

**CONTEÚDO/CONTENTS****Artigos/Articles**

- OS SISTEMAS DE INFORMAÇÃO E OS RISCOS E AMEAÇAS ORGANIZACIONAIS ..... 7  
 INFORMATION SYSTEMS AND ORGANIZATIONAL RISKS AND THREATS  
*MAURÍCIO PRATES DE CAMPOS FILHO*
- UMA VISÃO SOBRE ANÁLISE DE MUTANTES E O AMBIENTE DE TESTE PROTEUM ..... 11  
 AN OVERVIEW OF MUTATION ANALYSIS AND OF THE PROYEUM TESTING TOOL  
*JOSÉ CARLOS MALDONADO, MÁRCIO EDUARDO DELAMARO, MARCOS LORDELLO  
 CHAIM E MARIO JINO*
- SISTEMAS DE INFORMAÇÃO E SISTEMAS DE APOIO À DECISÃO ..... 24  
 INFORMATION SYSTEMS AND DECISION SUPPORT SYSTEMS  
*EDUARDO O. C. CHAVES, ORANDI MINA FALSARELLA*
- UMA EXPERIÊNCIA EM PORTABILIDADE DE INTERFACES GRÁFICAS ..... 32  
 AN EXPERIENCE IN GRAPHICAL USER INTERFACES PORTABILITY  
*MIGUEL ARGOLLO JR, SIMONE SCARPELINI*
- PLANEJAMENTO ESTRATÉGICO DE SISTEMAS DE INFORMAÇÕES UTILIZANDO A  
 REENGENHARIA DE PROCESSOS ..... 46  
 STRATEGIC PLANNING FOR INFORMATION SYSTEMS BASED ON PROCESS  
 REENGINEERING  
*JORGE LUIS CORDENONSI*
- GESTAÇÃO DA POLÍTICA NACIONAL DE INFORMÁTICA: DE JK AO GOVERNO  
 FIGUEIREDO ..... 55  
 THE FORMULATION OF BRAZIL'S NATIONAL INFORMATICS POLICY: FROM JK  
 TO FIGUEIREDO'S GOVERNMENT  
*PAULO DE MARTINO JANNUZZI*

**Revista do Instituto de Informática da PUCAMP. -**

Volume 1, n. 1 (1992) - . Campinas:

Semestral

1. Informática-Periódico

CDD 001.61

CDU 681.3

**Revista do Instituto de Informática da PUCAMP**

**Publicação Semestral**

**Editor-Executivo:** Prof. Ricardo Pannain

**Conselho Editorial:**

Profª Angela de M. Engelbrecht - Presidente  
Prof. José Oscar Fontanini de Carvalho - Vice-Presidente  
Dr. Arthur J. Catto (FCTI)  
Drª Beatriz M. Daltrini (UNICAMP)  
Dr. Carlos Mammana (FCTI/UNICAMP)  
Dr. Edmundo R. M. Madeira (UNICAMP)  
Dr. Eduardo O. C. Chaves (UNICAMP/PUCAMP)  
Dr. Geraldo Nonato Telles (UNICAMP/PUCAMP)  
Drª Geraldina Porto Witter (PUCAMP)  
Dr. Heitor Quintella (IBM Brasil)  
Dr. Hilton S. Pinto (UNICAMP)  
Dr. José Carlos Maldonado (USP)  
Dr. José M. da Mata (UFMG)  
Dr. Júlio S. Aude (UFRJ)  
Dr. Luiz F. B. Baptistella (TELEBRAS)  
Dr. Luiz Fernando Soares (PUCRJ)  
Dr. Manuel J. Mendes (UNICAMP/PUCAMP)  
Dr. Marcio Luiz de Andrade Netto (UNICAMP)  
Dr. Mario Jino (UNICAMP)  
Dr. Mario L. Cortes (TELEBRAS/UNICAMP)  
Dr. Maurício Magalhães (UNICAMP)  
Dr. Maurício Prates (UNICAMP/PUCAMP)  
Dr. Nelson J. Parada (UNICAMP/PUCAMP)  
Dr. Saul G. D'Ávila (UNICAMP)  
Drª Vera Sílvia M. Beraquet (PUCAMP)  
Dr. Wanderley L. de Souza (UFPb)

**Conselho Consultivo**

Prof. Ricardo Pannain  
Profª M. Cristina L. F. M. Aranha  
Prof. Dr. Maurício Prates  
Prof. Orandi Mina Falsarella

**Capa**

Máscara de circuito integrado cedida pela FCTI - Fundação Centro Tecnológico para Informática.

**Correspondência:**

A/C:  
Instituto de Informática - PUCAMP  
C. P. 317 - Campus I - Rod. D. Pedro I - Km 136 - CEP: 13020-904 - Campinas - SP - FAX: (0192) 52.8477

A "Revista do Instituto de Informática" tem uma tiragem de 2000 exemplares. É distribuída gratuitamente às Universidades, Centros de Pesquisa, Órgãos Governamentais e Empresas que nos solicitam.

**Composição e Impressão**

Gráfica PUCAMP



PONTIFÍCIA UNIVERSIDADE CATÓLICA DE CAMPINAS

INSTITUTO DE INFORMÁTICA

**REVISTA DO INSTITUTO DE INFORMÁTICA  
DA PUCCAMP**



---

## EDITORIAL

As notícias para este número são otimistas. No final do período letivo de 1994 o Instituto de Informática renovou a área administrativa, após uma eleição para escolha de diretor, vice-diretor, coordenadores de curso, coordenadores de departamento e coordenador de laboratório. Os novos integrantes da direção e das coordenações além da proposta de continuidade ao projeto do Instituto, capacitação dos docentes e realização de pesquisa, apresentaram proposta para melhorar todos os segmentos do Instituto, cursos de especialização; mestrado; cursos da graduação, bolsas de iniciação científica, monitoria, etc. O Conselho Executivo da Revista do Instituto de Informática também foi renovado, bem como o Editor Executivo, que passa a ser o prof. Ricardo Pannain. A nova administração da Revista pretende conseguir, entre outras coisas, a inclusão da revista em bancos de dados internacionais. As contribuições de artigos têm sido expressivas e a distribuição da Revista foi ampliada. Toda a comunidade do Instituto tem apoiado a nova administração e pretende colaborar para a obtenção de resultados cada vez melhores.

Neste número da Revista são apresentados os primeiros resultados do Mestrado em Gerenciamento de Sistemas de Informação, através da publicação de artigos resultantes de dissertações apresentadas em 1994 relacionadas à Sistemas de Apoio à Decisão e à Reengenharia. Ainda ligado às atividades do Mestrado há o artigo Sistemas de Informação e os Riscos e Ameaças Organizacionais cujo autor é o coordenador do Mestrado.

Há, ainda, um artigo que apresenta um estudo sobre um critério de teste para o uso na produção de software, denominado Uma visão sobre Análise de Mutantes e o Ambiente de Teste Proteum.

Apresenta-se, também, um artigo sobre interfaces gráficas, com o título Um exercício em Portabilidade de Interfaces Gráficas.

Finalmente apresenta-se um artigo sobre a política da informática no Brasil, com o título Geração da Política Nacional de Informática: de JK ao Governo Figueiredo.

Agradecemos a confiança e a colaboração de todos aqueles que já participaram da Revista de alguma forma e esperamos ter, cada vez mais, a colaboração dos integrantes da comunidade de informática.

**Profª M. Cristina L. F. M. Aranha**  
Conselho Executivo

---



---

# CONTEÚDO/CONTENTS

## Artigos/Articles

- OS SISTEMAS DE INFORMAÇÃO E OS RISCOS E AMEAÇAS ORGANIZACIONAIS ..... 7  
INFORMATION SYSTEMS AND ORGANIZATIONAL RISKS AND THREATS  
*MAURÍCIO PRATES DE CAMPOS FILHO*
  - UMA VISÃO SOBRE ANÁLISE DE MUTANTES E O AMBIENTE DE TESTE PROTEUM ..... 11  
AN OVERVIEW OF MUTATION ANALYSIS AND OF THE PROTEUM TESTING TOOL  
*JOSÉ CARLOS MALDONADO, MÁRCIO EDUARDO DELAMARO, MARCOS LORDELLO  
CHAIM E MARIO JINO*
  - SISTEMAS DE INFORMAÇÃO E SISTEMAS DE APOIO À DECISÃO ..... 24  
INFORMATION SYSTEMS AND DECISION SUPPORT SYSTEMS  
*EDUARDO O. C. CHAVES, ORANDI MINA FALSARELLA*
  - UMA EXPERIÊNCIA EM PORTABILIDADE DE INTERFACES GRÁFICAS ..... 32  
AN EXERCISE IN GRAPHICAL USER INTERFACES PORTABILITY  
*MIGUEL ARGOLLO JR, SIMONE SCARPELINI*
  - PLANEJAMENTO ESTRATÉGICO DE SISTEMAS DE INFORMAÇÕES UTILIZANDO A  
REENGENHARIA DE PROCESSOS ..... 46  
STRATEGIC PLANNING FOR INFORMATION SYSTEMS BASED ON PROCESS  
REENGINEERING  
*JORGE LUIS CORDENONSI*
  - GESTAÇÃO DA POLÍTICA NACIONAL DE INFORMÁTICA: DE JK AO GOVERNO  
FIGUEIREDO ..... 55  
THE FORMULATION OF BRAZIL'S NATIONAL INFORMATICS POLICY: FROM JK  
TO FIGUEIREDO'S GOVERNMENT  
*PAULO DE MARTINO JANNUZZI*
-



---

# OS SISTEMAS DE INFORMAÇÃO E OS RISCOS E AMEAÇAS ORGANIZACIONAIS

## INFORMATION SYSTEMS AND ORGANIZATIONAL RISKS AND THREATS

Maurício Prates de CAMPOS FILHO\*

### ABSTRACT

The risks and threats related to information systems include technological obsolescence, failure to achieve and/or retain competitive advantage, and vulnerability to theft, vandalism and accidents. Technological obsolescence occurs when an individual's or organization's special capabilities are rendered obsolete by new information technologies. Failure to achieve or retain competitive advantages occurs because there is no guarantee that an attempt to use information systems competitively will succeed. Vulnerability to theft, accident and vandalism exists because today's information systems are far from foolproof.

**KEY WORDS:** Technological obsolescence; competitive advantage; vulnerability; foolproof

### RESUMO

Os riscos e ameaças relacionados aos sistemas de informação das organizações informatizadas, incluem fatores como obsolescência tecnológica, incapacidade para atingir e/ou reter vantagens competitivas, e vulnerabilidade a acidentes, furtos e vandalismo. A obsolescência tecnológica ocorre quando a capacitação individual e/ou coletiva da organização se verifica obsoleta diante de novas e mais adequadas tecnologias de informação (**hardware**, **software**, redes, multimídia, etc.). A incapacidade para se alcançar e se manter vantagens competitivas ocorre quando desaparecem as garantias de um esforço para se utilizar com sucesso e competência os sistemas de informação da organização. A vulnerabilidade a furtos, acidentes e vandalismo existe porque, apesar de todo progresso na área, os sistemas de informação atuais ainda estão longe de serem suficientemente seguros.

**PALAVRAS-CHAVE:** Obsolescência tecnológica; vantagem competitiva; vulnerabilidade; segurança

## 1. INTRODUÇÃO

É muito fácil se verificar nas organizações empresariais, pessoas ou grupos que não se sentem nada confortáveis com os potenciais impactos trazidos pela implantação de sistemas de informação, ou que não conseguem acomodar adequadamente as eventuais mudanças organizacionais decorrentes da inserção de novas tecnologias de informação. Em muitos casos a questão é localizada e individualizada, como no exemplo do empregado que encontra dificuldade em se adaptar a um novo sistema que acaba de ser instalado, mas em muitos casos a questão envolve a organização como um

todo, como no exemplo da empresa que perde gradualmente seu mercado para suas concorrentes com melhor **marketing** ou tecnologia. Muitas vezes a situação de risco e ameaça pode até mesmo envolver empreendimentos empresariais de grande porte, cuja própria razão de existir e de sobreviver fica seriamente ameaçada pelas mudanças tecnológicas.

Na realidade, o uso de sistemas de informação pode também trazer riscos significativos às organizações, como no exemplo de sistemas computadorizados e automatizados, que podem ser vulneráveis a danos sérios causados por ações criminosas como furto, a vandalismos como vírus nos **softwares**, ou até mesmo

(\*) Professor Titular e Coordenador do Mestrado em Informática da Puccamp.

a acidentes como os **bugs** em programas de computador. Até agora, os eventos registrados mostram claramente que todos estes três tipos de vulnerabilidade são

as maiores ameaças à maioria das organizações que utilizam sistemas informatizados, como procura ilustrar a figura seguinte.

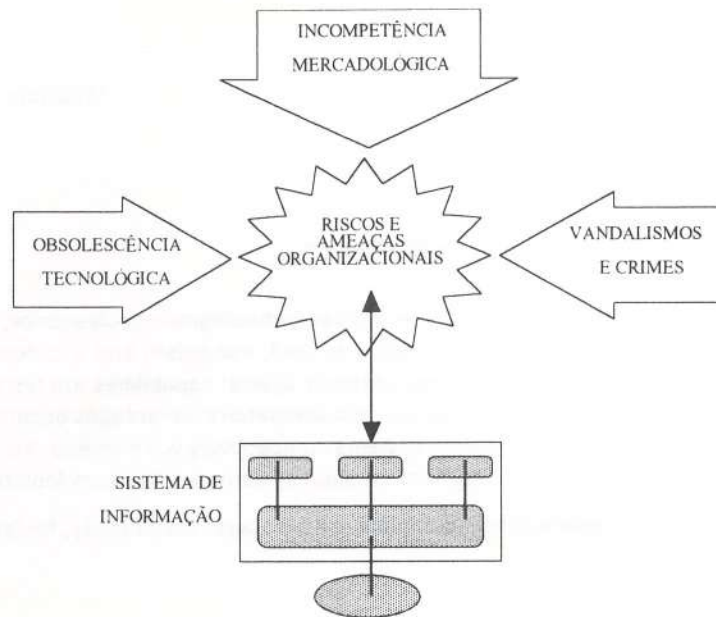


Figura 1 - Fatores desencadeantes de riscos e ameaças organizacionais

## 2. A OBSOLESCÊNCIA TECNOLÓGICA

A obsolescência tecnológica ocorre quando as capacitações específicas da organização ou dos indivíduos que a compõem são encontradas como obsoletas por novas tecnologias ou novas metodologias de produção. Esta situação de desencontro organizacional se verificou por décadas e acontece freqüentemente ainda hoje. Um exemplo clássico na literatura é o do **USPS - United States Postal Services** (Serviços Postais dos Estados Unidos), organização de grande porte que sempre lidou e ainda lida dramaticamente com a ameaça da obsolescência tecnológica.

Na década de 1980, cerca de 80% da entregas de primeira classe feitas pelos correios nos Estados Unidos eram constituídas de correspondência de negócios empresariais, tais como ordens de compra e notas fiscais (1). No final da década, essa grande massa de serviços começou a ser ameaçada pela correspondência eletrônica via telecomunicações, mais eficiente e rápida, cujo custo começou a cair a níveis de viabilidade comercial. Essa situação consiste numa ameaça de longo prazo ao **USPS**, que emprega cerca de 800.000 pessoas, e que tem oferecido serviços baratos e altamente confiáveis a uma massa incalculável de cidadãos americanos. Se

toda a correspondência empresarial vier a ser feita via computador/telefone, o custo da correspondência individual poderá aumentar dramaticamente, afetando aqueles cidadãos que não podem pagar as taxas ainda altas do correio eletrônico.

O **USPS** está longe de ser a primeira organização, na história americana das comunicações, a encarar de perto as ameaças da obsolescência ou das mudanças drásticas resultantes de inovações nos sistemas de informação. Em 1860, foi criada nos Estados Unidos a **Pony Express**, organização empresarial que teve por objetivo a entrega "rápida" de encomendas e correspondências, em lombo de cavalo, entre os estados do Missouri e da Califórnia. Enfrentado os maiores perigos característicos da região e da época, a empresa cresceu rapidamente, e se orgulhava em cruzar a distância em apenas "dez" dias. Entretanto, um ano e meio depois, outra empresa de comunicação, a **Westem Union**, completou uma ligação por telégrafo que cobria uma distancia de três mil quilômetros em segundos ao invés de dias. Apesar da imagem de alta respeitabilidade e confiabilidade, a **Pony Express** encerrou de imediato e melancolicamente suas atividades de negócio, suplantada que foi por uma nova e superior tecnologia de informação (2).

A **Western Union** sobreviveu muito mais que a **Pony Express**, mas teve sérios problemas para manter-se no mercado face a novas tecnologias que foram surgindo no setor das comunicações. Fundada em 1851, e baseada exclusivamente na tecnologia do telégrafo de Morse, a empresa recusou a oferta de Alexander Graham Bell para comprar sua patente do telefone por 100.000 dólares. Os negócios do telégrafo conheceram a derrocada na década de 1960, quando a tecnologia do telefone foi encontrando patamares de custos menores. Mais adiante, os negócios do telex também começaram a ser abandonados, tendo em vista que as tarifas telefônicas ficaram ainda mais reduzidas e as máquinas de Fax se tornaram cada vez mais baratas (3). O resultado foi que, em 1990, a **Western Union** entrou melancolicamente em concordata, após registrar sérios prejuízos em seus negócios.

As histórias dessas três marcantes organizações americanas do setor de comunicações, **US Postal Services**, **Pony Express** e **Western Union**, ilustram com clareza a forma com que os negócios podem ser fortemente ameaçados pelo surgimento de novas tecnologias.

### 3. A INCOMPETÊNCIA MERCADOLÓGICA

Apesar dos inúmeros exemplos de se alcançar vantagens competitivas nas organizações através da adoção de sistemas de informação, não existem garantias de que o uso desses sistemas possa ser bem sucedido. Um exemplo típico é o do fabricante de ferramentas e utensílios comerciais, que vendia a maior parte de sua produção para empreiteiros da construção civil, que compatibilizavam suas especificações técnicas em função das dimensões e capacidades dos utensílios. Na tentativa de auxiliar os empreiteiros e manter a fidelidade da clientela, o fabricante instalou um sistema **mail-in** de consulta, através do qual as especificações eram convertidas em instruções relativas à fiação elétrica, encaamentos, e outras montagens realizadas no local das construções. Para facilitar ainda mais o cliente, o sistema convertia as especificações em pedidos de compra dos utensílios correspondentes com as listagens das respectivas instruções de uso e montagem. Esse sistema de **marketing** por compartilhamento cresceu, mas foi logo revertido quando um fabricante concorrente implantou um sistema ainda melhor, o qual, além das mesmas facilidades, permitia ao cliente-empreiteiro acessar diretamente o computador do fabricante e ordenar os pedidos sem intermediários (4).

Em alguns casos, a tentativa de se usar sistemas de informação competitivamente pode falhar redondamente, em função da elevada rapidez com que surgem novas tecnologias. Um exemplo é o que aconteceu com o serviço **ZapMail** da **Federal Express** nos Estados Unidos, empresa bem sucedida que foi fundada na perspectiva da entrega rápida de correspondência, em prazos inferiores a 24 horas. O serviço **ZapMail** foi desenvolvido para se reduzir ainda mais estes prazos. Com a introdução de máquinas de Fax, o serviço utilizava o sistema de tomada-e-entrega já existente na empresa e acessava um satélite alugado para a transmissão da documentação de um ponto a outro. O serviço, inicialmente bem sucedido, acabou sendo abandonado em 1986 após amargar um prejuízo de 160 milhões de dólares, já que a **Federal Express** subestimou a taxa de desenvolvimento e de adoção da tecnologia de Fax, que tornou as máquinas em equipamentos rotineiros nos escritórios (5).

Mesmo quando a vantagem competitiva é alcançada, muitas vezes permanece apenas temporariamente, em função das tecnologias competitivas emergentes ou dos concorrentes que desenvolvem sistemas de informação com igual ou melhor desempenho. O ciclo competitivo típico se inicia quando um ou mais inovadores alcançam vantagens temporárias através do uso mais adequado dos sistemas de informação. Em seqüência, os competidores percebem as suas desvantagens, e partem para a aquisição de sistemas similares ou mais eficientes. Eventualmente, a inovação competitiva pode tornar-se um requisito novo para se realizar os negócios. Mesmo quando o novo sistema requer investimentos muito elevados, os competidores podem juntar-se para compartilhar os custos operacionais e de desenvolvimento. Grandes empresas podem juntar-se em consórcio, como fizeram os oito maiores bancos de Nova Iorque, quando viram o **Citibank** aumentar sua clientela e seus lucros devido a seu novo sistema de informação bancária. O consórcio acabou por associar-se a uma **softwarehouse** para o desenvolvimento e comercialização dos novos sistemas.

A natureza freqüentemente temporária das vantagens competitivas obtidas pelos sistemas de informação não deve significar o seu abandono como uma estratégia definitiva. Muito ao contrário, a natureza de avanço continuado dos sistemas de informação na liderança dos negócios, pode prover uma excepcional oportunidade para que a organização se mantenha a frente dos negócios. Na medida em que os competidores e concorrentes tentem se atualizar com novos sistemas de informação, os que já estão à frente podem mais rapidamente suplantá-los na corrida aos melhores negócios e mercados.

#### 4. A VULNERABILIDADE DOS SISTEMAS

Juntamente com seus preciosos benefícios, os sistemas automatizados trazem também perigos e vulnerabilidades. É preciso ter sempre em mente que todo sistema computadorizado apresenta falhas. Algumas delas são resultantes de erros nos projetos de **hardware**, ao passo que outras são decorrentes de erros no **software**, ou **bugs**.

Mesmo uma pequena falha pode causar colapsos catastróficos no sistema. Um exemplo dramático é o de um sinal de menos (**minus**) incorretamente colocado no **software** que computava a trajetória da primeira descida de um veículo na lua. Este **bug** faria com que o **software** comandasse repulsão do veículo, ao invés de atração, quando o mesmo tentasse alunar (6). Não tivesse esta falha sido descoberta cinco dias antes da missão, certamente os astronautas teriam sido vítimas de um acidente mortal, e a missão da **NASA** fracassaria de modo melancólico, tudo por causa de uma falha ínfima que só pode ser descoberta através de testes muito acurados. Pode-se imaginar quanto deve ser freqüente a presença de **bugs** nos sistemas de informação de uma organização informatizada, provocados por programadores improvisados ou inexperientes.

Danos sérios também podem ocorrer quando os sistemas são utilizados para atividades criminosas. São inúmeros os exemplos que aparecem na mídia sobre computadores piratas, vírus devastadores, falsificação eletrônica e transferências não-autorizadas de milhões de dólares para o exterior. As perdas anuais devido a crimes assistidos por computador chegam a centenas de milhões de dólares somente nos Estados Unidos. Um recente relatório da **National Academy of Sciences** (Academia de Ciências dos Estados Unidos) frisava a dependência crescente pelos computadores na sociedade, e citava a possibilidade futura de uma tentativa sistemática de se subverter sistemas computacionais críticos (7).

#### 5. CONCLUSÃO

O gerenciamento de sistemas de informação deve sempre levar em conta os três fatores principais de riscos e ameaças às modernas organizações informatizadas, quais sejam, a obsolescência tecnológica, a incompetência em alcançar e manter vantagens competitivas, e a vulnerabilidade dos sistemas. É sempre útil ter-se em mente que nenhuma organização, cujas atividades residem em sistemas computadorizados, pode garantir que os mesmos são projetados e gerenciados de forma adequada e segura.

#### REFERÊNCIAS BIBLIOGRÁFICAS

- (1) DIZARD Jr., Wilson P. **The Coming of Information Age**. Longman, N. York, 1982
- (2) PENZIAS, Arno. **Ideas and Information: Managing in a High-Tech World**. W. W. Norton & Co., N. York, 1989
- (3) SUTOR, Ruthanne. **Lost Horizons: Missed Opportunities Have Nearly Wrecked Venerable Western Union**. Financial World, Apr. 18, 1989, p. 60
- (4) VITALE, Michael R. **The Growing Risks of Information Systems Succes**, MIS Quarterly, Dec. 1986, p. 327
- (5) DAVIES, Stanley. **Future Peffect**. Addison-Wesley, Reading-MA, 1987
- (6) OLSON, Steve. **The Sage of Software**. Science 84, Jan.-Feb. 1984, p. 75
- (7) NATIONAL RESEARCH COUNCIL. **Computers at Risk Safe Computing in the Information Age**. Washington-DC, National Academy Press, 1991

---

# UMA VISÃO SOBRE ANÁLISE DE MUTANTES E O AMBIENTE DE TESTE PROTEUM

## AN OVERVIEW OF MUTATION ANALYSIS AND OF THE PROTEUM TESTING TOOL

Prof. Dr. José Carlos MALDONADO\*  
Márcio Eduardo DELAMARO\*\*  
Marcos Lordello CHAIM\*\*\*  
Prof. Dr. Mario JINO\*\*\*\*

### ABSTRACT

With the advance in hardware technology, Mutation Analysis - an error based testing criterion - has been recently investigated by many researchers, and evidences have been found that it is an attractive and practical testing criterion for software production. The necessity and relevance of testing tools are widely recognized and efforts have been carried out for developing tools to support application of this criterion.

In this paper, Mutation Analysis, as well as related work, is reviewed, aiming at specifying the main aspects of a multilanguage testing tool, named Proteum (**Program Testing Using Mutants**), which supports application of Mutation Analysis criterion for testing C programs. In addition to features usually available in similar tools, Proteum offers facilities for conducting experimental studies. It has been used in some empirical studies; a brief view of results obtained is given.

**KEY WORDS:** error based testing, mutation analysis, testing criterion, testing tool.

### RESUMO

Com o avanço da tecnologia de hardware, a Análise de Mutantes - um dos critérios de teste baseados em erros - tem sido mais recentemente investigada por diversos pesquisadores e se mostrado um critério de teste atrativo e factível para o uso na produção de software. A relevância e necessidade de ferramentas de teste são amplamente reconhecidas pela comunidade e esforços têm sido feitos para o desenvolvimento de ferramentas de apoio a este critério.

A Análise de Mutantes, bem como os principais trabalhos relacionados com este critério são revistos, procurando fornecer subsídios para evidenciar a necessidade de implementação de uma ferramenta de teste chamada Proteum (**Program Testing Using Mutants**). Essa ferramenta apóia a aplicação do critério Análise de Mutantes para programas C e, além de características comuns a outras ferramentas de teste, incorpora algumas características próprias como facilidades para a realização de experimentos empíricos. O Proteum foi utilizado em alguns experimentos empíricos cujos resultados são sucintamente descritos.

**PALAVRAS-CHAVE:** teste baseado em erros, análise de mutantes, critério de teste, ferramenta de teste.

---

(\*) Professor do Departamento de Ciências da Computação e Estatística do Instituto de Ciências Matemáticas de São Carlos - ICMSC - USP.

(\*\*) Aluno de Doutorado do Instituto de Física de São Carlos - IFSC - USP.

(\*\*\*) Pesquisador do Centro Nacional de Pesquisa Tecnológica em Informática para Agricultura da Empresa Brasileira de Pesquisa Agropecuária - EMBRAPA.

(\*\*\*\*) Professor do Departamento de Engenharia de Computação e Automação Industrial da Faculdade de Engenharia Elétrica - FEE - UNICAMP.

## 1 - INTRODUÇÃO

A Engenharia de Software constitui-se de um conjunto de métodos e técnicas que procuram dar ao desenvolvimento de software uma abordagem de engenharia de produto que requer planejamento, análise, projeto, implementação, teste e manutenção. Seu principal objetivo é produzir software de alta qualidade e com um custo baixo [PRE92].

Durante todo o ciclo de vida do software, paralelamente à sua construção, um conjunto de atividades denominadas de **Garantia de Qualidade de Software** é conduzido com o objetivo de assegurar que tanto o processo de desenvolvimento quanto o produto final atinjam níveis de qualidade especificados. Dentre essas atividades, a atividade de teste de software é uma das técnicas de validação mais utilizadas. A atividade de teste pode ser descrita essencialmente como: observação do comportamento do programa através de sua execução sobre um conjunto de dados.

A própria atividade de teste está sujeita a erros e deve ser submetida ao controle de qualidade, através da aplicação de técnicas de teste. Ferramentas de teste também são meios importantes para procurar níveis de qualidade satisfatórios na atividade de teste. Além disso, a disponibilidade de ferramentas que apoiem a aplicação de técnicas de teste permite que estudos sejam conduzidos com o intuito de avaliar e revelar características importantes das técnicas de teste e descobrir como elas se inter-relacionam, onde elas se sobrepõem e como elas podem se complementar.

Este artigo descreve um ambiente de teste denominado Proteum (**Program Testing Using Mutants**) que apóia a aplicação da **Análise de Mutantes**, um critério que se enquadra dentro da técnica de teste baseada em erros.

A Seção 2 apresenta uma revisão sobre técnicas e critérios de teste e a Seção 3 trata com mais detalhes o critério Análise de Mutantes. Na Seção 4 é descrita, através de um exemplo, a forma de aplicação deste critério dentro do ambiente de teste Proteum. A Seção 5 descreve alguns experimentos realizados utilizando-se a ferramenta e a Seção 6 traz as conclusões e trabalhos futuros.

