



Ministério da
**Ciência, Tecnologia
e Inovação**



sid.inpe.br/mtc-m19/2011/11.17.19.31-TDI

**PROPOSTA DE ADOÇÃO DE UM PROCESSO DE
CAPTURA E RASTREAMENTO DE REQUISITOS
BASEADA NUM ESTUDO DE CASO E NUM
HISTÓRICO DAS FASES DA ENGENHARIA DE
SISTEMAS DO INPE**

Mario Celso Padovan de Almeida

Dissertação de Mestrado do
Curso de Pós-Graduação em
Engenharia e Tecnologia Espaciais/
Gerenciamento de Sistemas
Espaciais, orientada pelo Dr. Marcelo
Lopes de Oliveira e Souza,
aprovada em 21 de dezembro de
2011.

URL do documento original:

<<http://urlib.net/8JMKD3MGP7W/3AQPAJ8>>

INPE
São José dos Campos
2011

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

Fax: (012) 3208-6919

E-mail: pubtc@sid.inpe.br

CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELLECTUAL DO INPE (RE/DIR-204):

Presidente:

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Membros:

Dr. Antonio Fernando Bertachini de Almeida Prado - Coordenação Engenharia e Tecnologia Espacial (ETE)

Dr^a Inez Staciarini Batista - Coordenação Ciências Espaciais e Atmosféricas (CEA)

Dr. Gerald Jean Francis Banon - Coordenação Observação da Terra (OBT)

Dr. Germano de Souza Kienbaum - Centro de Tecnologias Especiais (CTE)

Dr. Manoel Alonso Gan - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Dr^a Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação

Dr. Plínio Carlos Alvalá - Centro de Ciência do Sistema Terrestre (CST)

BIBLIOTECA DIGITAL:

Dr. Gerald Jean Francis Banon - Coordenação de Observação da Terra (OBT)

Deicy Farabello - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

EDITORAÇÃO ELETRÔNICA:

Vivéca Sant´Ana Lemos - Serviço de Informação e Documentação (SID)



Ministério da
**Ciência, Tecnologia
e Inovação**



sid.inpe.br/mtc-m19/2011/11.17.19.31-TDI

**PROPOSTA DE ADOÇÃO DE UM PROCESSO DE
CAPTURA E RASTREAMENTO DE REQUISITOS
BASEADA NUM ESTUDO DE CASO E NUM
HISTÓRICO DAS FASES DA ENGENHARIA DE
SISTEMAS DO INPE**

Mario Celso Padovan de Almeida

Dissertação de Mestrado do
Curso de Pós-Graduação em
Engenharia e Tecnologia Espaciais/
Gerenciamento de Sistemas
Espaciais, orientada pelo Dr. Marcelo
Lopes de Oliveira e Souza,
aprovada em 21 de dezembro de
2011.

URL do documento original:

<<http://urlib.net/8JMKD3MGP7W/3AQPAJ8>>

INPE
São José dos Campos
2011

Dados Internacionais de Catalogação na Publicação (CIP)

Almeida, Mario Celso Padovan de.
Al64p Proposta de adoção de um processo de captura e rastreamento de requisitos baseada num estudo de caso e num histórico das fases da engenharia de sistemas do INPE / Mario Celso Padovan de Almeida. – São José dos Campos : INPE, 2011.
xxiv + 198 p. ; (sid.inpe.br/mtc-m19/2011/11.17.19.31-TDI)

Dissertação (Gerenciamento de Sistemas Espaciais) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2011.
Orientador : Dr. Marcelo Lopes de Oliveira e Souza.

1. engenharia de sistemas espaciais. 2. engenharia de requisitos.
3. rastreabilidade de requisitos. I. Título.

CDU 629.78

Copyright © 2011 do MCT/INPE. Nenhuma parte desta publicação pode ser reproduzida, armazenada em um sistema de recuperação, ou transmitida sob qualquer forma ou por qualquer meio, eletrônico, mecânico, fotográfico, reprográfico, de microfilmagem ou outros, sem a permissão escrita do INPE, com exceção de qualquer material fornecido especificamente com o propósito de ser entrado e executado num sistema computacional, para o uso exclusivo do leitor da obra.

Copyright © 2011 by MCT/INPE. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, microfilming, or otherwise, without written permission from INPE, with the exception of any material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use of the reader of the work.

Aprovado (a) pela Banca Examinadora
em cumprimento ao requisito exigido para
obtenção do Título de Mestre em

Engenharia e Tecnologia
Espaciais/Gerenciamento de Sistemas
Espaciais

Dr. Petrônio Noronha de Souza

Presidente / INPE / SJCampos - SP

Dr. Marcelo Lopes de Oliveira e Souza

Orientador(a) / INPE / SJCampos - SP

Dr. Geilson Loureiro

Membro da Banca / INPE / São José dos Campos - SP

Dr. Rolf Henry Vargas Valdivia

Convidado(a) / Friuli / São José dos Campos - SP

Este trabalho foi aprovado por:

() maioria simples

unanimidade

Aluno (a): Mario Celso Padovan de Almeida

São José dos Campos, 21 de dezembro de 2011

“Our Age of Anxiety is, in great part, the result of trying to do today's job with yesterday's tools and yesterday's concepts”

Herbert Marshall McLuhan (1911-1980)

Cientista da mídia, criador do termo “Aldeia Global” (“Global Village”)

AGRADECIMENTOS

Há muitas instituições e pessoas a quem eu devo agradecer por terem tornado possível a realização deste trabalho.

Antes de todos, tenho que agradecer ao meu orientador, professor Marcelo Lopes, pela sugestão de tema do estudo, pelo incentivo e desafios propostos e por disponibilizar parte do seu escasso e concorrido tempo a interessantes discussões.

Tenho que agradecer ao INPE e ao LIT por terem me permitido cursar as aulas das disciplinas e a usar parte do meu horário de trabalho em estudos e na redação desta dissertação.

Agradeço aos colegas do Programa CBERS por me permitirem livre acesso a todo o arquivo de documentos do Programa e por me autorizarem a usar esses documentos no estudo de caso.

Agradeço aos meus colegas do Serviço de Documentação do INPE - SID pela ajuda em pesquisas na biblioteca do INPE e na localização de textos raros existentes em outros locais.

Agradeço à professora Glória pela meticulosa revisão no texto da dissertação.

Não poderia deixar de agradecer à National Instruments do Brasil - NI, por ter fornecido uma versão de demonstração da ferramenta Requirements Gateway; e à Aeroeletrônica Indústria de Componentes Aviônicos - AEL, por ter disponibilizado seus documentos para uso neste trabalho.

RESUMO

Há várias décadas o Instituto Nacional de Pesquisas Espaciais (INPE) emprega metodologias da Engenharia de Sistemas no desenvolvimento de sistemas espaciais. No entanto, o envolvimento do INPE com a Engenharia de Sistemas não ocorreu de forma uniforme, mas passou por três fases de maior e menor evolução. Em particular, a Engenharia de Sistemas do INPE não acompanhou a recente evolução ocorrida com a introdução de processos e ferramentas da Engenharia de Requisitos. Para atenuar isto, este trabalho apresenta uma proposta de adoção de um processo de captura e rastreamento de requisitos baseada num estudo de caso e num histórico das fases da Engenharia de Sistemas no INPE. Para isto, e para melhor entender a situação atual da Engenharia de Sistemas no INPE e as razões da não utilização de processos de Engenharia de Requisitos, este trabalho faz um estudo da emergência e evolução da Engenharia de Sistemas e da Engenharia de Requisitos, da introdução e evolução da Engenharia de Sistemas no INPE, e desenvolve um exercício de aplicação de uma ferramenta de rastreamento de requisitos sobre a documentação de um projeto de satélites do INPE. Este estudo mostra que processos e ferramentas da Engenharia de Requisitos reduzem os erros da documentação e melhoram sua qualidade. Este estudo pretende contribuir para a modernização dos processos de Engenharia de Sistemas do INPE e assim, por sugerir melhorias nos importantes processos de requisitos, refinar as aquisições e contratações e o cumprimento de metas de custos, prazos e qualidade.

A PROPOSAL FOR THE ADOPTION OF A REQUIREMENTS CAPTURE AND TRACEABILITY PROCESS BASED IN A CASE STUDY AND HISTORY OF SYSTEMS ENGINEERING PHASES AT INPE

ABSTRACT

For several decades the National Institute for Space Research (INPE) employs system engineering methodologies in the development of space systems. However, INPE's involvement with Systems Engineering has not occurred uniformly but has passed by three phases of higher and lower evolution. In particular, the Systems Engineering at INPE has not followed the recent evolution occurred by the introduction of requirements engineering processes and tools. To attenuate this, this work presents a proposal for the adoption of a requirements capture and traceability process based in a case study and in a history of systems engineering phases at INPE. To do that, and to better understand the current situation of Systems Engineering at INPE and the causes of the not utilization of requirements engineering processes and tools, this work studies the emergency and evolution of Systems Engineering and Requirements Engineering, of the introduction and evolution of Systems Engineering at INPE, and develops an exercise of use of a requirements traceability automation tool over the documentation of a satellite project. This study demonstrates that requirements engineering tools and processes reduce errors in the documentation and improve its quality. This study intends to contribute to the modernization of INPE systems engineering processes and so, because it suggests improvements in the important requirements processes, to refine the acquisitions and contracts and the compliance with cost, schedule and quality targets.

LISTA DE FIGURAS

	<u>Pág.</u>
Figura 2.1 - Modelo de Desenvolvimento de <i>Software</i> de Pequeno Porte.	9
Figura 2.2 - Modelo de Desenvolvimento de <i>Software</i> de Grande Porte.	9
Figura 2.3 - Processo da Engenharia de Requisitos.	20
Figura 2.4 - Atividades da Engenharia de Requisitos.	21
Figura 2.5 - Decomposição Hierárquica de um Sistema.	22
Figura 2.6 - Representação Gráfica do Rastreamento de Requisitos.	25
Figura 2.7 - Modelo de Rastreabilidade de Usuário "Low End".	30
Figura 2.8 - Modelo de Rastreabilidade de Usuário "High End".	31
Figura 2.9 - Decomposição das Atividades do Gerenciamento de Requisitos.	32
Figura 2.10 - Relações de um Requisito com Outras Informações.	33
Figura 2.11 - Atributos de um Requisito.	34
Figura 2.12 - Custo para Corrigir um Erro de <i>Software</i>	44
Figura 4.1 - Evolução das Normas de Controle de Configuração.	73
Figura 4.2 - Genealogia das Normas da Engenharia de Sistemas.	78
Figura 5.1 - Crescimento da Complexidade de Sistemas Operacionais.	89
Figura 5.2 - Crescimento do Custo da Fração de <i>Software</i>	91
Figura 5.3 - Evolução das Metodologias de Desenvolvimento de <i>Software</i>	93
Figura 5.4 - Frações de Tipos de Erros de <i>Software</i>	95
Figura 5.5 - Custo para Corrigir um Problema de <i>Software</i>	95
Figura 7.1 - Sumário das Ocorrências do Termo "Traceability" nos Programas MECB, CBERS 1 & 2 e CBERS 3 & 4.	132
Figura 8.1 - Documentos do Programa CBERS 3 & 4.	139
Figura 8.2 - Parte da Árvore de Especificações do Satélite CBERS 3 & 4.	144
Figura 8.3 - Compatibilidade da Requirements Gateway.	147
Figura 8.4 - Configuração de um Projeto na Ferramenta Requirements Gateway.	151
Figura 8.5 - Cobertura para Requisitos Específicos.	152
Figura 8.6 - Cobertura para Requisitos Gerais.	152
Figura 8.7 - Tela de Configuração do Projeto na Requirements Gateway.	164
Figura 8.8 - Tela de Visão Gerencial da Requirements Gateway.	165
Figura 8.9 - Tela de Análise de Cobertura da Requirements Gateway.	166
Figura 8.10 - Navegação entre a Ferramenta Requirements Gateway e um Documento.	167
Figura 8.11 - Tela de Análise de Impacto na Requirements Gateway.	168
Figura 8.12 - Tela Gráfica na Requirements Gateway.	169
Figura 8.13 - Congestionamento da Tela de Visão Gráfica para Grandes Projetos.	175

LISTA DE TABELAS

	<u>Pág.</u>
Tabela 2.1 - Características dos Requisitos.....	14
Tabela 2.2 - Requisitos Classificados pela sua Fonte.....	16
Tabela 2.3 - Classificação de Requisitos por Tipo.	17
Tabela 2.4 - Classificação dos Requisitos por Categoria.	18
Tabela 2.5 - Alocação de Requisitos de Sistemas aos Subsistemas.....	22
Tabela 2.6 - Alocação de Requisitos até Subsistemas.	23
Tabela 2.7 - Alocação de Requisitos de Equipamentos do Subsistema A.	24
Tabela 2.8 - Porcentagem de Sucesso em Projetos de Software.	43
Tabela 7.1 - Comparação de Documentos da USAF, MECB e CBERS 3 & 4	126

LISTA DE SIGLAS E ABREVIATURAS

AAF-SAB	Army Air Force-Scientific Advisory Board
AFR	Air Force Regulations
AFSC	Air Force Systems Command
AFSCM	Air Force Systems Command Manual
ARP	Aerospace Recommended Practices
ASCEND	Advanced System for Communication and Education in National Development
AT & T	American Telephone and Telegraph
ATS	Application Technology Satellite
BMDATC	Ballistic Missile Defense Advanced Technology Center
CAD	Computer-Aided Design
CAE	Computer-Aided Engineering
CAM	Computer-Aided Manufacturing
CASE	Computer-Aided Software Engineering
CAST	Chinese Academy of Spacecraft Technology
CBERS	China-Brazil Earth Resources Satellite
CMMI	Capability Maturity Model Integration
CNAE	Comissão Nacional de Atividades Espaciais
CNES	Centre National d'Études Spatiales
CNPq	Conselho Nacional de Pesquisas
COBAE	Comissão Brasileira de Atividades Espaciais
CPFF	Cost Plus Fixed Fee
CTA	Centro Técnico Aeroespacial
DJF	Design Justification File
DOD	Department of Defense
DTIC	Defense Technical Information Center
ECSS	European Cooperation for Space Standardization
EDA	Engineering Design Aid
EIA	Electronic Industries Association/Electronic Industries Alliance
EMFA	Estado Maior das Forças Armadas
EMI/EMC	Electromagnetic Interference/Electromagnetic Compatibility
EPS	Electrical Power Subsystem
ESA	European Space Agency
ESD	Electronic Systems Division
ETE	Coordenadoria de Engenharia e Tecnologia Espaciais
FBM	French Brazilian Microsatellite
FDA	Food and Drug Administration

(continua)

(continuação)

GE	General Electric Space Division
GOCNAE	Grupo de Organização da Comissão Nacional de Atividades Espaciais
GSFC	Goddard Space Flight Center
IBM	International Business Machine
ICBM	Intercontinental Ballistic Missile
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronic Engineers
INCOSE	International Council on Systems Engineering
INPE	Instituto Nacional de Pesquisas Espaciais
ISO	International Standardization Organization
ISS	International Space Station
ITA	Instituto Tecnológico da Aeronáutica
KWIC	Key-Word In Context
MECB	Missão Espacial Completa Brasileira
MIT	Massachusetts Institute of Technology
NACA	National Advisory Committee on Aeronautics
NAS	Núcleo de Análise de Sistemas
NASA	National Aeronautics and Space Administration
NDRC	National Defense Research Committee
NRL	Naval Research Laboratory
NSF	National Sciences Foundation
OTAN	Organização do Tratado do Atlântico Norte
PERT	Project Evaluation and Review Technique
PMBOK	Project Management Body of Knowledge
PMS	Project Management System
PROPESA	Projeto de Preparação de Especialistas da Área Espacial
PTH	Plated Through Holes
RAD	Rapid Application Development
RID	Review Item Discrepancy
RSL	Requirements Statement Language
SACI	Satélite Avançado de Comunicações Interdisciplinares
SACI	Satélite Científico
SAE	Society of Automotive Engineers
SAGE	Semi Automatic Ground Environment
SATEC	Satélite Tecnológico

