

# AMBIENTE DE CONTROLE DE CONFIGURAÇÃO DA CENTRAL TELEFÔNICA TRÓPICO-RA

Flávia M.V.Silva<sup>1</sup>  
M.Cristina Baraldi D.A.<sup>2</sup>  
Mario C.L.Curi<sup>3</sup>  
Jorge M.Souza<sup>4</sup>

CPqD-TELEBRÁS - CP 1579 - Campinas - SP - Brasil

## Resumo

Uma das atividades importantes para garantia de qualidade é a gerência de configuração que tem por objetivo assegurar a compatibilidade entre os diversos itens que compõem o produto assim como acompanhar e controlar sua evolução.

O presente trabalho descreve o Ambiente de Controle usado na gerência de configuração da Central Telefônica TRÓPICO-RA, equipamento de grande porte com mais de 5 Mbytes de software, 50 tipos de placas hardware e 400 tipos de tabelas.

## 1 Introdução

O desenvolvimento de sistemas complexos exige a definição de uma estrutura organizacional capaz de [1]:

- Gerenciar o projeto
- Definir o ciclo de desenvolvimento
- Garantir a qualidade do produto

Essas três atividades são necessárias para garantir a conformidade do produto aos atributos de desempenho e qualidade e também atender os planejamentos de custo e prazo.

O TRÓPICO-RA é uma central telefônica digital com uma arquitetura distribuída baseada em microcomputadores. Tem capacidade de comutar de 20.000 a 40.000 terminais telefônicos. Para isso sua estrutura permite a interligação de até 1024 microcomputadores através de uma via de sinalização que possibilita até 24 comunicações simultâneas.

<sup>1</sup>FTPTAT/CPqD

<sup>2</sup>STC TELECON/CPqD

<sup>3</sup>TELEBRÁS/CPqD

<sup>4</sup>TELEBRÁS/CPqD

No desenvolvimento do *TRÓPICO-RA* foi formalizada uma estrutura capaz de gerenciar o desenvolvimento de mais de 5 Mbyte de software e 50 tipos de placas hardware, cobrindo funções como sistema operacional, comutação, processamento de chamadas telefônicas, operação, manutenção, etc.

A garantia de qualidade do produto no caso de equipamentos de telecomunicações, deve receber atenção especial devido aos severos padrões internacionais recomendados pelo CCITT (The International Telegraph and Telephone Consultative Committee) [2,3].

A garantia de qualidade do produto é assegurada através da organização de atividades de controle que no caso de produto software correspondem a:

- Gerência da configuração do produto,
- Garantia da qualidade dos diversos itens que compõem o produto,
- Verificação e Validação dos requisitos funcionais e de qualidade do serviço percebido pelo usuário envolvendo aspectos de desempenho e confiabilidade.

A um conjunto coerente de itens que formam um produto chamamos de configuração.

A gerência de configuração do produto tem por objetivo garantir a compatibilidade entre os diversos itens que o compõem: documentação, software, firmware e hardware. A do *TRÓPICO-RA* é dividida em:

- Gerência de Documentação
- Gerência de Software e Firmware
- Gerência de Hardware.

O meio de comunicação entre as gerências de modo a acompanhar a evolução do produto e manter a coerência é o mecanismo de controle de alterações baseado num relatório padrão chamado Relatório de Falha (RF) que tem um ciclo de vida bem definido envolvendo as três gerências.

A garantia de qualidade dos itens hardware e software do *TRÓPICO-RA* é feita usando modelos matemáticos de falhas de componentes hardware e modelos de confiabilidade de software aplicados aos dados de falha coletados pelos RF's [4].

As atividades de Verificação e Validação do sistema são feitas durante todo o ciclo de vida e tem por objetivo, garantir:

- o desempenho correto das funções do sistema,
- o atendimento dos requisitos de degradação do desempenho devido a falhas de concepção,
- o atendimento dos requisitos de desempenho dentro das condições de uso especificadas,
- o atendimento dos requisitos de desempenho especificados para condições de sobrecarga.

O objetivo do presente trabalho é descrever a organização da gerência de configuração de software e firmware usada no *TRÓPICO-RA*.

As seguintes questões têm que ser respondidas pela estrutura organizacional [13]:

- quais os itens a serem controlados?
- como é feito o controle de correções/modificações?
- como são divididas as responsabilidades de aprovação das correções/modificações?
- que mecanismo garante que a correção/modificação foi feita corretamente?
- que mecanismo é utilizado para comunicar as correções/modificações?