## 2 - TESTE DE SOFTWARE

O teste de software, segundo Pressman [PRE92], é um elemento crítico para a garantia da qualidade de software e representa a última revisão da especificação,

projeto e codificação. Apesar das limitações características da atividade de teste (em geral, não se consegue, através de teste, mostrar que um programa está correto), sua aplicação de maneira sistematizada e bem organizada pode conferir ao software algumas características mínimas que são importantes no estabelecimento da qualidade do produto e relevantes para o seu processo de evolução.

A atividade de teste não pode ser conduzida na força bruta. Torna-se necessária a aplicação de técnicas que dêem indicações de como testar o software, quando parar o teste e que forneçam uma medida do nível de confiança e qualidade alcançados com os testes realizados. Duas questões são importantes:

*“Como os dados de teste devem ser selecionados?” e “Como se pode dizer que um programa foi testado suficientemente?”*

Assim, definem-se: **método de seleção de casos de teste**, como um procedimento para escolher casos de teste; e, **critério de adequação de casos de teste**, como um predicado usado para avaliar casos de teste. Em geral, um critério de adequação é um dos itens utilizados para decidir o encerramento das atividades de teste [MAL91].

Basicamente, pode-se classificar um critério dentro de três técnicas de teste: **funcional, estrutural e baseada em erros**. O que diferencia estas três técnicas é a origem da informação usada para estabelecer os requisitos para avaliar ou construir casos de teste.

A técnica funcional de teste, também chamada de “teste caixa preta”, é usada principalmente na construção de casos de teste; utiliza essencialmente a especificação funcional do programa para escolher os casos de teste a serem empregados, não considerando detalhes de uma particular implementação. Como exemplos de critérios funcionais podem ser citados: Grafo de Causa-efeito, Particionamento em Classes de Equivalência e Análise de Valor Limite [PRE92].

Na técnica estrutural, às vezes chamada de “teste caixa branca”, os aspectos de implementação são importantes na escolha dos casos de teste. Em geral, a técnica estrutural utiliza uma representação para programas conhecida como **grafo de fluxo de controle** ou **grafo de programa**; os critérios estruturais caracterizam-se através da determinação de quais componentes do grafo de programa devem ser exercitados pela aplicação dos casos de teste. Critérios estruturais podem basear-se em diferentes tipos de estruturas para determinar quais partes do programa têm sua execução requerida.

**Critérios baseados no fluxo de controle** usam apenas características do fluxo de execução, como comandos e desvios, para derivar os elementos requeridos. **Critérios baseados no fluxo de dados** exploram ainda associações entre pontos do programa onde atribui-se valor a uma variável e pontos do programa onde este valor é utilizado; com base nestas associações são determinados os elementos a serem exercitados. **Critérios baseados na complexidade** utilizam informação sobre a complexidade do programa para determinar os requisitos de teste [PRE92].

Os critérios que se enquadram na técnica de teste baseada em erros utilizam informação sobre erros mais freqüentes no processo de desenvolvimento de software e sobre tipos específicos de erros que se deseja localizar [DEM91]. Dentre estes critérios está a Análise de Mutantes, descrita na próxima seção.

### 3 - CRITÉRIO ANÁLISE DE MUTANTES

A Análise de Mutantes [DEM91, DEL93a] foi proposta por pesquisadores da Universidade YALE e do Georgia Institute of Technology no final da década de 70. Pode-se descrever a aplicação do critério da seguinte maneira: o testador fornece um programa **P** a ser testado e um conjunto de casos de teste **T** cuja qualidade deseja-se avaliar. O programa **P** é executado sobre **T**; se apresentar resultados incorretos, foi revelado um erro e o teste termina. Caso contrário, o programa pode conter erros que o conjunto **T** não conseguiu revelar.

São criados então programas  $P_1, P_2, \dots, P_k$  que são **mutantes** de **P** e que diferem de **P** apenas na existência de erros simples. Por exemplo, se **P** contém a expressão  $A < B$ , um dos mutantes  $P_i$  irá ter no lugar desta, a expressão  $A \leq B$ . Em seguida, cada um dos mutantes é executado sobre **T**. Se um mutante  $P_i$  apresenta resultados diferentes de **P** diz-se que esse mutante está **morto**; nesse caso, **T** conseguiu identificar o "erro" no mutante ou, mais precisamente, conseguiu revelar a diferença entre **P** e  $P_i$ . Por outro lado, se  $P_i$  apresenta os mesmos resultados para todos os casos de teste de **T**, diz-se que ele continua **vivo**. Isto pode ocorrer por dois motivos: ou porque **T** não contém casos de teste capazes de distinguir  $P_i$  de **P** ou porque ambos os programas, apesar de sintaticamente diferentes, são funcionalmente equivalentes. No primeiro caso, novos casos de teste podem ser adicionados a **T** para "matar" o mutante. Já no

caso de mutantes equivalentes, nenhum caso de teste será capaz de distingui-los, pois seus resultados serão sempre iguais aos resultados de **P**.

O objetivo é achar um conjunto de casos de teste que consiga matar o maior número possível de mutantes não equivalentes. Em outras palavras, a adequação de **T** é medida pela sua capacidade de "matar mutantes". Através do **escore de mutação** tem-se uma medida objetiva da adequação de **T**:

$$ms(T) = \frac{\# \text{ mutantes mortos por } T}{\# \text{ mutantes não equivalentes a } P}$$

Alguns dos pontos críticos na aplicação prática do critério Análise de Mutantes são sua complexidade e seu custo de aplicação, em termos de tempo. Mesmo para programas simples, pode ser gerado um número muito grande de mutantes, o que já descarta a possibilidade de aplicar-se o critério manualmente. Em muitos casos, mesmo com o auxílio de uma ferramenta de teste, tornam-se necessárias estratégias para minimizar o tempo de aplicação do critério. Algumas dessas estratégias podem ser encontradas na literatura e, em geral, procuram otimizar o custo de aplicação do critério, diminuindo de forma criteriosa o número de mutantes [MAR90, MAT93, WON94] ou utilizando arquiteturas de alto desempenho na condução do teste [MAT88, KRA91, CHO89].

### 4 - AMBIENTE PROTEUM E A APLICAÇÃO DA ANÁLISE DE MUTANTES

O Proteum [DEL93b, DEL94] é um ambiente de teste, implementado em plataforma de estações de trabalho SUN, que tem como objetivo apoiar a aplicação do critério Análise de Mutantes. Além disto, incorpora características que auxiliam na condução de experimentos empíricos, permitindo que avaliem-se as vantagens da aplicação do critério e que sejam realizados estudos comparativos com outros critérios.

Na definição do Proteum procurou-se levantar características que permitissem alcançar esses dois objetivos principais, além de identificar em outras ferramentas de teste existentes, aspectos e funcionalidades importantes para ambientes deste tipo.

Uma importante característica do Proteum é ser uma ferramenta multilinguagem. Com algum esforço de

programação e alteração de tabelas, principalmente as tabelas que dirigem a análise sintática, pode-se configurar a ferramenta para tratar diferentes linguagens alvo. Inicialmente o ambiente está configurado para tratar de programas escritos em linguagem C.

#### 4.1- Sessões de Teste

Para realizar o teste de um programa P, utilizando como critério a Análise de Mutantes, um conjunto essencial de tarefas deve ser executado:

- . definição do conjunto de casos de teste T;
- . execução do programa P utilizando T;
- . geração de mutantes;
- . execução dos mutantes;
- . comparação com os resultados de P;
- . análise dos mutantes vivos; e
- . cálculo do escore de mutação.

Para permitir que essas tarefas sejam realizadas, algumas automaticamente e algumas com a intervenção do testador, a ferramenta Proteum mantém uma base de dados que contém informação sobre os casos de teste a serem utilizados e sobre os mutantes gerados e seus

estados atuais. Inicialmente, para testar um programa P, o usuário deve criar um teste para P, fornecendo à ferramenta dados pertinentes como: nome dos arquivos do programa fonte e do programa executável, diretório onde estão estes arquivos, etc. Por exemplo, para testar o programa da Figura 1 que está no arquivo **myprog.c**, o usuário deve selecionar no menu principal do Proteum o botão **Program Test** e em seguida a opção **New** (Figura 2).

```
#include <stdio.h>

main()
{
  int    a, b;

  scanf("%d%d", &a, &b);
  if (a > b)
    printf("\n%d eh maior que %d", a, b);
  else
  if (a < b)
    printf("\n%d eh menor que %d", a, b);
  else
    printf("\n%d eh maior que %d", a, b);
  return 0;
}
```

Figura 1 - Programa myprog.c

Feito isso, o usuário recebe um painel onde deve fornecer as características do teste que deseja criar (Figura 3).



Figura 2 - Trabalhando com Sessões de Teste

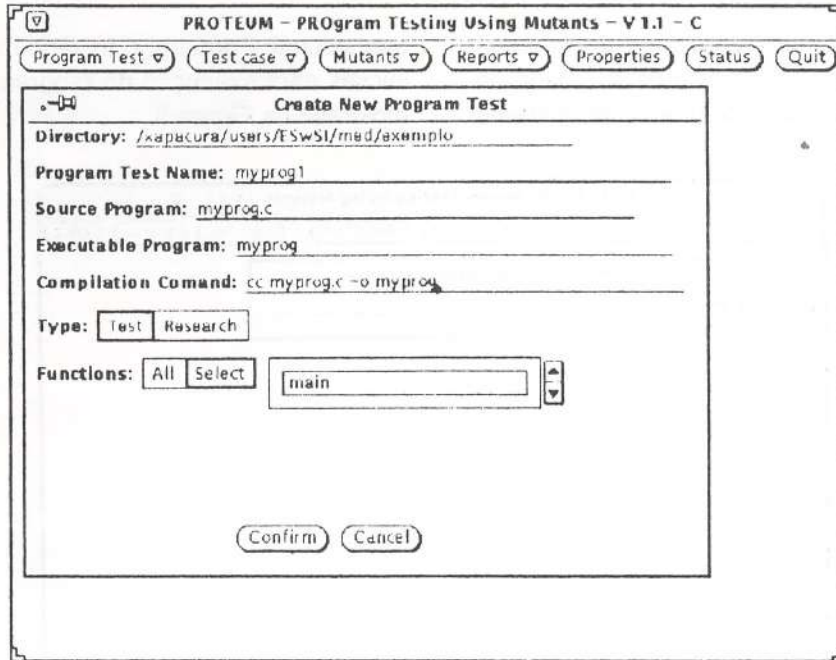


Figura 3 - Criando o Teste de um Programa

A operação do Proteum é dirigida por **sessões de teste**, ou seja, uma vez criado o teste de um programa, o testador pode interromper sua execução em qualquer ponto e futuramente retomá-la do ponto em que foi abandonada através da opção **Load**. Através da opção **Save** o testador garante que todas operações realizadas durante a sessão de teste serão realmente efetivadas e salvas para que a sessão possa ser retomada futuramente.

#### 4.2 Casos de Teste

A definição do conjunto de casos de teste T é feita através de algumas operações, mostradas na Figura 4, que permitem ao testador incluir e excluir casos de teste ou determinar quais devem ser efetivamente avaliados. Embora seja possível apoiar de forma automatizada a geração de casos de teste [DEM91], o Proteum não o faz, ficando esta tarefa por conta do testador.

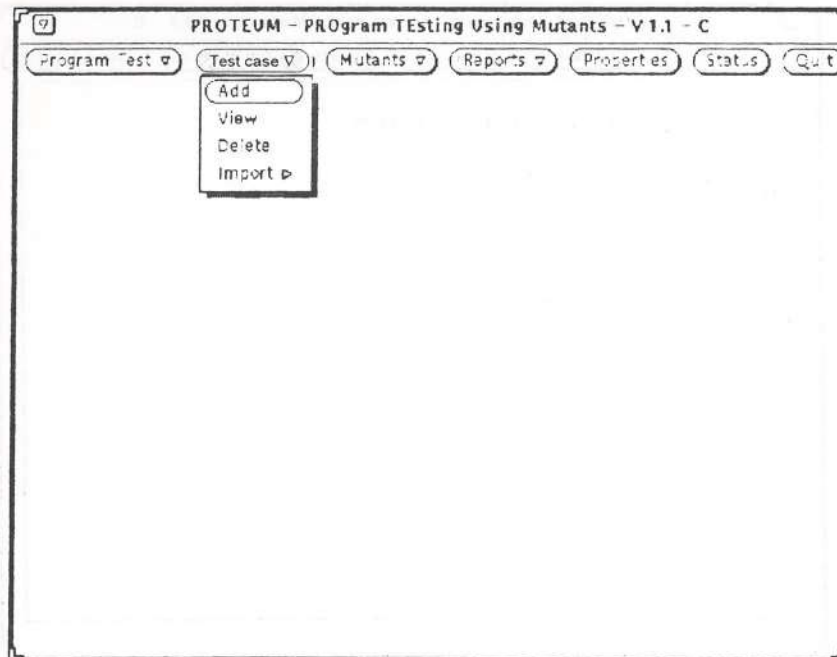


Figura 4 - Trabalhando com Casos de Teste

A inclusão de casos de teste é feita interativamente, de maneira simples, como se fosse uma chamada ao programa **P** através de um comando do sistema

operacional. Primeiro, o usuário fornece os parâmetros iniciais para execução do programa, através do painel mostrado na Figura 5.

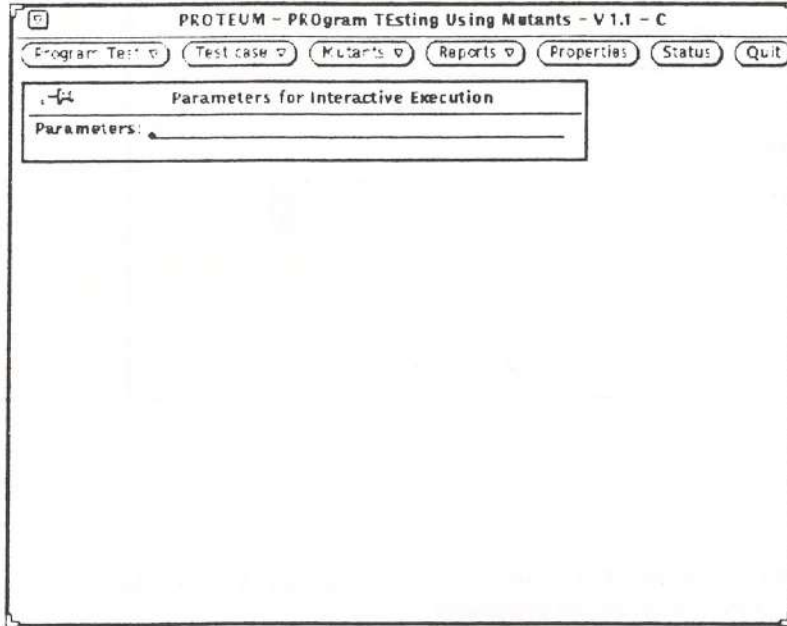


Figura 5 - Fornecendo Parâmetros para um Caso de Teste

Em seguida, inicia-se a execução do programa numa janela que simula um terminal, onde o usuário interage com o programa. Por exemplo, no caso do programa **myprog**, o programa aguarda a digitação dos dois número a serem comparados; é a situação que exibe a Figura 6.

Após fornecer todos os dados, ao final da execução, o usuário pode avaliar se o resultado apresentado é correto ou não, escolhendo entre incluir ou não o caso de teste. Para isso, o Proteum pede ao usuário a confirmação

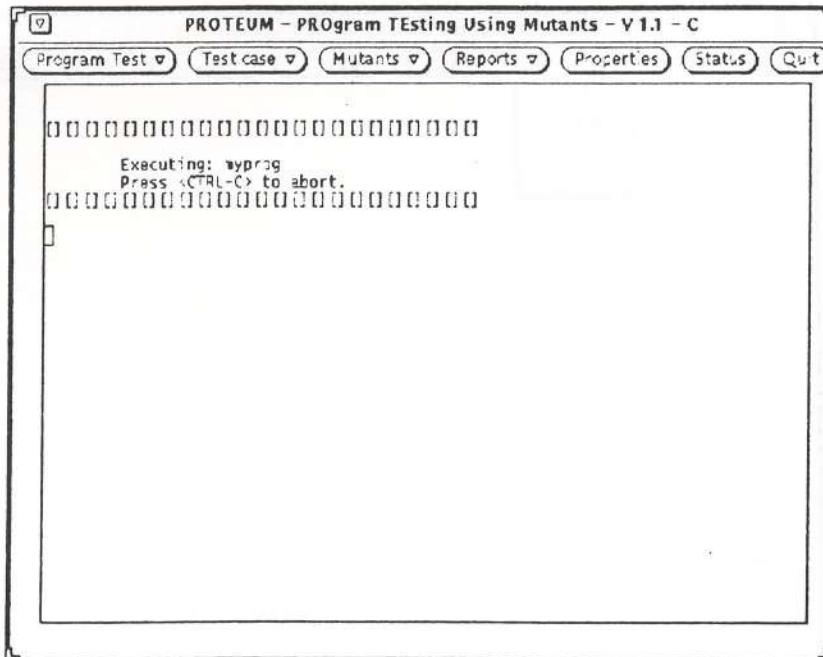


Figura 6 - Incluindo um Caso de Teste

ção, como mostra a Figura 7. Desta maneira o usuário tem a oportunidade de verificar se o resultado apresentado por P para o caso de teste fornecido é o esperado ou se um erro foi revelado.

Os dados fornecidos pelo usuário e os resultados apresentados na tela pelo programa em teste são armazenados pelo Proteum, caracterizando os casos de teste a serem empregados no teste corrente.

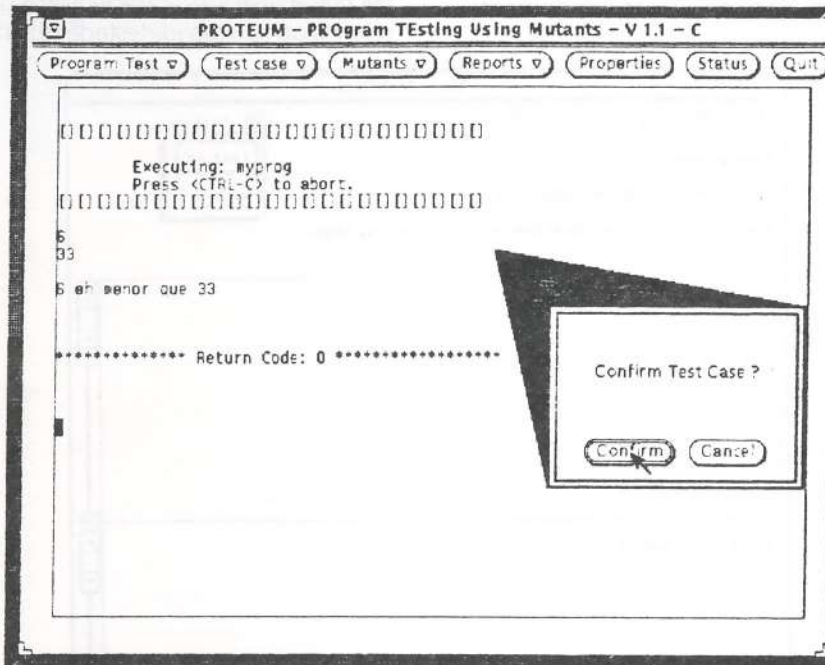


Figura 7 - Confirmando um Caso de Teste

Podem ser incluídos também casos de teste através da importação a partir de outras sessões de teste do Proteum ou da POKE-TOOL, ferramenta de teste que apóia a aplicação de critérios de teste baseados em fluxo de dados [CHA91] ou ainda a partir de arquivos ASCII convencionais.

A exclusão de casos de teste pode ser realizada física ou logicamente. Para eliminar fisicamente, através da opção **Delete**, o testador fornece os números dos casos de teste a serem eliminados (Figura 8), que então deixarão de pertencer ao conjunto T.

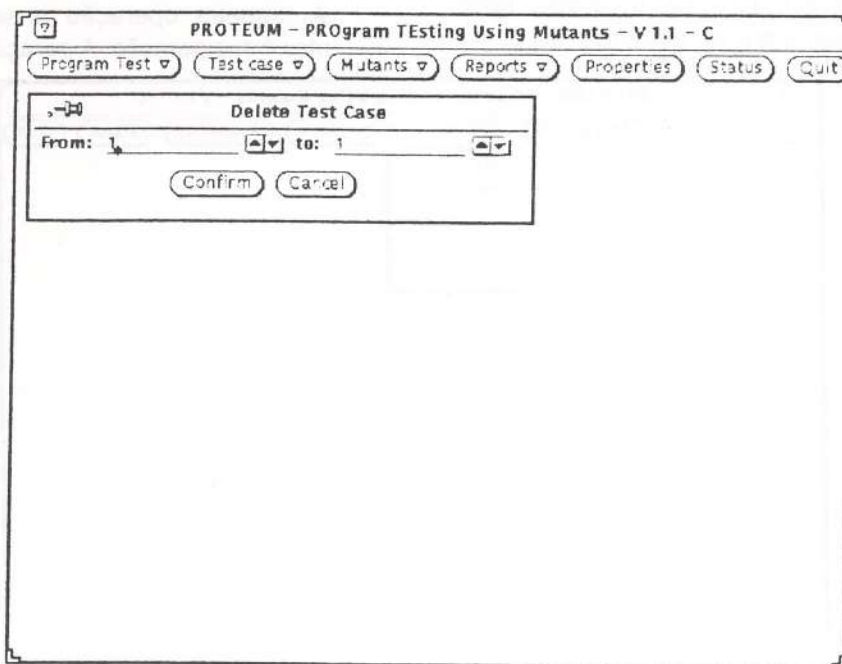


Figura 8 - Eliminando Casos de Teste

Tem-se também a possibilidade de desabilitar/habilitar, ou seja, excluir/reincluir logicamente casos de teste. Um caso de teste desabilitado continua fisicamente presente no conjunto T mas não logicamente, não sendo utilizado na execução dos mutantes e, conseqüentemente, na avaliação de T.

Para habilitar/desabilitar casos de teste, o testador deve selecionar a opção View do menu de casos de teste. Pode então selecionar o número do caso de teste a ser visualizado, através de um painel como o mostrado na Figura 9. Neste painel, que apresenta as características de um caso de teste, é possível indicarse o caso de teste deve ser habilitado ou desabilitado.

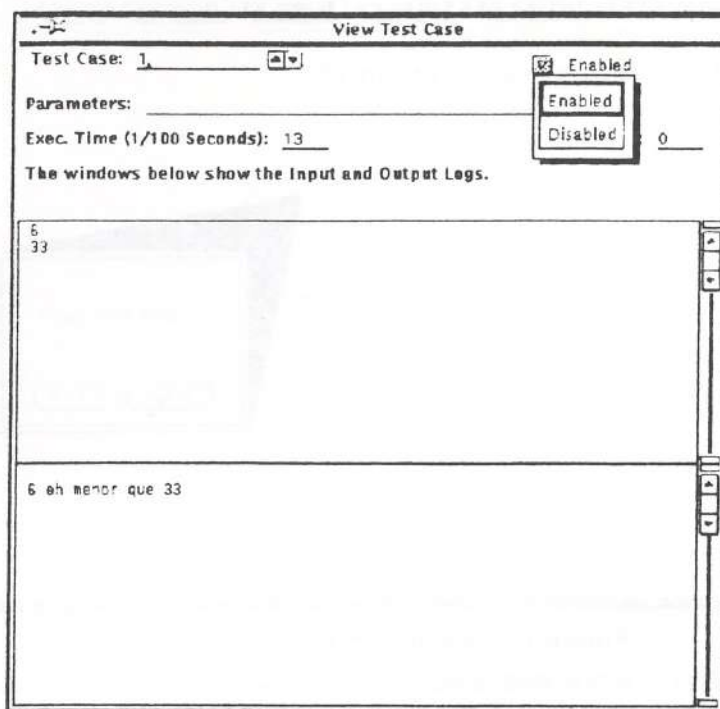


Figura 9 - Visualizando um Caso de Teste

### 4.3 - Mutantes

Conforme mostra a Figura 10, o usuário pode realizar cinco operações sobre os mutantes de P. As

principais são: gerar (**Generate**), visualizar (**View**) e executar (**Execute**) os mutantes.

A primeira operação a ser efetuada sobre os mutantes é sua geração. A geração dos mutantes de P é

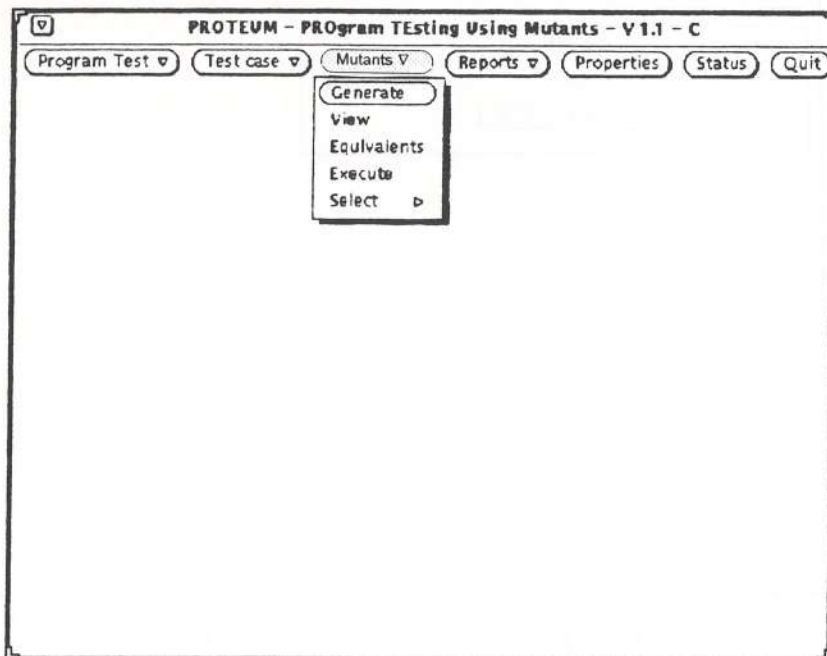


Figura 10 - Trabalhando com Mutantes

feita automaticamente. Cabe ao testador apenas selecionar quais os tipos de mutantes que devem ser gerados. Para determinar as mutações a serem feitas em P foi definido um conjunto de **operadores de mutação** [AGR89]; cada operador de mutação determina um tipo de alteração sintática que deve ser feita em P para criar os mutantes, procurando modelar uma classe específica de erros ou garantir características mínimas de qualidade ao conjunto T (por exemplo, a cobertura de todos os ramos do programa).

No Proteum foram implementados 71 operadores de mutação, divididos em quatro classes: muta-

ções de comandos; de variáveis; de constantes; e de operadores. Para gerar os mutantes o testador pode selecionar, por classes ou individualmente, para cada operador de mutação, uma porcentagem do número de mutantes a serem gerados (Figura 11). Com isso implementou-se uma estratégia de minimização do número de mutantes com o objetivo de diminuir o custo de aplicação do critério. Experimentos têm mostrado que mesmo utilizando uma pequena porcentagem dos mutantes ou apenas alguns dos operadores de mutação, consegue-se obter bons casos de teste [MAT93].

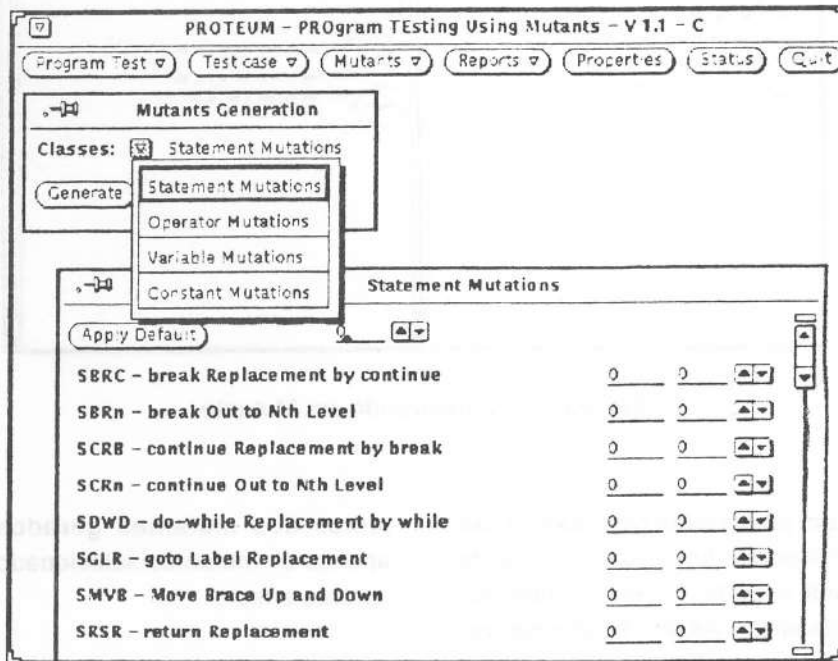


Figura 11 - Gerando Mutantes

O segundo passo para a avaliação do conjunto de casos de teste é a execução dos mutantes com este conjunto. A execução dos mutantes e comparação dos resultados também é feita de forma automática, sem intervenção do usuário. Um ponto importante a ser destacado é que ao efetuar a operação de execução dos mutantes, tem-se refletida a qualidade do conjunto de casos de teste existente no momento da execução. Isso permite que casos de teste sejam incluídos, excluídos, habilitados ou desabilitados e, ainda assim, mantenha-se a coerência entre o conjunto T e o estado dos mutantes. Por exemplo, considerando-se um mutante M que foi morto unicamente pelo caso de teste N que, em seguida, é desabilitado. Ao efetuar-se novamente a operação de execução dos mutantes, M deve ser "resuscitado" pois o caso de teste que o matou não pertence

mais ao conjunto de casos de teste (pelo menos logicamente).