(continua)

(conclusão)

SBTS	Sistema Brasileiro de Televisão por Satélite
SCD	Satélite de Coleta de Dados
SEBOK	Software Engineering Body of Knowledge
SEGI	Sistema Eletrônico de Gerenciamento de Informações
SMD	Surface Mounted Devices
SPOT	Système Probatoire d'Observation de la Terre
SREM	Software Requirements Engineering Methodology
STD	Standard
STL	Space Technology Laboratories
TRW	Thompson-Ramo-Wooldridge Inc.
UML	Unified Modeling Language
USA	United States of America
USAF	United States Air Force

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	1
1.1 Contexto	1
1.2 Motivação	1
1.3 Objetivos.....	2
1.4 Organização do Documento	3
2 CONCEITOS BÁSICOS DA ENGENHARIA DE REQUISITOS	5
2.1 Definições da Engenharia de Requisitos	6
2.2 A Engenharia de Requisitos	8
2.3 A Origem da Engenharia de Requisitos.....	10
2.4 Dificuldades da Engenharia de Requisitos	11
2.5 A Importância dos Requisitos	13
2.5.1 Características dos Requisitos	14
2.5.2 Taxonomia dos Requisitos	15
2.5.2.1 Classificação dos Requisitos por sua Fonte	16
2.5.2.2 Classificação dos Requisitos por Tipos	17
2.5.2.3 Classificação dos Requisitos por Categorias.....	18
2.6 O Processo da Engenharia de Requisitos	19
2.6.1 Atividades da Engenharia de Requisitos	20
2.7 Rastreabilidade de Requisitos	21
2.7.1 A Necessidade da Rastreabilidade de Requisitos	26
2.7.2 Indefinição da Rastreabilidade de Requisitos.....	26
2.7.2.1 Exemplos de Usuários de Rastreabilidade de Requisitos	28
2.8 O Gerenciamento de Requisitos.....	32
2.8.1 A Automatização do Gerenciamento de Requisitos	34
2.8.2 A Utilidade de Ferramentas de Gerenciamento de Requisitos	35
2.8.3 Os Custos do Gerenciamento de Requisitos.....	37
2.8.3.1 Limites do Gerenciamento de Requisitos	39
2.8.4 Ferramentas Comerciais de Gerenciamento de Requisitos	40
2.8.4.1 Tipos de Ferramentas de Gerenciamento de Requisitos.....	42
2.8.5 Engenharia de Requisitos na Atualidade.....	43
2.8.5.1 Gerenciamento de Requisitos e Certificações.....	46
2.8.6 O Futuro das Ferramentas de Gerenciamento de Requisitos	47
2.9 Comentários Finais.....	48
3 FORMULAÇÃO DO PROBLEMA E ABORDAGENS PARA SUA SOLUÇÃO	49
3.1 Definição do Problema	49
3.2 Metodologia	49
3.3 Limitações do Estudo	52

3.4	Resultados Esperados.....	54
4	A ENGENHARIA DE SISTEMAS	55
4.1	Origem da Engenharia.....	56
4.2	A Emergência da Engenharia de Sistemas	57
4.3	Aquisições pela Força Aérea dos Estados Unidos	60
4.4	A Ciência como Arma de Guerra	62
4.5	O Desenvolvimento de Mísseis ICBM	65
4.5.1	Uma Abordagem de Sistemas no Desenvolvimento de Mísseis ..	66
4.6	O Impacto do Lançamento do Sputnik.....	68
4.7	Os Documentos das Séries AFSCM 375 e AFR 375	70
4.7.1	A Atenção aos Requisitos no Documento AFSCM 375-5.....	74
4.7.2	Críticas à Série AFSCM 375.....	75
4.8	A Introdução da Engenharia de Sistemas na NASA.....	78
4.8.1	A Automatização dos Processos da Engenharia de Sistemas na NASA.....	80
4.9	Frustração da Aplicação da Engenharia de Sistemas em Problemas Sociais.....	81
4.10	Comentários Finais.....	83
5	A ENGENHARIA DE REQUISITOS	85
5.1	A Origem do <i>Software</i>	85
5.2	Metodologias de Engenharia de Sistemas Aplicadas a <i>Software</i> ..	86
5.3	A Emergência da Engenharia de <i>Software</i>	88
5.4	A Origem da Engenharia de Requisitos.....	94
5.5	A Introdução da Automatização dos Processos da Engenharia de Requisitos	98
5.6	Comentários Finais.....	99
6	A ENGENHARIA DE SISTEMAS NO INPE	101
6.1	A Primeira Fase da Engenharia de Sistemas no INPE.....	101
6.1.1	O Início das Atividades Espaciais no Brasil.....	101
6.1.2	O Projeto SACI e suas Consequências	102
6.1.3	A Introdução da Engenharia de Sistemas na CNAE	104
6.1.3.1	A Disseminação da Engenharia de Sistemas pela CNAE	106
6.1.3.2	Os Manuais de Engenharia de Sistemas do INPE	107
6.1.3.3	Os Requisitos no Manual de Engenharia de Sistemas do INPE ..	108
6.1.3.4	A Documentação Conforme o Manual de Engenharia de Sistemas do INPE.....	110
6.1.3.5	Ferramentas de Engenharia de Sistemas no Projeto SACI.....	111
6.1.4	O Fim do Satélite SACI.....	111
6.2	A Segunda Fase da Engenharia de Sistemas no INPE.....	112
6.2.1	A Missão Espacial Completa Brasileira	112
6.2.2	A Reintrodução da Engenharia de Sistemas no INPE.....	113
6.2.3	A Importância dos Requisitos no Programa MECB	115
6.2.4	O Programa CBERS.....	117

6.2.5	Outros Programas Internacionais	118
6.3	Falta de Treinamento em Engenharia de Sistemas no INPE	119
6.4	A Terceira Fase da Engenharia de Sistemas no INPE	120
6.5	Comentários Finais	123

7 A EVOLUÇÃO DAS EXIGÊNCIAS DE RASTREABILIDADE NO DESENVOLVIMENTO DE SATÉLITES DO INPE 125

7.1	As Origens do Sistema de Documentação dos Programas de Desenvolvimento de Satélite do INPE	125
7.2	O Processo de Documentação nos Programas Espaciais do INPE	126
7.3	A Evolução da Rastreabilidade nos Programas de Desenvolvimento de Satélites do INPE	128
7.4	Comentários Finais	134

8 ESTUDO DO SUBSISTEMA DE ENERGIA ELÉTRICA DO SATÉLITE CBERS 3 & 4 137

8.1	O Subsistema de Energia Elétrica dos Satélites CBERS 3 & 4 ..	140
8.2	A Documentação do Subsistema EPS do Satélite CBERS 3 & 4 ..	142
8.3	Limitações do Estudo	145
8.4	A Ferramenta Requirements Gateway	146
8.5	Requisitos nas Especificações dos Satélites CBERS 3 & 4	153
8.6	Preparação dos Documentos para Uso da Ferramenta	156
8.7	Uso da Ferramenta Requirements Gateway com a Documentação do EPS	159
8.7.1	Configuração do Projeto	159
8.7.2	Descrição das Telas e das Análises da Requirements Gateway ..	160
8.7.3	Observações Durante a Preparação dos Documentos	170
8.7.4	Observações Sobre o Caso Estudado	171
8.8	Vantagens de Uso da Ferramenta Requirements Gateway	172
8.9	Limitações Observadas na Ferramenta Requirements Gateway ..	173
8.10	Comentários Finais	176

9 CONCLUSÕES, RECOMENDAÇÕES E SUGESTÕES PARA CONTINUIDADE DO TRABALHO 179

9.1	Conclusões	179
9.2	Recomendações	183
9.3	Sugestões para a Continuidade do Trabalho	184

1 INTRODUÇÃO

1.1 Contexto

Desde o início da década de 1960, o Instituto Nacional de Pesquisas Espaciais (INPE) está envolvido em atividades de pesquisa pura e aplicada na área espacial. No entanto, esse envolvimento não ocorreu de forma contínua, mas passou por três fases de maior e menor atividade fazendo com que os conhecimentos não tenham se acumulado uniformemente, mas tenham ficado estacionados ou sido perdidos.

Introduzido em Engenharia de Sistemas no final da década de 1960, o INPE somente se envolveu decididamente no desenvolvimento de um satélite em meados da década de 1980 quando uma fase de intensa atividade, notada pela instalação de um centro de controle, estações de rastreamento e laboratórios especializados, foi concluída em 09 de Fevereiro de 1993, com o lançamento de seu primeiro satélite, o SCD1.

Apesar do ambiente de rápida evolução, a partir do início dos anos 1990, provocada pela introdução de microcomputadores em todas as atividades de engenharia, ocorreu na Engenharia de Sistemas do INPE uma fase de estabilidade, em que novos processos e ferramentas de automatização não foram introduzidas.

1.2 Motivação

O INPE está envolvido com desenvolvimento de sistemas espaciais há várias décadas. Os desenvolvimentos desses sistemas são caracterizados por elevados níveis de riscos e altos custos. Anteriormente responsável por todas as fases do desenvolvimento de um satélite, atualmente o INPE concentra-se em atividades de sistemas e delega o desenvolvimento de subsistemas a empresas contratadas.

O gerenciamento de requisitos é um processo de Engenharia de Sistemas que contribui para diminuir os riscos e custos, mas que é impossível de ser feito de forma manual, sem o auxílio de uma ferramenta computacional.

A inexistência no INPE de processos adequados de preparação dos requisitos impede a utilização de ferramentas de automatização e, conseqüentemente, impossibilita o gerenciamento dos requisitos. A introdução de processos adequados de preparação de requisitos, do gerenciamento de requisitos, e de ferramentas de automatização pode ajudar a reduzir riscos e custos, e fazer melhor uso da escassa mão-de-obra disponível.

1.3 Objetivos

Este trabalho apresenta uma proposta de adoção de um processo de captura e rastreamento de requisitos baseada num estudo de caso e num histórico das fases da Engenharia de Sistemas no INPE. Para isto, o trabalho faz uma análise de evolução da Engenharia de Sistemas em geral e no INPE, em particular, para entender o estado atual da Engenharia de Sistemas no INPE e apresenta um estudo de caso de uso de uma ferramenta de automatização de rastreamento de requisitos para demonstrar sua utilidade em um projeto de desenvolvimento de satélites.

O estudo de caso é um exercício de uso da ferramenta de automatização do rastreamento de requisitos aplicado a um caso próximo do real. O caso é considerado próximo do real pois é aplicado sobre a documentação preexistente de um satélite do INPE. Esse estudo apresenta uma técnica de preparação de requisitos e uma ferramenta comercial de automatização do rastreamento de requisitos. Essa técnica e ferramenta mantêm as informações no formato tradicional do INPE, de documentos de texto eletrônico, mas usa técnicas de preparação de requisitos que permitem o uso de ferramentas de automatização do rastreamento de requisitos.

Este trabalho pretende demonstrar que essa técnica e ferramenta melhoram a qualidade dos documentos, permitem um melhor gerenciamento das

informações, fazem melhor uso da mão-de-obra, e diminuem os riscos de não-cumprimento de metas de custos, prazos e qualidade.

1.4 Organização do Documento

Este estudo pode ser dividido em duas partes. A primeira parte apresenta um levantamento histórico da emergência e evolução da Engenharia de Sistemas e da Engenharia de Requisitos, e de sua introdução e evolução no INPE. A segunda parte apresenta o estudo de caso do exercício de aplicação de ferramenta à documentação de um projeto de satélites.

O Capítulo 2 apresenta um sumário dos tópicos da Engenharia de Requisitos relacionados com o gerenciamento de requisitos e necessários ao entendimento da parte prática do trabalho.

O Capítulo 3 apresenta a abordagem de estudo empregada neste trabalho.

O Capítulo 4 e o Capítulo 5 apresentam, respectivamente, um sumário do contexto histórico da formalização da Engenharia de Sistemas e da emergência da Engenharia de Requisitos. Esses capítulos são importantes para a compreensão do contexto do estudo e por demonstrarem que ferramentas computacionais foram rapidamente introduzidas e utilizadas no processamento de grande quantidade de informações em Engenharia de Sistemas e em Engenharia de *Software*.

O Capítulo 6 apresenta um sumário do histórico da Engenharia de Sistemas no INPE com um enfoque sobre a atenção dada aos requisitos e ao uso de ferramentas de automatização. Esse capítulo é interessante por demonstrar a importância dada no INPE, desde o início da década de 1970, aos requisitos e às ferramentas computacionais de Engenharia de Sistemas.

O Capítulo 7 apresenta um sumário da evolução da maneira usada no INPE para registrar informações e as exigências de rastreabilidade ao longo de três programas de desenvolvimento de satélites. Esse capítulo é interessante por

demonstrar que a rastreabilidade de requisitos já era exigida no programa CBERS 1 & 2, mas que essas exigências não puderam ser atendidas porque processos apropriados de preparação de requisitos não foram desenvolvidos.

O Capítulo 8 apresenta um estudo da documentação de projeto de um subsistema de satélites do INPE. Nesse capítulo são apresentados e discutidos exemplos de problemas na redação de requisitos e um exemplo de aplicação de uma ferramenta comercial de automatização do gerenciamento de requisitos sobre essa documentação. São também apresentados sumários de resultados das análises feitas pela ferramenta.

O Capítulo 9 apresenta uma avaliação dos resultados do estudo, conclusões e sugestões para o prosseguimento deste trabalho.

2 CONCEITOS BÁSICOS DA ENGENHARIA DE REQUISITOS

Diferentemente das áreas tradicionais da engenharia, a Engenharia de Requisitos não é baseada em conceitos universais indiscutíveis e princípios físicos e matemáticos precisos. Sem deixar de ser uma disciplina técnica, a Engenharia de Requisitos, assim como a Engenharia de Sistemas, tem interseções com muitas áreas das ciências sociais e humanas, como Comunicação, Psicologia, Comportamento, Direito, Política, etc (KOSSIAKOFF et al, 2011, p. 34). Essa circunstância, associada ao fato que, a Engenharia de Requisitos:

- é uma disciplina relativamente nova;
- ainda está em evolução;
- é praticada de forma diferente em diferentes locais e contextos;

faz com que haja uma grande variabilidade de conceitos, terminologia e modelos da Engenharia de Requisitos, que dificultam sua padronização.

A nomenclatura da Engenharia de Requisitos está tão indefinida que o Software Engineering Body of Knowledge declara que não vai usar no seu texto os termos "requirements engineering" e "requirements engineer" (SEBOK, 2004, p. 2-1). E sobre uma tentativa de padronização através da norma Recommended Practice for Software Requirements Specifications (IEEE 830, 1998) que define conceitos da Engenharia de Requisitos aplicada a *software* e define um formato padrão de especificação de requisitos de *software*, Sommerville (2007, p. 92) afirma:

Ele (o padrão IEEE 830) é muito geral para funcionar como padrão de uma organização. É um "framework" geral que pode ser configurado e adaptado para definir um padrão dirigido às necessidades de determinada organização (SOMMERVILLE, 2007, p. 92).

Diversos autores comentam sobre essa dificuldade da Engenharia de Requisitos. E, para demonstrá-la sem ter que recorrer a muitos exemplos, basta citar o autor que talvez seja o mais crítico sobre essa situação:

"Confusion about requirements terminology extends even to what to call the whole discipline " (WIEGERS, 2003, p. 12)

Apesar da discordância em termos de conceitos, terminologia, processos, modelos, há dois aspectos da Engenharia de Requisitos em que existe absoluta concordância entre os autores: a importância dos requisitos e a dificuldade da Engenharia de Requisitos.

Não é escopo deste capítulo fazer uma revisão da literatura ou comparações entre diferentes concepções, terminologias, conceitos e modelos da Engenharia de Requisitos, apresentados por diferentes autores - a Engenharia de Requisitos é muito ampla, tem vertentes muito diferentes e não pode ser tratada adequadamente aqui. Sendo assim, aqui serão somente apresentados os conceitos importantes e necessários ao contexto e entendimento deste estudo.