Na seção 2 são identificados os itens que fazem parte da configuração de software/firmware. Na seção seguinte o ciclo de controle de correções usando o RF e a interação com as gerências de documentação e hardware. Finalmente, na seção 4 é descrito o Ambiente de Controle de Configuração (ACC) que define uma configuração software e monitora a sua evolução e sua implementação.

## 2 Identificação dos Itens de uma Configuração

Uma configuração é um conjunto de itens que necessita ter sua evolução controlada de modo a manter a integridade do produto. A gerência de configuração do software tem que definir o conjunto de itens que compõem cada configuração. Esse conjunto varia de um sistema para outro porque depende da forma como o software está estruturado.

O Sistema TRÓPICO é baseado em uma estrutura modular cujos objetos físicos software e hardware são chamados de Blocos de Implementação (BI) e são definidos na fase do mapeamento funcional-físico. Um determinado firmware está associado a um BI hardware. Um BI software tem as seguintes características [5]:

- é composto por processos instanciáveis,
- cada processo é descrito por uma máquina de estado finita,
- formato padrão,
- comunicação entre processos feita através de mensagens,
- bases de dados independentes,
- utilização/prestação de serviço através de interfaces padrão.

Com o objetivo de aumentar a produtividade e qualidade, durante as fases de desenvolvimento e implementação são identificados Elementos Padrão de Implementação Software (EPIS), por exemplo subrotinas, que implementam serviços usados por vários BI's software ou por outro EPIS.

Os conceitos e interfaces identificados durante o desenvolvimento são definidos no Modelo de Informação. O Modelo de Informação é estruturado em objetos funcionais cujo elemento-básico para gerar contextos de BI e EPIS é o Bloco de Serviço (BS).

O BS contém três módulos de informação:

- Módulo de Conceitos (MC): contém as definições de tipos de variáveis e de constantes que são comuns a mais de um BS ou interface. O MC pode importar objetos definidos em MC's do mesmo ou de outro BS.
- Módulo de Interfaces (MI): contém as definições das mensagens associadas a uma interface. Um MI pode importar objetos definidos em MC's do mesmo ou de outro BS.
- Módulo Prestador (MP): contém as definições de tipos de variáveis e constantes do sistema. Além disso contém a definição das relações de uso (utilização/prestação de serviço) das interfaces do BS.

Existem BI's software que saem do formato padrão. São os BI's do sistema operacional e da comunicação entre processadores.

No caso do sistema *TRÓPICO-RA* decidiu-se incluir também a gerência de firmware de modo a aproveitar a estrutura de controle criada para o software que, caso contrário, deveria ser criada também para a gerência de hardware.

O conjunto de BI's hardware, software e a documentação associada aos BI's e ao sistema constituem um sistema de Aplicação pois possui características funcionais bem definidas.

De um sistema de Aplicação são geradas Instalações que tem características diferentes quanto ao número de processadores, terminais, periféricos, etc. Os dados de configuração de uma Instalação são gerados com o auxílio de Geradores de Tabelas (GT) que auxiliam no preenchimento do total de tabelas do *TRÓPICO-RA* (em torno de 400 tipos, aproximadamente 5000 tabelas) com dados default e dados de configuração. Os GT's, devido a sua íntima ligação com o sistema, também são objetos de gerenciamento.

Os BI's software são escritos em CHILL (CCITT High Level Language) [12]. O Ambiente CHILL coloca à disposição dos usuários uma Biblioteca de Utilitários cuja evolução deve ser compatível com a configuração do produto [11].

Podemos, então, resumidamente indicar os itens que são objeto de gerenciamento de configuração no sistema *TRÓPICO-RA*:

- fontes de BI's software
- firmware dos BI's hardware
- EPIS e conceitos dos BS's
- BI's não padrão (S.O. e sinalização entre processadores)
- Geradores de tabela
- Bibliotecas CHILL

### 3 Controle de Correção/Modificação

O controle de correção/modificação é a atividade fundamental da Gerência de Configuração principalmente quando se lida com grande quantidade de software desenvolvido por vários implementadores.

Um item é informalmente controlado pelo implementador enquanto não é liberado para uso. Nesse momento, após ser testado e/ou revisado, o item é congelado, ou seja, toda correção/modificação passa por um controle formal.