Tradicionalmente, ao executar-se um mutante com um caso de teste que o mate, esse mutante deve ser descartado, sem que seja necessária sua execução com os demais casos de teste. No Proteum o usuário pode determinar um outro comportamento: aplicar sobre cada mutante todos os casos de teste, permitindo observar-se quais casos são eficientes para matar cada mutante. Este procedimento propicia o levantamento de dados importantes para condução de estudos empíricos.

A tarefa de analisar os mutantes vivos consiste em procurar identificar quais mutantes são equivalentes e quais podem ser mortos; esta tarefa, no Proteum, é feita totalmente pelo testador. No primeiro caso, os mutantes

podem ser marcados como equivalentes. Ao marcar um mutante como equivalente, o usuário altera o valor calculado para o escore de mutação, uma vez que o número de mutantes equivalentes é usado no cálculo. Já para os mutantes vivos não equivalentes, o testador pode construir novos casos de teste para eliminá-los e

adicionar estes casos de teste a T. A análise dos mutantes é feita através da operação **View** do menu de mutantes. Assim como é feito para os casos de teste, o usuário pode visualizar um mutante fornecendo seu número ou "navegando" seqüencialmente através do conjunto de mutantes gerados, como mostra a Figura 12.

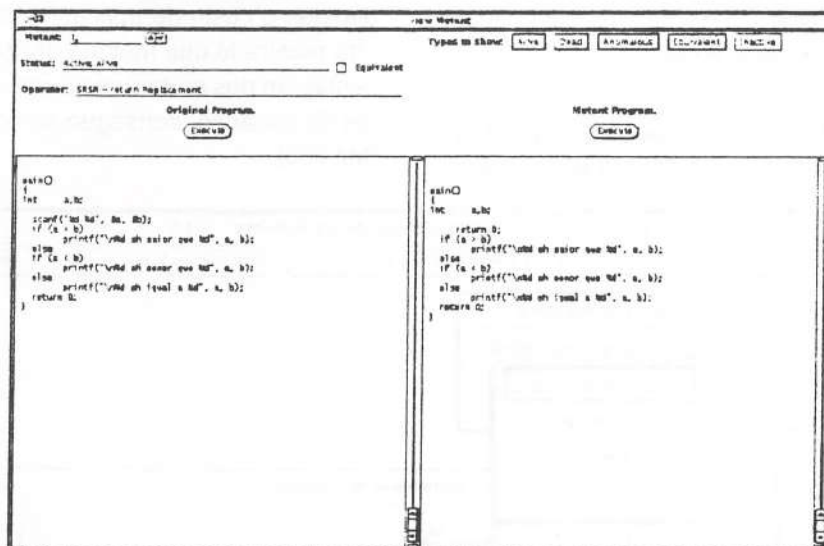


Figura 12 - Visualizando um Mutante

A Figura 12 exibe um painel onde os mutantes são apresentados. No canto superior direito do painel existe uma lista em que o usuário pode escolher o tipo dos mutantes que deseja visualizar. Assim, se desejar ver apenas os mutantes que continuam vivos, o usuário deve "desligar" os botões **Dead**, **Equivalent**, **Anomalous** (mutantes anômalos são aqueles que apresentam erros sintáticos, introduzidos pela mutação efetuada em P, e que por isso não podem ser executados; eles são desprezados no cálculo do escore de mutação). As janelas na base do painel mostram o programa original e o mutante. Sobre cada uma delas existem botões rotulados **Execute**; ao acioná-los o usuário inicia a execução interativa do programa correspondente podendo desta maneira "experimentalmente" casos de teste para tentar matar o mutante, antes de incluí-los.

A opção **Equivalent** do menu de mutantes é utilizada para que o testador, conhecendo a priori uma lista de mutantes equivalentes, possa marcá-los como equivalentes, todos de uma vez. A opção **Select** permite que sejam selecionados sub-con-

juntos dos mutantes gerados, fazendo com que apenas os mutantes selecionados sejam considerados durante o teste.

#### 4.4 - Outras Funções

Na seção anterior está descrita, basicamente, a forma de conduzir-se uma sessão de teste no Proteum. Algumas outras operações complementam estas operações principais. A primeira delas é iniciada ao selecionar-se o botão **Status**. O Proteum fornece informações sobre o andamento do teste. A Figura 13 mostra quais são essas informações; a mais importante delas é o escore de mutação que indica o quanto o conjunto de casos de teste está próximo da adequação para o programa em teste.

Dados sobre o andamento do teste podem também ser obtidos através de relatórios produzidos pelo Proteum. Atualmente o único tipo implementado é o relatório sobre casos de teste; nele o usuário obtém informação sobre o conteúdo e a capacidade de matar mutantes de cada

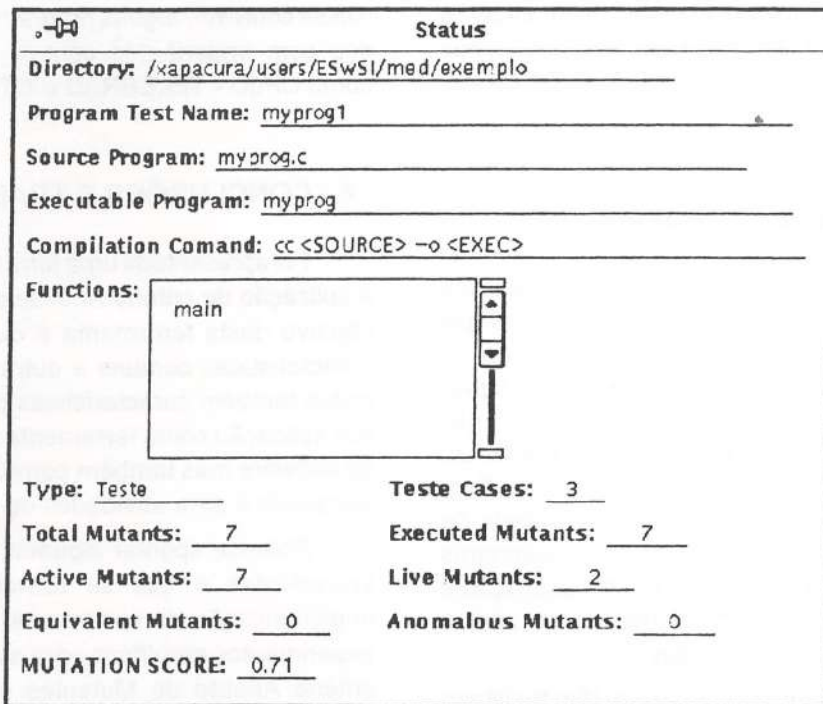
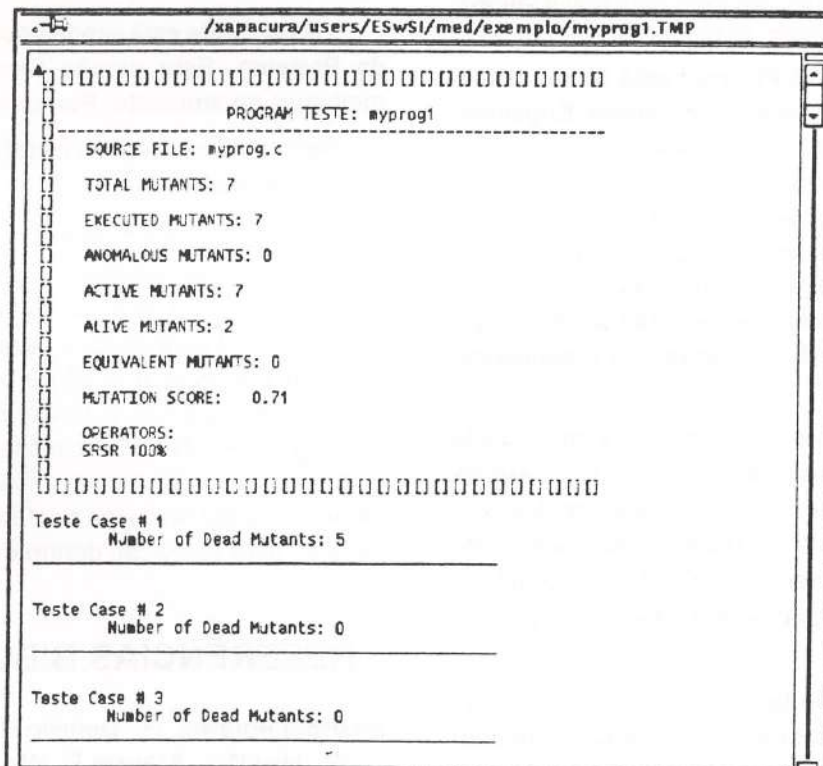


Figura 13 - Status de uma Sessão de Teste

caso de teste. O nível de detalhes de um relatório pode ser definido pelo usuário; a Figura 14 mostra um relatório que contém, para cada caso de teste, o número de mutantes mortos por ele.

O botão **Properties** permite ao usuário determinar algumas características operacionais do ambiente, como o diretório padrão, onde os arquivos de trabalho são armazenados. A opção **Quit** termina a execução da



ferramenta; se houver um teste sendo executado, o usuário deve escolher entre salvar as operações efetuadas durante a sessão de teste ou sair sem salvar, cancelando as operações realizadas.

## 5 - EXPERIMENTOS EMPÍRICOS

Alguns experimentos empíricos foram realizados utilizando o Proteum. O primeiro teve como principal objetivo avaliar a ferramenta tanto em termos funcionais como de desempenho [DEL93]. Foram testados 19 programas, em geral extraídos da literatura, com diferentes características. São programas de cálculo numérico, formatação de texto e um arbitrador de barramento. Este experimento serviu para demonstrar a factibilidade de utilização da Análise de Mutantes dentro de parâmetros aceitáveis de custo. Permitiu também uma avaliação a partir da qual novas características desejáveis para a ferramenta puderam ser identificadas.

Algumas características do Proteum têm facilitado a execução de alguns experimentos. É o caso, por exemplo, do trabalho apresentado no VIII SBES [WON94]. Esse experimento utilizou 10 diferentes programas para avaliar a capacidade de revelar erros de alguns subconjuntos de operadores de mutação. O procedimento básico foi o de aplicar sobre os programas em teste, que contêm erros, a Análise de Mutantes com diferentes conjuntos de operadores de mutação e avaliar a capacidade de revelar erros desses operadores.

Também no ensino o Proteum está sendo usado. No segundo semestre de 1994, na disciplina "Engenharia de Software Apoiada por Computador", ministrada no ICMSC-USP, a ferramenta foi utilizada para mostrar na prática o que é e como deve ser usada uma ferramenta de teste. Os alunos, utilizando o Proteum, desenvolveram experimentos com alguns programas selecionados; os resultados desses experimentos estão sendo compilados e brevemente serão divulgados à comunidade científica.

Em alguns centros o Proteum tem sido utilizado como ferramenta de auxílio a pesquisas. É o caso da Universidade Purdue nos Estados Unidos, um dos centros de destaque na área de teste de software, onde alguns trabalhos vêm sendo realizados utilizando o Proteum para aplicar a Análise de Mutantes em programas C.

Tem-se buscado também realizar um trabalho de transferência tecnológica, procurando colocar o Proteum em ambientes reais de desenvolvimento de software.

Nesse contexto, alguns programas têm sido desenvolvidos com importantes centros de pesquisa nacionais como CPqD - TELEBRÁS e CTI.

## 6 - CONCLUSÕES E TRABALHOS FUTUROS

Foi apresentada uma ferramenta de teste que apóia a aplicação do critério Análise de Mutantes. O principal objetivo desta ferramenta é que ela, além de possuir características comuns a outras ferramentas de teste, inclua também características próprias, visando não só sua aplicação como ferramenta para o desenvolvimento de software mas também como instrumento para novas pesquisas e para atividades de ensino.

Pode-se apontar algumas atividades a serem desenvolvidas e que se tornaram possíveis com a implementação da ferramenta Proteum: realização de experimentos empíricos para avaliação da aplicação do critério Análise de Mutantes e sua comparação com outros critérios; utilização do Proteum em cursos de graduação e pós-graduação como instrumento para demonstração prática de conceitos ligados às atividades de desenvolvimento e teste de software; implementação de outras estratégias de minimização de mutantes; configuração para outras linguagens, com características diversas como C++, OCCAM e, em especial, linguagens de aplicação comercial como COBOL e CLIPPER.

Atualmente está sendo implementada a versão 1.2 do Proteum. Esta versão deve incorporar algumas melhorias no ambiente. Por exemplo:

- . implementação de outras estratégias para minimização seletiva de mutantes;
- . execução batch para usuários experientes

Pretende-se também estender os conceitos do teste baseado na Análise de Mutantes - que se limitam ao teste em nível de unidade - para outros níveis de teste, em particular para o nível de integração, procurando, além disso, observar a tendência atual que aponta o paradigma orientado a objetos como uma abordagem poderosa e prática, o que possivelmente deve implicar na necessidade de rever e reformular alguns conceitos de teste para utilização dentro desse paradigma.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [AGR89] Agrawal, H., Demillo R. A., Hathaway B., Hsu W., Hsu Wy., Krauser E. W., Martin R. J., Mathur A.

- P., Spafford E., "Design of Mutant Operators for the C Programming Language", Technical Report SERC-TR-41-P, Software Eng. Research Center, Purdue University, Mar 1989.
- [CAR91] Carnassale, M., GFC - Uma Ferramenta Multilinguagem para Geração de Grafo de Programa, Dissertação de Mestrado, DCA/FEE/UNICAMP, Fev 1991.
- [CHA91] Chaim, M. L., POKETOOL - Uma Ferramenta para Suporte ao Teste Estrutural de Programas Baseado em Análise de Fluxo de Dados, DCA/FEE/UNICAMP, Abr 1991.
- [CHO89] Choi, B. J., Mathur, A. P. e Pattison, B., "pMothra: Scheduling Mutants For Execution on a Hypercube", Third Symposium on Software Testing, Analysis and Verification, Key West, Dez 1989.
- [DEL93a] Delamaro, M. E., "Uma Visão sobre a Aplicação da Análise de Mutantes", Notas do ICMSC, ICMSC-USP, São Carlos - SP, 1993.
- [DEL93b] Delamaro, M. E., "Proteum: Um Ambiente de Teste Baseado na Análise de Mutantes", Dissertação de Mestrado, ICMSC-USP, São Carlos - SP, Out 1993.
- [DEL94] Delamaro, M. E., "Proteum - Manual do Usuário, Versão 1.1 C", Relatório Técnico, ICMSC-USP, São Carlos - SP, 1994.
- [DEM91] DeMillo, R. A. e Offut, A. J., "Constraint-Based Automatic Test Data Generation", IEEE Trans. on Software Eng., vol 17, Nº 9, Set 1991.
- [KRA91] Krauser, E. W., Mathur, A. P. e Rego, V., "High Performance Software Testing on SIMD", IEEE Trans. on Software Eng., Vol 17, Nº 15, Mai 1991.
- [MAL91] Maldonado, J. C., Critérios Potenciais Usos: uma Contribuição ao Teste Estrutural de Software, Tese de Doutorado, FEE/UNICAMP, Campinas - S. P., 1991.
- [MAR90] Marshall, A. C., Hedley, D., Riddell, I. J. e Hennell, M. A., "Static Dataflow-aided Weak Mutation Analysis (SDAWM)", Information and Software Technology, vol 32, Nº 1, Jan/Fev 1990.
- [MAT88] Mathur, A. P. e Krauser, E. W., "Modeling Mutation on Vector Processor", Proc. of the Second Workshop on Software Testing, Verification and Analysis, Banff-Canada, 1988.
- [MAT93] Mathur, A. P., Wong, E. W., "Reducing the Cost of Mutation Testing: A Case Study", Tech Report SERC-TR-138-P, Software Engineering Research Center, Purdue University, Jun 1993.
- [PRE92] Pressman, R. S., Software Engineering - A Practitioner's Approach (3rd. edition), McGraw-Hill, 1992.
- [WON94] Wong, E. W., Maldonado, J. C., Delamaro, M. E., Mathur, A. P., "Constrained Mutation in C Programs", VII Simpósio Brasileiro de Eng. de Software, Curitiba - PR, Out 1994.

---

# SISTEMAS DE INFORMAÇÃO E SISTEMAS DE APOIO À DECISÃO\*

## INFORMATION SYSTEMS AND DECISION SUPPORT SYSTEMS

Eduardo O. C. CHAVES\*\*  
Orandi Mina FALSARELLA\*\*\*

### ABSTRACT

This work seeks to define Information Systems (IS) and to categorize the various types of IS described in the technical literature. The model of stages in the evolution of IS usage in organizations, proposed by Richard Nolan, is used as an instrument of categorization: the categories of IS found in the literature are related to the stages suggested by Nolan. Special attention is given to Decision Support Systems (DSS), in view of the increasingly important role they perform in organizations and of the fact that they are frequently confused in the literature with other types of IS.

**KEY WORDS:** Information Systems, Decision Support Systems.

### RESUMO

Este trabalho busca definir Sistemas de Informação (SI) e categorizar os vários tipos de SI descritos na literatura técnica. Como instrumento de categorização é usado o modelo dos estágios da evolução da informática nas organizações, proposto por Richard Nolan; as categorias de SI encontradas são relacionadas aos estágios sugeridos por Nolan. A seguir, dá-se destaque a Sistemas de Apoio à Decisão (SAD), em decorrência do papel cada vez mais importante que exercem nas organizações e do fato de que frequentemente são confundidos, na literatura, com os demais tipos de SI.

**PALAVRAS-CHAVE:** Sistemas de Informação, Sistemas de Apoio à Decisão

## 1. INTRODUÇÃO

A área de Gerenciamento de Sistemas de Informação é bastante abrangente. Por isso, encontramos nela uma grande quantidade de termos, usados em tentativas de caracterizar e classificar os Sistemas de Informação (SI). Geralmente essa proliferação de termos mais complica do que ajuda o entendimento da área. Existem tantas siglas, como EDP ("Electronic Data Processing"), MIS ("Management Information Systems"), EIS ("Executive Information Systems"), ES ("Expert Systems"), DSS ("Decision Support Systems"), que fica

difícil diferenciar, com certeza, os conceitos e as aplicações a que se referem. Na verdade, muitas vezes parece não existirem fronteiras claras, bem definidas, que limitem e demarcam corretamente esses conceitos.

Além disso, muitas das definições de SI encontradas na literatura, algumas das quais são discutidas neste trabalho, podem induzir o leitor a confundir os sistemas que descrevem como Sistemas de Apoio à Decisão (SAD), dando a impressão de que qualquer SI pode ser considerado como um SAD ou ser transformado em um SAD.

Pelos motivos apresentados, este trabalho tem os seguintes principais objetivos:

---

(\*) Este artigo é baseado em dissertação de Mestrado apresentada ao Instituto de Informática da PUCAMP em Novembro de 1994 por Orandi Falsarella, com a orientação de Eduardo Chaves.

(\*\*) Professor Titular da Faculdade de Educação da UNICAMP e do Instituto de Informática da PUCAMP.

(\*\*\*) Professor do Instituto de Informática da PUCAMP.

- . Definir e categorizar os vários tipos de SI descritos na literatura técnica;
- . Definir e demarcar SAD.

## 2. SISTEMAS DE INFORMAÇÃO

### 2.1. Definição

Sistemas de Informação (SI) são sistemas que permitem a coleta, o armazenamento, o processamento, a recuperação e a disseminação de informações. SI são, hoje, quase sem exceção, baseados no computador e apoiam as funções operacionais, gerenciais e de tomada de decisão existentes na organização. Os usuários de SI são provenientes tanto do nível operacional, como do nível tático e mesmo estratégico e utilizam SI para alcançar os objetivos e as metas de suas áreas funcionais.

Como bem ressalta Maurício Prates, em seu artigo "**Conceituação de Sistemas de Informação do Ponto de Vista do Gerenciamento**" [PRAT94], SI são formados pela combinação estruturada de vários elementos, a saber: a **informação** (dados formatados, textos livres, imagens e sons), os **recursos humanos** (pessoas que coletam, armazenam, recuperam, processam, disseminam e utilizam as informações), as **tecnologias de informação** (o hardware e o software usados no suporte aos SI) e as **práticas de trabalho** (métodos utilizados pelas pessoas no desempenho de suas atividades), organizados de tal modo a permitir o melhor atendimento dos objetivos da organização.

### 2.2. Características

Como observado na Introdução, uma grande quantidade de termos é usada para classificar e caracterizar SI. Uma forma de limitar e demarcar corretamente as fronteiras conceituais entre esses termos é relacionar cada tipo de SI com o estágio de utilização da informática em que uma organização se encontra.

Segundo Richard Nolan, em seu livro **Management Accounting and Control of Data Processing** [NOLA77], a evolução da informática numa organização ocorre em seis estágios:

#### . Iniciação

Neste estágio o usuário é resistente ao uso da informática e seu envolvimento com a tecnologia é superficial. A organização encoraja o uso da informática e se preocupa com o aprendizado, mas poucas atividades são automatizadas.

#### . Contágio

Neste estágio começam a proliferar SI informatizados, que automatizam atividades antes desenvolvidas manualmente, sem, porém, se preocupar com a integração das informações.

#### . Controle

Neste estágio o crescimento do uso de SI na organização passa a ser explosivo, o usuário sendo a força propulsora. Por isso, a organização passa a exigir melhor gestão dos recursos de informática.

#### . Integração

Neste estágio, em resposta à pressão por melhor gestão, os SI passam a ser orientados para atender às necessidades dos níveis gerenciais, as informações são de melhor qualidade e é exigida maior integração entre elas.

#### . Administração de Dados

Neste estágio, os SI começam a ser organizados em termos de sistemas que interessam à organização como um todo (chamados corporativos) e sistemas de uso setorial ou especializado, havendo cuidado, em qualquer hipótese, com a correta administração dos dados, de modo a evitar redundâncias.

#### . Maturidade

Neste estágio, a informação passa a ser considerada como patrimônio da organização, o usuário é participativo e responsável e o crescimento da informática é ordenado.

Na Seção 2.3, que descreve os vários tipos de SI, busca-se relacionar cada tipo de SI a um - ou mais de um - desses estágios.

### 2.3. Tipos

Embora possa haver muitas maneiras de categorizar os SI, uma forma interessante é a que os classifica em:

- . Sistemas Transacionais
- . Sistemas Gerenciais
- . Sistemas Executivos
- . Sistemas Especialistas
- . Sistemas de Apoio à Decisão

### 2.3.1. Sistemas Transacionais

O processo inicial de informatização de qualquer organização é baseado fundamentalmente no desenvolvimento e na implantação de SI transacionais (também chamados de operacionais). Esses SI são também identificados pela expressão "Electronic Data Processing" (EDPs), e eles são necessários para o controle operacional das organizações [TOM 91].

No modelo da evolução da informática nas organizações proposto por Richard Nolan, SI transacionais se enquadram nos estágios de iniciação e contágio. São sistemas operacionais, não integrados, atendem em geral à área administrativo-financeira, controlam, na maioria das vezes, o fluxo de informações financeiras, e os usuários finais esboçam uma certa resistência a utilizá-los. Os sistemas de folha de pagamento, contabilidade, controle de estoques, contas a pagar e a receber, faturamento, etc., são exemplos de SI transacionais.

As principais funções e características desses sistemas são:

- . Coletar, via digitação, os dados existentes nos documentos operacionais das organizações, validando-os;
- . Armazenar esses dados em meio magnético;
- . Ordenar ou indexar esses dados, de modo a facilitar o acesso a eles;
- . Permitir consultas, on-line ou em batch, aos dados, detalhados ou agregados, que permitam retratar diferentes aspectos das operações;
- . Gerar relatórios que possam ser distribuídos a outras pessoas que não os usuários diretos dos SI.

Muito embora esses sistemas só controlem o fluxo de informações operacionais, eles também disponibilizam informações para a tomada de decisão. Um exemplo disso pode ser um sistema de controle de estoques que

fornece informações sobre a movimentação do estoque para o departamento de compras. Este departamento poderá, através dessas informações, tomar decisões sobre quais produtos deverão ser comprados e em que quantidade. Um EDP pode, portanto, fornecer informações para apoio à decisão. Isso, porém, não o torna, apenas em decorrência desse fato, um SAD.

### 2.3.2. Sistemas Gerenciais

A evolução natural da informatização das organizações, após a implantação dos EDP, é o desenvolvimento de sistemas que forneçam informações integradas e sumarizadas, provenientes de diversos sistemas transacionais. Essas informações têm capacidade de prover material para análise, planejamento e suporte à decisão [DAVI85] e possibilitam a gerentes de médio escalão visualizar o desempenho de seu departamento e mesmo da organização como um todo. Esses sistemas que suprem com informações a média gerência são geralmente chamados de "Management Information Systems" (MIS).

O surgimento desses sistemas acontece nos estágios de controle e integração, no modelo proposto por Richard Nolan, onde o usuário é força propulsora e exige informações em maior quantidade, menor tempo e com melhor nível de integração.

Um bom exemplo de MIS pode ser encontrado em um sistema que analisa as receitas e as despesas de uma organização e possibilita que gerentes as relacionem e comparem com o que foi planejado no orçamento.

As principais funções e características desses sistemas são:

- . Integrar dados de diversas aplicações e transformá-los em informação;
- . Fornecer informações para o planejamento operacional, tático e até mesmo estratégico da organização;
- . Suprir gerentes com informações para que estes possam comparar o desempenho atual da organização com o que foi planejado;
- . Produzir relatórios que auxiliem os gerentes a tomar decisões.

A grande maioria das informações produzidas por um MIS, quer seja para análise de tendências, quer seja para planejamento ou revisão, auxilia os gerentes no

processo de tomada de decisão. Isso significa que um MIS pode ter funções específicas que façam parte de ambientes de apoio à decisão.

### 2.3.3. Sistemas Executivos

Com base nos dados existentes nos EDPs, nas informações disponíveis nos MIS e em informações coletadas de fontes externas à organização, é possível construir sistemas de informação dirigidos para a alta gerência. Esses sistemas que abastecem a alta gerência de informações são geralmente chamados de "Executive Information Systems" (EIS) e permitem que o executivo tenha ou ganhe acesso a informações internas e externas à organização que sejam relevantes para controlar os fatores críticos de sucesso [WATS92].

Segundo Henry C. Lucas Jr., em seu livro **Information Systems Concepts for Management** [LUCA90], um EIS não tem maiores diferenças conceituais em relação a um sistema de apoio à decisão. O que o diferencia é, em geral, a interface com o usuário, que deve permitir que um executivo o utilize com facilidade.

Os EIS começam a ser desenvolvidos nas organizações nos estágios de administração de dados e maturidade, no modelo definido por Richard Nolan. Nesses estágios os sistemas de informação existentes refletem o fluxo de informações da organização, o usuário participa integralmente do desenvolvimento dos sistemas, as informações passam a ser consideradas patrimônio da organização, o crescimento da informática é ordenado, a informática passa a ter função de apoio estratégico para a organização e não se tomam decisões sem base nas informações produzidas por um EIS.

As principais funções e características desses sistemas são:

- . Gerar mapas, gráficos e dados que possam ser submetidos a análise estatística para suprir os executivos com informações comparativas, fáceis de entender;
- . Fornecer dados detalhados sobre passado, presente e tendências futuras das unidades de negócios em relação ao mercado para auxiliar o processo de planejamento e de controle da organização [WATS92];
- . Possibilitar a análise das informações obtidas;

- . Permitir que o executivo se comunique com o mundo interno e externo através de interfaces amigáveis (correio eletrônico, teleconferência, etc.) que sejam flexíveis a ponto de se ajustarem ao seu estilo pessoal;
- . Oferecer ao executivo ferramentas de organização pessoal (calendários, agendas eletrônicas, etc.) e de gerenciamento de projetos, tarefas e pessoas.

### 2.3.4. Sistemas Especialistas

O conhecimento e as experiências que uma pessoa detém sobre determinada área do conhecimento precisa ser, muitas vezes, preservado e disseminado para que pessoas com menos conhecimento e experiência possam deles se valer para resolver seus problemas [ROCK86].

Existem sistemas de informação que armazenam e disponibilizam o conhecimento e as experiências de especialistas. Esses SI são geralmente conhecidos como "Expert Systems" (ES), quando fornecem, eles mesmos, soluções para determinados problemas, e como "Expert Support Systems" (ESS), quando fornecem informações extraídas das bases de conhecimento a profissionais e executivos para auxiliá-los no processo de tomada de decisão.

Normalmente, o desenvolvimento desses sistemas não depende da existência de outros SI e, portanto, eles podem ser desenvolvidos em qualquer um dos estágios da evolução da informática no modelo definido por Richard Nolan.