O fato de a Engenharia de Requisitos ter emergido e evoluído por forças oriundas da Engenharia de *Software* dificultou encontrar um padrão de referência adequado para este estudo, já que este é dirigido para a Engenharia de Sistemas Espaciais. A grande maioria das publicações existentes é voltada para *software*; portanto, usam conceitos, terminologias, modelos, processos e jargões próprios do *software* que, muitas vezes, não se aplicam adequadamente em outras áreas. Adicionalmente, os livros sobre Engenharia de Requisitos voltados para *software* geralmente são excessivamente práticos, i.e., ensinam técnicas, regras, lições aprendidas e apresentam minimamente os conceitos básicos gerais. A necessidade de usar terminologia e conceituação gerais e uniformes, e preferencialmente não de *software*, forçou o uso de referências de um número reduzido de autores.

2.1 Definições da Engenharia de Requisitos

Da impossibilidade de encontrar definições adequadas e pela constatação que:

- as normas existentes são voltadas para *software* e estão desatualizadas;
- a maioria dos autores usa suas próprias definições;

decidiu-se, no contexto deste trabalho, compilar e usar as seguintes definições:

Documento de Requisitos:

Um documento formal que relaciona de forma ampla e genérica o que se deseja. Numa relação comprador-fornecedor, é o documento do comprador. Pode ser usado como base para uma concorrência de fornecimento, pedido de propostas etc.

Engenharia de Requisitos:

A subdisciplina da Engenharia de Sistemas e da Engenharia de *Software* que se preocupa em determinar, analisar, clarificar, documentar, validar e gerenciar as finalidades, os atributos, as capacidades, as características, as qualidades, as restrições e as funções de um sistema ou *software* durante seu ciclo de vida.

Especificação de Requisitos:

Um documento formal que relaciona de forma restrita e precisa todos os requisitos de um sistema. Numa relação comprador-fornecedor, é um documento do fornecedor. Descreve o que vai ser feito, e serve de base para o contrato de fornecimento.

Gerenciamento de Requisitos:

A ação de registrar, armazenar, alterar, manter histórico e recuperar informações atualizadas sobre requisitos, atributos de requisitos, as relações entre requisitos e entre outras informações sobre os requisitos de um sistema durante o seu ciclo de vida.

Processo:

O conjunto de funções e responsabilidades designadas a operadores, procedimentos e métodos que definem as tarefas e as relações entre as tarefas e, equipamentos e ferramentas, que são necessários à execução de um trabalho.

Rastreabilidade de Requisitos:

A capacidade de associar informações sobre as relações entre requisitos em diferentes documentos nos diferentes níveis hierárquicos de um sistema.

Requisito:

Qualquer representação (textual, matemática, tabular, gráfica, pictórica etc), informal ou formal, ampla e genérica ou restrita e precisa, de uma necessidade de um *stakeholder*, ou finalidade, atributo, capacidade, característica, qualidade, função ou restrição de um sistema.

Stakeholder:

Qualquer pessoa, grupo de pessoas, entidade ou órgão que é afetado ou tem interesses (positivos ou negativos), influencia ou impõe requisitos para um sistema ou produto. São exemplos de *stakeholders*: quem desenvolve, opera, usa, e repara o sistema, quem paga pelo sistema e por suas despesas, agências reguladoras, seus proprietários, vítimas, fornecedores, patrocinadores, defensores, competidores. Alguns autores distinguem *stakeholders* de produto (que afetam somente o produto) e de processo (que afetam os processos de obtenção do produto).

2.2A Engenharia de Requisitos

Provavelmente uma das primeiras indicações da origem da Engenharia de Requisitos está na fundamental publicação de Royce (1970) onde este apresenta e critica diferentes modelos de ciclo de vida de *software* e, entre eles, aquele que ficaria conhecido como modelo "Waterfall".

Nessa publicação, o autor apresenta um modelo de desenvolvimento de *software* de pequeno porte, mostrado na Figura 2.1, e um modelo de desenvolvimento de *software* de grande porte, apresentado na Figura 2.2.

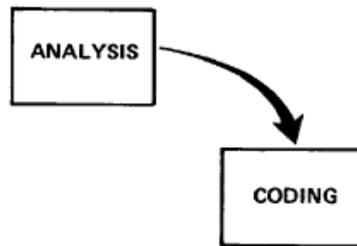


Figura 2.1 - Modelo de Desenvolvimento de *Software* de Pequeno Porte.
 Fonte: Royce (1970).

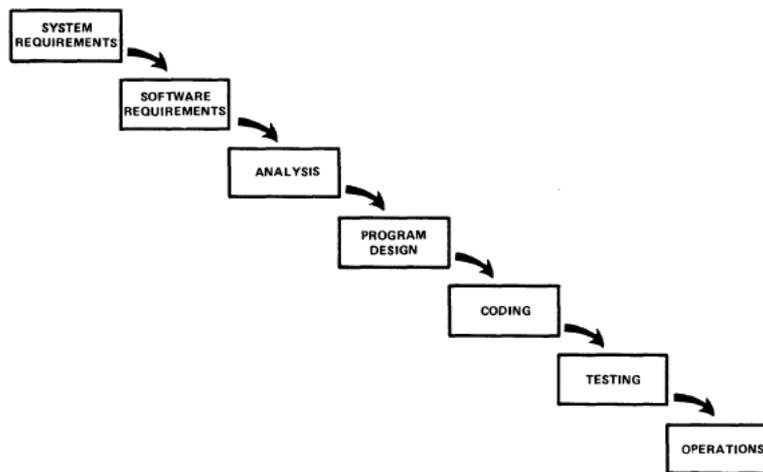


Figura 2.2 - Modelo de Desenvolvimento de *Software* de Grande Porte.
 Fonte: Royce (1970).

Comparando as Figuras 2.1 e 2.2, nota-se que no modelo de desenvolvimento de *software* de grande porte foram introduzidos blocos relacionados com requisitos de sistema e requisitos de *software*. No texto, o autor alertava que, devido à forma em cascata como o desenvolvimento é feito, no caso de haver problemas devido a requisitos nas fases finais do ciclo de desenvolvimento:

The required design changes are likely to be so disruptive that the software requirements upon which the design is based and which provides the rationale for everything are violated. Either the requirements must be modified, or a substantial change in the design is required. In effect the development process has returned to the origin and one can expect up to a 100-percent overrun in schedule and/or costs. (ROYCE, 1970).

A Engenharia de Requisitos é a responsável pelos três blocos iniciais da representação da Figura 2.2 e, desde então, aparece de formas variadas nas representações dos modelos de ciclo de vida de sistemas.

Como será visto nos parágrafos seguintes, passados mais de quarenta anos desde que a importância dos requisitos foi reconhecida, estes continuam sendo um desafio para a Engenharia de Sistemas e para a Engenharia de *Software*.

2.3 A Origem da Engenharia de Requisitos

No final dos anos 1960 devido, principalmente ao/à:

- vertiginoso aumento da capacidade dos computadores seguida à substituição de válvulas termiônicas por dispositivos eletrônicos semicondutores;
- rápida disseminação do uso de computadores em aplicações comerciais;
- aumento da complexidade do *software*, principalmente de *softwares* para aplicações militares;
- facilidade de iniciar a tarefa de codificação, que pode ser feita somente com papel e lápis;
- aparente facilidade de modificar o *software*, que fazia com que este tivesse que absorver a maior parte das modificações;
- não-existência de metodologias específicas para desenvolvimento de *software*,

ocorreu a chamada "crise do *software*", que alguns autores compararam a um "tar pit" (um campo de areia movediça onde, quanto mais se tenta escapar, mais se afunda), que se caracterizava pela grande dificuldade de cumprir metas de prazos e custos e pela baixa qualidade dos desenvolvimentos de *softwares* de grande porte.

Estudos que se seguiram demonstraram que a principal causa da crise era o mau tratamento dado aos requisitos de *software*, i.e., à má e incompleta definição do que o *software* vai fazer. Dessa crise emergiram a Engenharia de *Software*, interessada em todos os processos envolvidos no desenvolvimento de *software*, e a Engenharia de Requisitos, interessada somente nos processos relacionados com requisitos.

2.4 Dificuldades da Engenharia de Requisitos

Apesar da grande discordância em conceitos, terminologia, processos e modelos, há grande concordância a respeito das dificuldades da Engenharia de Requisitos. Vários autores expressaram as dificuldades, criticalidade e desafios da Engenharia de Requisitos, conforme se nota nas citações abaixo:

If you have not already found from experience in industry that it is hard work to write requirements, you will eventually (GRADY, 1993, p. 22).

The purpose of this book is to help you improve the practice of requirements engineering. Requirements engineering is difficult. It's not just a simple matter of writing down what the customer says he wants (YOUNG, 2004, p. 1)

Requirements analysis and requirements writing are necessary but difficult, sometimes tedious, tasks (GRADY, 2006, p. 48).

Requirements engineering is one of the most challenging aspects of software development. It is also arguably the most important aspect, as it lays the foundation for all the subsequent project work (WIEGERS, 2006, p. 3).

Requirements engineering has proven to be one of the most difficult and critical activities for the successful development of software and software intensive systems (BERENBACH et al, 2009, p. xxi).

Para justificar algumas das dificuldades com Engenharia de Requisitos, Eric Honour, no prefácio de Young (2004, p. xi), afirma que a comunicação humana é muito imprecisa e exemplifica essa afirmação com a conhecida brincadeira de sussurrar uma frase no ouvido de seu vizinho, num círculo de pessoas, e comparar a frase da última pessoa com a frase inicial. Eric Honour segue

afirmando que requisitos são uma tentativa de transportar uma idéia complexa de uma mente para outra e é falha por pretender obter precisão de simples palavras escritas e, conclui, dizendo que é absolutamente impossível escrever um requisito, mesmo muito simples, que não possa ser, honestamente, mal-entendido por outra pessoa.

Para Wiegers, não somente escrever requisitos é difícil, mas também descobrir quais são eles. Wiegers também apresenta suas "Dez Verdades Cósmicas sobre Requisitos de *Software*" (WIEGERS, 2006, p. 11):

- 1) Se você não obtiver os requisitos corretamente, não importa quão bem você executa o resto do projeto;
- 2) O desenvolvimento de requisitos é um processo de descoberta e invenção, não um processo de coleta;
- 3) Mudanças sempre acontecem;
- 4) Os interesses dos *stakeholders* se interceptam no processo de requisitos;
- 5) Envolvimento do cliente é o mais importante contribuinte para a qualidade do *software*;
- 6) O cliente nem sempre está correto, mas ele sempre tem uma opinião;
- 7) A primeira questão que um analista deve perguntar sobre um novo requisito proposto é: "Isso deve fazer parte do escopo?".
- 8) Mesmo o melhor documento de requisitos não pode - e não deve - substituir o diálogo;
- 9) O requisito pode ser vago, mas o produto deve ser específico;
- 10) Você nunca vai ter requisitos perfeitos.

Outros autores expressam outras causas das dificuldades com a Engenharia de Requisitos (algumas se sobrepõem parcialmente):

- Não é possível obter todos os requisitos corretamente no início do projeto;
- Requisitos vão mudar e novos requisitos vão surgir;
- Diferentes produtos exigem diferentes abordagens;

- Alguns problemas são inerentemente difíceis de descrever e exigem linguagens específicas;
- As prioridades do cliente mudam com o tempo;
- Nem sempre é possível satisfazer todos os *stakeholders*;
- Diferentes *stakeholders* têm diferentes interesses, culturas, visões, vocabulários;
- Uma vez que os requisitos de baixo nível são satisfeitos, outros se tornam de alta prioridade;
- Há dificuldades de comunicação com os *stakeholders*;
- Não há uma linguagem comum entre todos os *stakeholders*.

2.5A Importância dos Requisitos

Qualquer pessoa com experiência sabe avaliar a importância de se ter bons requisitos no início de um projeto, pois eles são a base para todo o trabalho que se realizará a seguir. Mal-entendidos em requisitos geralmente resultam em trabalho e esforço perdidos. Apesar disso, a abundância de citações e de casos judiciais de projetos que tiveram prazos e custos excedidos ou foram cancelados por motivos relacionados a requisitos são exemplos tanto de sua importância quanto da insuficiente atenção que lhes é dada.

Eric Honour, no prefácio de Young (2004, p. xiii), afirma que requisitos são a melhor forma de comunicar a complexidade de uma ideia, e que:

- Requisitos são uma ferramenta contratual:
Sendo este o seu mais óbvio propósito, para proteção do comprador e do fornecedor, os requisitos devem ser tão claros quanto possível;
- Requisitos são uma ferramenta de gerenciamento de configuração:
Pois um conjunto de requisitos define, de forma única, a configuração de um sistema;
- Requisitos são uma ferramenta de engenharia:
Esse papel dos requisitos geralmente não é muito percebido, pois é obscurecido por sua importância nos contratos e no controle de

configuração, mas é na engenharia que os requisitos demonstram sua força, sendo usados para:

- Descrever uma necessidade operacional;
- Definir a arquitetura do produto;
- Permitir comunicação entre desenvolvedores, compradores, usuários e outros;
- Capturar decisões sobre soluções técnicas;
- Verificar a completude dos elementos do produto;
- Verificar a completude do produto.

2.5.1 Características dos Requisitos

Requisitos têm de ser escritos ou representados de alguma forma para poderem ser comunicados entre todos os *stakeholders*. Vários autores determinam que os requisitos devem ter algumas características e devem ser escritos de uma forma imperativa e objetiva. A lista da Tabela 2.1, baseada em tabelas similares apresentadas por Bahill (2009, p. 210), Young (2004, p. 8) e Wiegers (2003, p. 182), apresenta diversas características dos requisitos.

Tabela 2.1 - Características dos Requisitos.

Característica	Descrição
Identificável	Um requisito deve estar associado a somente um identificador.
Necessário	Um requisito é, por definição, não-opcional.
Não-prescritivo	Um requisito não deve descrever uma solução.
Verificável	Um requisito deve ser verificável de alguma maneira.
Factível	Um requisito deve ser possível de ser implementado ou executado.
Correto	Um requisito deve estar corretamente redigido.
Rastreável	Um requisito deve ser rastreável à sua fonte e a outros requisitos.
Indivisível	Um requisito não deve poder ser dividido em mais requisitos e não deve usar “e”.
Completo	Um requisito deve conter todas as informações necessárias.
Conciso	Um requisito não deve ter palavras e texto desnecessários.
	(continua)

	(conclusão)
Compreensível	Um requisito deve permitir entendimento.
Claro	Um requisito deve ser facilmente entendido.
Preciso	Um requisito não deve deixar dúvidas e não usar palavras vagas como "bom", "adequado", "geralmente", "tipicamente", "eficiente" etc.
Finito	Um requisito não deve ser aberto e usar palavras "todo", "tudo", "muitos" etc.
Mensurável	Um requisito deve especificar valores e limites.
Não-ambíguo	Um requisito deve ter somente um entendimento e não deve usar "ou".
Consistente	Um requisito não deve ser contraditório.
Modificável	Uma modificação em um requisito não deve causar impactos desnecessários.
Não-obsoleto	Um requisito deve estar atualizado.
Priorizável	Uma prioridade deve estar associada a cada requisito pois facilita em caso de necessidade de redução do escopo do projeto.

Além das características dos requisitos individuais, um conjunto de requisitos deve ser:

- **Completo:** Todos os requisitos necessários devem estar contidos no conjunto;
- **Consistente:** Contradições entre os requisitos do conjunto não devem acontecer;
- **Modificável:** O texto do documento pode ser alterado mantendo-se o histórico de mudanças;
- **Rastreável:** O conjunto de requisitos deve ser rastreável a outros conjuntos.
- **Possível:** O conjunto de requisitos deve definir uma solução possível.

2.5.2 Taxonomia dos Requisitos

Diferentes autores classificam os requisitos de formas muito diferentes. Bahill e Dean (2009, p. 219) demonstram que há muita discordância em relação à classificação dos requisitos, quando afirma que Wymore distingue seis tipos de requisitos (entrada-saída, tecnologia, desempenho, custo, *trade-off* e teste); Grady distingue cinco tipos (funcional, desempenho, restrição, verificação e programática); a norma EIA-632 distingue apenas três tipos (funcional, desempenho e restrições) e a UML distingue somente outros três tipos (requisitos funcionais, requisitos não-funcionais de desempenho e requisitos suplementares).