Depois de congelado, um item só pode ser alterado através de dois mecanismos:

- Relatório de Falha (RF) [6] e
- Proposta de Modificação (PM).

A "PM" documenta as sugestões de melhoramentos a serem feitas oportunamente no produto.

O "RF" serve para:

- registrar as falhas detectadas
- documentar o processo de depuração e correção dos defeitos causadores dessas falhas,

- identificar os itens a serem corrigidos,
- controlar a evolução do produto pelas gerências de configuração.

Vamos descrever apenas a forma de controle de RF's porque seu número é muito maior do que PM's durante o desenvolvimento.

O RF contém as seguintes informações:

1. Data da ocorrência da falha.
2. Origem da falha: Protótipo/Configuração onde foi detectada e condições de ocorrência.
3. Tipo de falha: Hardware, Software, Firmware, Documento, Empacotamento Eletromecânico, Sistema (Especificação ou Projeto de Sistema) e Procedimento.
4. Análise: identificação e classificação do defeito que provocou a falha (codificação, especificação, interface, etc...).
5. Soluções: as soluções propostas e a que é recomendada.
6. Controle das modificações: controle efetuado pelos responsáveis indicando os itens (inclusive documentos) a serem alterados.
7. Resultado dos testes aplicados ao item corrigido.

O RF é o principal mecanismo usado para o controle de configuração. É necessário, portanto, definir as responsabilidades quanto a sua geração, controle e aprovação.

O RF é gerado durante os testes de integração e validação do produto quando uma falha é detectada.

O ciclo de vida do RF é mostrado na figura-1 indicando a interseção entre os responsáveis pelas diversas fases cujas atribuições são:

**ELABORADOR** : Pessoa ou grupo que detectou a falha em qualquer fase do desenvolvimento ou mesmo durante a fase de industrialização.

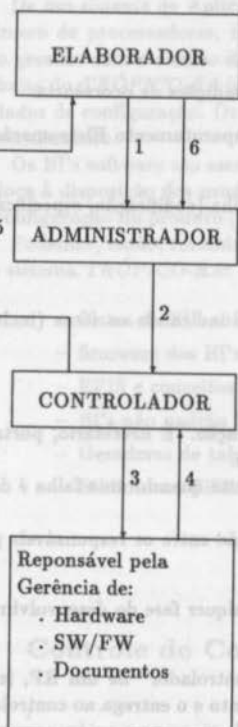
**ADMINISTRADOR** : É a interface entre o "elaborador" e o "controlador" de um RF, isto é, fornece o formulário em branco ao elaborador; registra o documento e o entrega ao controlador. Quando do encerramento do RF, informa ao elaborador o destino do mesmo e mantém os arquivos de controle sempre atualizados.

**CONTROLADOR** : Analisa a falha detectada e verifica sua abrangência com o objetivo de enviá-la aos responsáveis pelo projeto. Faz a interface entre o elaborador, o administrador e os responsáveis pelas gerências.

**RESPONSÁVEL** : Supervisiona as fases de detalhamento da falha, elaboração de propostas de solução, análise das soluções, aprovação e implementação das modificações.

Cada gerência (SW, HW e documentos) tem o seu controlador.

Uma falha que afeta itens controlados por gerências diferentes exige um controle adicional para garantir que passará pelos responsáveis das diversas gerências para efetuarem as correções necessárias antes de ser encaminhado para o elaborador na fase de teste.



1. Pede o registro do RF
2. Registra e encaminha
3. Analisa a falha e encaminha para a(s) gerência(s) responsável(is)
4. Aprovação e implementação da solução adotada
5. Colocar disponível a última versão da biblioteca para o implementador efetuar a correção
6. Se a implementação está correta pede o encerramento e arquivo

Figura - 1

Afim de auxiliar o controle dos RF's foi criada uma ferramenta software chamada Monitor de Relatório de Falha (MONRF).

Essa ferramenta é usada pelo administrador para registrar o RF, atualizar os dados e protocolar o RF, isto é, registrar para qual responsável o controlador passará o RF.

Além desse controle o MONRF possui relatórios que auxiliam no controle e na obtenção de dados relativos ao andamento de cada RF gerado.

Os relatórios estão divididos em dois tipos, os de acompanhamento gerencial e os estatísticos dos RF's.

Como exemplo de relatórios gerenciais usados temos:

- RF's não encerradas por TIPO DE FALHA
- RF's encerradas por TIPO DE FALHA
- Ciclo de um determinado RF com os respectivos responsáveis.