As principais funções e características desses sistemas são:

- . Armazenar o conhecimento e as experiências de especialistas em bases de conhecimento;
- . Utilizar mecanismos de inferência integrados às bases de conhecimento para resolver - ou auxiliar a resolver - problemas;
- . Possibilitar a inclusão de novos conhecimentos nas bases de conhecimentos sem eliminar os conhecimentos já armazenados.

### 2.3.5. Sistemas de Apoio à Decisão

Os SI até aqui descritos podem ter funções que forneçam informações para tomada de decisão. Segun-

do Ralph H. Sprague e Hugh J. Watson, no livro **Sistemas de Apoio à Decisão** [SPRA91], qualquer SI que forneça informações para auxílio à decisão é um sistema de apoio à decisão (SAD).

Essa afirmação é, porém, bastante questionável, pois SAD são sistemas que não só fornecem informações para apoio à tomada de decisão, mas que contribuem para o **processo de tomada de decisão**. A obtenção da informação é apenas parte do processo, como veremos detalhadamente adiante.

SAD, que também são conhecidos como "Decision Support Systems" (DSS), possuem funções específicas, não vinculadas aos sistemas existentes, que permitem buscar informações nas bases de dados existentes e delas retirar subsídios para o processo de tomada de decisão. SAD começam a ser desenvolvidos na organização a partir dos estágios de controle e integração no modelo proposto por Richard Nolan.

SAD serão discutidos em maior detalhe a seguir.

### 3. SISTEMAS DE APOIO À DECISÃO

#### 3.1. Conceituação

A necessidade dos SAD surgiu na década de 70, em decorrência de diversos fatores, como, por exemplo, os seguintes:

- . Competição cada vez maior entre as organizações;
- . Necessidade de informações rápidas para auxiliar no processo de tomada de decisão;
- . Disponibilidade de tecnologias de hardware e software para armazenar e buscar rapidamente as informações;
- . Possibilidade de armazenar o conhecimento e as experiências de especialistas em bases de conhecimentos;
- . Necessidade de a informática apoiar o processo de planejamento estratégico empresarial.

Esses fatores contribuíram para que as organizações começassem a desenvolver SI que pudessem fornecer informações para auxiliar no processo de tomada de decisão.

A literatura disponível sobre SAD não deixa muito claro o que vem a ser um SAD. Existem muitas definições que são contraditórias e que podem, inclusive, ser confundidas com definições de outros tipos de SI.

Dentre as definições consultadas, ilustremos três:

- . "SAD é um sistema de informação que apoia qualquer processo de tomada de decisão em áreas de planejamento estratégico, controle gerencial e controle operacional" [SPRA91].
- . "SAD é um sistema baseado em computador que auxilia o processo de tomada de decisão utilizando dados e modelos para resolver problemas não estruturados" [LUCA90].
- . "SAD é uma estratégia de implementação que torna o computador útil ao gerente" [ROCK86].

Analisando essas definições, algumas questões podem ser levantadas, como, por exemplo:

- . Um EIS também auxilia o processo de tomada de decisão na área de planejamento estratégico e, nem por isso, é chamado de SAD. O mesmo acontece com MIS na área de controle gerencial e com EDP na área de controle operacional.
- . Um SAD também serve para auxiliar a resolução de problemas estruturados.
- . Todo SI pode ser útil ao nível gerencial e, nem por isso, todo SI será um SAD.

Um EIS, um MIS e um EDP podem ter funções que forneçam **informações** para apoio à decisão. Porém, esses SI não foram construídos com o objetivo de auxiliar o **processo** de tomada de decisão. Quando se fala em auxiliar o processo de tomada de decisão, isso não significa somente fornecer informações para apoio à decisão, mas, também, analisar alternativas, propor soluções, pesquisar o histórico das decisões tomadas, simular situações, etc.

O processo de tomada de decisão se desenrola, portanto, através da interação constante do usuário com um **ambiente de apoio à decisão** especialmente criado para dar subsídio às decisões a serem tomadas. Esse ambiente, representado na Figura 3.1, é constituído por:

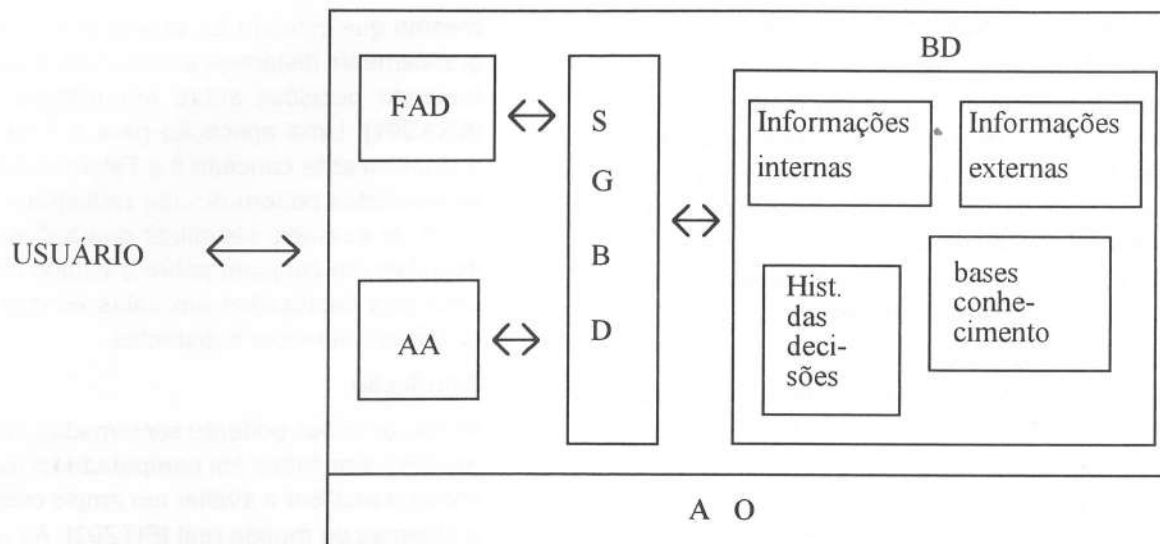


Figura 3.1. - Ambiente de Apoio à Decisão

- . **Bancos de Dados (BD)** - São formados por informações internas e externas à organização, por conhecimentos e experiências de especialistas e por informações históricas acerca das decisões tomadas.
- . **Sistema Gerenciador de Banco de Dados (SGBD)** - Após os dados estarem instalados no BD, o SGDB deve possibilitar o acesso às informações e a sua atualização, garantindo a segurança e a integridade do BD [AWAD88].
- . **Ferramentas de Apoio à Decisão (FAD)** - São softwares que auxiliam na simulação de situações, na representação gráfica das informações, etc.
- . **Ambiente Aplicativo (AA)** - São sistemas aplicativos ou funções acrescidas aos sistemas existentes que fazem análise de alternativas e fornecem soluções de problemas.
- . **Ambiente Operacional (AO)** - É composto por hardwares e softwares que permitem que todos os componentes do ambiente sejam integrados.

A todo esse ambiente, que fornece subsídios para que o usuário tome decisões, é que daremos neste trabalho o nome de SAD.

### 3.2. Características

As principais características dos SAD são:

- . Possibilidade de desenvolvimento rápido, com a participação ativa do usuário em todo o processo;
- . Facilidade para incorporar novas ferramentas de apoio à decisão, novos aplicativos e novas informações.
- . Flexibilidade na busca e manipulação das informações [BURC89];
- . Individualização e orientação para a pessoa que toma as decisões, com flexibilidade de adaptação ao estilo pessoal de tomada de decisão do usuário [MITT86];
- . Real pertinência ao processo de tomada de decisão, ajudando o usuário a decidir através de subsídios relevantes;
- . Usabilidade, ou seja, facilidade para que o usuário o entenda, use e modifique de forma interativa. [AWAD88].

### 3.3. Motivação para Uso

Quando uma organização não possui sistemas de informação que auxiliem o processo de tomada de decisão, as decisões são baseadas em dados históricos e em experiências individuais. Quando existe um SAD apoiando esse processo, as informações fornecidas por ele são incorporadas aos dados históricos e experiências individuais, possibilitando melhores condições para a tomada de decisão.

Em qualquer das situações citadas, a saída ou a mudança do usuário poderá causar grande impacto na

organização. Isto acontecerá devido à perda da história de como as decisões foram tomadas.

Partindo do princípio de que um SAD auxilia o processo de tomada de decisão, é importante que seus conceitos retratem a cultura da organização e façam parte integrante dela, não servindo apenas para atender às necessidades específicas de um usuário. Este é o principal motivo para que as empresas implantem SI que auxiliem o processo de tomada de decisão.

O sucesso de um SAD, a sua continuidade, e, principalmente, a motivação para que as pessoas responsáveis pela tomada de decisão o utilizem dependem dos seguintes fatores:

- . O modelo construído deve atender às necessidades gerais da organização e não somente às necessidades específicas de um usuário;
- . Eventuais mudanças no sistema devem ser feitas rapidamente pelo analista de sistemas para atender a novas necessidades de informação para apoio à decisão;
- . Informações sobre as decisões tomadas devem ser armazenadas e estar disponíveis para que outras pessoas as utilizem em novos processos de tomada de decisão;
- . A interface com o usuário deve ser a mais amigável possível;
- . A obtenção das informações, internas e externas à organização, deve ser imediata;
- . Os benefícios da utilização de SAD devem ser disseminados na organização através de cursos, palestras, etc.

### 3.4. Tendências no Uso de SAD

À medida que a informática vai evoluindo dentro das empresas (os sistemas transacionais já estão implantados, já existem sistemas que fornecem informações gerenciais, etc.), a tendência natural é que aumente a demanda por SAD. O grande incentivo para a utilização de SAD se dará, porém, quando a eles forem incorporadas algumas importantes tecnologias de informação, já disponíveis ou emergentes, como as seguintes:

#### . **Groupware ou CSCW (Computer-Supported Cooperative Work)**

Essa tecnologia permitirá que duas ou mais pessoas trabalhem em tarefas comuns, em ambientes computacionais completamente diferentes,

mesmo que estejam localizadas em locais geograficamente distantes, interagindo, discutindo e tomando decisões sobre um mesmo assunto [KRAS91]. Uma aplicação para a área médica que utiliza esse conceito é a Telemedicina, onde especialistas podem discutir radiografias, resultados de exames, visualizar operações e tomar decisões em conjunto sobre o estado clínico de pacientes localizados em salas ou mesmo em hospitais diferentes e distantes.

#### . **Simulação**

Muitas decisões poderão ser tomadas através de modelos simulados em computadores que servirão para analisar e avaliar um amplo conjunto de problemas do mundo real [FITZ93]. As alternativas de decisão poderão ser analisadas e validadas através de simulação antes que a decisão seja tomada.

#### . **Gerenciadores de Informação**

Esses sistemas permitirão obter, integrar e manipular, além dos dados quantitativos, os dados qualitativos de todos os SI existentes na organização.

#### . **Multimídia**

Multimídia possibilitará que um SAD possa guardar e buscar informações contidas em gerenciadores de banco de dados através de hiperdocumentos, ou seja, documentos computadorizados que contenham diagramas, imagens, sons, animação, vídeo e texto, disponibilizados através de formas de acesso totalmente flexíveis [MART92]. A informação para auxílio à tomada de decisão poderá ser encontrada e visualizada de forma mais rápida e fácil.

#### . **Expert Systems**

Através da área da inteligência artificial, que se ocupa da construção de sistemas que manipulam informações armazenadas em bases de conhecimento e que retratam o raciocínio de especialistas [ROCK86], será possível guardar as decisões tomadas e o raciocínio que foi utilizado para se chegar a elas. Os SAD poderão buscar informações nessas bases de conhecimento para auxiliar futuras decisões.

#### . **Interfaces Amigáveis**

O acesso às informações oferecidas por um SAD poderá ser feito através de ícones acionados por mouses ou tocados com os próprios dedos. Os

teclados e mesmo os mouses futuramente deverão ser substituídos pela voz humana. A representação da informação será exibida através de várias mídias (voz, sons, imagens, animação, texto, etc.). Isto tornará a interface bem mais agradável.

#### Redes de Comunicação

O avanço das redes de comunicação permitirá a transmissão simultânea e sincronizada de sons, imagens, vídeos, dados e textos em alta velocidade, com informações totalmente digitais e com alto grau de segurança. A obtenção das informações para auxílio à decisão será instantânea.

## 4. CONCLUSÕES

Com base na discussão realizada, podemos extrair as seguintes conclusões do presente trabalho:

- Para classificar e caracterizar um SI é útil analisar o estágio da evolução da informática em que a organização se encontra. Se a organização estiver no estágio de iniciação ou contágio, dificilmente os SI implantados poderão ser classificados e caracterizados como MIS, SAD, ou EIS.
- O fato de um SAD fornecer informações e subsídios que contribuem para o processo de tomada de decisão o diferencia dos demais tipos de SI: somente SAD possuem essa característica.
- Para desenvolver um SAD é necessário construir um ambiente de apoio à decisão (AAD). Construir um AAD, em alguns casos, não significa construir um novo SI, mas, sim, incorporar aos sistemas existentes Ambientes Aplicativos (AA) e/ou Ferramentas de Apoio à Decisão (FAD) que forneçam informações e subsídios para o processo de tomada de decisão.

## 5. REFERÊNCIAS BIBLIOGRÁFICAS

- [AWAD88] Awad, Elias M., **Management Information Systems**, Benjamin/Cummings, 1988.
- [BURC89] Burch, John G. e Gary Grudnitski, **Information Systems - Theory and Practice**, John Wiley & Sons, 1989.
- [DAVI85] Davis, Gordon B. e Margrethe H. Olson, **Management Information Systems**, McGraw-Hill, 1985.
- [FITZ93] Fitzpatrick, Kathy E., Joanna R. Baker e Dinesh S. Dave, "An Application of Computer Simulation to Improve Scheduling of Hospital Operating Room Facilities in the United States", **International Journal of Computer Applications in Technology**, 1993.
- [KRAS91] Krasner, Herb, John McInroy e Diane B. Walz, "Groupware Research and Technology Issues with Application to Software Process Management", **IEEE Transactions on Systems, Man, and Cybernetics**, July/August, 1991.
- [LUCA90] Lucas, Henry C. Jr., **Information Systems Concepts for Management**, McGraw-Hill International, 1990.
- [MART92] Martin, James, **Hiperdocumentos e Como Criá-los**, Campus, 1992.
- [MITT86] Mitra, Sitansu S., **Decision Support Systems Tools and Techniques**, John Wiley & Sons, 1986.
- [NOLA77] Nolan, Richard L., **Management Accounting and Control of Data Processing**, National Association of Accountants, 1977.
- [PRAT94] Prates, Maurício, "Conceituação de Sistemas de Informação do Ponto de Vista do Gerenciamento", **Revista do Instituto de Informática**, PUCCAMP, Março/Setembro, 1994.
- [ROCK86] Rockart, John F. e Cristine V. Bullen, **The Rise of Managerial Computing**, Dow Jones-Irwin, 1986.
- [SPRA91] Sprague, Ralph H. e Hugh J. Watson, **Sistemas de Apoio à Decisão**, Campus, 1991.
- [TOM91] Tom, Paul L., **Managing Information as a Corporate Resource**, Harper Collings Publishers, 1991.
- [WATS92] Watson, Hugh J., R. Kelly Rainer e George Houdeshel, **Executive Information Systems**, John Wiley & Sons, 1992.

---

# UMA EXPERIÊNCIA EM PORTABILIDADE DE INTERFACES GRÁFICAS

## AN EXERCISE IN GRAPHICAL USER INTERFACES PORTABILITY

Miguel ARGOLLO JR.\*  
Simone SCARPELINI\*\*

### ABSTRACT

During the last few years Graphical User Interfaces (GUI for short) became a key factor for the success of a software system in almost any application area. Software portability has always been a great concern for developers, and GUI portability became a reality with multi-platform GUI design tools; however, these tools are quite new for the Brazilian software industry. This work describes an experience in developing portable GUI with such a tool, XVT.

**KEY WORDS:** Graphical User Interfaces (GUI), portability.

### RESUMO

Nos últimos anos interfaces gráficas (GUI) tornaram-se um fator chave para o sucesso de um sistema de software em praticamente qualquer domínio de aplicação. Portabilidade de software sempre foi uma preocupação para os desenvolvedores, e a portabilidade de interfaces gráficas tornou-se uma realidade com ferramentas para o projeto de interfaces gráficas multi-plataformas; entretanto, a utilização das mesmas é bastante recente para a indústria nacional de software. Este trabalho descreve uma experiência no desenvolvimento de interfaces gráficas portáteis com uma ferramenta desse tipo, XVT.

**PALAVRAS-CHAVE:** Interfaces Gráficas (GUI), portabilidade.

## 1 - INTRODUÇÃO

Desde o final da década de 80 o mercado de software passou a fazer uma forte pressão para que os aplicativos fossem desenvolvidos com interfaces gráficas. Esta pressão foi motivada tanto pela disponibilidade de computadores pessoais e estações de trabalho com capacidade gráfica a um preço acessível quanto pelo surgimento de poderosos ambientes gráficos.

Uma das principais vantagens de aplicativos com interfaces gráficas sob o ponto de vista de seus usuários é o fato de apresentarem aparência e funcionamento (**look-and-feel**) consistentes. Essa consistência não é obtida automaticamente, e sim através do uso de padrões, que podem ser públicos ou proprietários e que definem tanto os elementos gráficos das interfaces quanto o funcionamento dos mesmos.

Um padrão proprietário depende unicamente da companhia que o desenvolveu, e normalmente se encontra associado a um único ambiente operacional. Os sistemas MS-WINDOWS, IBM Presentation Manager e Macintosh são os exemplos de padrões proprietários mais utilizados atualmente.

Um padrão público não se encontra sobre o controle de uma única companhia, sendo sua evolução normalmente controlada por um conjunto de empresas ou um grupo de usuários. O sistema OSF/MOTIF, o padrão público de maior aceitação nesta área, foi definido pela Open Software Foundation, um consórcio formado pelos principais fornecedores de sistemas Unix.

Embora o sistema MS-WINDOWS domine o cenário nacional, em nível internacional outros ambientes possuem uma fatia considerável do mercado. A dinâmica da área de informática, com o surgimento de novas

---

(\*) Matemático (mod. Informática) pela UFRJ (1980) e mestre em Engenharia de Sistemas e Computação pela COPPE/UFRJ (1984). Atualmente chefe da Divisão de Engenharia de Software do Instituto de Computação da Fundação Centro Tecnológico para Informática - CTI.

(\*\*) Engenheira de Computação pela Unicamp (1992). Atualmente Analista de Software prestando serviços de reengenharia de sistemas no Bamerindus.

plataformas de software e hardware, bem como a possibilidade dos produtores nacionais passarem a disputar o mercado internacional, transforma o desenvolvimento de aplicações multi-plataformas em um mecanismo gerador de novos negócios. Um produtor de software que domine a tecnologia necessária para oferecer seus produtos em várias plataformas sem onerar substancialmente seus custos de desenvolvimento possui uma importante vantagem competitiva em suas mãos.

As áreas de linguagens de programação, sistemas de bancos de dados e sistemas de comunicações vêm sendo objetos de esforços de padronização há algum tempo e os padrões obtidos, embora com um uso ainda incipiente, são uma ferramenta poderosa para o desenvolvimento de software multi-plataforma [IEE93a]. Entretanto, o desenvolvimento de interfaces gráficas portáteis ainda é considerado, por boa parte dos produtores de software, como um desafio tecnológico a ser superado. Por exemplo, uma pesquisa sobre experiências práticas no desenvolvimento de interfaces gráficas publicada em 1992 mostrou que somente 3 dos 74 projetos que responderam à pesquisa tinham sido planejados para serem transportados para múltiplas plataformas [MR92].

Este artigo reflete a experiência do CTI na utilização de uma ferramenta empregada para produção de interfaces gráficas portáteis e está organizado na seguinte forma: a próxima seção apresenta o escopo no qual o trabalho foi realizado; a 4ª seção apresenta a arquitetura da ferramenta utilizada e o fluxo de desenvolvimento de um programa de computador através de seu uso; as duas seções seguintes apresentam os dois módulos principais da ferramenta, o editor de interfaces gráficas e a biblioteca multi-plataforma; o artigo termina com 2 seções que apresentam os resultados dos testes realizados e as conclusões obtidas.

Embora [BA] e [HS91] apresentem métodos de avaliação de ferramentas de suporte ao desenvolvimento de interfaces gráficas, estas referências não tratam do problema de portabilidade das interfaces obtidas, de modo que o enfoque adotado para o desenvolvimento do trabalho reportado aqui é mais pragmático e orientado para as necessidades de produção de software portátil do Programa de Qualidade e Produtividade em Software - PQPS.

## 2 - PROGRAMA DE QUALIDADE E PRODUTIVIDADE EM SOFTWARE - PQPS

O Programa de **Qualidade e Produtividade em Software do CTI - PQPS** - é composto por um conjunto coordenado de ações que visa a aquisição, o domínio e

a disseminação de métodos, técnicas, critérios e ferramentas integrados em ambientes de suporte à produção de software. Essas ações cobrem duas grandes áreas: produção de software portátil e avaliação de produtos e processos de desenvolvimento de software.

No campo de avaliação está sendo desenvolvido o projeto **Tecnologia para Avaliação da Qualidade de Software - TAQS**, cujo objetivo básico é o de desenvolver métodos para a avaliação da qualidade de produtos de software e do processo de desenvolvimento de software, além de instrumentos práticos para a sua aplicação ampla e facilitada. Desta forma, pretende-se aparelhar um avaliador com recursos para julgamento do nível de qualidade de software em relação a critérios e padrões internacionalmente reconhecidos.

Numa primeira fase, o projeto está estruturado para a geração de tecnologias para a avaliação de produtos de software - TAQS-PROD, onde se pretende atingir os seguintes pontos:

- . especificação de métodos, expressos em um manual do avaliador, contendo procedimentos para o desenvolvimento de um processo de avaliação e para estimar os custos do processo;

- . criação do Ambiente para Avaliação da Qualidade do Produto de Software - AAQPS, composto por:

1. Sistema de apoio à Especificação e ao Planejamento da Avaliação;
2. Sistema de gerenciamento de Bibliotecas de Módulos de Avaliação, Métricas e de um Banco de Dados Histórico;
3. Módulos de Avaliação para integrar a Biblioteca correspondente, compostos de procedimentos manuais, auxiliados por computador, ou automáticos.

Todo o ambiente deverá utilizar, sempre que possível, tecnologias já existentes, como ferramentas já comercializadas, e padrões já aceitos. Dentro deste contexto, atividades de pesquisa que colaborem para o aprimoramento e evolução deste ambiente e que ampliem a disponibilidade tecnológica deverão ser favorecidas.

No campo de produção de software portátil procura-se desenvolver técnicas para a produção de aplicativos que possam estar disponíveis em ambientes operacionais distintos através da utilização de padrões que forneçam a base sobre a qual os aplicativos são desenvolvidos. Tais padrões podem ser classificados segundo um modelo de referência, que deve funcionar como uma ferramenta de trabalho para os projetistas dos aplicativos,

oferecendo, para cada requisito do aplicativo, um componente que forneça mecanismos e serviços para o seu desenvolvimento.

Atualmente existem poucos modelos propostos, entre os quais se destaca o do grupo de trabalho da IEEE P1003.0, chamado OSE (**Open System Environment**) [IEE93a]. Este modelo está dividido em cinco componentes principais, aqui apresentados em inglês como definidos pelo grupo de trabalho:

1. "Application Software";
2. "Application Program Interface - API";
3. "Application Platform Service Areas";
4. "External Environment Interface";
5. "External Environment".

O terceiro componente, que trata dos serviços necessários nas plataformas nas quais os aplicativos devem estar disponíveis, está dividido em 4 componentes que são críticos para a obtenção de software portátil:

1. - "Serviços de Sistemas Operacionais e Linguagens";
2. - "Serviços de Interface com o Usuário";
3. - "Serviços de Informação";
4. - "Serviços de Comunicações".

Espera-se que a utilização dessa arquitetura sirva como base para a implementação de software portátil, satisfazendo as expectativas da comunidade de processamento de dados, entre as quais podem ser citadas:

- . aumento da produtividade dos programadores e usuários;
- . facilidade no uso e suporte dos produtos desenvolvidos pela consistência obtida entre os mesmos; e
- . aumento no retorno dos investimentos em sistemas de informação através da melhor utilização dos recursos de computação e da experiência de seus usuários.

A adoção dessa arquitetura pelo PQPS apresenta como vantagens relevantes, além de suas qualidades técnicas, o fato de ser independente de fornecedores e de estar sendo formalizada e padronizada por uma instituição de relevância mundial, a IEEE.

(1) Parte da funcionalidade que está sendo definida para a UAPI está baseada na funcionalidade da biblioteca XVT, como por exemplo a semântica das operações de criação e destruição de janelas (P1201.1 Draft, sc. 6.5.1).

(2) Os testes foram realizados com a versão 3.01 do sistema. No final da fase de elaboração deste trabalho foi lançada a nova versão do sistema, 4.0. Embora significativas, as mudanças realizadas nessa versão não alteram as conclusões obtidas à respeito do desenvolvimento de interfaces gráficas. Todas as referências relacionadas com a funcionalidade do sistema estão relacionadas com a versão anterior, a menos quando a nova versão é explicitamente citada.

No PQPS decidiu-se inicialmente atacar os componentes de interface com o usuário e gerenciamento de serviços de informações.

Essa escolha foi motivada pela experiência da equipe, pela relevância que esses componentes têm na produção de software de qualidade e no fato de todas as arquiteturas propostas na literatura contemplarem e realçarem a importância dos mesmos.

Consciente da relevância de se obter uma solução satisfatória para o problema de portabilidade de interfaces gráficas, a IEEE constituiu um grupo de trabalho com o objetivo de definir uma interface padrão que permitirá aos desenvolvedores de software implementarem programas portáteis entre múltiplos sistemas de janelas e toolkits, tais como OSF/Motif, OPEN LOOK, Microsoft Windows, OS/2 Presentation Manager e Apple Macintosh.

Esta interface, que pode ser implementada através de uma camada de software sobre os toolkits nativos de diversas plataformas, é uma interface de programação para o desenvolvimento de interfaces gráficas e está sendo chamada de UAPI - **Uniform Application Program Interface**.

O grupo de trabalho da IEEE, P1201.1, já iniciou suas atividades, e os resultados obtidos [IEE93b] estão sendo incorporados em algumas ferramentas existentes no mercado. Uma destas ferramentas, XVT, adquirida pelo CTI, é o objeto deste trabalho.<sup>12</sup>

### 3 - ARQUITETURA E FLUXO DE DESENVOLVIMENTO

O produto XVT é composto por 2 módulos principais: um editor de interfaces gráficas e uma biblioteca multi-plataforma.

O primeiro módulo, XVT-DESIGN, é um gerador de interface e de código C que auxilia ao projetista construir e testar sua interface gráfica. Este módulo também gera um arquivo que contém a definição dos elementos da interface, codificado em uma linguagem denominada de **Universal Resource Language - URL. Resources, recursos**, no âmbito de interfaces gráficas, representam informações específicas da interface, tais como rótulos de menus, posições e tamanhos de elementos, etc. A

linguagem URL só é utilizada por produtos XVT e é traduzida por um compilador também fornecido pela XVT para a linguagem de definição de recursos do ambiente alvo no qual o aplicativo irá ser executado.

O segundo módulo, a biblioteca multi-plataforma, fornece uma API com funcionalidade básica para a implementação de interfaces gráficas e está disponível entre as principais plataformas existentes: MS-Windows, Windows NT, OSF/Motif, Open Look, OS/2 Presentation Manager, Macintosh e para modo caracter para os ambientes UNIX e DOS.

A portabilidade fornecida pela biblioteca foi obtida através da implementação de uma camada de software sobre os toolkits das várias plataformas. A norma IEEE P1201.1 [IEE93b] utiliza o nome **Uniform Application Program Interface - UAPI** para se referenciar a essa interface. Essa solução é

conhecida na bibliografia pelo nome de mínimo denominador comum, uma vez que fornece suporte somente para os elementos encontrados em todas as plataformas [LEE92]. Desta forma, objetos e funções específicos de uma plataforma (como por exemplo os controles do tipo "gauge" encontrados no MOTIF) não podem ser diretamente suportados pela UAPI. O sistema XVT soluciona esse problema ao permitir o desenvolvimento de controles customizados (**custom controls**), que serão apresentados na 5ª seção. O sistema permite também que o programador acesse diretamente os elementos de um toolkit específico, mas, neste caso, a portabilidade de seu código fica comprometida.

O fluxo simplificado de desenvolvimento de um aplicativo para o ambiente MS-Windows no ambiente XVT é apresentado na Figura 1.