Como será visto, a classificação dos requisitos é complexa, mas é muito importante pois, além de ter uma função didática, é útil para quem está preparando requisitos porque faz lembrar onde se deve procurar por requisitos.

2.5.2.1 Classificação dos Requisitos por sua Fonte

Bahill e Dean (2009, p. 219) classificam os requisitos por sua fonte e distinguem mais de 20 diferentes fontes de requisitos. A Tabela 2.2, adaptada de Bahill e Dean (2009), apresenta um exemplo de classificação de requisitos por sua fonte.

Tabela 2.2 - Requisitos Classificados pela sua Fonte.

FONTES DE REQUISITOS	DESCRIÇÃO
Funções	As funções que o sistema deve prover.
Entrada-saída	As relações entre entrada e saída.
Tecnologia	O conjunto de componentes disponíveis para construir o sistema.
Desempenho	Quantidades (quanto, quantos), qualidades (quão bem), coberturas (quanta área, quão distante), tempo (quão freqüente), prontidão (confiabilidade, disponibilidade).
Custo	Vários tipos de custos, como custos do trabalho, custos de recursos, custos monetários.
<i>Trade-off</i>	Uma relação entre dois outros requisitos.
Testes de sistema	Requisitos de testes podem ser incluídos bem cedo.
Autoteste	Muitos dos sistemas atuais requerem autoteste.
Política da empresa	Política própria da empresa pode gerar requisitos.
Práticas de negócios	Vários tipos de práticas da empresa.
Engenharia de Sistemas	Requisitos sobre o gerenciamento de sistemas.
Gerenciamento de projeto	Requisitos de custos e cronograma, e programa de recompensas e punições.
<i>Marketing</i>	Requisitos que podem aumentar as vendas.
Processos de manufatura	Imposições de processos de manufatura.
Engenheiros de projeto	Engenheiros que detalham o projeto geram requisitos.
Confiabilidade	Confiabilidade pode ser um requisito de desempenho.
Segurança	Alguns requisitos podem se originar de considerações de segurança.
Ética	Ética profissional e outras.
Meio ambiente	Preocupações ambientais podem gerar requisitos.
Intangíveis	Requisitos difíceis de quantificar, como requisitos estéticos, prestígio, etc.
	(continua)

	(conclusão)
Senso comum	Há requisitos que não precisam ser expressos por serem de senso comum.
Leis e padrões	Requisitos de conformidade com determinadas normas e padrões.
Cliente	Requisitos do cliente em termos de missão, objetivos, restrições, medidas de efetividade etc.
Legado	Legados de sistemas anteriormente existentes.
Atividades anteriormente existentes	Pessoal ocupado com equipamentos similares preexistentes pode gerar requisitos para um novo sistema.
Abusos por humanos	Ações ou erros de humanos.
Correção política	A necessidade de ser politicamente correto impõe requisitos.
Aceitação de materiais	Requisitos sobre a qualidade da matéria-prima, partes, componentes comerciais.
Interface	As interfaces entre componentes do sistema e com o ambiente.
Outras fontes	Outras fontes de requisitos como fatores humanos, ambientais (temperatura, umidade, choque, vibração etc.), usuário final, operador, vítimas potenciais, gerenciamento, visão da empresa, expansão futura, cronograma, política, opinião pública, parceiros, falhas passadas, responsabilidade, religião, cultura, agências governamentais etc.

2.5.2.2 Classificação dos Requisitos por Tipos

A maior parte dos autores classifica os requisitos por tipos. Sommerville (2007) classifica conforme a Tabela 2.3.

Tabela 2.3 - Classificação de Requisitos por Tipo.

Requisitos de usuário e requisitos de sistema	Funcionais			
	não-funcionais	de produto	de facilidade de uso	
			de eficiência	de desempenho
				de espaço
			de confiabilidade	
		de portabilidade		
		organizacionais	de entrega	
			de implementação	
			de padrões	
		externos	de interoperabilidade	
	Éticos			
	Legais		de privacidade	
			de segurança	
de domínio				

Além da classificação por fontes, Bahill e Dean (2009, p. 216) distinguem requisitos obrigatórios, de *tradeoff* e de funções de pontuação.

Sommerville (2007) ainda classifica os requisitos quanto a sua estabilidade em permanentes e voláteis, e distingue os últimos em mutáveis, emergentes, consequenciais e de compatibilidade e, também distingue requisitos de legados e de interface. Wiegers (2003, p. 9) distingue requisitos de negócios, de usuários, de sistemas. O Software Engineering Body of Knowledge (SEBOK, 2004, p. 2-2) distingue requisitos de produto e de processo.

2.5.2.3 Classificação dos Requisitos por Categorias

O Goddard Space Flight Center (AZZOLINI, 2000) classifica os requisitos por categorias, conforme a Tabela 2.4.

Tabela 2.4 - Classificação dos Requisitos por Categoria.

Categoria do requisito	Descrição
Requisitos nível 1	Requisitos de nível mais alto acordados com o HQ da NASA e que definem sucesso da missão.
Requisitos operacionais	Requisitos que definem como usuários e operadores interagem com o sistema e seus produtos de comando e dados.
Requisitos repartidos	Requisitos que são distribuídos quantitativamente nos níveis inferiores e para o qual a unidade de medida deve permanecer.
Requisitos derivados	Requisitos definidos pela decomposição de um requisito de nível mais alto para o qual a unidade de medida pode mudar.
Requisitos refletidos	Requisitos descobertos no processo de análise de requisitos e que outro subsistema ou elemento deve atender.
Requisitos de interface	Requisitos que especificam detalhes de características de comando, dados, elétricos, térmicos e mecânicos nos contornos dos subsistemas e elementos.
Requisitos ambientais	Requisitos que são definidos de forma que o sistema atenda a ambientes de teste, transporte, lançamento, ascensão e órbita.
Requisitos de projeto	Requisitos que definem os padrões e guias que um determinado projeto deve seguir.
Requisitos programáticos	Requisitos que incluem tolerância à falha, riscos, custo, cronograma e outras restrições de recursos.

Como se nota, a classificação dos requisitos é bastante imperfeita e é tão confusa que Wiegers (2006, p. 4) afirma:

The terminology used when discussing software requirements causes all sorts of confusion. Ten people who read a single requirement statement might each call it something different. They might term it a business requirement, user requirement, system requirement, functional requirement, software requirement, product requirement, technical requirement, constraint, feature, or simply a requirement (WIEGERS, 2006, p. 4).

2.6 O Processo da Engenharia de Requisitos

No entendimento deste estudo, o processo da Engenharia de Requisitos é o conjunto de atividades que tem por finalidade determinar, analisar, clarificar, documentar, validar e manter as finalidades, as restrições e as funções de um sistema durante seu ciclo de vida.

Também há grande variabilidade na forma como o processo da Engenharia de Requisitos é representado por diferentes autores. Essa grande variabilidade é explicada pela grande diversidade existente entre as organizações que usam a Engenharia de Requisitos, nas condições a que estas organizações estão submetidas, nas suas culturas, nas suas metodologias, nos seus níveis de maturidade, e principalmente em seus produtos e mercados. As metodologias de engenharia de *software* conhecidas como RAD (Rapid Application Development) têm uma maneira própria de trabalhar requisitos: reconhecendo a extrema mutabilidade dos requisitos de *software*, e afirmando que os requisitos vão mudar mais rapidamente do que o *software* é produzido, usam processos que são apropriados para se acomodarem rapidamente às mudanças dos requisitos. Consequentemente, não existe um modelo padrão, canônico, ideal, certo ou errado para o processo da Engenharia de Requisitos. Cada organização usa o processo que melhor se ajuste às suas necessidades.

Entre os diversos modelos do processo da Engenharia de Requisitos presentes na literatura, escolheu-se o representado da Figura 2.3, adaptado de Bahill e Dean (2009, 228), por apresentar separadamente blocos com as funções "gerenciar requisito" e "gerenciar processo de requisitos" executados durante todo o ciclo de vida do projeto. Apesar de não representados na Figura 2.3, há

um grande número de realimentações entre os blocos, e cada bloco é de fato um subprocesso, com entradas, operações, saídas. A representação com perfeição de um processo complexo em um modelo pode ser muito difícil, e, conforme nos lembra Wiegers (2003), um modelo é sempre uma simplificação e nunca representa exatamente a realidade.

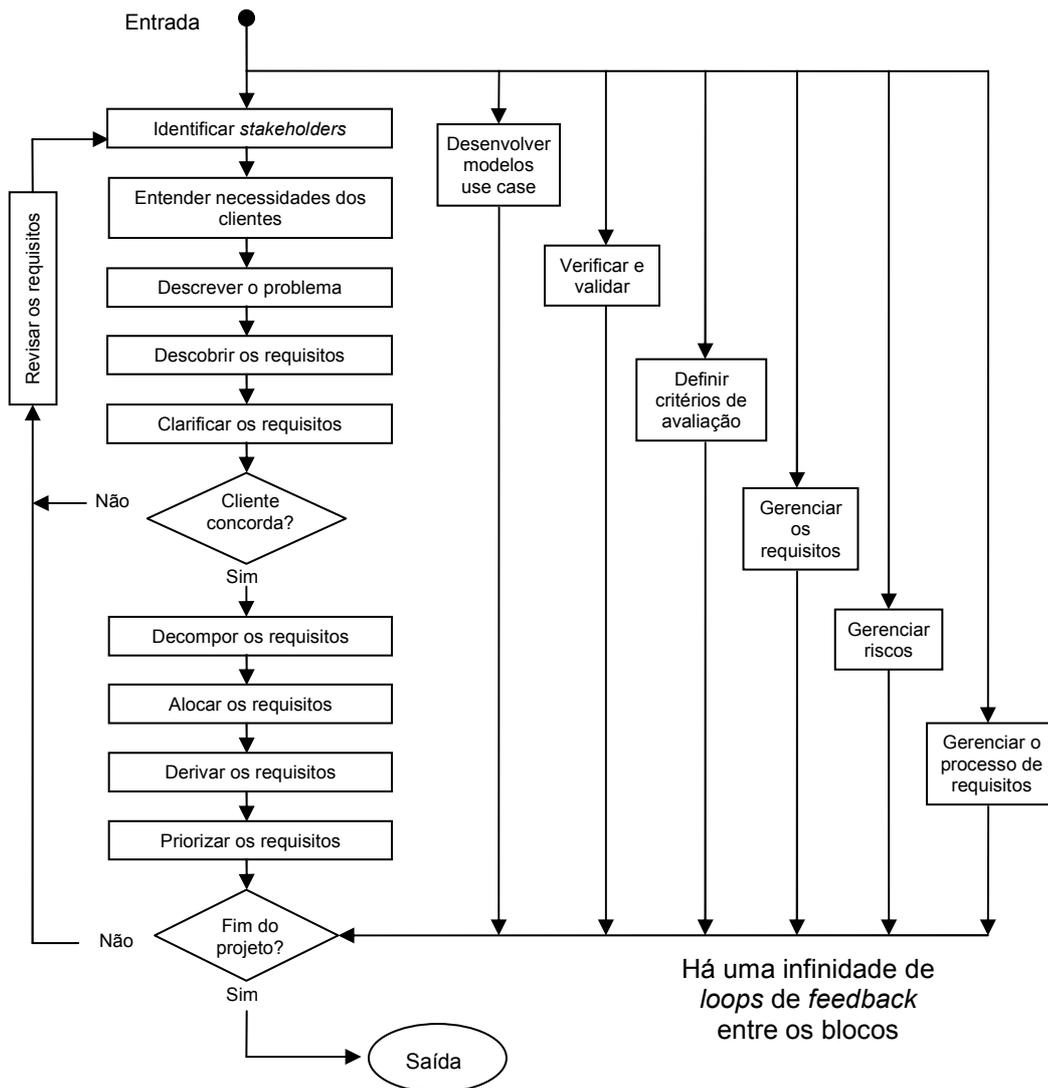


Figura 2.3 - Processo da Engenharia de Requisitos.
 Fonte: Adaptado de Bahill e Dean (2009, p. 228).

2.6.1 Atividades da Engenharia de Requisitos

As atividades executadas no processo da Engenharia de Requisitos obviamente são dependentes do modelo usado como referência. Portanto, da mesma forma como há uma grande variedade de modelos, também há uma

grande variedade nas atividades da Engenharia de Requisitos apresentadas por diferentes autores. Aqui preferiu-se apresentá-las de uma forma hierárquica como na Figura 2.4, baseada em Hokkanen (2001), por ser didática, apesar de não ser coerente na sua nomenclatura com o processo apresentado na Figura 2.3.

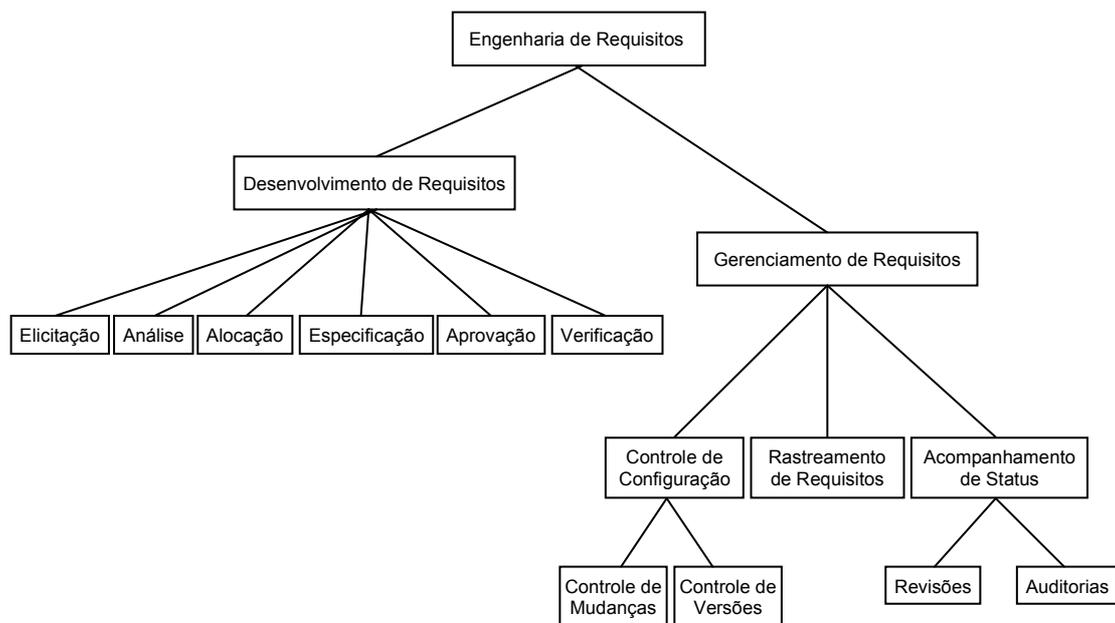


Figura 2.4 - Atividades da Engenharia de Requisitos.
Fonte: Adaptado de Hokkanen (2001).

2.7 Rastreabilidade de Requisitos

O número de requisitos aumenta rapidamente durante os processos de análise e alocação, já que cada nível hierárquico inferior tem mais elementos e maior detalhamento que o nível imediatamente superior. A Figura 2.5 apresenta, esquematicamente, esse aumento progressivo do número de requisitos, usando um exemplo baseado em Dorfman (1990), com a decomposição hierárquica de um sistema composto por três subsistemas e seis equipamentos.