Para análise qualitativa e quantitativa de confiabilidade do hardware e do software, são usados os seguintes relatórios:

- Número de RF's por CLASSE de PRODUTO a cada intervalo de N dias
- Número de RF's por PRODUTO
- Número de RF's por CLASSE de PRODUTO por intervalo tempo entre falhas
- Número de RF's por PRODUTO por intervalo tempo entre falhas
- Número de RF's por TIPO DE FALHA
- Número de Falhas por ATIVIDADE DE DETECÇÃO
- Número de Falhas por TIPO por ATIVIDADE DE DETECÇÃO
- Número de Falhas por TIPO DE CONSEQUÊNCIA NO FUNCIONAMENTO
- Número de Falhas por TIPO por CONSEQUÊNCIA NO FUNCIONAMENTO

Após seu encerramento o RF é analisado com o objetivo de fornecer dados para modelos matemáticos e de confiabilidade afim de assegurar a garantia de qualidade dos itens medidos, bem como subsidiar resposta às diversas questões como [7]:

- Quais as falhas mais comuns no sistema;
- Em que ponto do ciclo de vida um certo tipo de falha tem mais chance de ser introduzida;
- Comparar resultados obtidos com novos projetos e/ou ampliação de projetos já existentes;
- etc...

#### 4.1.1 A Organização das Bibliotecas CMIS

CMIS O 1.4

A área de EPIS e conselhos de RF's está organizada da seguinte forma:

- Biblioteca INRI SORA,
- Biblioteca INRI SORA CDU,



## 4 Implementação do Ambiente

A fim de controlar os itens software envolvidos no *TRÓPICO-RA*, assim como suas configurações e monitorar a correção e teste dos RF's gerados, foi criado o ACC.

O ACC (Ambiente de Controle de Configuração) está armazenado no Cluster Digital - VAX (Virtual Address Extension) do CPqD.

Para o controle de itens foram criadas bibliotecas CMS (Code Management System), ferramenta construída pela Digital e disponível no VAX [8].

Esta ferramenta controla todo o acesso feito aos itens e/ou às bibliotecas, facilitando a manutenção deles e de suas configurações.

São duas as áreas de controle:

- Área de EPIS e conceitos dos BS's (Blocos de Serviço) e
- Área dos elementos funcionais do projeto (basicamente BI's).

Quando existe um RF a ser implementado, o(s) item(itens) que deve(m) ser corrigido(s) é(são) reservado(s) na sua biblioteca correspondente e colocado(s) em uma área de trabalho. A correção é feita pelo implementador, que avisa o controle quando tiver terminado. O(s) item(itens) é(são) copiado(s) pelo controle novamente em uma área de teste, mas não é(são) recolocado(s) na sua biblioteca. Ele(s) precisa(m) ser recompilado(s), ligado(s) e testado(s).

Para realizar a compilação/ligação são usadas as seguintes ferramentas:

1. GERAMMS : que extrai as relações de dependência de um item com os demais (dentro do mesmo BI; entre BI e BS ou entre gerador de Tabela (GT), BI e BS); colocando-as em formato compreensível pelo MMS (Module Management System).
2. MMS (Module Management System) : ferramenta construída pela Digital e também disponível no VAX [9]. Realiza uma comparação entre data(s) de revisão do(s) fonte(s) e a de seu executável, fazendo com que apenas o necessário seja recompilado.
3. MAKE : coordena a chamada do GERAMMS, do compilador e do ligador CHILL (providos pelo Ambiente CHILL). As funções mais importantes do MAKE são orientar o ligador quanto ao PID (Process ID) do BI, ressaltando que cada BI da central possui um PID que o identifica, e seu outermost (processo principal da sua execução funcional).

O passo seguinte é testar o novo executável em conjunto com os demais BI's (teste de integração), para verificar e validar a correção feita. Caso seja necessário, o(s) item (itens) é(são) re-corrigido(s), recompilado(s), ligado(s) e re-testado(s) até sua aprovação. Quando passar no teste, o(s) item(itens) retorna(m) à biblioteca, justificando com o(s) número(s) do(s) RF(s) corrigido(s).

### 4.1 O CMS

O CMS é um sistema bibliotecário de desenvolvimento e manutenção de software.



