema estivesse sendo proposto.

8 – EPÍLOGO

Este texto tinha por objetivo apresentar uma visão abrangente do problema "desenvolvimento de sistemas automatizados". Procuramos ater-nos a este objetivo sem entrar em detalhes demasiados. O nível de abordagem é justificado, uma vez que visa ao estudo do problema como um todo, evitando soluções de problemas relativos a detalhes das fases de desenvolvimento, o que poderia levar a perda de visão do conjunto.

Certamente remanesce um número elevado de perguntas carecendo de respostas. Estas perguntas podem ser respondidas através de estudos detalhados dos problemas relativos a cada fase ou classe de atividades. Ou seja, deve ter ficado evidente que ainda existe muito por fazer para que consigamos tornar o desenvolvimento de sistemas automatizados realmente em uma indústria com padrões de qualidade, utilidade e rentabilidade adequados. Deve ter ficado claro, também, que necessitamos para isto do desenvolvimento de ferramentas poderosas e econômicas, bem como do desenvolvimento de padrões e métodos dos industriais que comprovadamente contribuam para a melhoria destes índices.

REFERÊNCIAS BIBLIOGRÁFICAS

- [Boehm73] Boehm, B. W.
"Software and its Impact: A Quantitative Assessment";
Datamation 19 (5), maio 1973; pags. 48-59
- [Brooks75] Brooks, F. P.
The Mythical Man-Month; Addison Wesley
Publishing Co.; 1975
- [Brown74] Brown, P. J.
"Programming and Documenting Software
Projects"; ACM Computing Surveys 6 (4), dez
1974; pags. 213-220
- [DeMarco78] DeMarco, T.

Structured Analysis and System Specification;
Prentice Hall, Englewood Cliffs; 1978

[Furtado78] Furtado, A. L.; Staa, A. v.
Diretório de Dados: Utilidade e Especificação;
Monografias em Ciência da Computação nº 11/78,
Departamento de Informática, Pontifícia Universidade
Católica; Rio de Janeiro; jul 1978

[Gane79] Gane, C.; Sarson, T.
Structured Analysis, Tools and Techniques;
Prentice Hall, Englewood Cliffs; 1979

[Leite79] Leite, J. C. S. P.
Contabilidade de Custos no Desenvolvimento de
Software; Dissertação de Mestrado, Departamento
de Informática, Pontifícia Universidade Católica, Rio de
Janeiro; Mar 1979

[Lucena78] Lucena, C. J. P.; Staa, A.
Projeto de Progamas ; SERPRO, Série Cadernos
de Programação, nº 1; Rio de Janeiro, RJ; 1978

[Metzger73] Metzger, P. W.
Managing a Programming Project; Prentice Hall,
Englewood Cliffs, NJ, EEUU; 1973

[Myers75] Myers, G. J.
Reliable Software Through Composite Design;
Petrocelli/charter, New York, NY, EEUU; 1975

[Putnam77a] Putnam, L. H.
"The Software Life Cycle: Evidence and
Foundation"; in [Putnam77b] ed.; 1977;
pags. 1-112

[Putnam77b] Putnam, L. H.; Wolverton, R. W. eds.
Quantitative Management. Software Cost
Estimating; Computer Software and Applications
Software Tutorial (COMPSA C77); LEEE, catalog
nº EHO 129-7; nov 1977

[Rocha78] Rocha, A. R. C.
Plano de Desenvolvimento de Projetos de
Sistemas Automatizados; Dissertação de Mestrado,
Departamento de Informática, Pontifícia Universidade
Católica, Rio de Janeiro; dez 1978

[Ross77] Ross, D. T.
"Structured Analysis (SA): A Language For
Communicating Ideas"; *LEEE Transactions on
Software Engineering SE- 3 (1)*, jan 1977; pags.
16-34

[Staa79] Staa, A. v.
Desenvolvimento Estruturado de Sistemas
Automatizados: Monografias em Ciência da
Computação 9/79; Departamento de Informática,
Pontifícia Universidade Católica, Rio de Janeiro; 1979

[Wolverton77] Wolverton, R. W.
"Management by Objectives,"; em [Putnam77b]
ed.; 1977; pags. 239-256

[Yohe74] Yohe, J. M.
"An Overview of Programming Practices";
ACM Computing Surveys 6 (4); dez 1974;
pags. 221-246.