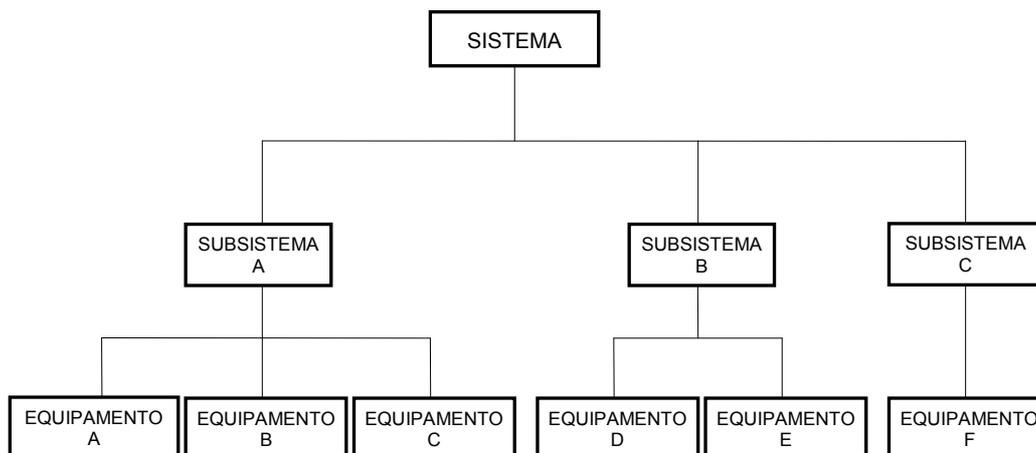


Figura 2.5 - Decomposição Hierárquica de um Sistema.

Quando o sistema é pequeno, com poucas dezenas de requisitos, ainda é possível fazer o rastreamento das relações entre os requisitos através de representações gráficas ou de tabelas de rastreabilidade. Existem várias possibilidades de tabelas, conforme mostrado a seguir, também usando exemplos baseados nos apresentados em Dorfman (1990, p. 9).

Referindo-se à decomposição hierárquica da Figura 2.5, cada marca da Tabela 2.5 indica que há uma relação entre o requisito de nível de sistema e um dado subsistema sem, no entanto, identificar qualquer requisito no nível de subsistemas.

Tabela 2.5 - Alocação de Requisitos de Sistemas aos Subsistemas.

REQUISITO DE SISTEMA	SUBSISTEMA A	SUBSISTEMA B	SUBSISTEMA C
SIS_REQ_001	X	X	
SIS_REQ_002	X		X
SIS_REQ_003		X	
SIS_REQ_004	X	X	X
SIS_REQ_005			X
SIS_REQ_006		X	
SIS_REQ_007	X		

Ainda com referência à decomposição hierárquica da Figura 2.5, a Tabela 2.6 apresenta, para cada subsistema, os requisitos no nível de subsistemas com o requisito no nível de sistemas a ele relacionado.

Tabela 2.6 - Alocação de Requisitos até Subsistemas.

REQUISITO DE SISTEMA	REQUISITO DO SUBSISTEMA A	REQUISITO DO SUBSISTEMA B	REQUISITO DO SUBSISTEMA C
SIS_REQ_001	SSA_REQ_001 SSA_REQ_002	SSB_REQ_001	
SIS_REQ_002	SSA_REQ_003 SSA_REQ_004 SSA_REQ_005		SSC_REQ_001 SSC_REQ_002
SIS_REQ_003		SSB_REQ_002 SSB_REQ_003	
SIS_REQ_004	SSA_REQ_006 SSA_REQ_007	SSB_REQ_004 SSB_REQ_005 SSB_REQ_006	SSC_REQ_003
SIS_REQ_005			SSC_REQ_004 SSC_REQ_005
SIS_REQ_006		SSB_REQ_007 SSB_REQ_008	
SIS_REQ_007	SSA_REQ_008 SSA_REQ_009		

Mais uma vez com referência à decomposição hierárquica da Figura 2.5, a Tabela 2.7 apresenta, somente para os equipamentos A, B e C do subsistema A, os requisitos de nível de equipamento relacionados com os requisitos de nível de subsistemas e com os requisitos de nível de sistemas.

Tabela 2.7 - Alocação de Requisitos de Equipamentos do Subsistema A.

REQUISITO DE SISTEMA	REQUISITO DO SUBSISTEMA A	REQUISITO DO EQUIPAMENTO A	REQUISITO DO EQUIPAMENTO B	REQUISITO DO EQUIPAMENTO C
SIS_REQ_001	SSA_REQ_001	EQA_REQ-001 EQA_REQ-002	EQB_REQ-001	
	SSA_REQ_002	EQA_REQ-003		EQC_REQ-001 EQC_REQ-002
SIS_REQ_002	SSA_REQ_003		EQB_REQ-002 EQB_REQ-003	
	SSA_REQ_004	EQA_REQ-004	EQB_REQ-004	EQC_REQ-003
	SSA_REQ_005	EQA_REQ-005	EQB_REQ-005 EQB_REQ-006	EQC_REQ-004 EQC_REQ-005
SIS_REQ_003				
SIS_REQ_004	SSA_REQ_006	EQA_REQ-006 EQA_REQ-007	EQB_REQ-007 EQB_REQ-008	EQC_REQ-006 EQC_REQ-007
	SSA_REQ_007	EQA_REQ-008		
SIS_REQ_005				
SIS_REQ_006				
SIS_REQ_007	SSA_REQ_008	EQA_REQ-009 EQA_REQ-010	EQB_REQ-009	
	SSA_REQ_009	EQA_REQ-011 EQA_REQ-012 EQA_REQ-013		EQC_REQ-008

E, finalmente, também com referência à decomposição hierárquica da Figura 2.5, a Figura 2.6 apresenta, em forma gráfica, a mesma informação da Tabela 2.7, i.e., relaciona os requisitos dos equipamentos A, B e C do subsistema A, com os requisitos nos nível de subsistemas e sistemas.

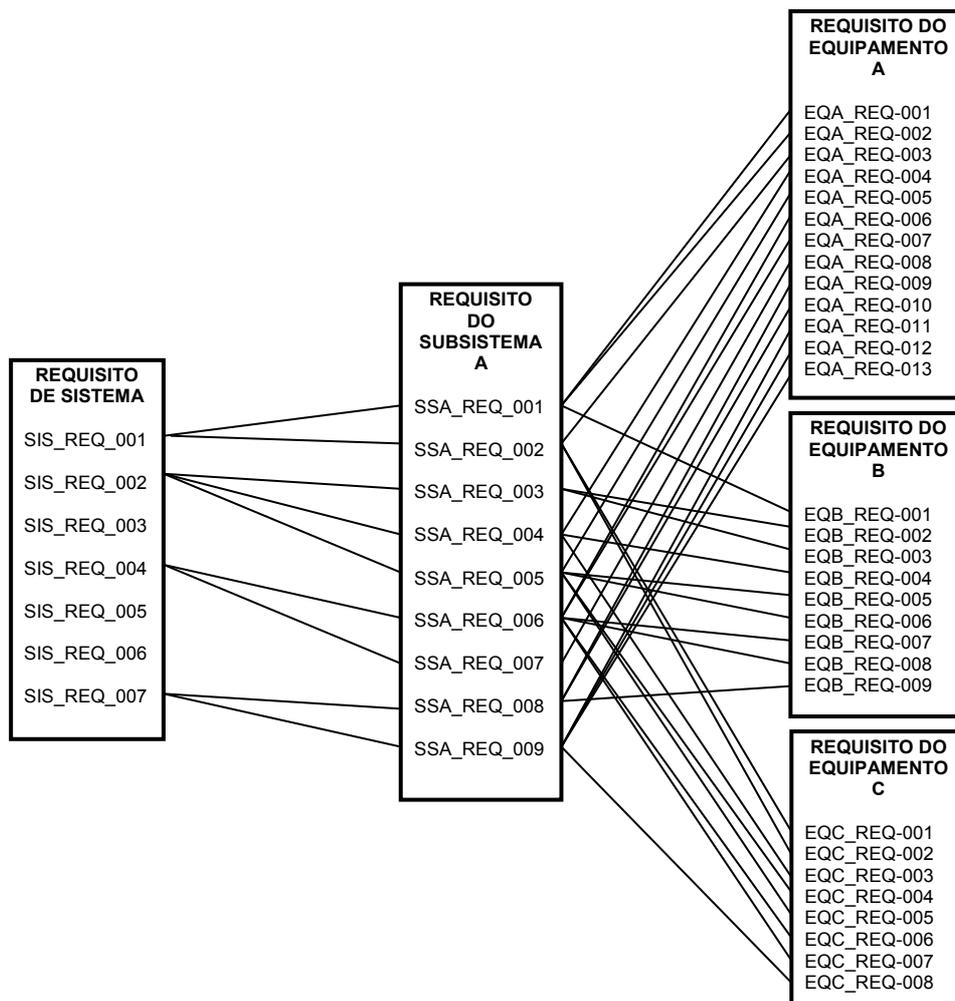


Figura 2.6 - Representação Gráfica do Rastreamento de Requisitos.

Como se nota, é bastante difícil representar as relações entre requisitos em papel, tanto em forma tabular ou quanto gráfica. Além de serem difíceis de usar, tabelas e gráficos apresentam somente as relações entre requisitos, mas não os textos dos requisitos, que precisam ser consultados. No caso do exemplo parcial da Figura 2.6, eles estão em cinco documentos; mas no sistema completo da decomposição hierárquica da Figura 2.5, eles estão em dez documentos. Isso faz com que o rastreamento manual de requisitos seja uma tarefa difícil, lenta e muito susceptível a erros.

2.7.1 A Necessidade da Rastreabilidade de Requisitos

A rastreabilidade de requisitos originou-se da necessidade de:

- Demonstrar que todos os requisitos em um nível hierárquico são cobertos por um ou mais requisitos no nível hierárquico inferior;
- Demonstrar que todos os requisitos estão sendo implementados;
- Demonstrar que todos os requisitos estão sendo verificados;
- Demonstrar que não há componentes desnecessários (não relacionados com requisitos);
- Demonstrar as relações entre requisitos, projeto, implementação e verificação;
- Demonstrar a conformidade do projeto com os requisitos;
- Auxiliar a análise de impacto de mudanças nos muito mutáveis requisitos de *software*;
- Auxiliar no controle a configuração do projeto.

2.7.2 Indefinição da Rastreabilidade de Requisitos

Apesar de ser praticada há quatro décadas, Winkler e von Pilgrim (2009, p. 1) consideram a rastreabilidade uma atividade emergente e longe da maturidade. Um dos reflexos dessa imaturidade pode ser a inexistência de uma definição universal para rastreabilidade de requisitos, fato observado por muitos autores, como Gotel e Finkelstein (1994), Ramesh et al. (1997), Hokkanen (2001), Ghazarian (2008) e Winkler e von Pilgrim (2009). Hokkanen (2001) afirma que as definições existentes são parciais, direcionadas, e válidas apenas para uma área de aplicação.

Como demonstração da falta de uma definição comum de rastreabilidade de requisitos, citam-se algumas definições encontradas:

a characteristic of a system in which the requirements are clearly linked to their sources and to the artifacts created during the system development life cycle based on these requirements (RAMESH; JARKE, 1999).

the ability to describe and follow the life of a requirement, in both a forward and backwards direction (i.e., from its origin, through its development and specification, to its subsequent deployment and use, and through periods of on-going refinement and iteration in any of these phases) (GOTEL; FINKELSTEIN, 1994).

the ability to define, capture, and follow the traces left by requirements on other elements of the software development environment and the traces left by those elements on requirements (PINHEIRO, 2004).

The ability of a software to provide a thread from the requirement to the implementation, with respect to the specific development and operational environment (IEEE STD 830, 1998).

The characteristic by which requirements at one level of a design may be related to requirements at another level (SAE, 1996).

The degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship to one another; e.g., the degree to which the requirements and design of a given system element match (IEEE STD 1233, 1998).

Uma lista mais completa seria excessivamente longa, pois, somente o site do INCOSE tem mais sete diferentes definições; e a Wikipedia tem outras quatro.

Há autores que usam o termo "traceability" para indicar a ação e não uma propriedade, enquanto outros, mais corretamente, usam "tracing". Alguns autores usam o termo "traceability" se referindo, de fato, a "requirements management".

Da mesma forma como não há uma definição única e comum para rastreabilidade de requisitos, também não há uma maneira única de fazer o rastreamento de requisitos. Complicando ainda mais, os documentos normativos que exigem rastreabilidade são genéricos e não determinam "o que rastrear" e "como rastrear". Para tentar resolver essas indefinições, Ramesh e Jarke (1999) propõem alguns modelos de referência para a rastreabilidade de requisitos.

2.7.2.1 Exemplos de Usuários de Rastreabilidade de Requisitos

A rastreabilidade de requisitos é praticada, em diferentes locais, de diferentes formas e com diferentes finalidades. Em 1993, Powers e Stubbs (1993) realizaram um interessante estudo de campo envolvendo 35 empresas usuárias de rastreabilidade de requisitos, com o fim de determinar o estado do seu uso. Esse estudo é interessante porque envolveu empresas com diferentes experiências, diferentes graus de comprometimento com rastreabilidade, e trabalhando para o Departamento de Defesa dos Estados Unidos com sistemas de diferentes graus de complexidade.

Esse estudo de Power e Stubbs (1993) classifica os usuários de rastreabilidade de requisitos em usuários "low end" e usuários "high end", sendo a distinção entre essas duas classes relacionada com a finalidade, o comprometimento, a forma como a rastreabilidade é praticada, e a experiência em seu uso e no desenvolvimento de sistemas.

A classe de usuários "low end":

- Tem 1,5 anos de experiência com ferramentas de gerenciamento de requisitos;
- Tem 10 anos de experiência desenvolvendo sistemas complexos;
- Trabalha com sistemas com menos de 1000 requisitos;
- Usa rastreabilidade como uma resposta a uma exigência do Departamento de Defesa para sistemas críticos;
- Faz o gerenciamento de forma manual por processadores de texto e planilhas.

A classe de usuários "high end":

- Tem 15 anos de experiência com ferramentas de gerenciamento de requisitos;
- Tem 30 anos de experiência desenvolvendo sistemas complexos;
- Trabalha com sistema com mais de 10.000 requisitos;

- Vê a rastreabilidade como um meio de melhorar a qualidade do projeto e do produto e de reduzir o tempo e o custo no ciclo de vida do projeto;
- Pratica a rastreabilidade pela maior probabilidade de produzir um sistema que satisfaça o cliente, que seja de fácil manutenção por manter as relações das informações importantes do projeto, como auxílio nas revisões periódicas.
- Pratica a rastreabilidade por todo o ciclo de vida do sistema;
- Faz o gerenciamento através de ferramenta específica de gerenciamento de requisitos.

Para representar a forma como a rastreabilidade de requisitos é implementada Powers e Stubbs (1993) criaram modelos de rastreabilidade "típicos", i.e., uma agregação das técnicas gerais implementadas e usadas pelos usuários "low end" e "high end" conforme representado pelas Figuras 2.7 e 2.8.

Observa-se que o estudo usou o termo "rastreabilidade de requisitos" para o sofisticado "gerenciamento de requisitos" praticado pelos usuários "high end".