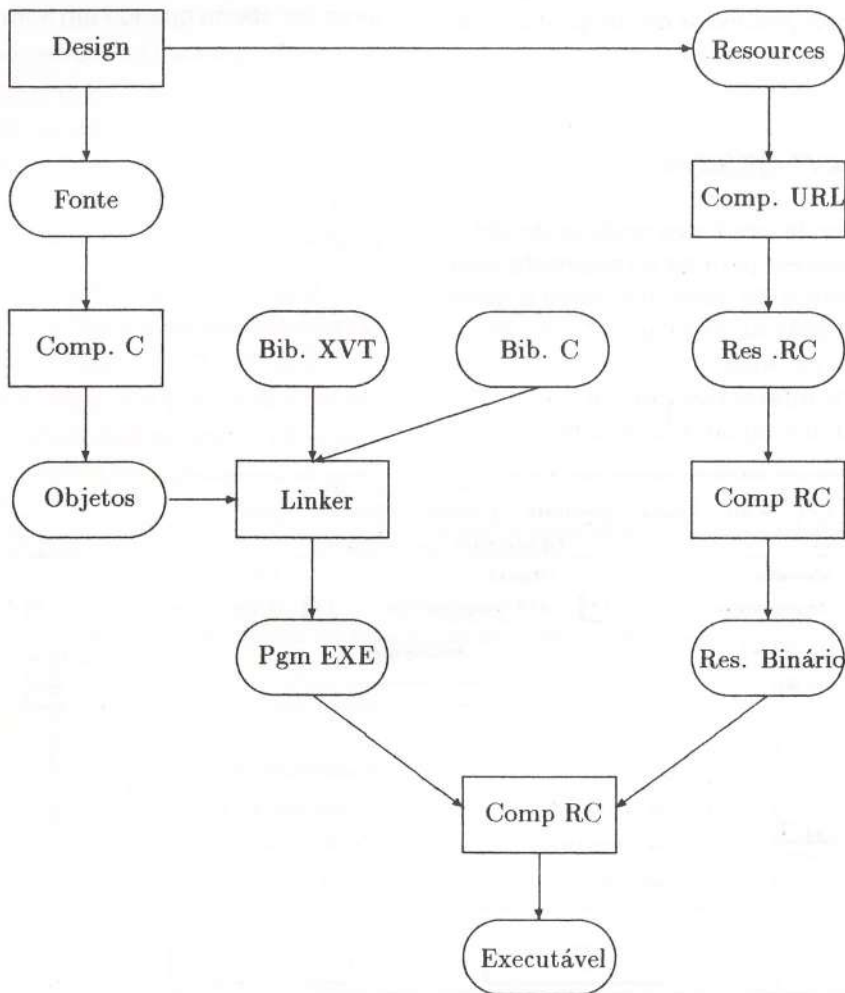


Figura 1 - Fluxo de Desenvolvimento

O aplicativo gerado pode ser distribuído sem a necessidade do pagamento de royalties.

Os testes realizados até o momento utilizaram a versão do editor gráfico que gera código C; existe outra versão que gera código C++.

Um arquivo do tipo "makefile" com os comandos para a compilação, link-edição e demais passos para a geração do aplicativo também é gerado automaticamente. O DESIGN fornece um template deste arquivo, possibilitando que o projetista da interface possa adaptá-lo, se necessário, para as características de seu projeto. Note que, com uma cópia do DESIGN disponível em um único ambiente de desenvolvimento, é possível gerar os arquivos necessários para a produção do aplicativo em todas as plataformas suportadas. Para tanto, os arquivos gerados devem ser transportados para o ambiente alvo, no qual deve haver uma cópia da biblioteca XVT, e onde os processos de compilação e de link-edição devem ser realizados novamente.

#### 4 - XVT-DESIGN

DESIGN é o editor de interfaces gráficas do sistema XVT e possui 2 funções: permitir ao projetista criar os objetos da sua interface de modo interativo e gerar um arcabouço da aplicação na linguagem C. Na literatura, uma ferramenta com esta funcionalidade recebe normalmente o nome de **Interactive Design Tool (IDT)**, editor de recursos ou prototipador de interfaces.

Usando operações básicas de um editor gráfico do tipo WYSIWYG, o projetista da interface pode criar os objetos de sua interface: janelas, menus, push-buttons, grupos, campos de edição, etc, e posicioná-los convenientemente. Cada objeto é criado com atributos padrão que podem ser alterados pelo projetista através de janelas de diálogo próprias.

Uma série de facilidades para o lay-out de interfaces é oferecida pelo DESIGN; por exemplo, o projetista pode selecionar um conjunto de controles e alinhá-los uniformemente. O projetista também pode utilizar um grid para auxiliá-lo a posicionar os controles nas janelas sendo definidas. A Figura 2 apresenta a tela do DESIGN durante a definição de uma janela de diálogo.

Enquanto o projetista vai definindo os objetos de sua interface, trechos de código C que realizam as operações padrão de acordo com as ações do projetista vão sendo gerados automaticamente pelo DESIGN. Por exemplo, se o projetista define que uma janela de diálogo deve ser aberta quando um item do menu for selecionado, o editor gera um trecho de código com a chamada da função da biblioteca XVT responsável pela abertura da janela com os parâmetros convenientemente preenchidos. A Figura 3 apresenta a tela do DESIGN com o código gerado automaticamente por esta ferramenta para a abertura da janela apresentada na Figura 2.

O projetista pode modificar ou completar o código gerado através de um editor de programas incorporado ao DESIGN. Cada objeto da interface recebe um identificador único que pode ser utilizado no código da aplicação se houver necessidade da aplicação manipular

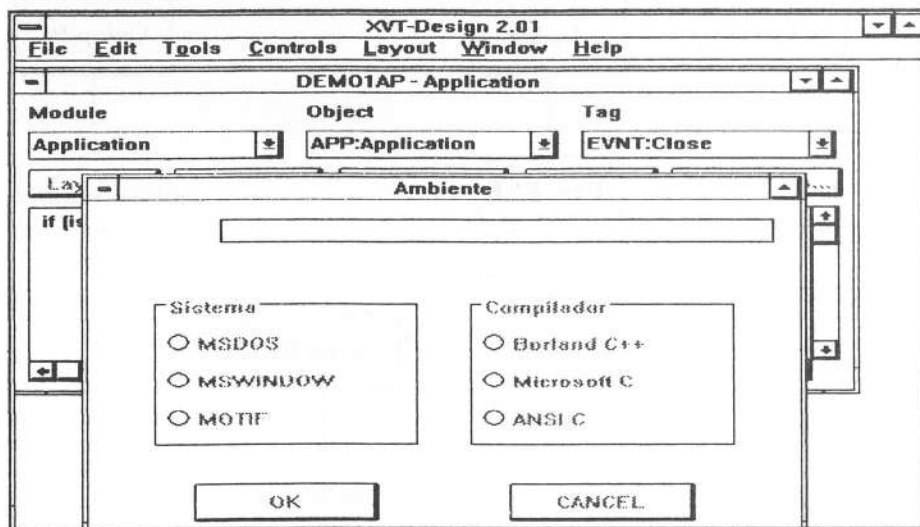


Figura 2 - Definição de Janelas Através do DESIGN

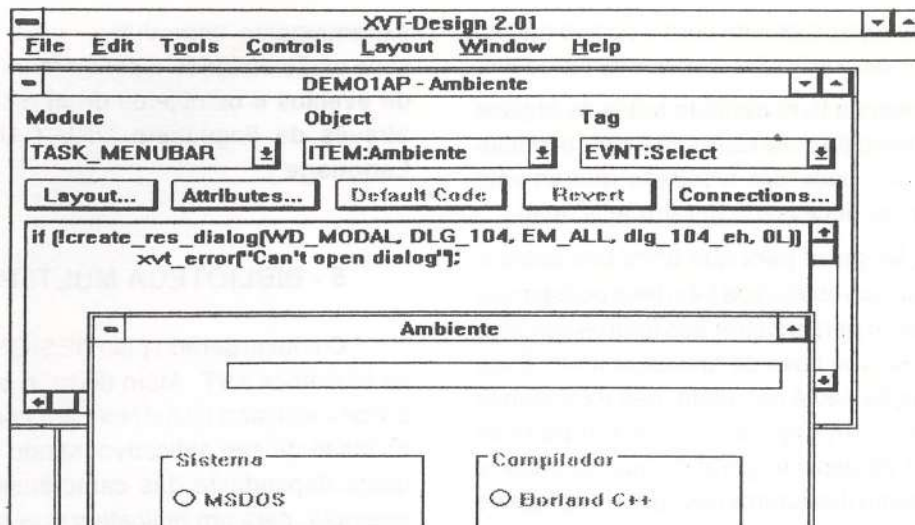


Figura 3 - Código de Abertura de Janela

os objetos da interface. Um exemplo típico deste caso ocorre quando a aplicação encontra uma situação de erro e informa essa condição ao seu usuário através de uma janela de alerta ou de erro, necessitando, para tanto, do identificador dessa janela.

O código gerado pelo DESIGN é suficiente para o controle de boa parte das operações da interface gráfica, tais como a abertura e o fechamento de janelas. Porém, alguns detalhes precisam ser acrescentados pelo projetista da interface, como por exemplo a manipulação da barra de rolagem ("scroll bar") e o sincronismo com o conteúdo da janela associada. O código acrescentado fica automaticamente incorporado ao código gerado pelo DESIGN.

Neste ponto o projetista deve ser bastante criterioso para não misturar o código relativo ao funcionamento da aplicação com o código responsável pelo controle da interface gráfica, de modo que decisões de projeto que afetem a interação com os usuários do aplicativo sejam isoladas das decisões que afetem somente a estrutura do aplicativo ou seu código correspondente. Essa independência entre o código da interface e do aplicativo, conhecida na literatura como **independência de diálogo** [HH89], é de extrema importância para o processo de desenvolvimento interativo e a futura manutenção do aplicativo. O projetista também deve usar o DESIGN para inserir chamadas para as rotinas da aplicação; no âmbito de interfaces gráficas essas chamadas são denominadas como **call-back's**.

Como um programa de interface gráfica típico, um programa XVT deve ser orientado por eventos, e deve ser estruturado de tal forma que exista um trecho de código associado a cada ação do usuário do aplicativo.

Ao passo que programas que não possuem interfaces gráficas recebem dados diretamente de seus usuários, arquivos, etc, programas com interface gráfica recebem dados na forma de eventos, tais como "mouse down", "character typed" ou "update window". Normalmente, cada evento é gerado a partir de ações realizadas pelo usuário (movimentação do mouse, manipulação de seus botões, digitação de caracteres, fechamento de janelas, ...).

Um programa orientado por eventos deve possuir uma rotina responsável por captar os eventos gerados por usuários do programa em tempo de execução, bem como as informações associadas aos eventos (posição do mouse na tela, botão que foi manipulado, tecla que foi pressionada, ...) e enviar essas informações para a rotina responsável pelo tratamento dos eventos do objeto que sofreu a ação do usuário; essa rotina é denominada na literatura como **gerente de eventos**.

O DESIGN gera um gerente de eventos para cada janela do aplicativo, e o mapeamento dos eventos para o gerente de eventos correspondente é realizado automaticamente por uma rotina de controle da biblioteca XVT.

Embora aparentemente possa parecer trabalhoso desenvolver um programa orientado por eventos, o fato do DESIGN gerar automaticamente os gerentes de eventos necessários simplifica bastante o trabalho do projetis-

ta, que deve se preocupar somente com o código necessário para responder cada ação dos usuários do aplicativo.

Quando o projetista tiver definido todos os objetos da interface e estabelecido conexões entre eles, o funcionamento da interface pode ser simulado diretamente pelo DESIGN sem ser necessário gerar o aplicativo.

Essa simulação serve para que itens tais como o tamanho e a posição dos objetos da interface possam ser verificados, bem como para mostrar aos futuros usuários do aplicativo parte de seu fluxo de funcionamento. Esse processo de simulação não é completo, não executando o código C associado com o aplicativo, mas é importante para dar uma idéia do aspecto geral de sua interface e permitir o envolvimento dos futuros usuários do aplicativo desde o início de seu desenvolvimento.

Um ponto positivo apresentado pela versão 4.0 do DESIGN é a possibilidade de trabalho em equipe, com a integração de módulos desenvolvidos separadamente. Dessa forma, se um projeto for dividido em sub-sistemas, a interface gráfica de cada sub-sistema pode ser implementada paralela e independentemente da implementação das interfaces gráficas dos outros sub-sistemas, da mesma forma que possivelmente ocorrerá com a implementação do restante dos sub-sistemas. O DESIGN oferece facilidades para a integração final das interfaces gráficas desenvolvidas separadamente.

De maneira geral, a utilização desta ferramenta é razoavelmente simples, exigindo somente um pequeno período de treinamento da parte de seus usuários. O desenvolvimento de interfaces gráficas com o XVT sem o emprego do DESIGN não é aconselhável por ser

extremamente improdutivo, uma vez que o projetista seria então obrigado a definir manualmente os gerentes de eventos e os objetos de apresentação da interface através da linguagem URL ("Universal Resource Language").

## 5 - BIBLIOTECA MULTI-PLATAFORMA

O código gerado pelo DESIGN utiliza várias rotinas da biblioteca XVT. Além disso, o programador também precisa utilizar a biblioteca para complementar a funcionalidade de seu aplicativo, sendo o esforço correspondente dependente das características do mesmo. Por exemplo, para um aplicativo que faça entrada e validação de dados o programador precisará somente desenvolver o código que apanha os valores digitados e passá-los para as rotinas de validação do aplicativo. Por outro lado, para um editor gráfico, todo o trabalho de manipulação de figuras deverá ser codificado pelo programador com a utilização das rotinas da biblioteca.

Como foi visto, a API XVT é mapeada sobre os toolkits das plataformas suportadas, de modo que as aplicações possuem o look-and-feel das plataformas nas quais são executadas.

Por exemplo, a seguinte chamada:

```
open-window ();
```

gera uma janela com o look-and-feel WINDOWS (Figura 4) ou com o look-and-feel MOTIF (Figura 5), dependendo somente do ambiente para o qual o aplicativo for gerado.

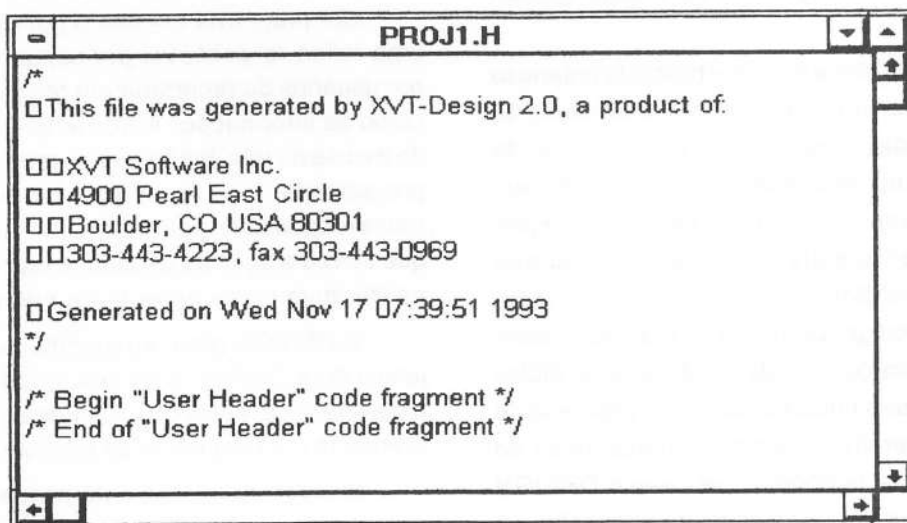


Figura 4 - Janela com look-and-feel Windows

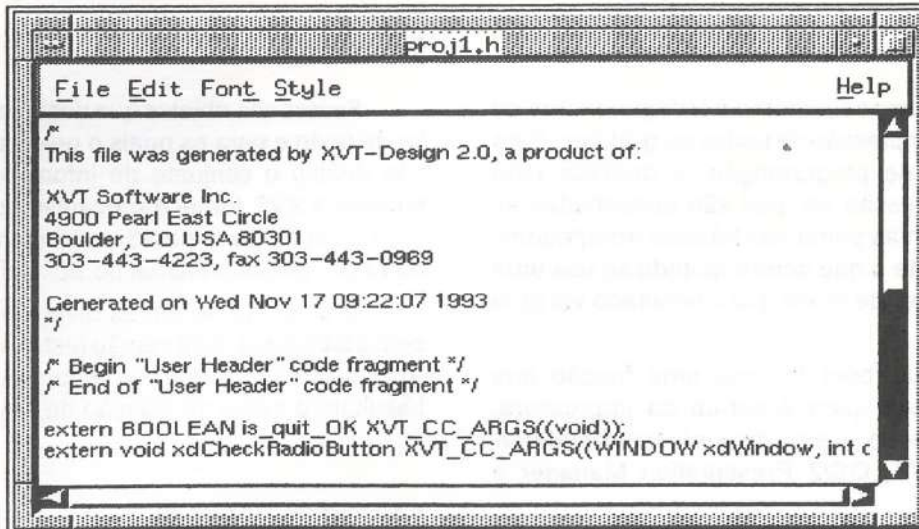


Figura 5 - Janela com look-and-feel MOTIF

Embora as barras de rolagem (scroll bar) das janelas tenham comportamentos distintos,<sup>3</sup> a rotina que trata os eventos desses objetos não precisa saber se estão recebendo eventos de uma janela WINDOWS ou MOTIF, pois em termos de programação é desenvolvido um código único, independente de plataforma.

A biblioteca dispõe de primitivas para a manipulação dos objetos básicos de interfaces gráficas:

- . Janelas
- . Controles
- . Eventos
- . Operações Gráficas
- . Menus
- . Manipulação do clip board

A biblioteca fornece também outras funções que serão sucintamente apresentadas a seguir:

### 5.1 Diálogos pré-definidos:

O XVT fornece diálogos pré-definidos para as seguintes ações:

- . solicitar uma resposta ao usuário do tipo sim/não;
- . ativar uma janela de erro ou de alerta;
- . solicitar uma cadeia de caracteres digitada pelo usuário;
- . ativar uma janela para escolha de nome de arquivo;
- . iniciar a operação de impressão

Esses diálogos são ativados a partir de rotinas da biblioteca; por exemplo, a janela mostrada na Figura 6 foi exposta simplesmente através da chamada da seguinte rotina:

open-file-dlg ();

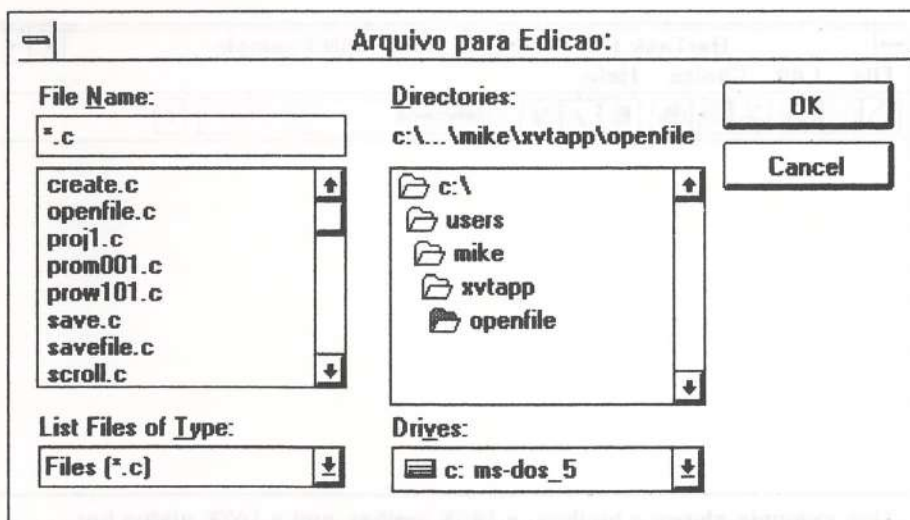


Figura 6 - Janela de Abertura de Arquivos em Windows

(3) O tamanho do controle da barra de rolagem de uma janela MOTIF varia de acordo com a proporção entre o tamanho do arquivo apresentado na janela e o tamanho da janela, enquanto que no ambiente Windows o tamanho do mesmo elemento é fixo.

## 5.2 Suporte às operações de impressão:

A biblioteca fornece primitivas independentes de plataforma para a impressão de textos ou gráficos. Para tanto, em termos de programação, é definida uma janela do tipo impressão na qual são desenhadas as figuras e textos com as primitivas básicas apropriadas, de forma semelhante a que ocorre quando se usa uma janela gráfica normal, de modo que o resultado vai para a impressora.

A biblioteca também fornece uma função que apresenta uma janela para o set-up da impressora; atualmente esta função só está disponível nas plataformas MS-WINDOWS, OS/2 Presentation Manager e Macintosh.

## 5.3 Compilador HELP:

O sistema fornece facilidades para a produção de janelas de auxílio com explicações para os usuários finais dos aplicativos.

Para tanto, é necessário que antes da geração do aplicativo seja definida uma lista de tópicos com uma explicação associada a cada tópico em um arquivo do tipo texto. Este arquivo é transformado pelo utilitário CHELP, distribuído junto com o sistema XVT, de forma que o resultado obtido possa ser utilizado em tempo de execução.

Embora portátil, o formato do arquivo de mensagens de auxílio obtido é bastante rudimentar, fornecendo um único nível de auxílio, não permitindo o emprego de gráficos nem de hipertexto.<sup>4</sup>

## 5.4 Fontes:

Fontes são objetos que designam um estilo particular de texto e para os quais o grupo de trabalho da IEEE não definiu o conjunto de informações associadas. A biblioteca XVT fornece mecanismos para que os usuários dos aplicativos possam selecionar fontes a partir da barra de menus principal do aplicativo.

O conjunto de fontes disponíveis varia de acordo com a biblioteca, e na versão testada é bastante limitado. Aplicativos que rodam no ambiente DOS-caracter não habilitam o menu de seleção de fontes.

## 5.5 Custom Controls:

Além de fornecer os elementos gráficos básicos de interfaces gráficas, o sistema permite que se desenvolvam novos controles customizados para atender necessidades específicas. Isto é obtido através de objetos do tipo "custom controls". Um objeto desse tipo, obtido a partir de primitivas existentes, também é portátil entre plataformas. Atualmente algumas firmas já comercializam alguns objetos deste tipo, e a própria XVT desenvolveu e comercializa à parte objetos custom control do tipo tabela ou planilha.

A existência de objetos desse tipo confere ao sistema XVT a característica de **extensibilidade**, definida em [HH89] como a possibilidade de incluir novos elementos gráficos aos elementos suportados diretamente pela ferramenta empregada para produção da interface gráfica.

A Figura 7 apresenta uma janela cuja barra de ícones foi obtida através de um objeto customizado.

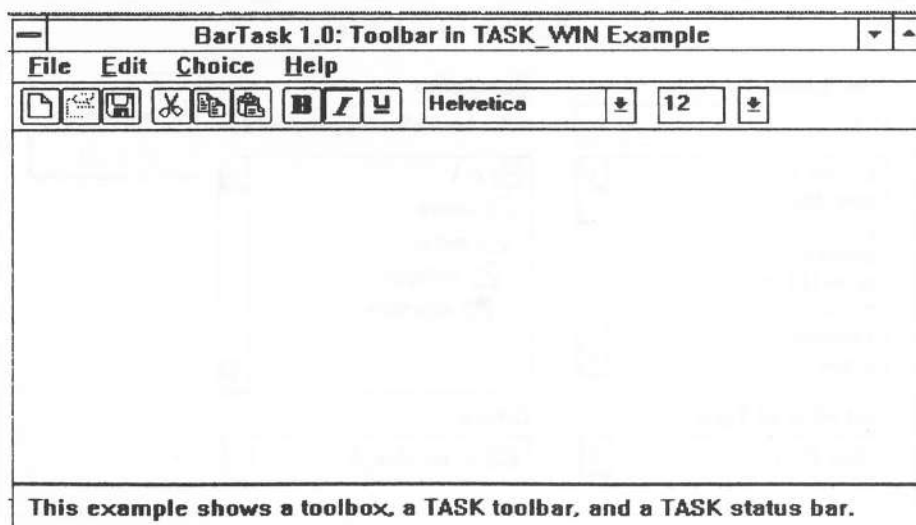


Figura 7 - Exemplo de Uso de Objeto Customizado

(4) A versão mais recente do sistema, 4.0, passou a gerar janelas de auxílio do tipo hipertexto.

## 5.6 Ícones:

Embora as operações de manipulação de ícones estejam disponíveis nas bibliotecas de todas as plataformas, esses objetos são tratados de maneira não portátil pela biblioteca XVT. O fato do formato destes objetos ser dependente das plataformas justifica, em parte, esta restrição, que deverá ser eliminada das futuras versões da biblioteca.

## 5.7 Suporte a operações específicas de plataformas:

A biblioteca possui uma rotina que fornece uma interface consistente mas não portátil para utilizar a funcionalidade específica das plataformas. Cada plataforma oferece códigos de escape que podem ser utilizados pelos aplicativos para realização de ações específicas da plataforma. Esses códigos são passados como parâmetros para a rotina **xvt-escape**, e para cada código estão definidos argumentos que também devem ser passados para essa rotina.

Por exemplo, para os ambientes MS-WINDOWS e Macintosh existem o código de escape **xvt-esc-get-printer-info** que devolve informações à respeito da resolução da impressora.

Uma segunda maneira de se manipular aspectos específicos das plataformas é através da rotina **xvt-set-value** com atributos não portáteis. Por exemplo, para o ambiente MS-WINDOWS, um aplicativo pode definir e usar fontes específicas deste ambiente com o uso dessa rotina.

Essas possibilidades, embora impactem a portabilidade dos aplicativos, são importantes por encapsularem os toolkits originais das plataformas e permitem que o sistema XVT possua a característica de **escapabilidade**, definida em [HH89] como a possibilidade de acessar o toolkit da plataforma alvo para implementar elementos não suportados diretamente pela ferramenta empregada para produção da interface gráfica.

## 6 - TESTES REALIZADOS:

Com a finalidade de verificar a funcionalidade oferecida pelo sistema bem como a portabilidade do código gerado pelo editor de interfaces DESIGN foram definidos e implementados alguns protótipos.

A escolha dos mesmos foi realizada de modo a se cobrir os principais elementos de uma interface gráfica.

Todos os testes foram desenvolvidos com o editor de interfaces do sistema.

Interfaces simples, que utilizem somente janelas de diálogos, radio-boutões, check-boxes e campos de edição, são rapidamente implementadas e podem ser obtidas sem um grande trabalho adicional de programação ao código gerado pelo DESIGN.

As operações gráficas, desenhos de figuras, por exemplo, também podem ser facilmente programadas e não exigem um profundo conhecimento da biblioteca.

As operações de rolagem de tela (scroll) não são implementadas automaticamente, de modo que devem ser programadas manualmente. Embora a quantidade de código que precisa ser desenvolvido não seja grande, a documentação relacionada com este assunto não é muito clara, de modo que foi necessário um tempo razoável para que esta operação pudesse funcionar corretamente em nosso protótipo.

A manipulação de fontes de caracteres de uma maneira portátil e independente de plataforma é um problema complexo; a funcionalidade fornecida pela biblioteca não permite se obter, por exemplo, todos os recursos fornecidos pelo ambiente MS-WINDOWS. Este é um ponto no qual a possibilidade de se utilizar diretamente primitivas específicas das plataformas pode justificar a perda de parte da portabilidade do aplicativo gerado.

A biblioteca não fornece rotinas para validação de campos de entrada de dados nem facilidades do tipo "picture". Embora a implementação dessas funções pelo projetista da interface não seja uma tarefa complexa, não deixa de ser um fator limitante em termos de produtividade.

Os manuais são simples, mas muitas vezes é necessário se recorrer aos exemplos distribuídos com o sistema para se obter uma visão completa de determinadas operações, como por exemplo a operação de rolagem de tela.

O primeiro protótipo desenvolvido permite que seus usuários, utilizando uma opção da barra de menus, abram 2 janelas de diálogo. As Figuras 8 e 9 apresentam uma das janelas do protótipo abertas respectivamente nos ambientes MS-WINDOWS e OSF/Motif.

Não foi programada nenhuma rotina de consistência para os dados digitados, e quando o botão "OK" é selecionado os mesmos são apresentados em uma janela.

Figura 8 - Janela do Protótipo em Ambiente Windows

O aplicativo foi desenvolvido inicialmente no ambiente MS-WINDOWS com o compilador Borland C/C++ versão 3.1 e foi transportado para os ambientes OSF/MOTIF e DOS-Character sem alterações em seu código fonte.

A eficiência dos programas obtidos é satisfatória não tendo sido percebido nenhum impacto no desempenho dos mesmos; o tamanho dos executáveis é mostrado a seguir:

- Ambiente -	- Tamanho -
MS-WINDOWS	31.170K
DOS	419.004K
Motif	1.146.800K

No ambiente MS-WINDOWS, os executáveis gerados utilizam bibliotecas DLL fornecidas junto com o sistema, o que explica o tamanho obtido.

O sistema tem problemas para apresentar os objetos da interface no ambiente DOS-Character quando os mesmos não têm sua origem coincidente com a origem de um caracter na tela. Como a interface deste protótipo foi desenvolvida com o DESIGN no ambiente MS-WINDOWS foi necessá-

rio que a opção de grid estivesse ligada com o tamanho das células igual a 1 caracter.