Uma biblioteca CMS é um diretório VMS (Virtual Memory System) contendo arquivos especialmente formatados. Esse diretório apenas pode ser acessado por comandos CMS, caso contrário seu conteúdo pode ser violado e, dependendo do acesso, esse conteúdo pode ser irrecuperável.

Essa biblioteca serve como um depósito de várias entidades CMS (chamados objetos).

Os conceitos usados pelo CMS são: elemento, grupo e classe.

Um elemento é sua unidade básica estrutural, e consiste de um arquivo (ASCII ou binário) e todas as suas versões. Uma geração de um elemento representa uma versão específica dele.

Ao se criar um elemento na biblioteca (equivalente à sua primeira inserção nela) esse arquivo se torna a geração 1 desse elemento. Nessa operação, o CMS armazena o texto inteiro do elemento; para as gerações sucessivas, o CMS armazena apenas as linhas de modificação de uma geração para a seguinte. Essas alterações são armazenadas num arquivo DELTA correspondente ao elemento alterado.

Cada item controlado é um elemento em alguma das bibliotecas existentes.

Um grupo é um conjunto de elementos (ou outros grupos) que se pode combinar e manipular como uma unidade. É usado para definir, por exemplo, os elementos que compõem um contexto de compilação (entende-se por contexto de compilação os elementos da área de EPIS e conceitos dos BS's que devem ser compilados antes da compilação dos BP's que seguem esse contexto).

Uma classe é um conjunto de gerações particular de elementos. É usado para definir uma configuração do sistema sob controle (a geração de cada elemento que participa da configuração gerada é inserida na classe correspondente a ela).

As operações possíveis de serem feitas nos arquivos contidos em biblioteca são:

- armazenar elementos (ASCII e binários),
- retirar elementos, modificá-los e testá-los na área do usuário,
- controlar modificações concorrentes no mesmo elemento,
- realizar merge de modificações concorrentes a um elemento,
- criar versões concorrentes (chamadas gerações) de um elemento,
- comparar duas gerações de um mesmo elemento dentro da biblioteca,
- organizar em grupos, elementos relacionados da biblioteca,
- definir um conjunto de gerações de elementos como uma classe para construir uma configuração ou versão de release de um projeto,
- discriminar quais usuários estão trabalhando em quais elementos da biblioteca,
- manter uma conta do histórico das transações efetuadas nos elementos e na biblioteca,

Sobre o CMS foi criada uma ferramenta chamada TESIS, que facilita o acesso às bibliotecas e a sintaxe de alguns comandos mais usados. Ela foi escrita em DCL (DIGITAL Command Language) [10].

#### 4.1.1 A Organização das Bibliotecas CMS

A área de EPIS e conceitos dos BS's está organizada da seguinte forma :

- Biblioteca DIHLSORA,
- Biblioteca DIHLSORA.CDU,

#### - Biblioteca SOPAO.

Nas duas primeiras estão armazenados módulos básicos de acesso ao hardware.

Na terceira estão armazenados módulos de conceitos, definição de estruturas de dados e EPIS (Elemento Padrão de Implementação Software) usados pelos BI's.

A área dos elementos funcionais do projeto está organizada da seguinte forma:

- Bibliotecas dos BI's software(59),
- Bibliotecas dos Geradores de Tabelas (30),
- Biblioteca dos BI's hardware (12),
- Biblioteca SISTEMAS,
- Biblioteca CHILL,
- Biblioteca de Tabelas,
- Biblioteca SINALIZAÇÃO.

As bibliotecas dos BI's software possuem fontes dos módulos do BI, executável com e sem depuração.

As bibliotecas dos Geradores de Tabelas possuem fontes dos módulos do Gerador, seu executável com e sem depuração.

As bibliotecas dos BI's hardware possuem o executável referente a cada firmware.

A biblioteca SISTEMAS possui os executáveis dos sistemas operacionais.

A biblioteca CHILL possui os executáveis dos utilitários do ambiente CHILL para os sistemas operacionais.

A biblioteca de tabelas possui tabelas usadas pelos BS's (Blocos de Serviço).

A biblioteca SINALIZAÇÃO possui os executáveis dos BI's não padrão relativos à função de sinalização entre processadores, que recebem tratamento diferenciado na sua compilação.

## 4.2 O MMS

É uma ferramenta que atualiza arquivos através da comparação das datas de revisão dos arquivos especificados num arquivo de descrição. Se um arquivo fonte é mais recente que seu objeto (seu executável), o MMS o atualiza usando os arquivos dos quais ele depende e que foram descritos no arquivo de descrição.