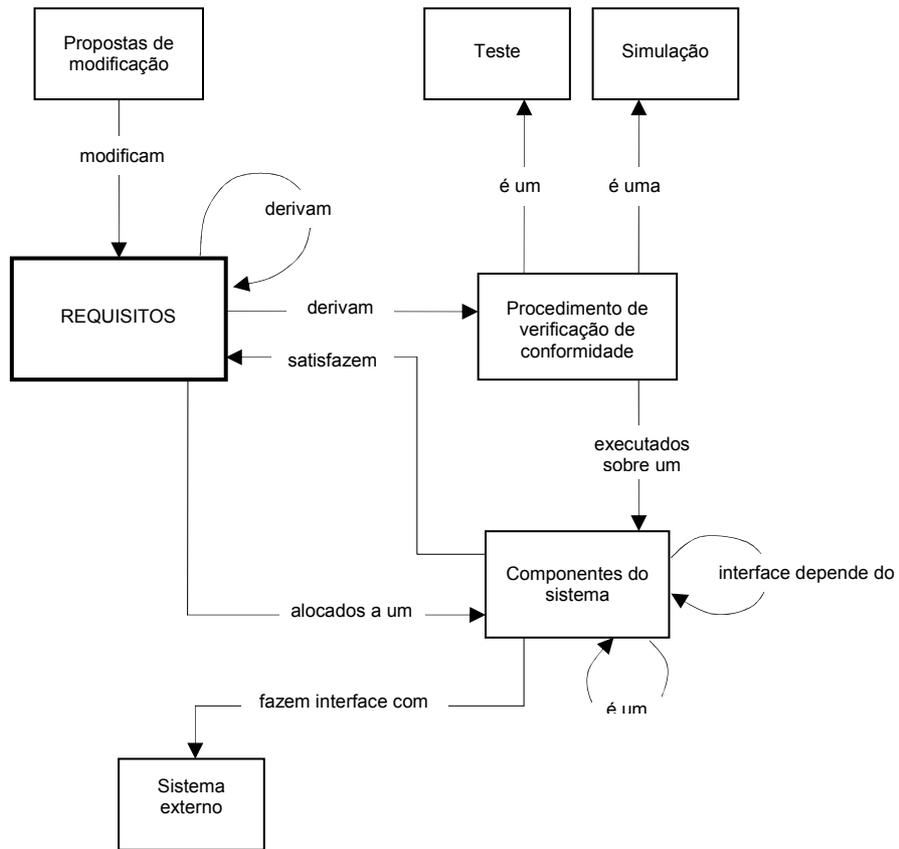


Figura 2.7 - Modelo de Rastreabilidade de Usuário "Low End".
 Fonte: Adaptado de Power e Stubbs (1993).

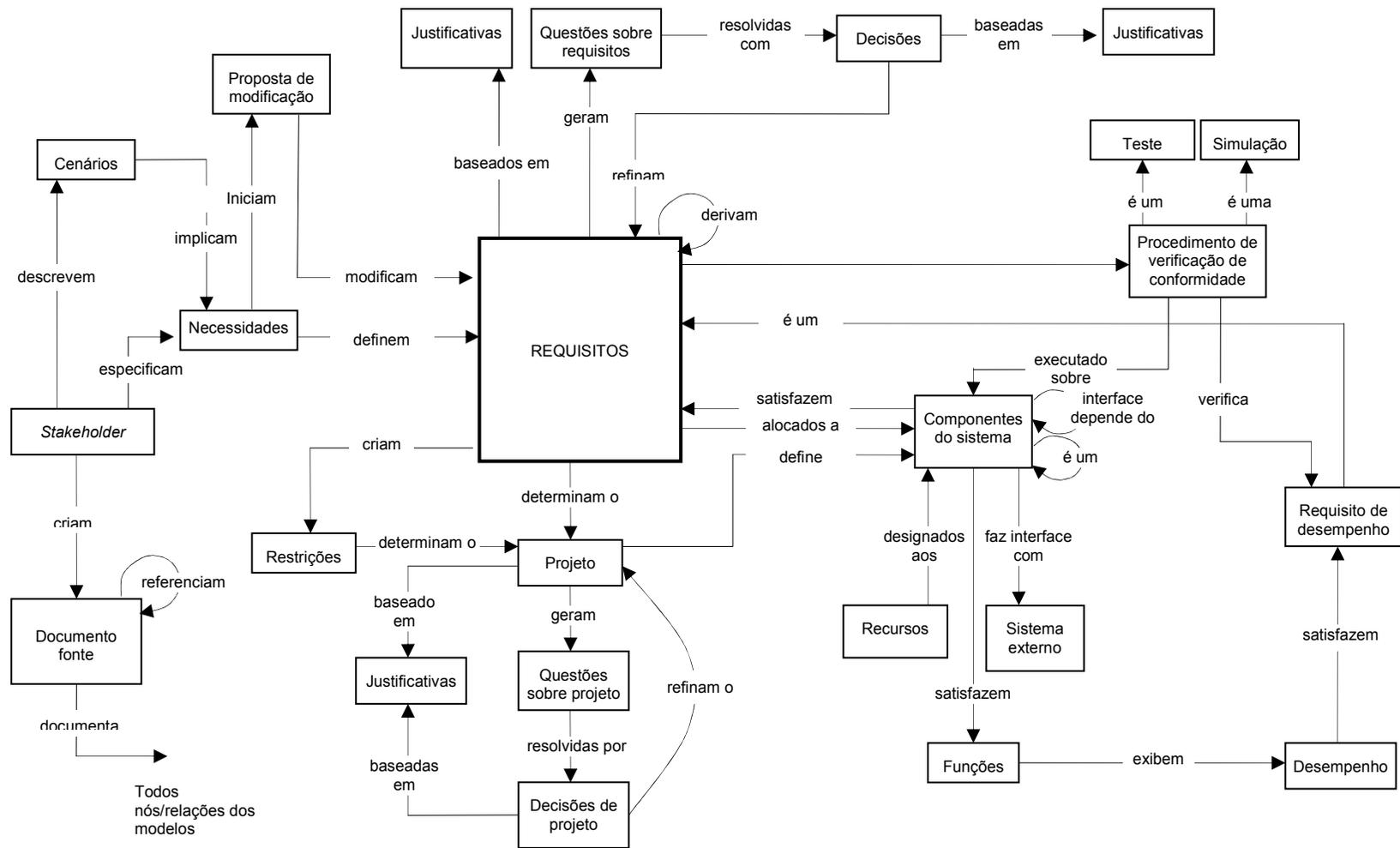


Figura 2.8 - Modelo de Rastreabilidade de Usuário "High End".
 Fonte: Adaptado de Powers e Stubbs (1993).

2.8 O Gerenciamento de Requisitos

Usa-se o termo “gerenciamento de requisitos” para o processo que, além do rastreamento de requisitos, inclui outras atividades, e.g., controle de configuração, acompanhamento de status, etc. O gerenciamento de requisitos é mais complexo que o rastreamento de requisitos por ter características multidimensionais. As outras dimensões estão relacionadas com as informações adicionais sobre os requisitos: *What?*, *Who?*, *Where?*, *How?*, *Why?*, e *When?* (RAMESH; JARKE, 1999). Curiosamente, não há muitas publicações sobre gerenciamento de requisitos e há um número muito menor de diferentes definições.

Wiegiers (2003, p. 314) propõe a decomposição apresentada na Figura 2.9 para as atividades do gerenciamento de requisitos:

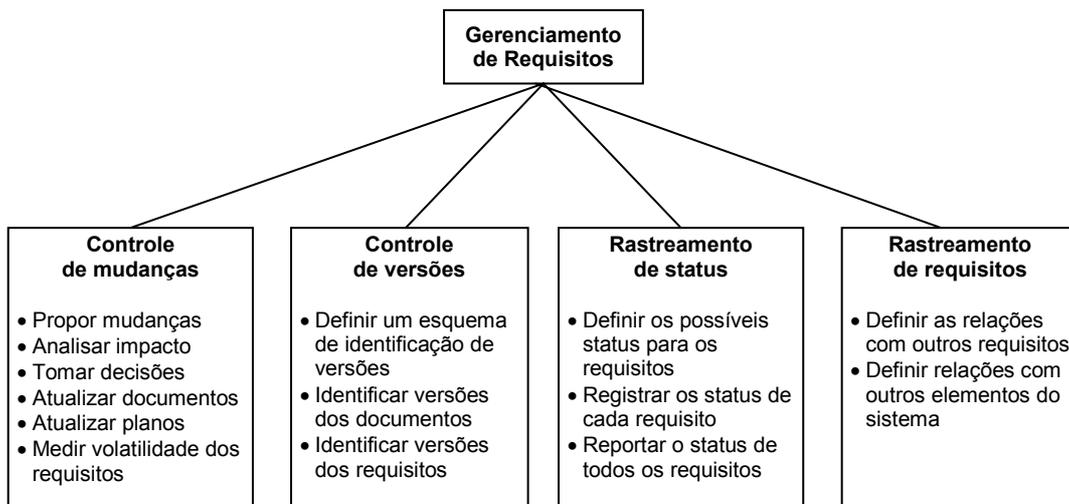


Figura 2.9 - Decomposição das Atividades do Gerenciamento de Requisitos.
Fonte: Adaptado de Wiegiers (2003, p. 314).

Como já demonstrado para o rastreamento de requisitos, é ainda mais difícil fazer manualmente o gerenciamento de requisito devido às muitas dimensões e à grande quantidade de informações sobre requisitos que precisam ser armazenadas.

A Figura 2.10 mostra exemplos de tipos de documentos que podem estar relacionados com um requisito. Observa-se que requisitos podem estar relacionados com muitas outras informações, e não apenas outros requisitos em uma cadeia hierárquica de documentos de requisitos. Dessa forma, as designações "rastreamento de requisitos" e "gerenciamento de requisitos" podem ser inadequadas para designar "gerenciamento de informações" de uma maneira ampla e geral.



Figura 2.10 - Relações de um Requisito com Outras Informações.

Adicionalmente, cada requisito ou qualquer informação pode ter muitos atributos, como mostrado na Figura 2.11.

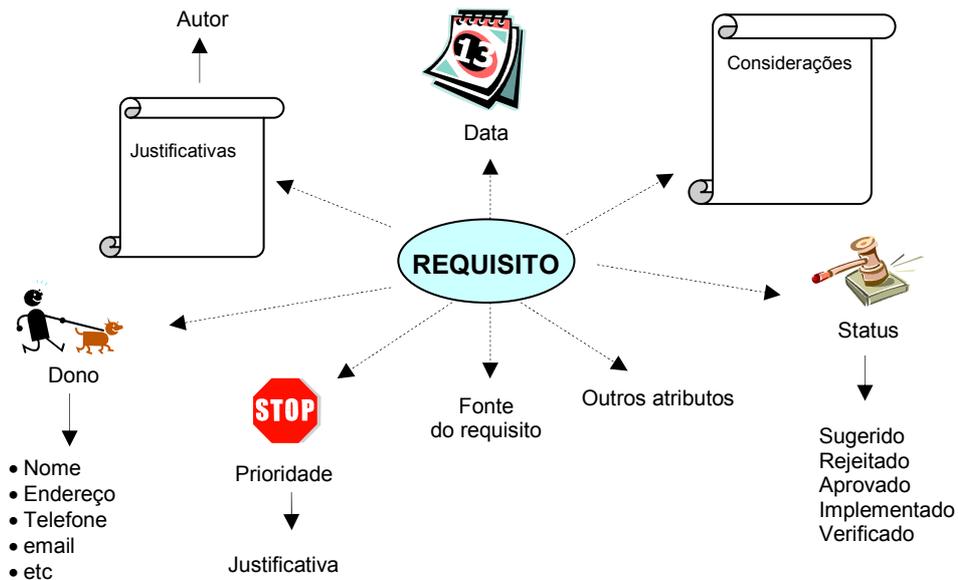


Figura 2.11 - Atributos de um Requisito.

2.8.1 A Automatização do Gerenciamento de Requisitos

A mais antiga descrição encontrada de uma ferramenta de automatização do rastreamento de requisitos é a definição simples e clara da ferramenta ARTS (*Automated Requirements Traceability System*), apresentada em Dorfman e Flynn (1984):

ARTS is a bookkeeping program that operates on a data base consisting of system requirements and their attributes. It provides upward and downward traceability in a hierarchical structure, as well as data base management and output operations on any requirement-related attributes selected by the user. (DORFMAN; FLYNN,1984)

Nessa publicação Dorfman e Flynn afirmam que apesar de não haver disputa com relação à necessidade de rastreabilidade de requisitos, poucos projetos estabelecem rastreabilidade, e explicam as razões para a sua não-implementação:

- Os parágrafos em prosa das especificações contêm vários requisitos;
- Apesar de haver rastreabilidade entre componentes do sistema, não há rastreabilidade entre os requisitos dos componentes;

- O enorme volume de requisitos dos sistemas de grande porte impossibilita o rastreamento manual.

Apesar de o nome da ferramenta se referir a "traceability", de fato a ARTS era uma sofisticada ferramenta de gerenciamento de requisitos pois, além de registrar as relações entre requisitos, tinha um banco de dados que armazenava atributos dos requisitos e dava suporte para controle de configuração.

Curiosamente, essa descrição do ARTS somente foi publicada em abril de 1984 enquanto o trabalho informa que a ferramenta estava operacional em várias plataformas desde outubro de 1980. Não publicar informações sobre essa ferramenta durante pelo menos quatro anos concorda com a informação, de um trabalho posterior dos mesmos autores, que diz que ferramentas de rastreamento de requisitos eram usadas por empresas da área aeroespacial como "arma secreta" para obter vantagens comerciais em relação a seus concorrentes (FLYNN; DORFMAN, 1990, p. 424).

2.8.2A Utilidade de Ferramentas de Gerenciamento de Requisitos

Manter os requisitos em documentos de papel não permite o gerenciamento de requisitos pois:

- Dificulta fazer alterações nos documentos;
- Dificulta manter as informações atualizadas em todas as cópias;
- Não permite registrar informações adicionais sobre os requisitos;
- Dificulta registrar relações entre as informações;
- Dificulta rastrear as informações;
- Dificulta avaliar o impacto de mudanças.

O gerenciamento de requisitos com uma ferramenta computacional tem as seguintes vantagens:

No gerenciamento do projeto:

- Aumenta a produtividade;
- Aumenta a velocidade do desenvolvimento;
- Reduz o desvio de escopo;
- Diminui problemas de comunicação;
- Diminui o retrabalho;
- Reduz o caos do projeto;
- Reduz os riscos;
- Permite melhores estimativas de custos e prazos;
- Permite melhor avaliação do impacto de mudanças;
- Permite melhor avaliação do status e do progresso do projeto;
- Permite acompanhar o status do projeto ao nível de requisito (aprovado, rejeitado, implementado, verificado etc.);
- Permite controlar o acesso às informações;
- Registra a memória do projeto;
- É fundamental em projetos de longa duração devido à saída e substituição de pessoal e à falha da memória das pessoas.

Na engenharia:

- Facilita fazer mudanças nas informações;
- Reduz erros nos requisitos;
- Permite armazenar atributos e outras informações sobre os requisitos;
- Facilita comunicar modificações aos *stakeholders*;
- Reduz o impacto de mudanças na equipe de projeto;
- Permite visualizar e comparar diferentes versões das informações;
- Facilita manter o documento atualizado;
- Permite recuperar o histórico de cada requisito;
- Permite rápida manipulação e visualização das informações;
- Permite definir relações entre informações em diferentes documentos;
- Permite melhor avaliação do impacto de mudanças;
- Permite melhor gerenciamento das informações;

- Demonstra as relações entre requisitos, projeto, implementação e verificação.

No gerenciamento da configuração:

- Automatiza o controle de configuração;
- Permite alterações somente através de processo rigorosamente controlado;
- Armazena todas as versões dos requisitos em uma base de dados única atualizada;
- Assegura que todos enxerguem a mesma versão das informações;
- Minimiza erros;
- Auxilia encontrar erros;

Na verificação do projeto:

- Auxilia os processos de verificação e validação;
- Facilita demonstrar a conformidade do projeto com os requisitos;
- Facilita demonstrar a completude do projeto;

Na manutenção:

- Facilita a manutenção;
- Possibilita a recuperação de todas as informações sobre o projeto.

2.8.3 Os Custos do Gerenciamento de Requisitos

Ninguém questiona que o gerenciamento de requisitos é importante, mas o gerenciamento de requisitos tem custos:

- Custo de aquisição da ferramenta;
- Custo de treinamento de operadores no uso da ferramenta;
- Custo de manter operadores durante todo o projeto;
- Custo do *hardware* (servidor) que vai hospedar a ferramenta e onde vão ficar armazenadas todas as informações;
- Custo de manutenção, suporte, administração e *upgrades*.

Diferentemente de arquivos de papel que não são facilmente destruídos, dados eletrônicos podem ser facilmente corrompidos ou apagados. Portanto, deve haver preocupação constante com a criação de *backups* e com o acesso de pessoas não autorizadas aos dados dos arquivos.

Ferramentas são somente auxiliares do gerenciamento de requisitos, i.e., requisitos e outras informações têm de ser corretamente introduzidos na ferramenta. Portanto, o uso de uma ferramenta de gerenciamento de requisitos necessita de operadores que conheçam os recursos da ferramenta e de pessoal que conheça o sistema sendo desenvolvido para poder registrar as informações corretamente. Dados entrados erradamente na ferramenta podem ser impossíveis de serem corrigidos no futuro, pois pode não haver nenhum outro registro exceto aqueles que estão na ferramenta.