## 7 - CONCLUSÕES E PERSPECTIVAS - PRÓXIMOS PASSOS

Os testes realizados até o momento foram suficientes para a avaliação da ferramenta; os principais resultados serão apresentados a seguir.

A portabilidade da biblioteca foi bastante testada e nunca foi encontrado nenhum problema de transporte de suas rotinas, exceto pelo fato dela não suportar a manipulação de ícones de uma maneira independente de plataforma. Isso significa que grande parte do problema de transporte do código do aplicativo fica resolvida, restando ao programador tomar os cuidados necessários para garantir que seu código também pode ser transportado para outros ambientes sem problemas. Nesse campo, a utilização do padrão POSIX é fundamental, e técnicas como as apresentadas em [LEV91] e [ADR90] são de extrema importância.

Embora o código da aplicação possa ser transportado sem problemas, nem sempre o mesmo ocorre com alguns elementos da interface, pois em determinadas situações o tamanho ou a posição desses elementos precisam ser ajustados quando a

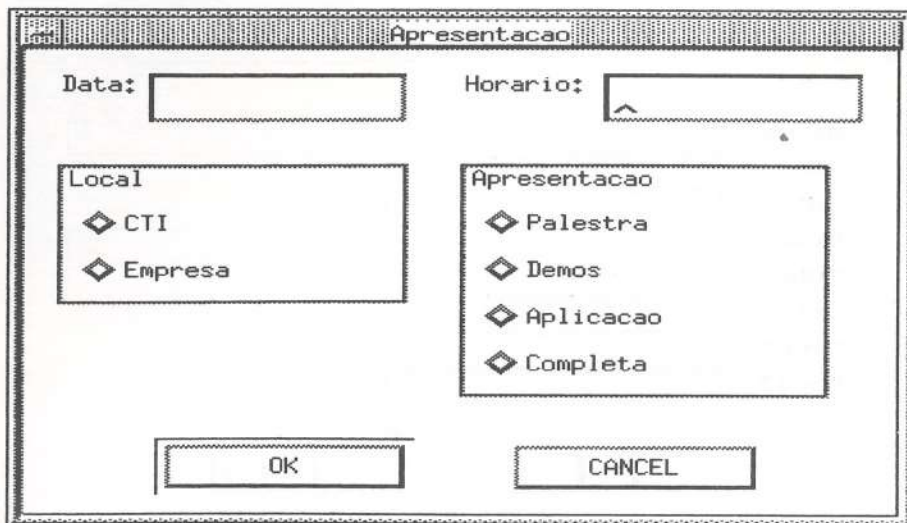


Figura 9 - Janela do Protótipo em Ambiente Motif

interface muda de plataforma. A Figura 10 apresenta uma janela de diálogo originalmente definida pelo Design no ambiente MS-WINDOWS. Note que quando o aplicativo foi transportado para o ambiente OSF/MOTIF os campos de entrada de dados ficaram sobrepostos, conforme pode ser observado na Figura 11.

Essa situação ocorre porque o DESIGN gera os elementos da interface com o look-and-feel da plataforma na qual ele está sendo executado; no exemplo apresentado, a relação entre a distância e a altura dos campos de edição apresentava um aspecto agradável no ambiente MS-WINDOWS, o que deixou de ocorrer quando o aplicativo foi transportado para o ambiente MOTIF. Existem soluções simples e parciais para esse problema, como por exemplo a manutenção de um arquivo com a definição dos elementos da interface em cada plataforma para qual o aplicativo é transportado. De qualquer forma, o fato do sistema não possuir um gerente de geometria, que permitiria que a distância entre elementos da interface pudesse ser definida, não deixa de ser uma limitação.

Nenhum dos testes realizados apresentou problemas em termos de desempenho, não tendo sido identificada nenhuma situação na qual a camada da biblioteca XVT tenha provocado degradação no tempo de resposta do aplicativo desenvolvido, de modo que o desempenho da interface gráfica é dependente do desempenho do toolkit no qual o aplicativo será executado. Os testes realizados também constataram que a biblioteca é suficientemente eficiente para a implementação de editores gráficos em máquinas PC 386/486.

A qualidade das interfaces geradas foi bastante satisfatória, e em nenhum dos testes realizados foi encontrado algum elemento que pudesse identificar a utilização ou não da biblioteca XVT. De forma análoga do que ocorre

com a eficiência dos aplicativos, a qualidade da interface também é dependente dos toolkits nativos.

Um ponto que, de certa forma, compromete a qualidade dos aplicativos gerados é o gerenciamento de fontes. A biblioteca XVT fornece mecanismos para o usuário dos aplicativos selecionar fontes a partir da barra de menus principal, mas o conjunto de fontes disponíveis varia de acordo com a biblioteca. Por exemplo, a biblioteca para o ambiente WINDOWS fornece somente as fontes Modern, Roman, Script, Courier, Helvética e Times Roman, nos estilos Bold, Italic e Underline, de modo que usuários de aplicativos desenvolvidos pelo XVT para a plataforma MS-WINDOWS não poderão contar com os outros fontes que por ventura estejam instalados no ambiente.

Embora o sistema XVT seja suficientemente genérico para poder ser utilizado em qualquer área de aplicação, algumas ferramentas existentes são mais voltadas para áreas específicas, permitindo uma maior produtividade para o desenvolvimento de sistemas nas mesmas.

Por exemplo, para o desenvolvimento de aplicações do tipo cliente-servidor para as plataformas MS-WINDOWS e Motif que manipulem bases de dados através da linguagem SQL existem no mercado ferramentas que fornecem uma produtividade bastante superior à oferecida pelo sistema XVT, uma vez que tratam essas áreas de aplicação de forma mais completa. Aplicações genéricas, entretanto, continuam sendo melhor tratadas como um produto mais abrangente como o XVT.

O tempo necessário para que um programador utilize o editor de interfaces DESIGN de uma maneira produtiva é razoavelmente curto (aproximadamente uma ou duas semanas). O domínio total da biblioteca que possui cerca de 300 rotinas, é mais demorado, levando entre 2 e 3 meses, dependendo principalmente da expe-

**CADCLN.DBF**

**Nome:**

**RG:**

**CIC:**

**Rua:**

**Cidade:**

**CEP:**

**Fone1:**

**Fone2:**

|< < > >|

**Figura 10** - Janela com look-and-feel Windows

riência que o programador possua no desenvolvimento de aplicativos com interfaces gráficas.

A experiência do CTI permite afirmar que, embora exista um custo associado, é técnica e economicamente viável o desenvolvimento de interfaces gráficas multi-plataformas.

O Programa de Qualidade e Produtividade em Software do CTI continuará a absorver tecnologia para a produção de interfaces gráficas portáteis e a transferir esta tecnologia para a indústria nacional de software. Entre as metas propostas nesta área, destacam-se as seguintes:

#### · Tecnologia de Objetos

Além de ser considerada uma tecnologia avançada para a produção de software, o desenvolvimento de interfaces gráficas foi um dos primeiros campos nos quais a tecnologia de objetos foi empregada com sucesso. A decisão de se iniciar o estudo do sistema XVT com sua versão não orientada a objetos foi motivada pelo fato de se procurar inicialmente atuar em um patamar tecnológico mais próximo ao praticado em nível nacional. Procurando avançar neste patamar está prevista a compra da versão do sistema XVT orientada a objetos.

#### · Teste de Interfaces Gráficas

Testar aplicativos com interface gráfica é uma atividade mais trabalhosa do que testar aplicativos com interface baseada em caracteres [THE92], [MYE93] e [HS93]. Existem no mercado algumas ferramentas desenvolvidas para automatizar esta

atividade. O PQPS já adquiriu uma ferramenta deste tipo e está iniciando sua avaliação.

#### · Métodos

Devido ao forte conteúdo semântico associado a uma interface gráfica, seu projeto é mais complexo do que o projeto de uma interface tradicional [LR93], [MAY92]. Espera-se iniciar o desenvolvimento de uma metodologia que busque indicar os principais passos que devem ser seguidos para a obtenção de uma interface gráfica de qualidade.

Um projeto deste tipo desenvolvido em uma instituição de pesquisa com as características do CTI, que tem como um dos seus objetivos induzir e apoiar a introdução de tecnologia nas empresas nacionais, só pode obter sucesso se conseguir repassar efetivamente seus resultados para o setor produtivo do país. O PQPS escolheu como método preferencial para repassar sua experiência o desenvolvimento de projetos conjuntos com empresas nacionais.

A atividade de divulgação do ambiente para a produção de interfaces gráficas bem como a busca de empresas interessadas em desenvolvimentos conjuntos deverá continuar dentro das atividades do PQPS. Os contatos já realizados com associações como a Assespro e o Programa Softex 2000 apresentaram resultados iniciais animadores e deverão ser aprofundados. O CTI procurará facilitar o acesso dos pequenos produtores de software aos resultados obtidos através de, por exemplo, convênios com o SEBRAE e com o Programa SOFTEX 2000.

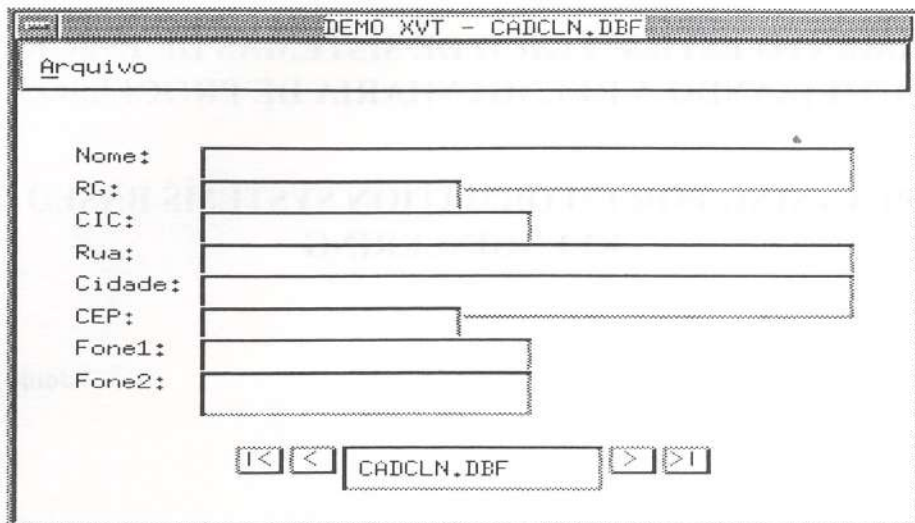


Figura 11 - Janela com look-and-feel Motif

O CTI espera que o caminho de integração com a indústria nacional de software seja percorrido com sucesso e se coloca à disposição dos interessados para os contatos que forem necessários.

Agradecimentos:

A José Carlos Maldonado, pelas sugestões e incentivos dados durante o desenvolvimento deste trabalho.

### REFERÊNCIAS BIBLIOGRÁFICAS

[ADR90] D. Keppel A. Dolenc, A Lemmke and G. V. Reilly. **Notes on writing portable programs in C.**, Technical report, CSE, University of Washington, Novembro 1990.

[API94] Steve Apiki. **Paths to platform independence.** Byte, 19(01): 172 - 178, 1994.

[BA] L. Bass and G. Abowd. **Issues in the evaluation of user interfaces tools.** Technical report, Software Engineering Institute.

[HH89] H. Rex Hartson and Debora Hix. **Human-computer interface development: Concepts and systems for its management.** ACM Computer Surveys, 21(1) 5 - 92, 1989.

[HS91] Debora Hix and Robert S. Schulman. **Human-computer interface development tools: a methodology for their evaluation.** Communications of the ACM, 34(3) 75 - 87, 1991.

[HS93] Ellis Horowitz and Zafar Singhera. **Graphical user interface testing.** Technical report, Department of

Computer Science - University of Southern California, 1993.

[IEE93a] IEEE. **P1003.0 Draft 16.1 - Draft Guide to the POSIX Open System Environment,** October 1993.

[IEE93b] IEEE. **P1202.1/Draft 7 - Draft Standard for Information Technology-Uniform Application Program Interface - Graphical Users Interfaces,** March 1993.

[LEE92] Geoff Lee. **Object-Oriented GUI Application Development.** PTR Prentice-Hall, 1992.

[LEV91] Donald Levine. **POSIX Programmer's Guide - Writing Portable UNIX Programs.** O Reilly and Associates, 1991.

[LR93] Clayton Lewis and John Rieman. **Task-Centered User Interface Design.** Livro em Domínio Público disponível via ftp anônimo em ftp.cs.colorado.edu, 1993.

[MAY92] Deborah J. Mayhew. **Principles and Guidelines in Software User Interface Design.** PTR Prentice-Hall, 1992.

[MR92] Brad A. Myers and Mary B. Rosson. **Survey on user interface programming.** Technical report, Computer Science Department - Carnegie Mellon University, Fevereiro 1992.

[MYE93] Brad A. Myers. **Why are human-computer interfaces difficult to design and implement.** Technical report, Computer Science Department - Carnegie Mellon University, Julho 1993.

[THE92] Lee The. **Stress tests for gui programs.** Datamation, pages 5 - 92, 1992.

---

# PLANEJAMENTO ESTRATÉGICO DE SISTEMAS DE INFORMAÇÕES UTILIZANDO A REENGENHARIA DE PROCESSOS

## STRATEGIC PLANNING FOR INFORMATION SYSTEMS BASED ON PROCESS REENGINEERING

Jorge Luis CORDENONSI\*

### ABSTRACT

Strategic Planning for Information Systems is traditionally not integrated with formal or not formal Strategic Planning of the Organization. Besides, it is usually driven and managed only by Information Systems experts, without any end-user participation. So, Organizations that do not have this integration, will have additional difficulties in being successful on implementing Strategic Planning for Information Systems, due to excessive complex methodologies, its very theoretical approaches, and having as their main goal the huge Organizations. Beyond this scenario, the present work main objective, is to show that process reengineering is emerging as a very effective alternative for prioritizing the Strategic Planning for Information Systems, making it be considered as a fundamental activity in Organizations of any size. Taking in account this approach, the Information Systems area will have a new role, a new organizational structure and, in this way, will become a strategic element of the Organization.

**KEY WORDS:** Information Systems (IS); Process Reengineering; Strategic Planning; Strategic Planning for Information Systems (SPIS), Information Technology.

### RESUMO

Tradicionalmente o Planejamento Estratégico de Sistemas de Informações não está integrado ao Planejamento Estratégico da Organização (formal ou informal), e geralmente é conduzido apenas por especialistas da área de Sistemas de Informações, praticamente sem nenhuma participação dos usuários finais. Além disso, as Organizações que não possuem esta integração, dificilmente chegam a alcançar o sucesso na implementação do Planejamento Estratégico de Sistemas de Informações, devido a fatores agravantes ligados à complexidade das metodologias existentes, aos enfoques excessivamente teóricos das mesmas, e com seus desenvolvimentos voltados principalmente para Organizações de grande porte. Levando em conta este cenário, o presente trabalho tem como objetivo principal, mostrar que a reengenharia de processos surge como uma excepcional alternativa, para fazer com que o Planejamento Estratégico de Sistemas de Informações seja priorizado e passe a ser considerado uma atividade fundamental na Organização, independente de seu porte. Com esta ferramenta, a área de Sistemas de Informações passará a ter um novo papel, uma nova estrutura organizacional e irá tomar-se um elemento estratégico na Organização.

**PALAVRAS-CHAVE:** Sistemas de Informação (SI); Reengenharia de Processos; Planejamento Estratégico; Planejamento Estratégico de Sistemas de Informações (P. E. S. I.), Tecnologia da Informação.

(\*) Gerente de Sistemas de Informação da S/A Têxtil Nova Odessa

## 1. INTRODUÇÃO

O surgimento de novas tecnologias e a necessidade da busca de vantagens competitivas, formam a nova lógica de transformação do P. E. S. I., onde técnicas tradicionais de planejamento estão ultrapassadas, e os responsáveis pelo planejamento chegam a conclusão que precisam reformular suas funções, para atingirem o potencial competitivo da era da informação. Os responsáveis pelo P. E. S. I. estão forçados pela importância crescente, nos negócios, dos S. I., a mudarem o escopo, os objetivos, o estilo, as fontes de especialização e a natureza do valor do planejamento. Embora as novas diretrizes não tenham ainda produzido uma metodologia universal única e integrada, muitas organizações estão hoje apresentando tendências novas e incrivelmente dominantes para o P. E. S. I.. Mas a maior parte das organizações que se interessam na implementação de um P. E. S. I., encontram inúmeras dificuldades [SUL87], entre as principais:

a) As metodologias atuais de P. E. S. I. são complexas, de difícil implementação e não abordam o tamanho da organização, ou seja, não descrevem como proceder a sua implementação em organizações de diferentes portes, uma vez que as mesmas possuem um enfoque voltado principalmente para organizações de grande porte;

b) Falta de credibilidade no P. E. S. I., por tentativas infrutíferas no passado de desenvolvimento do tradicional PDI (Plano Diretor de Informática). Porque além de não existir uma integração ao negócio da organização, o enfoque era apenas direcionado a Tecnologia da Informação;

c) Durante o desenvolvimento do P. E. S. I., geralmente ocorre a participação apenas de especialistas da área de S. I.;

d) Não existe um comprometimento da alta direção da organização no processo de P. E. S. I.;

e) O difícil controle da ansiedade generalizada na organização em ver resultados práticos durante o processo de P. E. S. I.;

f) Geralmente o que a alta direção da organização quer saber é o custo, o retorno do investimento e a tecnologia a ser escolhida;

g) A existência de barreiras nas organizações que não possuem um Planejamento Estratégico formal, em

demonstrar para a alta direção a real necessidade da integração do P. E. S. I. com o mesmo.

### 1.1. Conceituação

Com o objetivo de uniformizar os conceitos utilizados neste artigo, onde enfocamos uma importante área do Gerenciamento de S. I., que é o P. E. S. I., e principalmente mostrar as diferenças entre S. I. e Tecnologia da Informação, iremos conceituar o Planejamento Estratégico, o Sistema de Informação, a Reengenharia de Processos e o P. E. S. I..

#### 1.1.1. Planejamento Estratégico

Planejamento Estratégico é o planejamento conduzido pelos objetivos e pelas necessidades do negócio e orientado para o fornecimento de melhores condições de competição à organização [CHA92].

#### 1.1.2. Sistema de Informação

O Sistema de informação é hoje um elemento indispensável para dar apoio às operações e a tomada de decisão na organização moderna. Ainda não foi encontrada uma definição que seja universalmente aceita para o termo sistema de informação, utilizamos uma que ao nosso entendimento reflete de forma integrada os reais elementos de um Sistema de Informação.

Sistema de Informação é uma combinação estruturada de Informação, Recursos Humanos, Tecnologias da Informação e Práticas de Trabalho, conforme Figura 1, organizados de tal forma a permitir o melhor atendimento dos objetivos da organização. As práticas de trabalho podem ser consideradas como os métodos utilizados pelos Recursos Humanos para desempenharem suas tarefas. A Informação pode consistir de dados formatados, textos, imagens e som. As tecnologias da Informação consistem do hardware e software que executam as tarefas de processamento de dados, tais como capturar, transmitir, estocar, recuperar, manipular ou exibir dados concernentes [PRA94].

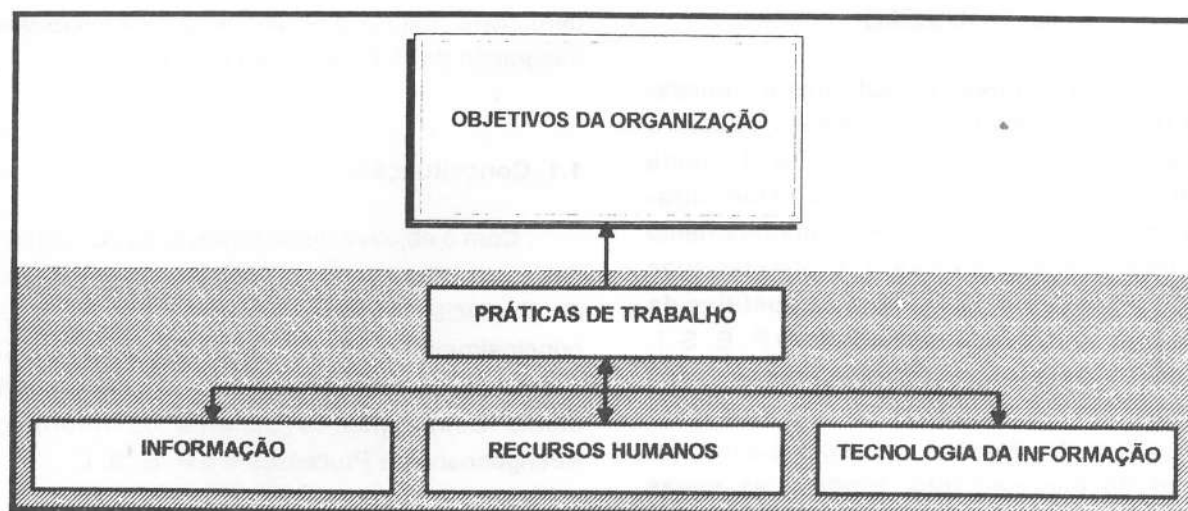


Figura 1 - Representação gráfica de um Sistema de Informação

### 1.1.3. Reengenharia de Processos

Reengenharia de Processos é melhorar radicalmente um conjunto de atividades estruturadas e medidas destinadas a resultar num produto especificado para um determinado cliente ou mercado [DAV94], ou seja, é melhorar radicalmente os processos chaves da organização através de uma abordagem sistemática [JOH93].

### 1.1.4. Planejamento Estratégico de Sistemas de Informações

Com base nos conceitos anteriores, podemos definir o P. E. S. I. como sendo a união dos componentes: Planejamento Estratégico e Sistema de Informação, sendo um processo onde se definem os S. I. necessários de forma integrada ao negócio de uma determinada organização.

Para [FUR91], o P. E. S. I. deve refletir as funções e dados necessários para suportar o negócio, os objetivos, os fatores críticos de sucesso, e as necessidades de Informação da alta administração da organização. Da mesma forma, deve retratar como a Tecnologia da Informação pode ser utilizada para criar novas oportunidades ou vantagens competitivas.

No contexto deste artigo, a Reengenharia de Processos irá ser utilizada como ferramenta para o desenvolvimento e implementação do P. E. S. I., indicando o portfólio de S. I. necessários para o negócio da organização.

## 2. A EVOLUÇÃO DO P. E. S. I.

O P. E. S. I. se caracteriza pela tradição de mudanças. Durante os últimos anos, as alterações foram tão frequentes que o planejamento às vezes parecia apenas um modismo, porém, o presente artigo mostra que o surgimento de idéias importantes de planejamento geralmente coincide com avanços da Tecnologia da Informação e dos S. I..

Inicialmente, o P. E. S. I. ou como tradicionalmente conhecido P. D. I. - Plano Diretor de Informática, abordava pontos-chaves envolvidos na administração de uma função empresarial desconhecida que estava sendo desenvolvida em um ambiente de computador central. O planejamento da criação de um centro de dados, do desenvolvimento dos primeiros S. I., e da crescente conscientização dos usuários, adotou a noção de desenvolvimento organizacional segundo a qual todas as organizações seguiram, acertadamente, uma trajetória semelhante na adoção da Tecnologia da Informação com uma série previsível de problema e soluções características. Tais padrões de mudança são conhecidos como estágios de crescimento de Nolan [SUL87].

O gerenciamento dos estágios de crescimento predominou quando muitas organizações limitavam seu investimento na Tecnologia da Informação a um computador com S. I. contábeis e processamento centralizado de dados. No final dos anos 60, as coisas mudaram porque o número de S. I., transações e o backlog dos projetos de desenvolvimento e expansão aumentaram

drasticamente. Como conseqüência, surgiu uma nova técnica de planejamento de sistemas que deixou de enfatizar os sistemas e o computador para considerar os próprios dados como o principal recurso, dirigindo a atenção da gerência para a importância dos dados no aumento da flexibilidade e produtividade. A metodologia de Planejamento de Sistemas da IBM (BSP-Business Systems Planning) é a mais popular dentre os processos de planejamento orientado para os dados [IBM-84].

Mais recentemente, os S. I. começaram a se preocupar com as necessidades do trabalho gerencial e profissional. Os responsáveis pela tomada de decisão receberam uma enxurrada de informações gerada pelos S. I.. Em resposta, surgiu um terceiro conceito de planejamento, no final dos anos 70 onde os S. I. deviam dar suporte a tomada de decisões através de dados e análises de indicadores importantes de desempenho. O planejamento de S. I. deveria, portanto, se concentrar na identificação de fatores críticos de sucesso [ROC79].

Muitas outras técnicas de planejamento derivam ou são uma combinação dessas três técnicas originais. Embora todas tenham contribuído para tornar o planejamento mais interativo e útil, mudanças recentes no uso dos S. I. fizeram com que as metodologias tradicionais se tornassem insatisfatórias. Por exemplo, as técnicas tradicionais de planejamento antecedem a crescente função estratégica dos S. I. em relação ao negócio da organização. Além disso, os métodos tradicionais de planejamento geralmente eram indicados para ambientes onde a Tecnologia da Informação era aplicada de forma mais centralizada. Segundo a IBM a metodologia BSP não sofreu atualizações e ainda está sendo indicada somente para ambientes centralizados.

Constatamos que diante das metodologias e abordagens existentes para o P. E. S. I. identificam-se problemas e respostas inovadoras de planejamento, adotadas por várias organizações. Tais tendências são as precursoras de uma nova disciplina de planejamento para a era da informação. Como ainda não conseguimos descrever de forma definitiva esse novo ambiente de planejamento, este artigo demonstra uma proposta de evolução na forma de planejar S. I. utilizando para isso uma ferramenta inovadora que é a Reengenharia de Processos.

### 3. A ESTRUTURA DO P. E. S. I.

A partir das metodologias e abordagens estudadas e analisadas descritas anteriormente durante a evolu-

ção, uma das etapas fundamentais do processo de P. E. S. I. é definir as etapas em que o mesmo deverá abordar, ou seja, definir uma abordagem de trabalho, adaptando as metodologias a realidade da organização.

A primeira etapa do processo de P. E. S. I. deve ser o planejamento do P. E. S. I. em seguida a transformação do conjunto de estratégias da organização para a obtenção dos requisitos de Informação a nível organizacional. Os requisitos de informação são derivados diretamente dos objetivos da organização. Para isso, deve existir uma integração entre o P. E. S. I. e o Planejamento Estratégico da organização, gerando a missão, objetivos e estratégias para a área de S. I., por exemplo, se o objetivo da organização é de aumentar os lucros e a estratégia selecionada é mudar o plano de vendas, o sistema de informação necessário para este objetivo poderá ser um sistema que possibilite a análise do plano de vendas frente a contribuição marginal de cada produto.

BLUMENTHAL foi um dos primeiros autores a apresentar o P. E. S. I. como similar ao Planejamento Organizacional. Inúmeros autores advogam, hoje em dia, este como sendo o método ideal para se planejar S. I.. Nesta etapa o objetivo principal é de transformar o conjunto de estratégias da organização em um conjunto de estratégias para S. I. [DIAS-85]. Portanto, para que os S. I. contribuam para os objetivos da organização o P. E. S. I. deve estar integrado ao Planejamento Estratégico da organização formal ou informal. A não integração pode conduzir o P. E. S. I. ao insucesso, pois dificilmente refletirá as estratégias da organização.

#### 3.1. O Planejamento do P. E. S. I.

Uma das etapas fundamentais no desenvolvimento do P. E. S. I., consiste em elaborar um planejamento do P. E. S. I., ou seja, uma vez definida a necessidade da elaboração do P. E. S. I., os seguintes passos se fazem necessários:

##### 3.1.1 A formação do Steering Committee

A formação do Steering Committee é necessária para gerenciar todo o processo do desenvolvimento e implementação do P. E. S. I.. Um dos grandes objetivos da formação deste Comitê é de fazer com que a área de S. I. tome conhecimento das estratégias de negócios da organização, que geralmente é do conhecimento apenas da alta direção da organização, integrando desta forma

o P. E. S. I. com o Planejamento Estratégico da organização [WYS90]. Também neste Comitê a alta direção da organização deve ter o conhecimento das possibilidades de aliar as Tecnologias da Informação e os Sistemas de Informação aos seus objetivos estratégicos. Basicamente as atribuições definidas para o **Steering Committee**, são as seguintes:

a) Estabelecer as prioridades dos trabalhos a serem executados pela área de S. I., entre as diversas áreas da organização;

b) Avaliar as propostas de investimentos e determinar o nível de recursos para S. I. e Tecnologia da Informação;

c) Definir e aprovar políticas para a utilização dos S. I. e Tecnologia da Informação;

d) Definir e avaliar o P. E. S. I.;

e) Definir os grupos de usuários que estariam auxiliando o P. E. S. I.;

f) Avaliar o desempenho da área de S. I..

### 3.2. Estruturação da arquitetura de S. I.

Acompanhando a evolução dos S. I., o P. E. S. I. deve contemplar em uma de suas principais etapas a estruturação da arquitetura de S. I. necessárias para a organização [IAD93], a mesma deve englobar os seguintes S. I., conforme Figura 2.