Um exemplo de relação de dependência descrita ao MMS :

```
ACETAS.REL : -  
  SOPAO_ACETAS.CHL*, -  
  DADT_TREC.REL
```

onde o alvo ACETAS.REL é obtido através do fonte SOPAO\_ACETAS.CHL(\*) e o executável DADT.TREC.REL.

(\*) o prefixo SOPAO denota que o fonte ACETAS.CHL se encontra na biblioteca CMS do SOPAO (área de EPIS e conceitos de BS's); prefixo análogo é utilizado nas demais bibliotecas.

Para automatizar seu uso foi construída uma ferramenta chamada GERAMMS, que atualiza a cadeia de dependência dos módulos envolvidos (internamente ao SOPAO ou internamente ao BI, ou entre eles).

#### 4.3. A compilação do Software

Os módulos que compõem o Software do *TRÓPICO-RA* se interdependem da seguinte forma:

- internamente ao SOPAO,
- internamente ao BI,
- entre módulos dos BI's e módulos do SOPAO,
- entre módulos dos Geradores de Tabelas e módulos do SOPAO,
- entre módulos dos Geradores de Tabelas e módulos dos BI's.

Para levantar as cadeias de dependências é utilizado o GERAMMS, que também faz a ligação do CMS com o MMS, indicando que os fontes deverão ser buscados na biblioteca antes de serem compilados. Dessa forma garante-se que apenas serão compilados os fontes alterados e os que deles dependem.

### 5 Conclusão

A Gerência da configuração é uma atividade fundamental para a Garantia da Qualidade de sistemas complexos.

Ela exige uma estrutura formal que permite o controle da evolução dos itens que compõem o produto e defina as responsabilidades quanto as aprovação, teste e comunicação das correções/modificações.

O mecanismo adotado no desenvolvimento do *TRÓPICO-RA* é geral e pode ser usado em outros projetos. Suas características principais são:

- adoção de um mecanismo único, o Relatório de Falha (RF), para identificação, solução, aprovação, teste e comunicação das correções dos defeitos.
- atribuição de responsabilidades e definição do ciclo de vida do RF
- utilização de ferramentas de gerenciamento dos arquivos fontes disponíveis no mercado (no nosso caso CMS e MMS da Digital)
- desenvolvimento de ferramentas complementares de auxílio à gerência (MONRF) e à manipulação de arquivos (TESIS e GERAMMS).

## REFERÊNCIAS

- [ 1 ] E.H.Bersoff, "Elements of Software Configuration Management",  
IEEE Trans. on Software Engineering, vol. SE-10, nº 1, Jan 1984,
- [ 2 ] "Handbook on Quality of Service, Network Management and Network Maintenance" - CCITT -  
1984
- [ 3 ] "Digital Local, Transit, Combined and International Exchanges in IDN and Mixed Analogue -  
Digital Networks"  
CCITT - vol VI - Fasc VI.5 - Rec. Q.543 - Nov 1988
- [ 4 ] K.Kanoun, M.R.Bastos Martini, J.Moreira de Souza. " A Method for Software Reliability  
Analysis and Prediction: Application to the TRÓPICO-RA Switching System",  
IEEE Trans. Software Engineering, Junho 91
- [ 5 ] "Metodologia para Desenvolvimento de Sistemas TRÓPICO"  
DCT - CPqD - TELEBRÁS - Abril 1991
- [ 6 ] "Metadocumento: Relatório de Falha" - DCT - CPqD - TELEBRÁS - 1990
- [ 7 ] M.R.Bastos Martini, J.A.Vallegas Pereira " Estudo dos RF's Encerrados do TRÓPICO-RA -  
DCT - CPqD - TELEBRÁS - Nov 90
- [ 8 ] "Guide to VAX DEC/Code Management System" - Digital Equipment Corp. - Abril 1988
- [ 9 ] "Guide to VAX DEC/Module Management System" - Digital Equipment Corp. - July 1990
- [ 10 ] "VMS DCL Concepts Manual" - Digital Equipment Corp. - April 1988
- [ 11 ] APCC V3.3 Documentação do Usuário - CPqD TELEBRÁS - Set 1990
- [ 12 ] Recomendação do CCITT Z200 - Livro Azul - Melbourne - 1988
- [ 13 ] R.S. Pressman, Software Engineering, A Practitioner's Approach, McGraw-Hill, 1987