Hokkanen (2001, p. 36) alerta que o uso de uma ferramenta de gerenciamento de requisitos é um compromisso que não pode ser interrompido, pois pode ser impossível recuperar o status das informações se houver um intervalo sem entrada de registros na ferramenta; e mesmo com diretrizes eficientes, corretas e claramente definidas, não há garantias que os operadores farão uso correto da ferramenta.

Wieggers (2003, p. 278 e p. 298) ainda lembra que ferramentas não substituem processos e não conseguem superar as deficiências de um processo, e a escolha, criação e implantação dos processos devem ser feitos antes da seleção da ferramenta para que esta seja compatível com aqueles processos.

Shroff (2001, p. 39) expõe dessa forma esse problema:

Requirements Management tools are enablers that drive the existing requirements management process. They do not define the requirements management process. They are tools to assist existing requirements management processes. A common term associated with requirements management tools in the software industry is 'silver bullet syndrome'. The user becomes enamored with a particular tool and expects it to solve all the problems based on the marketing pitch of a particular vendor. However, the user does not realize that a tool is only a tool. If

people write poor requirements, test badly, and haphazardly do impact assessment, the tool is not going to help. The tool cannot solve communication problems between the programmer, manager, marketing, etc. Any requirements management tool is only as good as the discipline of the people using it. (SHROFF, 2001, p. 39)

Adicionalmente, o uso de uma ferramenta de gerenciamento de requisitos pode exigir mudanças de cultura da empresa, pois tais ferramentas presumem a prática rigorosa do controle de configuração, às quais algumas empresas não estão acostumadas. Wiegers (2003) relata alguns casos interessantes, e supostamente reais, de mal-uso dessas ferramentas.

2.8.3.1 Limites do Gerenciamento de Requisitos

Devido às dificuldades e aos custos do gerenciamento de requisitos, deve haver um limite na quantidade de informações que se deve rastrear. Infelizmente, esse assunto não é bem tratado na literatura. O sumário abaixo é a visão de Fetters, Hsu e Smeed (1999) de como adaptar um ambiente de rastreamento às necessidades de um projeto:

- Definir quais são os dados de projeto que serão rastreados e a estratégia de rastreamento, considerando:
 - O uso esperado do rastreamento;
 - As obrigações contratuais;
 - As leis;
 - As experiências de projetos similares.
- Os dados a serem rastreados podem ser classificados nos seguintes tipos:
 - Dados do produto, que representam os requisitos do sistema;
 - Dados suplementares, que explicam ou justificam os requisitos do sistema;
 - Dados observacionais dos processos;
 - Dados de dependência que representam relações entre os três tipos de dados.

- Sobre a estratégia de rastreamento:
 - Deve ser especificado que dado de rastreamento vai ser usado em cada situação;
 - Deve ser decidido como o dado deve ser registrado;
 - Deve ser definida a fonte de dados de rastreabilidade;
 - Deve ser definida em que situação a informação de rastreamento registrada deve ser considerada;
 - Deve ser definido o uso de informações agregadas, simplificadas ou generalizadas;
 - A definição da estratégia de rastreamento é determinada pela experiência do gerente do projeto.

2.8.4 Ferramentas Comerciais de Gerenciamento de Requisitos

A necessidade de rastrear requisitos de *software* e de sistemas fez com que muitas empresas desenvolvessem produtos comerciais de gerenciamento de requisitos. As primeiras ferramentas comerciais de gerenciamento de requisitos surgiram entre o final dos anos 1980 e o início dos anos 1990 (GOTEL; MADER, 2009, p. 2) e sua importância atual é demonstrada pelo grande número de ferramentas comerciais atualmente disponíveis. Um levantamento feito pelo INCOSE e disponibilizado no seu website (INCOSE, 2011) lista quase meia centena de ferramentas de gerenciamento de requisitos.

Shroff (2001, p. 38) lista algumas das características desejáveis das ferramentas comerciais de gerenciamento de requisitos:

- Dar suporte para funcionalidades comuns como: identificação, visualização e edição de requisitos, rastreamento de requisitos até sua origem, análise de impacto de mudanças, verificação de completude e consistência, análise de métricas, controle de mudanças (registro de qualquer adição, supressão ou mudança nos requisitos), visualização de subconjuntos de requisitos (listar, filtrar, ou listar subconjuntos com determinado valor de atributo), gerar relatórios;

- Dar ajuda na comunicação dos requisitos entre os membros das equipes do sistema em desenvolvimento;
- Facilitar a configuração para se adaptar aos processos existentes na organização;
- Ser baseado em tecnologia comercialmente disponível, provada e escalável;
- Dar suporte a plataformas operacionais heterogêneas;
- Dar suporte ao gerenciamento de configuração.

Kannenber e Saiedian (2009, p. 17) informam que muitas das ferramentas atuais de rastreamento de requisitos são vendidas na forma de pacotes completos, sendo o rastreamento somente uma das funcionalidades. Kannenberg e Saiedian (2009) também apontam exemplos de algumas das suas deficiências e dificuldades de uso:

- Algumas ferramentas dão um suporte muito simplista ao rastreamento;
- Não automatizam todo o processo de rastreamento;
- A funcionalidade de rastreabilidade somente funciona se a metodologia empregada na empresa é criada em torno da ferramenta;
- A mesma ferramenta deve ser usada desde o início do projeto;
- Indicam as partes afetadas pelas mudanças, mas não dão qualquer auxílio na execução das alterações das relações de rastreamento;
- Tem problemas de integração com outras ferramentas;
- Aparentam terem sido desenvolvidas a partir de ferramentas experimentais de pesquisa;
- Tem alto preço;
- Mesmo que uma ferramenta seja usada, nem sempre a informação é mantida adequadamente.

E concluem que, mesmo que uma ferramenta seja usada, não é possível confiar que a informação que é fornecida está atualizada ou é correta.

Não seria possível descrever a grande variedade de funcionalidades oferecidas pelas ferramentas disponíveis no mercado pois cada ferramenta tem sua maneira de operar e apresentar análises e resultados.

2.8.4.1 Tipos de Ferramentas de Gerenciamento de Requisitos

Todas as ferramentas de gerenciamento de requisitos são baseadas em bancos de dados de requisitos, atributos e relações com outras informações. No entanto, distinguem-se principalmente dois tipos de ferramentas de gerenciamento de requisitos pelas diferentes maneiras como são estruturadas e usadas:

a) Ferramentas centradas em documentos:

Esse tipo de ferramenta é adequado para as organizações que, por razões culturais e de tradição, preferem trabalhar com os requisitos em formato de documento de texto. Uma vez que o documento está dentro da ferramenta de gerenciamento, os requisitos passam a ser elementos de um banco de dados e é possível armazenar relações, atributos e outras informações que ficam armazenados somente na ferramenta e são filtrados quando se visualiza o documento de texto. É possível fazer alterações no documento sem sair do ambiente da ferramenta.

b) Ferramentas centradas em base de dados:

Nesse tipo de ferramenta todos os requisitos, relações e atributos são armazenados num banco de dados. Não há necessariamente documentos impressos, mas relatórios impressos podem ser criados em vários formatos padronizados, inclusive como uma especificação de requisitos, com requisitos ou qualquer outra informação que é selecionada do banco de dados.

Shroff (2001, p. 40) distingue as ferramentas centradas em base de dados em dois tipos: baseadas em base de dados relacionais e baseadas em base de dados orientadas a objetos.

2.8.5 Engenharia de Requisitos na Atualidade

A importância dos requisitos é demonstrada pelo fato de o Departamento de Defesa dos Estados Unidos gastar 4% do custo do ciclo de vida do sistema em captura e rastreamento de requisitos (RAMESH et al., 1997). No entanto, apesar da contínua evolução e das inúmeras metodologias e ferramentas hoje existentes, os problemas observados no passado continuam a se manifestar nos desenvolvimentos de *software*. Dados do “Chaos Report” - um conhecido relatório do estado da indústria de *software* publicado pelo Standish Group - apresentados na Tabela 2.8, demonstram que a porcentagem de sucesso em projetos de desenvolvimento de *software* é baixa.

Tabela 2.8 - Porcentagem de Sucesso em Projetos de Software.

Fonte: Eveleens e Verhoef (2010).

Standish project benchmarks over the years			
Year	Successful (%)	Challenged (%)	Failed (%)
1994	16	53	31
1996	27	33	40
1998	26	46	28
2000	28	49	23
2004	29	53	18
2006	35	46	19
2009	32	44	24

E, a julgar pela similaridade da Figura 2.12 com a Figura 5.5, extraídas de referências separadas 33 anos no tempo, requisitos continuam a ser os principais responsáveis pelos problemas com desenvolvimento de *software*. A possibilidade de ambos os gráficos basearem-se nos mesmos dados não invalida o argumento.

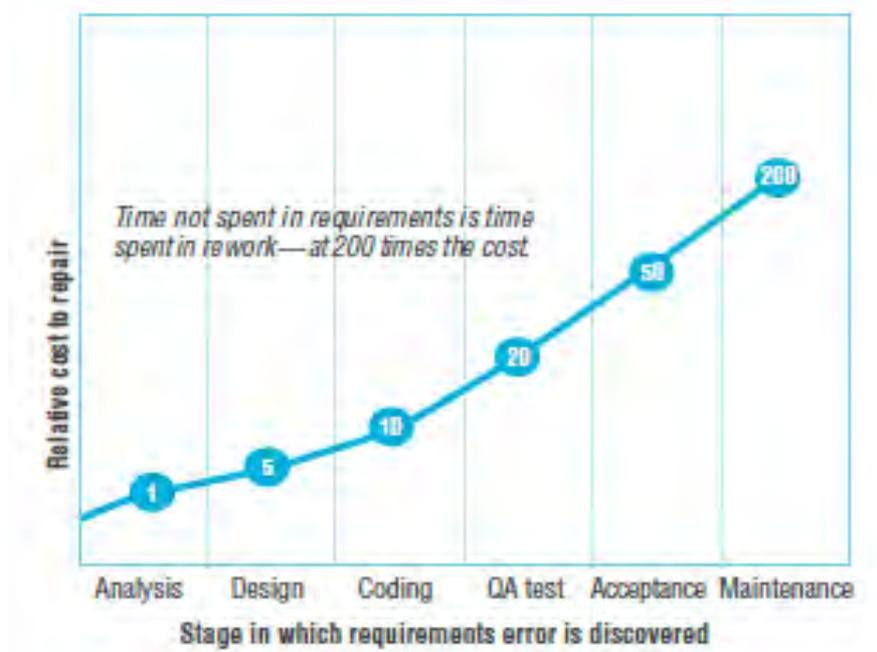


Figura 2.12 - Custo para Corrigir um Erro de *Software*.

Fonte: Material publicitário Rational Doors (IBM, 2009).

A Engenharia de Requisitos é, atualmente, de grande importância no desenvolvimento de sistemas. No entanto, as principais forças que atualmente agem e que dão importância à Engenharia de Requisitos são de natureza bastante diferente daquelas que lhe deram origem, i.e., enormes sistemas de defesa. As principais forças evolutivas atuais vêm da área comercial e podem ser resumidas em/no/na:

- Engenharia de Requisitos é a primeira fase da Engenharia de Produto; se for malfeita, comprometerá todo o trabalho posterior;
- Onipresença de produtos comerciais com *software* embutido e muitas funcionalidades;
- Desenvolvimento de *softwares* e sistemas comerciais grandes, complexos e integrados;
- Maior exigência dos consumidores;
- Enorme competição por mercados forçando desenvolvimentos com custo e em tempo reduzidos;
- Chegar ao mercado antes, ou ter uma função adicional ou mais bem planejada, pode fazer a diferença entre o sucesso ou o fracasso comercial de um produto;

- Rápida evolução tecnológica causa rápida obsolescência e substituição de produtos;
- Pressões exercidas por outros *stakeholders* e que podem afetar o sucesso do produto;
- Desenvolvimentos globalizados, com partes desenvolvidas em diferentes partes do mundo por diferentes empresas, exigem rigoroso gerenciamento;
- Desenvolvimentos de produtos para aplicações críticas, regulados por processos de certificação que exigem demonstração de processos de gerenciamento de requisitos.

Grady (2006, p. 39), afirma que o próprio DoD reconhece que hoje há ascendência do setor comercial sobre o militar, pois ele faz aquisições de equipamentos e *softwares* comerciais para uso em aplicações militares.

A importância atual da Engenharia de Requisitos no desenvolvimento de *softwares* e sistemas é também demonstrada pelo grande número de recentes publicações de livros voltados para requisitos e Engenharia de Requisitos, como pode ser visto pela lista abaixo:

- Requirements Engineering; Elizabeth Hull, Ken Jackson, Jeremy Dickson, Springer-Verlag London Limited, 2011;
- Relating Software Requirements and Architecture; Paris Avgeriou, John Grundy, Jon G. Hall, Patricia Lago, Ivan Mistrík (editores), Springer-Verlag, Berlin Heidelberg, 2011;
- Requirements Engineering Fundamentals: A Study Guide for the Certified Professional for Requirements Engineering Exam - Foundation Level - IREB compliant; Klaus Pohl and Chris Rupp, Rocky Nook Inc., 2011;
- Requirements Engineering: from Systems Goal to UML Models to Software Specifications, Axel van Lamsweerde, John Wiley, 2009;
- Discovering Requirements, Ian Alexander e Ljerka Beus-Dukic, John Wiley and Sons, Chichester, England, 2009;

- Software and System Requirements Engineering in Practice, Brian Berenbach, Daniel J. Paulisch, Juergen Kazmeier, Arnold Rudorfer, McGraw Hill, 2009;
- Requirements Engineering for Software and Systems, Phillip A. Laplante, CRC Press, 2009;
- Requirements Management: The Interface Between Requirements Development and All Other Systems Engineering Processes, Colin Hood, Simon Wiedemann, Stefan Fichtinger, Urte Pautz, Springer-Verlag Berlin Heidelberg, 2009;
- More About Software Requirements, Thorny Issues and Practical Advices; Wiegers, K., Microsoft Press, 2006;
- The Requirements Engineering Handbook; Young R. R., Artech House, Inc., 2004;
- Systems Requirements Analysis; Grady, J. O., Academic Press, 2006;
- Perspectives in Software Requirements; Leite, J. C. S. P.; Doorn, J. H., Kluwer Academic Publishers, 2004.

Devido à atual importância da Engenharia de Requisitos notam-se muitas publicações na última década sobre Engenharia de Requisitos e rastreabilidade de requisitos. Mas, comparando-se os trabalhos recentes com os trabalhos anteriores, conclui-se que a parte mais importante do trabalho foi publicada antes de 1995.

2.8.5.1 Gerenciamento de Requisitos e Certificações

Atualmente, são exigidas demonstrações de uso de processos de gerenciamento de requisitos para obtenção de certificações no desenvolvimento de *softwares* e sistemas para aplicações “safety critical” (usinas nucleares, aviões, equipamentos e produtos médico-farmacêuticos, alimentos, automóveis, etc), como, por exemplo, as regidas pelas seguintes regulamentações:

- IEC-61513 Nuclear power plants - Instrumentation and control for systems important to safety - General requirements for systems, para desenvolvimento de sistemas de segurança centrais nucleares;
- DO-178C Software Considerations in Airborne Systems and Equipment Certification, para desenvolvimento de *software* aeronáutico;
- DO-254 Design Assurance Guidance for Airborne Electronic Hardware, para desenvolvimento de *hardware* complexo de uso aeronáutico;
- FDA 21 CFR Part 11 Electronic Records; Electronic Signatures, para registros eletrônicos de produtos farmacêuticos;
- EN5012x/IEC 62425 IEC 62425. Railway applications. Communication, signaling and processing systems. Safety related electronic systems for signaling, para desenvolvimento de sinalização ferroviária de segurança.
- ISO 26262 Road Vehicle – Functional Safety, para desenvolvimento de sistemas de segurança de veículos;
- IEC-61508-3 Functional safety of electrical/electronic/programmable electronic safety-related systems - Part 3: Software requirements, para desenvolvimento de sistemas de segurança;
- CMMI level 2 e acima, para desenvolvimento de *software* em geral;

2.8.6O Futuro das Ferramentas de Gerenciamento de Requisitos

Na opinião de Finkelstein e Emmerich (2000), os grandes pacotes de gerenciamento de requisitos podem evoluir a ponto de eliminarem todos os documentos de papel, que serão substituídos por um banco de dados e uma ferramenta de gerenciamento.