#### (I) EDP (Electronic Data Processing)

Tradicionalmente são os primeiros S. I. desenvolvidos. Sua função principal é dar suporte operacional a tarefa de registrar determinadas operações, com a finalidade de controle ou para atender exigências fisco-contábeis. Esses sistemas têm seu foco nos dados (grande volume de dados e pequeno volume de informações), pelo que não são ferramentas adequadas no processo decisório, o qual necessita manipular informações.

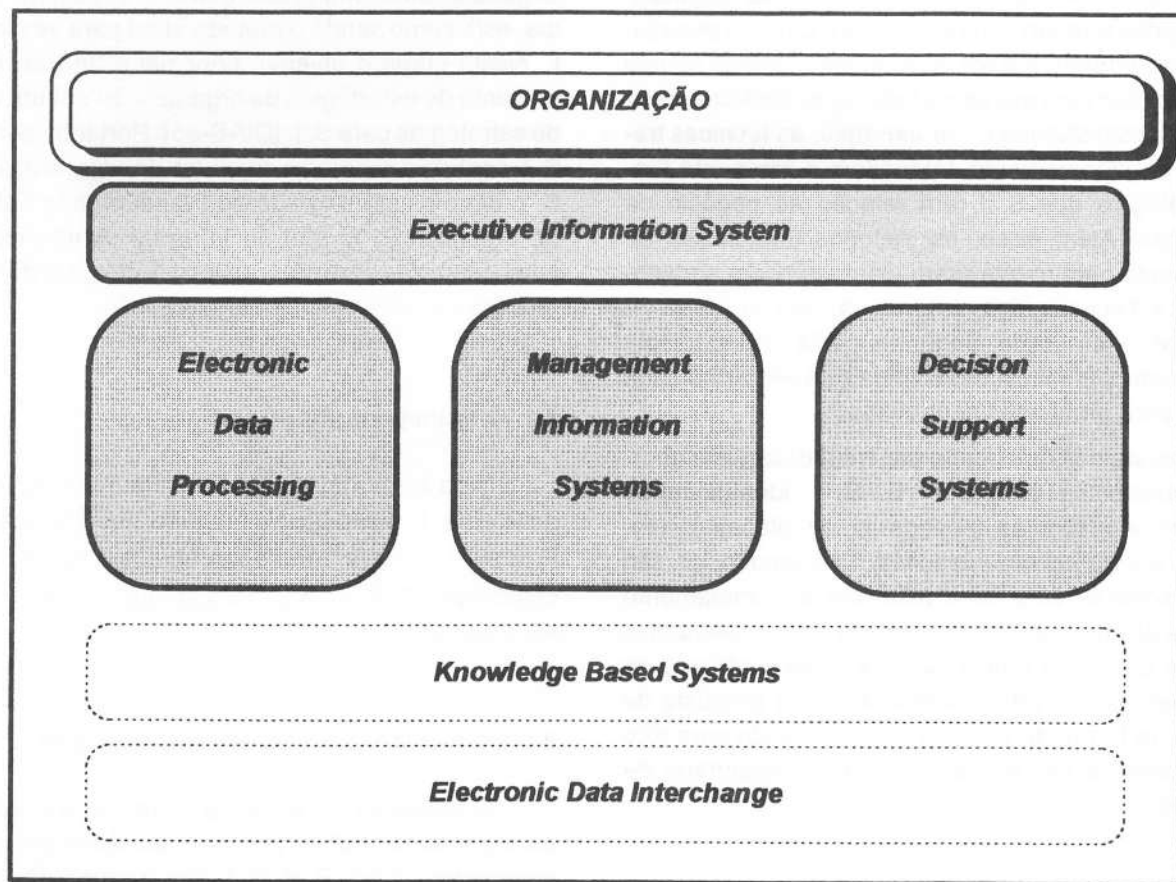


Figura 2 - Arquitetura de Sistemas de Informações

## (II) MIS (Management Information Systems)

Os sistemas de gerenciamento de informações atendem às necessidades gerenciais mas ainda são bastante voltados para fluxo de informações e arquivos de dados. Todavia já têm um equilíbrio entre o volume de dados e o de informações, pois têm foco centrado nas informações. Em suma, é um sistema formal numa organização, que fornece à gerência os relatórios necessários para serem utilizados no processo de tomada de decisão.

## (III) DSS (Decision Support Systems)

Trata-se de uma categoria de sistemas apropriados para a tomada de decisão nos níveis mais altos da organização. É um sistema interativo que proporciona ao usuário, acesso fácil a modelos decisórios e dados a fim de dar apoio às atividades de tomada de decisões semi-estruturadas ou não estruturadas. Em suma, pode ser qualquer Sistema de Informação que consiga transformar dados em informações e suporte o processo de tomada de decisão, complementando desta forma os S. I. EDP e MIS.

## (IV) EIS (Executive Information Systems)

São S. I. para os executivos, com foco na decisão, baseado em informações internas e externas da organização, dando suporte a gestão empresarial. O EIS é um Sistema de Informação rápido, dinâmico, flexível, personalizado, de fácil utilização e integração, que recebe, processa e fornece pequenos volumes de informações de vários tipos, que sejam comparativas ou não a partir de uma variedade enorme de fontes de dados internas e externas de diferentes formatos quer seja na forma gráfica, textual, de imagem, etc.

## (V) EDI (Electronic Data Interchange)

É o intercâmbio eletrônico de dados de negócio, estruturados em um formato padronizado, entre os S. I. de parceiros comerciais, com o mínimo de intervenção manual.

## (VI) KBS (Knowledge Based Systems)

São sistemas que simulam o processo de resolução de problemas de especialistas humanos em um determinado domínio de conhecimento. Têm como objetivo principal tomar o conhecimento de especialistas disponíveis para outros e suportar o processo de resolução de problemas, em áreas onde o conhecimento de um especialista se faz necessário.

A construção da arquitetura de S. I. de uma organização segue uma abordagem proposta e tem suas bases conceituais em comum com as metodologias existentes

de P. E. S. I.. Define-se inicialmente a missão da organização, que estabelece seu propósito mais amplo. Em seguida definem-se os objetivos da organização, que deverão ser a base e apresentar compatibilidade com a missão estabelecida. Estabelecidos a missão e objetivos da organização, levanta-se o modelo organizacional de sustentação, de forma a identificar os executivos dos diversos níveis da organização, que estabelecerão os objetivos, metas e desafios de cada área funcional.

As metas podem ser definidas como resultados quantificados que se esperam atingir para cada objetivo. Os desafios são quantificações e ações mais específicas das metas estabelecidas. Definidos os objetivos, metas e desafios, devem-se determinar os fatores críticos de sucesso, que são aquelas coisas que devem andar corretamente na organização, para que os objetivos dos diversos níveis da organização sejam alcançados. Em seguida, cada executivo relacionará as suas necessidades de Informação, que darão suporte aos fatores críticos de sucesso definidos para a sua área.

Durante toda a fase de estruturação da arquitetura de S. I., a ferramenta fundamental para redefinir e organizar os processos que darão origem as necessidades de S. I. é a Reengenharia de Processos. De acordo com [EPS94] a reengenharia de processos é um tema presente no mundo atual dos negócios, o relacionamento entre reengenharia de processos e Tecnologia da Informação é muito importante, principalmente na utilização como estratégia pelas organizações até o final da década. [HAM94] define a reengenharia de processos como o "repensar fundamental e o replanejamento radical dos processos de negócios para alcançar melhorias dramáticas nas atuais medidas críticas de performance, tais como custo, qualidade, serviço e velocidade".

Um ponto em que [HAM94] e [EPS94] diferem é na abordagem com relação a reengenharia de processos. [HAM94] descreve a "reengenharia como sendo colocar velhos sistemas de lado e recomeçar tudo". [EPS94], ao contrário, acredita que "jogar tudo fora e começar de novo é a mesma coisa que tentar reconstruir um avião em pleno vôo". Ele acrescenta que o procedimento correto é construir os novos sistemas fora do antigo ambiente e conectá-los novamente aos antigos sistemas, separando alguns dos sistemas existentes, de forma que se possa continuar a colher seus benefícios. O importante é não jogar fora os que tenham utilidade prática para a organização. Na aplicação da reengenharia de processos em casos práticos, o que se observa é que a solução a ser adotada poderá ser a de [HAM94] ou a de [EPS94], o que vai indicar a melhor

solução são as características do processo em questão.

A partir das necessidades de informações que darão suporte aos fatores críticos de sucesso definidos pelos executivos mais os processos identificados, analisados e redesenhados utilizando a Reengenharia de Processos, obtém-se o portfólio de S. I. necessários, conforme Figura 3.

#### 4. A NOVA ESTRUTURA DA ÁREA DE S. I.

O processo de reestruturação da área de S. I. é necessário, para fazer com que o P. E. S. I. ressuscite. Para isso, o enfoque deve ser os processos da organização, e uma nova ferramenta deve escolhida para isso, a reengenharia de processos. Para se conseguir grandes ganhos em desempenho é necessário iniciar uma reformulação de toda a maneira de realizar o trabalho, baseado no novo conceito que a organização

deve se organizar em torno dos seus processos básicos em vez de criar uma estrutura organizacional ao redor de funções ou departamentos [BYR94] Desta forma a área de S. I. passa a ter uma nova-estrutura contando agora com a participação de pessoas envolvidas nos processos básicos da organização sejam eles burocráticos, industriais ou específicos da organização. Para implantação desta nova estrutura para a área de S. I., é necessário a redefinição das atividades e objetivos da área.

#### 5. A IMPLEMENTAÇÃO DO P. E. S. I.

A abordagem proposta de elaboração do P. E. S. I. baseia-se na utilização de uma estrutura organizacional diferente para a área de S. I., subdividindo-se em: processos e tecnologia, com o objetivo de estabelecer uma compreensão comum do negócio e identificar as necessidades de informação que suportem os fatores

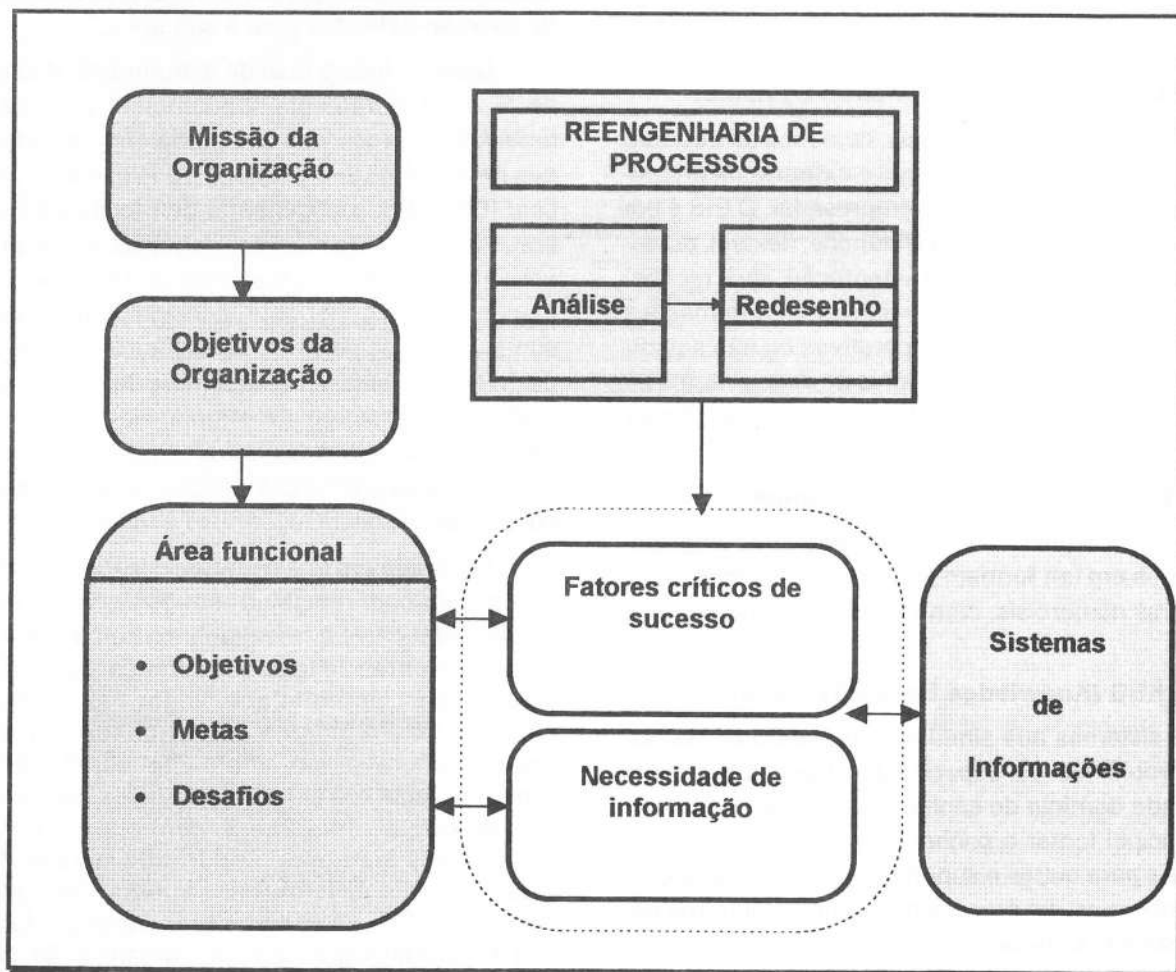


Figura 3 - Abordagem para a construção da estrutura de S. I.

críticos de sucesso da organização. O enfoque integra os conhecimentos do negócio, as necessidades de informação e uso dos S. I., através do envolvimento e comprometimento dos usuários e alta administração. O escopo de estudo para o P. E. S. I. pode ser uma organização, uma divisão, um departamento, ou mesmo uma seção. Este planejamento se dá através da indagação e resposta de três questões básicas:

- . Onde estamos?
- . Para onde iremos? e
- . Como iremos?

A questão "Onde estamos?" visa determinar o posicionamento atual da organização no mercado, incluindo uma análise dos negócios e verificação da efetividade dos S. I. atuais da organização. A questão "Para onde iremos?" visa determinar as expectativas dos executivos com relação ao futuro da organização, incluindo a sua participação futura de mercado e necessidades adicionais de negócio. Para essa questão a determinação do horizonte estratégico a ser adotado, por ser dinâmico, não deve ser considerado como diretriz para a elaboração do P. E. S. I., uma vez que as prioridades mudam constantemente fazendo com que o P. E. S. I. torne-se uma atividade diária da área de S. I..

Por fim, a questão "como iremos" visa determinar os meios e recursos necessários para que a organização atinja os seus objetivos de curto, médio e longo prazos em termos de S. I. [FUR91].

## 6. CONCLUSÃO

Diante do fato que a atividade de P. E. S. I. é essencialmente orientada para o futuro direcionando decisões no curto prazo visando o atendimento de objetivos de longo prazo, torna-se necessário considerar o efeito que a rápida evolução tecnológica trará para a área de S. I..

O surgimento de novas tecnologias da informação e S. I. e a integração dessas tecnologias vem propiciar a formação de ambientes distribuídos, nos quais se verifica a mudança da ênfase da gerência orientada para a tecnologia para a gerência orientada a informações, a qual, segundo NOLAN, representa alcançar estágios mais evoluídos da área de S. I. nas organizações. A necessidade de compreender esta realidade e de modificá-la se torna premente à medida que as organizações necessitam aumentar seus níveis de produtividade e de

competência, o que atualmente só é possível mediante o uso eficiente e eficaz do recurso **INFORMAÇÃO**. Neste cenário conclui-se que:

- a) Os P. E. S. I. das organizações são fortemente dependentes do tamanho da organização e do porte da área de S. I.;
- b) A ausência de um P. E. S. I. ou mesmo a existência de algum esforço aplicado em Planejamento de S. I., poderá resultar em S. I. que não contribuam ao negócio da organização, usuários frustrados, custos elevados, fazendo com que a área de S. I. não represente um elemento estratégico da organização;
- c) A aplicação da Reengenharia de Processos é necessária durante o desenvolvimento do P. E. S. I. a fim de agilizar o seu desenvolvimento e conduzir os S. I. as necessidades reais do negócio da organização, tornando-se desta forma uma evolução das metodologias de planejamento existentes;
- d) O P. E. S. I. se faz necessário independente do porte da organização;
- e) O papel e a estrutura organizacional da área de S. I. sofrerá alterações, passando a ser reconhecida como um elemento estratégico da organização;
- f) Através da integração do P. E. S. I. com o Planejamento Estratégico da organização, será possível atrelar os esforços da área de S. I. aos objetivos da organização;
- g) A simples aplicação de uma metodologia "pronta", sem adaptações para a realidade da organização envolvida, poderá resultar em mais um P. E. S. I. deficiente, complexo e que dificilmente sairá do papel.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [BYR94] BYRNE, John A., **A corporação horizontal**, Gazeta Mercantil, 1994.
- [CHA92] CHAVES, Eduardo, **Anotações de aula da disciplina Gerenciamento de Sistemas de Informações**, Mestrado em Informática - Gerenciamento de Sistemas de Informações, Pontifícia Universidade Católica de Campinas, São Paulo, aula 4, 1992.
- [DAV94] DAVENPORT, Thomas H., **Reengenharia de Processos: Como inovar a empresa através da tecnologia de informação**, 2ª Edição, Editora Campus, Rio de Janeiro, 1994.
- [DIA85] DIAS, Donaldo de Souza, **O Sistema de Informação e a Empresa**, Editora Livros Técnicos e Científicos, 1985.

- [EPS94] EPSTEIN, Bob & BARRY John A. & STRUKHOFF Roger, **Reengenharia e Tecnologia de Informação**, SYBASE - A Revista da Arquitetura Cliente/Servidor, abril/maio 1994, p. 12-18.
- [FUR91] FURLAN, José Davi, **Como Elaborar e Implementar Planejamento Estratégico de Sistemas de Informação**, Makron Books do Brasil Editora, São Paulo, 1991.
- [HAM94] HAMMER, Michael & CHAMPY, James, **Reengenharia: revolucionando a empresa em função dos clientes, da concorrência e das grandes mudanças da gerência**, Editora Campus, Rio de Janeiro, 1994.
- [IAD93] IADEROZA, Luís A., **Anotações de aula da disciplina Sistemas de Informação para gerenciamento**, Mestrado em Informática - Gerenciamento de Sistemas de Informações, Pontifícia Universidade Católica de Campinas, São Paulo, 1993.
- [IBM84] IBM Corporation, **Information Systems Planning Guide: Business Systems Planning: Information Systems Planning Guide**, Fourth Edition, GE20-0527-4, IBM, 1984.
- [JOH93] JOHANSSON, Henry J., **Business Process Reengineering - Breakpoint Strategies for market dominance**, John Wiley & Sons, New York, 1993, p. 57-83.
- [PRA94] PRATES, Maurício, **Conceituação de Sistemas de Informação (SI) do Ponto de Vista do Gerenciamento**, Revista do Instituto de Informática - PUCCAMP, volume 2, número 1, março/setembro 1994, p. 7-12.
- [ROC79] ROCKART, John F., **Chief executives define their own data needs**, Harvard Business Review, volume 57, número 2, março/abril 1979, p. 81-93.
- [SUL87] SULLIVAN, Jr., Cornelius H., **Redefinindo o Planejamento Estratégico de Sistemas**, MIS - O Relatório de Gerenciamento da Informação, novembro 1987, p. 35-42.
- [WYS90] WYSOCKI, Robert K. & YOUNG, James, **Information Systems: Management Principles in Action**, John Wiley & Som, New York, 1990.

---

# GESTAÇÃO DA POLÍTICA NACIONAL DE INFORMÁTICA: de JK ao Governo Figueiredo

## THE FORMULATION OF BRAZIL'S NATIONAL INFORMATICS POLICY: FROM JK TO FIGUEIREDO'S GOVERNMENT

Paulo de Martino JANNUZZI\*

### ABSTRACT

The present article presents a historical summary of the national effort to formulate an informatic industry policy during the last 3 decades, showing its vinculation with the national development goals and bringing out the importance of the governmental agencies on the process. The origins of process of formulation of this policy goes back to the 60's when the military importance of technological domain of computer architecture became clear. On the 70's, specially after II PND ( 2nd National Development Plan), federal government perceived the strategical value of informatic industry to the National Prosperity, as much for international trade and for the technological improvement of Brazil's manufacturies. Besides the bureaucracy and authoritarianism of the agencies responsible to conduct the process, the national effort to formulate a informatic policy allowed the qualification of considerable number of technicians and the development of a national knowhow that puts Brazil in a distint position in some international markets of hardware and software.

**KEY WORDS:** National Informatics Policy; Brazil; Technological Development; Informatics.

### RESUMO

O presente artigo apresenta uma síntese histórica do processo de formulação da Política Nacional de Informática nas últimas 3 décadas, mostrando sua vinculação com os objetivos nacionais de desenvolvimento e resgatando o papel das agências governamentais no processo de desenvolvimento da Informática Brasileira. Originado a partir de motivações estratégico-militares nos anos 60, o processo de formulação de uma Política Nacional de Informática passou a receber menção substantiva dentro da estratégia nacional de desenvolvimento apenas a partir do II PND (1974/79). A partir de então, a informática passou a ser assimilada como um recurso estratégico para viabilizar a Prosperidade Nacional, como fonte de divisas geradas pela substituição das importações e pela exportação de componentes aqui produzidos, e como insumo para a atualização tecnológica do parque industrial brasileiro, a fim de mantê-lo em níveis de competitividade com o mercado produtor internacional. Em que pese os vieses autoritários e excessos regulamentatórios das agências encarregadas da formulação da política e gerenciamento dos programas de desenvolvimento tecnológico, a PNI permitiu a formação de quadros técnicos de alta qualificação e a acumulação de uma massa crítica que coloca, hoje, o Brasil em condições de competitividade internacional em vários nichos de mercado de equipamentos e software.

**PALAVRAS-CHAVE:** Política Nacional de Informática; Brasil; Desenvolvimento tecnológico; Informática.

(\*) Professor FCECA/PUCCAMP, Mestre em Administração Pública na EAESP/FGV.

## 1. INTRODUÇÃO

Os porta-vozes do ideário neoliberal, que transitam com invejável desenvoltura até nos círculos acadêmicos, têm se ocupado em propagar que a experiência de Planejamento Governamental no Brasil e na América Latina é a grande responsável pelo atraso em que se encontram nossas sociedades. No entendimento deles, tivesse o Mercado a liberdade irrestrita para agir conforme sua "mão invisível" melhor julgasse, o quadro social e econômico seria muito diferente, e melhor!

Não nos ocuparemos em rebater as simplificações e falta de visão histórica do processo de desenvolvimento que tais porta-vozes manifestam. Tavares (1988) já demonstrou, de modo brilhante, a incoerência e inconsistência deste discurso. Nossa contribuição é, neste sentido, mais pontual.

Especificamente, nosso objetivo neste artigo é o de apresentar uma síntese histórica do processo de formulação da Política Nacional de Informática nas últimas 3 décadas, mostrar sua vinculação com os objetivos nacionais de desenvolvimento e resgatar o papel das agências governamentais no processo de desenvolvimento da Informática Brasileira. Ocupamo-nos, neste momento, de uma descrição do processo e não de um balanço dos aspectos positivos e negativos da PNI. Isso vem sendo encaminhado por diversos autores, e demanda um texto específico. Mas, em última instância, o que se procura mostrar é que, em que pese o viés tecnocrático que marcou o processo de formulação desta política - assim como todas as outras nos governos militares -, o legado proporcionado para a Informática Brasileira é muito superior àquilo que teria sido possível se as "forças do mercado" tivessem agido livremente no período.

## 2. ANTECEDENTES DA PNI

Embora o setor de informática tenha recebido menção substantiva dentro da estratégia nacional de desenvolvimento apenas a partir do II PND (1974/79), as origens do processo de formulação de uma política nacional para a área datam de meados dos anos 60, quando começa haver um reconhecimento, por parte do governo, da importância estratégica do desenvolvimento tecnológico nacional.

Durante o período que se estende de 1950 até boa parte da década de 70, a política nacional de desenvolvimento econômico se pautou na expansão dos investimentos em infra-estrutura de transportes e energia (prin-

cipalmente energia elétrica e petróleo). Os documentos oficiais deste largo período, que compreende vários governos diferentes, ressaltam, em seus diagnósticos, a necessidade destes fatores para viabilizar o crescimento industrial do país, e por conseguinte, elevar a renda nacional e o padrão de bem-estar da população.

A importância da tecnologia como fator dinâmico da produção ainda não era plenamente reconhecida ou pelo menos não figurava em posição de destaque. No Plano de Metas de JK, o que havia de referência à política tecnológica nacional era uma ressalva para a necessidade de expandir a oferta de energia através de usinas nucleares. Essa preocupação foi preservada nos planos subsequentes.

No curto governo de João Goulart, já parece haver um reconhecimento explícito da questão tecnológica, embora o provimento de energia elétrica ainda se constituísse no principal fator para aumento da produtividade. No Plano Trienal de Desenvolvimento Econômico e Social para o período de 1963 a 1965, figurava um diagnóstico de que a composição dos investimentos (em equipamentos ou em construções) variara na década anterior. E acrescentava:

*"Em primeiro lugar, comprova-se o processo de tecnificação, pela ascendência dos equipamentos em geral. Em segundo lugar, o início da mecanização agrícola, que seguramente tenderá a intensificar-se. Em terceiro, o aumento relativo da oferta de energia elétrica, que permite a substituição de formas menos eficientes de energia e o aumento geral da produtividade do trabalho."*[pag.50]

O PAEG, retratando o direcionamento "laissez-faire" de seus idealizadores, não contemplou a questão tecnológica na órbita do planejamento federal. Os interesses estratégico-militares de preservar a integridade, unidade e soberania nacionais, somados a necessidade de expandir a fronteira do desenvolvimento econômico levam a incorporação das Telecomunicações na estratégia nacional de crescimento.

A partir de 1968, a questão tecnológica passa a ter posição de destaque nos planos governamentais. No documento-síntese do Programa Estratégico de Desenvolvimento há um capítulo exclusivo para o tema ("O Progresso Tecnológico") onde se diagnostica que "Será difícil encontrar experiência de algum país em que o crescimento rápido e auto-sustentado não tenha sido apoiado por um processo interno de desenvolvimento tecnológico..."[pag.121]. O debate sobre o progresso tecnológico, que nos meios universitários já vinha desde

os anos 50, tem seu conteúdo incorporado na esfera do governo federal. O progresso tecnológico passa a ser entendido em três dimensões mais amplas:

- do ponto de vista do desenvolvimento industrial, passa a ser encarado como um fator de produção tão necessário quanto o capital e trabalho;

- do ponto de vista estratégico, passa a ser cada vez mais importante a capacitação tecnológica nacional, a fim de assegurar a soberania nacional;

- do ponto de vista conceitual, passa a ser entendido não apenas como novos equipamentos, novos insumos ou novos métodos de produção e mas também novas e melhores formas de organização (tecnologia administrativa).

A informática, dentro deste contexto, ainda não adquirira dinâmica e importância próprias dentro do "Progresso Tecnológico". No Governo Costa e Silva, a questão da informática vinha no bojo do Subprograma de "Modernização das Indústrias Eletrônicas, de Navios, Aeronaves e Infra-estrutura de apoio", coordenado pela Marinha de Guerra. Rodrigues [1984b:16] coloca que nesta época "... a Marinha se dá conta de que, para manter em funcionamento navios modernos, precisaria de uma aparelhagem eletrônica que não tínhamos capacidade nem de produzir nem para manter." Formava-se, assim, a consciência nas Forças Armadas de que a produção local de componentes eletrônicos (primeiro passo para a indústria nacional de informática) era fundamental para garantir a independência e a soberania nacional.

A motivação estratégico-militar para criação da indústria de computadores não foi uma especificidade brasileira, não foi uma distorção engendrada pela "doutrina de segurança nacional" dos protagonistas do movimento de 64. A própria origem dos computadores na década de 40 nos EUA foi "fruto das necessidades geradas pelas Forças Armadas norte-americanas durante a II Guerra Mundial." [Cobra 1988]. O que existiu de específico foi o atraso em reconhecer a importância deste ramo industrial, que desde os anos 50, no Japão, e desde os anos 60 na França e Inglaterra, já recebia pesados investimentos governamentais.

### 3. INÍCIO DO PROCESSO DE FORMULAÇÃO DA PNI

A partir dos anos 70, a informática começa a ser colocada como um tema destacado nas referências

governamentais sobre política tecnológica e industrial. O início é tímido, mas adquire grande relevância em menos de 5 anos.

A timidez do início deriva da origem subordinada da área às diretrizes dos programas estratégicos da Marinha. Mas já nos primeiros anos da década de 1970, o Ministério de Planejamento passa a reconhecer a importância do setor tecnológico da informática para a Estratégia de Desenvolvimento Econômico. Assim tanto no documento "Metas e Bases para a Ação de Governo", que definiu as diretrizes da atuação do Governo Médici nos dois primeiros anos, como I PND (1972/74), a indústria Eletrônica recebeu menção como uma das áreas prioritárias para o desenvolvimento científico e tecnológico nacional, ao lado da pesquisa espacial e energia nuclear. Ao menos, como item programático, a informática deixava de ser apenas uma preocupação da Marinha.

Em 1971, por iniciativa da Marinha, que queria uma solução rápida para a produção nacional de componentes eletrônicos para suas fragatas, e apoio da SEPLAN, que ansiava por uma oportunidade de financiar o desenvolvimento de tecnologia de ponta, foi criado o GTE, Grupo Especial de Trabalho, com o objetivo de promover o projeto, desenvolvimento e construção de protótipos de computadores eletrônicos para aplicação naval. As medidas concretas não tardaram a aparecer, e através dos recursos liberados pelo BNDE, O GTE assinou contrato com a USP em 1972, para desenvolvimento de um microcomputador em dois anos, e estabeleceu convênio com a PUC-RJ, para desenvolvimento de software.

As divergências entre a Marinha, coordenadora do GTE, e os técnicos da SEPLAN e BNDE, viabilizadores dos recursos financeiros, surgiram tão logo se iniciaram os contatos com grupos multinacionais estrangeiros para formação de uma joint-venture nacional. A Marinha defendia a associação com uma companhia inglesa, Ferranti, que já fornecia equipamentos para modernização da esquadra. A SEPLAN e BNDE achavam que o mais adequado seria a fabricação de computadores de uso geral - que a Ferranti não tinha - para viabilizar a sustentação financeira da empresa emergente. Havia um grande mercado potencial para absorção de computadores no Brasil, que nas projeções do Ministro Reis Velloso teria "crescimento anual da ordem de 30%" [Rodrigues 1984, pag.30]. Julgavam, assim, que a união com a Fujitsu seria mais vantajosa.

Os embates se intensificaram no interior do GTE, até que em 1974, criou-se a Cobra - Computadores e Sistemas Brasileiros S.A., com participação acionária do BNDE, a Ferranti inglesa e a empresa privada nacional

Equipamentos Eletrônicos (E.E.), com o objetivo de fabricar o computador Argus 700, um equipamento de uso em controle de processos industriais. O que pode ter aparecido uma vitória da Marinha, logo se transformou em malogro, quando já em 1975, a Cobra se viu em dificuldades de comercializar o seu produto. A Marinha perdia, com este desenlace, papel preponderante no direcionamento da política industrial de informática.

No documento "Plano Básico de Desenvolvimento Científico e Tecnológico 1973/74" a desvinculação da política de desenvolvimento industrial da informática dos programas da Marinha já parece mais nítida. A indústria de minicomputadores é um projeto prioritário dentro do Programa Especial de Tecnologia Industrial, e sua "produção está projetada inicialmente para atender ao mercado interno, mas existem boas perspectivas de exportação". [pag.61]. O enfoque é claramente diferente: a informática não é apenas mais um instrumento estratégico ou mais um fator de produção para a economia, mas um bem para capitalização de divisas.

Mas o enfraquecimento da Marinha, e o conseqüente ocaso do GTE, na definição dos rumos da informática deve-se também, e talvez com maior intensidade, às iniciativas institucionais da SEPLAN na regulação do setor. Crescia no interior do governo, o reconhecimento do impacto social e econômico trazido pelo uso dos computadores. Nos dizeres do próprio Ministro do Planejamento, as projeções de expansão do mercado mundial e no Brasil "indicam a necessidade de uma planificação que estabeleça as diretrizes e metas para o setor, no contexto nacional..." [Rodrigues 1984, pag.30].

Já em 1972, o Ministro Reis Velloso conseguia a aprovação presidencial para criação da CAPRE, Comissão de Coordenação das Atividades de Processamento Eletrônico, com objetivo de racionalizar o uso de computadores na administração pública federal, levantar o parque computacional instalado no Brasil, opinar sobre novas aquisições, coordenar programas de treinamento na área e propor medidas para formulação de uma política de financiamento governamental para a atividade de processamento de dados.

A Capre tinha uma composição mais pluralista que o GTE, compreendendo representantes da SEPLAN (presidente), do EMFA, do Ministério da Fazenda, do BNDE, do SERPRO e IBGE.

Com tais atribuições, e dotada de representatividade ministerial e poder político (conferido pela SEPLAN), a

Capre começa a reunir condições institucionais para formulação de uma política setorial na área de informática.

#### 4. INSTITUCIONALIZAÇÃO DO PROCESSO

Isto é o que de fato ocorre durante o Governo Geisel. Embora se chegue ao final do mandato sem um documento sistematizado em forma de Lei Nacional, muitas iniciativas institucionais foram tomadas, delineando a política oficial no setor de informática. Neste período a Capre torna-se um órgão permanente de reflexão e ação sobre o tema.

Quando da elaboração do II PND, havia, nos meios acadêmicos, nas Forças Armadas e na própria tecnocracia federal, um relativo consenso de que o domínio nacional da tecnologia de componentes eletrônicos só se efetivaria com a criação de empresas de capital predominantemente brasileiro. Esta constatação está espelhada no II PND, no capítulo sobre Estratégia Industrial e Agropecuária, nos seguintes termos:

*".. será implantada a indústria e transferida efetivamente a tecnologia, no campo da eletrônica digital. Isso se fará pela implantação da indústria brasileira de minicomputadores, sob controle de capital nacional, pela fabricação de processadores de centrais eletrônicas de comutação, na área de telecomunicações, e pela implantação de sólida indústria nacional de componentes eletrônicos sofisticados, como os circuitos integrados."* [pag. 40]

Fica assim estabelecido a opção estratégica de constituir um modelo nacionalista no campo da informática, iniciando pelo fabrico de minicomputadores. Nas outras áreas, em função do reconhecimento das limitações para atacá-las simultaneamente, não fica explícito a necessidade de domínio nacional do capital. Enfatiza-se a produção local, mas não necessariamente a empresa de capital majoritário brasileiro. Este é o caso dos computadores de grande porte, montados pela IBM e Bourroughs no país, que se enquadravam, dentro da Estratégia Industrial, como fonte de divisas de exportações (como já vinha ocorrendo na época).

De qualquer forma, a posição oficial sobre a necessidade de controle de capital das indústrias de minicomputadores representa um avanço em relação a disposição anterior, consubstanciada no PBDCT. Neste documento propugnava-se a associação do Governo, com fabricante nacional e fabricante estrangeiro, para a

produção de microcomputadores, sem referência à composição do capital.

Esse posicionamento nacionalista no II PND teria, como se viu depois, influência decisiva na política que se delineava para o setor de informática. Assim, em 1975, em uma decisão sem precedentes, a CACEX "delegou à Capre competência para analisar e autorizar todos os pedidos de importação de equipamentos de processamento de dados e partes e peças para fabricação de produtos de Informática." [Brízida 1983, pag. 8]. É bem verdade que havia uma motivação de ordem econômica para o controle das importações, que era a de garantir o equilíbrio do balanço de pagamentos, afetado seriamente pelo primeiro choque do petróleo. Mas como infere Rodrigues [1984, pag.27-28], o equilíbrio das contas era uma justificativa econômica, para impor uma medida de cunho nitidamente político, de forçar a nacionalização dos componentes usados nos computadores aqui montados.

Em 1976, acentuando o caráter intervencionista que o Governo vinha assumindo no mercado de informática, há uma reestruturação da Capre, que passa a ter um Conselho Deliberativo com poderes para formular uma Política Nacional de Informática. A Capre, de órgão regulador interno da informática na Administração Pública Federal, e de controle das importações na área (a partir de 1975), passa a ter poder político para fazer recomendações e baixar resoluções normativas válidas fora do âmbito interno do Governo.

A primeira resolução do Conselho da Capre foi recomendar o estabelecimento da reserva de mercado para a faixa dos minis, microcomputadores, periféricos e terminais, sob a oposição veemente das multinacionais aqui instaladas. Mas a Capre não fazia mais do que expressar a vontade política dominante no Governo. Um ano depois a Capre convidava empresas nacionais a apresentarem projetos de fabricação de minicomputadores, e liberar recursos para as selecionadas.

Além destas medidas concretas da Capre, as resoluções baixadas, em janeiro de 77, pelo Conselho de Desenvolvimento Econômico constituíram o embrião para sistematização das diretrizes de uma Política Nacional de Informática. A contribuição do CDE foi de estabelecer critérios para o estabelecimento de projetos na área de computação, quais sejam:

*"1 - Prioridade às empresas que estivessem dispostas a desenvolver ou usar tecnologia nacional para conceber e fabricar seus produtos.*

*2 - Prioridade às empresas que não possuíssem vínculos permanentes com fornecedores externos...*

*3 - O controle do capital teria que ser nacional.*

*4 - Evitar-se-ia a formação de monopólios ou o domínio do mercado por algumas empresas.*

*5 - Prioridades às empresas que apresentassem perspectivas mais favoráveis ao país, no que se refere à balança comercial."*[Cobra 1988]

## 5. A POLÍTICA NACIONAL DE INFORMÁTICA COMO LEI

Com o Governo Figueiredo, e a emergência de novos agentes no cenário, tem início um processo de "fritura" da Capre. De um lado, as multinacionais e seus asseclas, brandindo o conhecido discurso da excessiva intromissão do Estado, da ineficácia macroeconômica da reserva de mercado e outras balelas mais (como se nos seus países de origem o Estado não as tivessem garantido proteção frente a concorrência externa). Do outro lado, o SNI, o Conselho de Segurança Nacional (CSN), CNPq e Ministério das Relações Exteriores (MRE), criticando o que consideravam (imaginem só!) "a excessiva liberdade da Capre em relação às multinacionais" [Rodrigues 1984, pag.47].

Na verdade tratava-se de uma disputa de poder dentro do próprio governo, para ver quem distribuía as cartas do baralho da informática. Se a "comunidade de informações" fizera o Presidente da República, porque não haveria de querer influenciar esta área estratégica para a soberania nacional. A Capre havia adquirido, ao longo dos anos, uma certa independência, e contava, aos olhos do SNI, com excessiva participação da sociedade civil, que poderia desvirtuar-lhe os caminhos.

O relatório da comissão do CNPq/SNI/MRE, encarregada, no início do governo, de fazer um diagnóstico setorial da informática e propor diretrizes para a PNI, legitimava tecnicamente as críticas à Capre. Constava no relatório que a Capre não possuía os instrumentos de ação necessários a uma atuação mais abrangente e integrada, e que portanto, dada a relevância do assunto, era necessário a criação de uma secretaria ou comissão de informática junto à Presidência da República.

Os resultados desta pressão logo viriam a se manifestar. Um decreto presidencial, datado de outubro de 1979, criava a SEI, Secretaria Especial de Informática, com o objetivo de "assessorar na formulação da Política Nacional de Informações e coordenar sua execução,

como órgão superior de orientação, planejamento, supervisão e fiscalização, tendo em vista especialmente o desenvolvimento científico e tecnológico do setor"[DOU de 09/10/79]. O centro das articulações da PNI passava agora da Capre, para a SEI.

O curioso em todo esse processo, é que informática não recebeu tratamento proporcional a sua aparente importância dentro do Governo, no documento que descreve o III PND. Talvez isso decorra da própria inversão de prioridades no processo de planejamento na época. Os PNDs anteriores destacam, nos seus capítulos, as Estratégias Nacionais nos vários setores da economia e da sociedade, expondo as diretrizes que devem nortear a ação governamental (em suas macro-políticas) a médio e longo prazos. O III PND, ao contrário, ressalta os mecanismos de política macroeconômica, para combater à inflação e equilíbrio da balança de pagamentos. As políticas setoriais ganham pouca importância e ficam condicionadas aos sucessos da política anti-inflacionária.

O conteúdo programático das políticas setoriais é visivelmente esvaziado. O termo "informática" ou "eletrônica" não são nem mesmo referenciados no texto, mas englobados implicitamente dentro das diretrizes para o item "Ciência e Tecnologia", do capítulo "Outras Políticas Governamentais" (a pobreza do documento se revela até mesmo pelos títulos e forma de organizar os assuntos temáticos). A julgar pelo tratamento conferido à informática no plano, a preocupação governamental estava retrocedendo aos tempos do PAEG. Felizmente, a PNI transitava por canais competentes e comprometidos com o longo prazo.

O estilo autoritário que marcou o período inicial da SEI foi sendo, gradativamente, abrandado com a inclusão da comunidade acadêmica e empresarial (nacional), nos anos 80, na discussão sobre a PNI. Afinal, começava-se a respirar os ares menos tensos da abertura "lenta e gradual" iniciada no Governo Geisel.

O processo de formulação da PNI na SEI culminou com o seu envio, como projeto de lei do Governo Federal, ao Congresso Nacional em 1984. Formou-se uma comissão de parlamentares para exame e discussão do assunto com professores, empresários, cientistas e autoridades. Votada, recebeu alguns vetos, sendo finalmente sancionada a 24 de outubro de 1984.

Segundo a Lei (n.7.232/84), a Política Nacional de Informática tem por "objetivo a capacitação nacional nas atividades de informática, em proveito do desenvolvimento social, cultural, político, tecnológico e econômico da sociedade brasileira..." [Brasil-PR-SEI

1987, pag.7]. Para que este objetivo seja alcançado, ela assegura ao Estado o poder de intervir no mercado para proteger a produção nacional de "determinadas classes e espécies de bens e serviços". É assegurado também ao Poder Executivo, o controle das importações por oito anos, a contar da publicação da lei, e o direito de adotar "restrições de natureza transitória em relação à produção, operação, comercialização e importação de bens e serviços técnicos de informática"[Cobra 1988, pag. 23].

Do ponto de vista institucional, a lei criou, vinculado diretamente ao Presidente da República, o Conin, Conselho Nacional de Informática e Automação, formado por representantes de diversos ministérios e oito representantes de entidades não governamentais ligadas à área. Ao Conin cabe, além de assessorar o Governo na formulação da PNI, propor um Plano Nacional de Informática e Automação - Planin, a cada três anos. O aparato institucional se complementa com a SEI, que passa a ser a secretaria executiva do Conin, e o CTI- Centro Tecnológico para Informática, cuja a finalidade é incentivar o desenvolvimento da pesquisa científica e tecnológica nas atividades de informática, através de convênios com universidades e empresas nacionais.

Como se vê, a lei veio consubstanciar as principais idéias e proposições que surgiram nos vários governos militares, por intermédio do GTE, Capre e SEI. Foi um percurso longo, de quase 20 anos, para que os objetivos de produzir localmente componentes eletrônicos para Marinha se transformassem em uma Política Nacional.

## 6. A PNI E SUA VINCULAÇÃO AOS OBJETIVOS NACIONAIS

A autodeterminação, a integração e a prosperidade nacional são, na ideologia geo-política de Silva [1981:251-252], os objetivos permanentes básicos de uma nação, isto é, os requisitos indispensáveis a sobrevivência de um "grupo nacional". Seriam estes valores, disseminados em maior ou menor grau, que teriam levado os povos a se constituírem em Estados-Nação, e a buscarem o progresso material. É por esta razão que o autor coloca estes valores nacionais básicos como pontos de partida do processo de Planejamento Governamental (PG).

No Brasil, estes três ONPs (Objetivos nacionais permanentes) se fazem presentes desde as primeiras experiências de PG, nos tempos de JK, até os governos militares. Embora a ênfase na autodeterminação e integração possam ter variado, assumindo maior proe-

minência nos PNDs, a prosperidade, da nação ou do grupo dominante, foi, sem dúvida, a meta mais permanentemente visada.

Em cada governo, dadas as condições históricas e principalmente, as diretrizes políticas, a Prosperidade Nacional se cristalizou como um objetivo nacional atual (ONA) diferente, como "Crescer 50 anos em 5", no Plano de Metas de JK, ou "Crescimento com melhoria das condições de vida", no Plano Trienal de João Goulart, ou "Aceleração do crescimento com contenção da inflação", no Programa Estratégico de Desenvolvimento de Costa e Silva, ou ainda o "Crescimento econômico a taxas superiores a médias históricas", no I PND.

Em um processo de Planejamento Governamental coerente é de se esperar que as políticas setoriais na indústria, agricultura, energia ou tecnologia se enquadrem dentro deste esforço nacional de prosperidade econômica. De fato, podemos verificar uma certa coerência entre os objetivos nacionais de Prosperidade definidos em cada plano com as diretrizes traçadas no campo da tecnologia e informática, como o quadro nas páginas seguintes procura sintetizar.

De JK até o Governo de Castello Branco, o que se poderia classificar como iniciativas na área tecnológica (entendendo-se este termo em uma acepção mais ampla), foram os investimentos governamentais para expansão da capacidade geradora de energia elétrica e de petróleo, bem como a consolidação de uma ampla infra-estrutura de transportes. Estes eram os fatores que, no entender do governo, eram necessários para a aceleração do desenvolvimento econômico nacional, na medida que ofereciam atrativos e vantagens para a instalação de indústrias e multinacionais no país, além dos incentivos fiscais.

Como já citamos anteriormente, a tecnologia recebe um tratamento diferenciado a partir do Governo Costa e Silva. Passa a ser entendida como um fator de produção tão importante quanto o capital e trabalho, mas a referência específica à informática não é ainda tão magnânima. A diretriz para a área contempla a produção de componentes eletrônicos para a Marinha, revelando, portanto, uma motivação antes estratégico-militar (ligada ao ONP de autodeterminação) do que desenvolvimentista.

No I PND, a Eletrônica deixa de ser um item programático apenas das Forças Armadas, para se vincular na estratégia nacional de "colocar o Brasil na categoria das nações desenvolvidas, no espaço de uma geração". Assim, esta área, bem como a Energia Nuclear, a Pesquisa Espacial, a Química Fina e a indústria

Aeronáutica, são encaradas como prioritárias dentro da política tecnológica, a fim de que o Brasil pudesse, além de se desenvolver, garantir sua posição de destaque no cenário internacional no futuro, ao lado das potências industrializadas.

A partir de então, a informática passa a ser assimilada como mais um meio para viabilizar a Prosperidade Nacional, como fonte de divisas geradas pela substituição das importações e pela exportação de componentes aqui produzidos, e como insumo para a atualização tecnológica do parque industrial brasileiro, a fim de mantê-lo em níveis de competitividade com o mercado produtor internacional.

A vinculação da informática com a Autodeterminação Nacional não deixa de existir, no entanto. Na verdade, nos Governos Geisel e Figueiredo estes dois ONPs (Prosperidade e Autodeterminação) se mostram determinantes e presentes no processo de formulação da PNI, a julgar pela ênfase na necessidade de capacitação nacional da tecnologia envolvida na área e na dominação do capital das indústrias que estavam se formando. As medidas concretas justificavam-se tanto por um lado, como por outro. Assim, o controle das importações de componentes eletrônicos era um desdobramento operacional da política de equilíbrio da Balança de Pagamentos (para garantir a manutenção do crescimento acelerado), mas também um recurso para estimular a fabricação local destes produtos (e preservar a autonomia nacional). Um outro exemplo é o estabelecimento da reserva de mercado para mini e microcomputadores, que por um lado, visava a independência nacional com a instalação de indústria com capital e tecnologia brasileiros, e por outro, aproveitar uma grande oportunidade de crescimento auto-sustentado, usando recursos financeiros advindos das próprias vendas internas crescentes.

No Governo Figueiredo, com a extinção da Capre e sua substituição pela SEI, como órgão coordenador do processo de formulação da PNI, vinculada ao Conselho de Segurança Nacional, reforça o caráter autodeterminista que o processo se revestia. A pressão das indústrias nacionais de informática e da comunidade acadêmica não deixam, no entanto, o governo perder de vista a importância da informática dentro da estratégia nacional de desenvolvimento.

Como se vê, o processo de formulação da política nacional de informática guardou uma grande coerência com os objetivos nacionais definidos nos planos de governo, sobretudo aqueles relacionados ao atingimento dos ONPs da Prosperidade e Autodeterminação Nacionais.

## Quadro sintético: a PNI e sua vinculação aos Objetivos Nacionais

### Obj. Nacional Permanente: Prosperidade nacional

Plano/Período Governo	Objetivo Nacional de Prosperidade	Estratégia na área tecnológica ou na informática
Plano de Metas 1958/1960 JK	"Crescer 50 anos em 5"	Investimentos públicos em energia/transportes
Plano Trienal 1963/1965 Goulart	"Assegurar uma tx. de crescimento da renda nacional compatível com expectativas de melhoria das condições de vida..."	Ênfase no provimento de energia e transportes
PAEG 1964/1966 Castello Branco	"Acelerar o ritmo de desenvolvimento econômico do país, interrompido no biênio 1962/63"	Incorpora as Telecomunicações como fator para desenvolvimento
Prog. Estratégico de Desenvolvimento 1968/1970 Costa e Silva	"Aceleração do desenvolvimento econômico, simultaneamente com a contenção da inflação"	Progresso tecnológico é fator básico para desenvolvimento. Produção de componentes eletrônicos assume importância estratégico-militar
Metas e Bases para Ação de Governo 1970/1973 Médici	"Crescimento Econômico: taxa crescente, que alcance no mínimo 7 a 9% ao ano no período, e evolua para 10%."	Aceleração dos proj. tecnol. e científicos em áreas selecionadas: Pesquisa Espacial, Energia Nuclear, Eletrônica
I PND 1972/1974 Médici	"Colocar o Brasil, no espaço de uma geração, na categoria das nações desenvolvidas, duplicando até 1980 a renda per capita (em rel. 1969)"	Desenvolver áreas tecnológicas prioritárias: Energia Nuclear, Pesquisa Espacial, Química, Eletrônica, Aeronáutica
II PND 1975/1979 Geisel	"Manter o crescimento acelerado dos últimos anos, com taxas de expansão do emprego superiores à década passada..."	Executar projetos de ponta, tecnologias de alta sofisticação. Desenvolver no país a tecnologia da eletrônica digital, em empresa de capital nacional
III PND 1980/1985 Figueiredo	"Crescimento acelerado da renda e do emprego, concomitante ao combate à inflação"	Desenvolver e ampliar a capacidade de absorção e geração de tecnologias

## 7. CONSIDERAÇÕES FINAIS

Entre as origens do processo de formulação de uma Política Nacional para Informática e sua institucionalização em forma de Lei Nacional, transcorreram-se quase 20 anos de discussões e avanços.

Comparando-se com outras políticas nacionais, como as voltadas ao setor urbano, como habitação e transportes coletivos, a política para a informática apresentou um grau bem maior de coerência entre os diagnósticos e as medidas concretas implementadas. Talvez deva-se debitar essa coerência a uma certa unidade no diagnóstico, de objetivos e de ação entre as diversas forças representadas no governo federal.

O diagnóstico para a área não foi sempre unânime dentro do governo, mas caminhou nesta direção ao longo dos planos. Como já tivemos a oportunidade de relatar, a origem do processo de formulação da PNI está ligada ao diagnóstico estratégico-militar feito pela Marinha, motivada pelo seu programa de modernização das fragatas. Mas já a esta época, dentro da SEPLAN, se reconhecia a importância da tecnologia dentro da estratégia nacional de Desenvolvimento Econômico. E é essa última visão que vai gradativamente se consolidando dentro das outras esferas de governo, e influenciando as diretrizes para a política tecnológica nos PNDs.

Os três órgãos governamentais criados para tratar assuntos do setor de informática, GTE, Capre e SEI, tiveram papéis distintos no processo de formulação da PNI. Cada um deles cumpriu um papel de relevância dentro do que o contexto histórico e político lhes reservou.

O GTE, criado em 1971 e dissolvido em 1976, lançou as bases para estabelecimento de um programa de capacitação tecnológica e de uma indústria nacional na área. É assim que podem ser entendidas as iniciativas de apoio à USP e PUC-RJ, para produção de um minicomputador e software, e a fundação da Cobra Computadores em 1974. A sua vinculação e subordinação à Marinha, ilustrada pela sua composição interna e pelos objetivos para os quais foi criado, não permitiram que o GTE assimilasse a responsabilidade de definir diretrizes gerais para uma política nacional para o setor. Seu objetivo era mais imediatista, qual seja, a de viabilizar a produção local de componentes eletrônicos para a modernização da frota da Marinha.

Desta forma, embora tivesse assento no GTE, o Ministério do Planejamento sentiu a necessidade de criação de um novo instrumento institucional para tratar a questão da informática. Era a Capre, criada em 1972. Já desde o início, a Capre acumularia certas funções que lhe outorgariam, no Governo Geisel, papel proeminente na definição dos rumos da PNI. Pelo decreto de sua fundação, a Capre era responsável pela racionalização dos recursos computacionais da Administração Pública Federal e pelo cadastramento dos computadores existentes no Brasil, dentro e fora do governo. A estas funções seriam acrescentadas mais tarde, o controle das importações de componentes eletrônicos (1975), e a incumbência explícita de formular diretrizes para uma política nacional de informática.

Como se vê, as funções da Capre foram gradativamente ampliadas, tornando-a o fórum privilegiado para institucionalização do processo de formulação da PNI. Seu sucesso advinha não apenas de seu poder político, conferido pela SEPLAN, mas também da composição pluralista de seu corpo de dirigentes, com representantes de vários ministérios, agências governamentais, e da sociedade civil.

A SEI, que substituiu a Capre no Governo Figueiredo, preservou o caminho já delineado. Avançou na definição das diretrizes da PNI, chegando em 1984, a um projeto de lei. Encaminhado ao Congresso, recebeu poucas modificações e vetos, já que consubstanciava a vontade nacional para o setor.

O que se seguiu é assunto para outro artigo....

## REFERÊNCIAS BIBLIOGRÁFICAS

- (1) Brasil-Ministério do Planejamento e Coordenação Econômica, Programa de Ação Econômica do Governo 1964-1966, Rio de Janeiro, Fibge, 1964.
- (2) Brasil-Presidência da República; Metas e Bases para a Ação de Governo 1970/73, Rio de Janeiro, Fibge, 1970.
- (3) Brasil-Presidência da República; Plano de Metas 1958/60, Rio de Janeiro, Ed. Jornal do Comércio, 1958.
- (4) Brasil-Presidência da República; Plano Trienal de Desenvolvimento Econômico e Social 1963-1965, Rio de Janeiro, Depto. Imprensa Nacional, 1963.

- (5) Brasil-Presidência da República; I Plano Nacional de Desenvolvimento (1972/74), Rio de Janeiro, Fibge, 1972.
- (6) Brasil-Presidência da República; II Plano Nacional de Desenvolvimento (1975/79), Rio de Janeiro, Fibge, 1974.
- (7) Brasil-Presidência da República; III Plano Nacional de Desenvolvimento (1980/85), Brasília, Seplan, 1981.
- (8) Brasil-Presidência da República; Plano Básico de Desenvolvimento Científico e Tecnológico ((1973/74), Rio de Janeiro, Fibge, 1973.
- (9) Brasil-Presidência da República; Programa Estratégico de Desenvolvimento: Síntese 1968-1970, local ignor., mimeografado, sem data.
- (10) Brasil-Presidência da República- Sec.Especial de Informática, Boletim Informativo, vol. 7. n. 16, Brasília, SEI, 1987.
- (11) Brasil-Presidência da República- Sec. Especial de Informática, Boletim Informativo, vol. 8. n. 17, Brasília, SEI, 1988.
- (12) Brízida, Joubert de O.; "Informática: Modelo Institucional Brasileiro", Palestra apresentada na Escola Superior de Guerra, apostilada, 1983.
- (13) Cobra-Computadores e Sistemas Brasileiros; "O sentido, a história e os resultados da Política Nacional de Informática", Rio de Janeiro, Cobra, 1988.
- (14) Rodrigues, Silvia Helena [1984]; "A indústria de Computadores: Evolução das decisões governamentais", in Reflexões sobre a Informática Brasileira: uma coletânea", Rio de Janeiro, Ed. Abicomp, 1984.
- (15) Rodrigues, Silvia H. V. [1984b]; "Rastro de Cobra", local e edit. não citados, 1984.
- (16) Silva, Golbery do C.; "Conjuntura política nacional: o Poder Executivo & Geopolítica do Brasil", Rio de Janeiro, José Olympio Ed., 3a. ed., 1981.
- (17) Tavares, M. C.; "O Planejamento em economias mistas", in SEMINÁRIO INTERNACIONAL ESTADO E PLANEJAMENTO: SONHOS E REALIDADE, Brasília, Cendec, 1988.

# **Revista do Instituto de Informática**

## **Publicação Semestral do Instituto de Informática**

### **PUCCAMP**

A Revista do Instituto de Informática aceita colaborações que lhe forem espontaneamente enviadas, reservando-se o direito de publicá-las ou não, conforme avaliação dos Editores. Os temas abordados serão relacionados com as várias áreas de Informática. Os trabalhos podem ser artigos científicos (textos com 30.000 caracteres), ou opiniões (texto com 8.000 caracteres) e remetidos em 3 vias, seguindo o formato dos artigos e opiniões aqui publicados\*.

\* Os nomes dos autores, endereço e vinculação profissional devem aparecer em folha separada do texto, de modo a possibilitar, sem identificação, um julgamento do trabalho.

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE CAMPINAS**

**Magnífico Reitor: *Prof. Gilberto Luiz M. Selber***

**Vice-Reitor para Assuntos Administrativos: *Prof. Alberto Martins***

**Vice-Reitor para Assuntos Acadêmicos: *Pe. José Benedito A. David***

**Diretora do Instituto de Informática: *Prof. Angela M. Engelbrecht***

**Vice-Diretor do Instituto: *Prof. José Oscar Fontanini de Carvalho***