Por sua capacidade de registrar e armazenar todas as informações de um projeto, Finkelstein e Emmerich (2000) preveem que, no futuro, as ferramentas de gerenciamento de requisitos poderão evoluir até serem capazes de registrar todas as informações de uma empresa tornando-se a “corporate memory”.

2.9 Comentários Finais

Dificuldades com requisitos de *software* de grande porte levaram à "crise do *software*" que se caracterizava pela dificuldade de cumprir metas de custos e prazos e pela baixa qualidade dos produtos de *software* de grande porte. Dessa crise emergiram a Engenharia de *Software* e a Engenharia de Requisitos, sendo esta responsável por disciplinar as atividades relativas aos requisitos de *software*.

Devido ao fato que a Engenharia de Requisitos ser relativamente nova, ainda estar em evolução e ser usada de forma bastante diferente em diferentes locais e contextos, não há uma padronização de terminologia, conceitos, processos e modelos. Similarmente, também não há uma definição universal e um modelo de referência para a rastreabilidade de requisitos.

A rastreabilidade de requisitos facilita analisar o impacto e os custos de mudanças em requisitos e é a parte mais importante do gerenciamento de requisitos. No entanto, o rastreamento de requisitos só pode ser feito sem o auxílio de ferramentas para sistemas muito simples e com poucos requisitos. Ferramentas de gerenciamento de requisitos armazenam todas as informações relacionadas com os requisitos de um projeto e permitem fazer o rastreamento de requisitos para sistemas complexos.

O uso de uma ferramenta de gerenciamento de requisitos pode exigir mudanças na cultura da empresa em relação aos requisitos e seus processos e em relação ao controle de configuração. Uma ferramenta de gerenciamento de requisitos exige compromisso de uso, pois pode não ser possível recuperar o status do projeto se registros deixarem de ser entrados na ferramenta por um período.

Ferramentas de gerenciamento de requisitos não são uma panacéia ou "bala mágica", mas Wieger (2003, p. 378) afirma que "uma vez que você faça uma ferramenta trabalhar para você, você nunca mais vai voltar a usar papel".

3 FORMULAÇÃO DO PROBLEMA E ABORDAGENS PARA SUA SOLUÇÃO

3.1 Definição do Problema

Ferramentas de automatização de processos da Engenharia de Requisitos foram introduzidas em Engenharia de *Software* e em Engenharia de Sistemas na década de 1970.

Com o advento dos microcomputadores, no início dos anos 1990, muitas ferramentas computacionais, representadas por ferramentas EDA, CAD, CAE e CAM, foram introduzidas em atividades de subsistemas dos desenvolvimentos espaciais do INPE. No entanto, pouca evolução ocorreu nas atividades de nível de sistemas do INPE e as ferramentas atualmente empregadas são manuais e praticamente as mesmas daquelas usadas nos anos 1980.

Houve também pouca alteração nos processos e ferramentas usados no registro e armazenamento de informações. Esses processos são ainda quase totalmente manuais, tendo havido somente a substituição de máquinas de escrever, documentos em papel, pastas e arquivos, por processadores de texto, documentos eletrônicos, pastas e arquivos eletrônicos.

A escassez de mão de obra e a necessidade de reduzir riscos e custos exigem uma maior automatização dos processos da Engenharia de Sistemas do INPE.

3.2 Metodologia

Observando que apenas um exercício de aplicação de uma ferramenta seria insuficiente para entender as necessidades de uso e as causas do pouco uso de ferramentas de automatização da Engenharia de Sistemas no INPE, este trabalho pesquisou as origens da necessidade de atenção à qualidade dos requisitos e do uso de ferramentas de automatização do gerenciamento de requisitos. Para isso, foram pesquisadas referências relacionadas à

emergência e formalização da Engenharia de Sistemas, à emergência da Engenharia de Requisitos, e à introdução da Engenharia de Sistemas no INPE.

Portanto, este estudo foi realizado da seguinte forma:

a) Foi feito um estudo da origem e do estado atual dos tópicos da Engenharia de Requisitos relacionados com o gerenciamento de requisitos;

b) Foi realizado um estudo das origens e da formalização da Engenharia de Sistemas, e da introdução de ferramentas computacionais no gerenciamento de informações em Engenharia de Sistemas Espaciais, com o objetivo de melhor entender a necessidade de aplicação de ferramentas computacionais nos projetos espaciais do INPE;

c) Foi realizado um estudo das origens da Engenharia de Requisitos e do rastreamento de requisitos e da aplicação de ferramentas de gerenciamento de requisitos na Engenharia de Software, com o objetivo de melhor entender a importância da aplicação de ferramentas computacionais da Engenharia de Requisitos nos projetos espaciais do INPE;

d) Foi realizado um estudo da introdução e evolução da Engenharia de Sistemas Espaciais do INPE, para melhor entender o estado atual da Engenharia de Sistemas Espaciais no INPE e algumas das causas da não-aplicação de processos da Engenharia de Requisitos e de ferramentas de gerenciamento de requisitos nos programas espaciais do INPE;

e) Foi selecionada uma parte da documentação de um subsistema de satélite desenvolvido pelo INPE para, através de um exercício, demonstrar uma técnica de preparação de requisitos e a aplicação de uma ferramenta comercial de gerenciamento de requisitos.

O levantamento bibliográfico do estudo foi feito, em grande parte, por meio de pesquisas pela Internet. A Internet permitiu, em alguns casos, encontrar informações importantes em documentos originais do período da emergência

da Engenharia de Sistemas e da Engenharia de Requisitos. Referências recentes foram obtidas através de aquisições de livros.

Apesar da maior parte da documentação sobre o INPE estar localizada em sua biblioteca, alguns documentos interessantes sobre o INPE e sobre o Programa Espacial Brasileiro foram obtidos de outros locais pela Internet.

O exemplo para o exercício de uso da ferramenta de gerenciamento de requisitos é parte da documentação de um subsistema dos satélites CBERS 3 & 4 do Programa Sino-Brasileiro de Satélites de Recursos Terrestres (CBERS - China-Brazil Earth Resources Satellite). O Programa CBERS 3 & 4 foi escolhido por ser o maior e mais bem documentado programa de desenvolvimento de satélites do INPE e por ter grande parte de sua documentação em formato eletrônico, sendo, por esse motivo, o melhor exemplo disponível no INPE para um exercício de aplicação de uma ferramenta de automatização de gerenciamento de requisitos. Uma parte da documentação relativa ao Subsistema de Energia Elétrica (EPS - Electrical Power Subsystem) foi escolhida para o estudo por ser este subsistema de responsabilidade do INPE e, portanto, ter-se acesso a toda a sua documentação e ter-se competência técnica para interpretá-la. O uso adequado de ferramenta de gerenciamento de requisitos exige, além de conhecimentos dos seus recursos, o entendimento das informações sendo introduzidas. Esses conhecimentos são dominados pelo autor, pois este teve responsabilidades de gerência sobre este subsistema durante 18 anos, tendo sido também responsável pela primeira versão de muitos dos documentos usados no estudo.

Todos os documentos do programa CBERS 3 & 4 foram disponibilizados do arquivo do sistema SEGI (Sistema Eletrônico de Gerenciamento de Informações) da Coordenadoria Geral de Tecnologia e Engenharia Espaciais (ETE) do INPE. Os documentos não disponíveis nos arquivos do INPE foram disponibilizados pela empresa Aeroeletrônica – Indústria de Componentes Aviónicos S.A. sucedida pela AEL Sistemas Ltda. em 2011, fabricante dos equipamentos eletrônicos do subsistema.

A ferramenta de gerenciamento de requisitos utilizada no estudo é a Requirements Gateway, da National Instruments do Brasil. Foi usada uma instalação de demonstração da versão 1.1 (*build* 586) obtida por *download* diretamente do *site* da National Instruments Corporation. A instalação de demonstração da versão 1.1 da Requirements Gateway tem, por um período de 30 dias, todas as funcionalidades de uma ferramenta adquirida.

3.3 Limitações do Estudo

Este estudo não é um caso real, mas apenas um exercício sendo feito a *posteriori* sobre documentos pré-existentes. No entanto, é o mais próximo de um caso real, pois é baseado em documentos reais.

Muitos dos documentos analisados foram preparados e publicados em 2004 e fizeram parte da documentação de concorrência pública. Por esse motivo, tinham sido preparados com grande atenção e cuidado. No entanto, o fato de terem sido preparados sem preocupação com as características de qualidade dos requisitos, impede que sejam diretamente usados com uma ferramenta de automatização do gerenciamento de requisitos. Portanto, antes de introduzir os documentos na ferramenta, houve a necessidade de revisar e alterar manualmente seu texto. Os documentos foram minimamente alterados para permitir o uso da ferramenta pelo isolamento dos requisitos em requisitos individuais e pela inclusão de identificadores e referências de coberturas. Obviamente foram encontrados erros que não puderam ser corrigidos sem afetar grandemente os documentos.

Apesar de o estudo se limitar a apenas uma parte da documentação do satélite, seus resultados e conclusões são bastante gerais, pois os processos usados na preparação da documentação analisada foram os mesmos usados na preparação do restante da documentação, sendo grande a similaridade entre toda a documentação.

Procurou-se usar todos os documentos disponíveis relativos ao subsistema de energia elétrica, mas alguns documentos não puderam ser usados, seja porque somente existiam em formatos não-compatíveis com a ferramenta ou porque existiam somente em formatos que não permitiam alterações, e.g., documentos escaneados. Obviamente, não foram criadas informações não existentes na documentação; e, preferencialmente, foram usados documentos existentes no sistema SEGI, tendo sido raramente consultados arquivos particulares.

Como o Programa CBERS 3 & 4 não pratica rastreabilidade de requisitos, não há, na documentação, qualquer informação de rastreabilidade. Portanto, as informações de rastreabilidade entre documentos foram criadas pelo autor com base no seu conhecimento do subsistema estudado. Como o sistema SEGI somente disponibiliza a última versão dos documentos, não seria possível fazer rastreabilidade de mudanças entre diferentes versões de um mesmo documento. A maior parte dos documentos usados no estudo são documentos de especificação; apenas alguns documentos são documentos de cobertura, i.e., documentos que não contém requisitos, mas cobertura de requisitos.

Como não foi possível fazer um treinamento formal no uso da ferramenta de gerenciamento de requisitos, o seu aprendizado foi feito somente por tutoriais, exemplos e recursos de ajuda da própria ferramenta. Dessa forma, há recursos disponíveis que não puderam ser usados por serem desconhecidos. Há recursos da ferramenta que não foram usados por não haver na documentação informações que necessitem o uso desses recursos.

A ferramenta Requirements Gateway, assim como todas as ferramentas de gerenciamento de requisitos conhecidas, não é totalmente automática, isto é, as informações e relações de rastreabilidade entre informações devem ser manual e corretamente introduzidas na ferramenta.

3.4 Resultados Esperados

Através de um exercício de aplicação de uma ferramenta de gerenciamento de requisitos ao subsistema de energia elétrica, o estudo analisa o processo de Engenharia de Requisitos dos satélites CBERS 3 & 4, sugerindo mudanças que melhorem a qualidade dos requisitos e permitam o gerenciamento automatizado.

O estudo detecta a existência de deficiências no processo de preparação de requisitos do Programa CBERS 3 & 4 e também demonstra que, apesar de preparados com muita atenção, havia erros nos documentos.

O estudo demonstra as vantagens da introdução de processos da Engenharia de Requisitos na Engenharia de Sistemas Espaciais, em particular a melhoria da documentação e o uso de ferramentas de gerenciamento de requisitos.

O estudo pode contribuir para melhorar os processos da Engenharia de Sistemas do INPE e para uma futura decisão sobre processos e ferramentas de gerenciamento de requisitos a serem empregados em projetos de Engenharia Espacial do INPE.

4 A ENGENHARIA DE SISTEMAS

Alguns livros de Engenharia de Sistemas dedicam algumas de suas páginas iniciais para informar que a Engenharia de Sistemas é praticada de forma empírica há muitos milênios e que sua formalização somente ocorreu entre as décadas de 1950 e 1960. No entanto, nenhum livro dá detalhes das forças, eventos que agiram e de como ocorreu sua formalização. A formalização da Engenharia de Sistemas é um acontecimento tão recente que alguns dos seus protagonistas ainda estão vivos. No entanto, por sua formalização ter ocorrido durante programas secretos de defesa e no período da Guerra-Fria, sua história é pouco conhecida e é difícil encontrar informações sobre como ela ocorreu. Felizmente, recentemente surgiram publicações produzidas por historiadores que hoje têm acesso a arquivos que já foram secretos.

A Engenharia de Sistemas Espaciais é historicamente ligada à Engenharia de Sistemas, usa com rigor suas metodologias e foi a primeira a empregar seus conceitos e processos num contexto não-militar. Para entender as razões da necessidade de uso de metodologias rigorosas de gerenciamento em Engenharia de Sistemas Espaciais, este capítulo apresenta um resumo histórico dos acontecimentos que levaram à emergência e formalização da Engenharia de Sistemas e a sua introdução na Engenharia de Sistemas Espaciais. A intenção desse resumo obviamente não é contribuir para a história da Engenharia de Sistemas, mas auxiliar a entender a origem das forças evolutivas e a seqüência de eventos que causaram a sua emergência e formalização; e determinar quando surgiu a necessidade de automatização dos processos da Engenharia de Sistemas. Por não ser um trabalho com foco em história, datas e personagens somente serão informados quando forem considerados relevantes para o entendimento.

A visão apresentada neste estudo é similar àquelas que aparecem de forma muito breve em livros-texto de Engenharia de Sistemas, mas é baseada em publicações de historiadores que estudaram a evolução tecnológica ocorrida nos Estados Unidos no período aproximado entre 1940 e 1970, durante o

desenvolvimento de armas avançadas, e que diretamente influenciou a Engenharia de Sistemas Espaciais aplicada no INPE. Essa evolução tecnológica foi chamada em diferentes épocas e por diferentes autores com os nomes de “systems management”, “systems engineering management”, ou simplesmente “systems engineering”.

4.1 Origem da Engenharia

A humanidade cria conscientemente objetos e conhecimentos desde quando atingiu níveis de tecnologia e cultura suficientemente elevados. Qualquer criação consciente, tangível ou intangível, mesmo a mais tosca ferramenta ou arma, de madeira ou pedra, pode ser chamada de um projeto por ser um empreendimento organizado, temporário e com um objetivo definido (PMBOK, 1996, p. 4).

Escrevendo no século I AC, Vitruvius nos deixou um manual que foi praticamente a única referência no Ocidente sobre Arquitetura e Engenharia até o período da Renascença. Para Vitruvius, o arquiteto era integralmente responsável por um projeto e, o idealizava, contratava os construtores e operários, e administrava a construção em nome do dono da obra. Para Vitruvius, o arquiteto não precisava ser excelente nas artes do Desenho, Geometria, Óptica, Aritmética, Letras, Música, Pintura, Escultura, Medicina e outras ciências, mas tinha de ter bons conhecimentos nessas áreas para ser capaz de planejar, administrar e concluir um projeto (VITRUVIUS, 1960, pp. 5-13).

Até o século XVIII, os projetos de engenharia exigiam somente conhecimentos relativamente básicos da Engenharia Civil, Metalúrgica, Mecânica, Hidráulica e Naval que podiam ser dominados por apenas uma pessoa dotada de inteligência e habilidade. A baixa complexidade dos projetos de engenharia, independentemente das suas dimensões e da quantidade de trabalho necessário para sua conclusão, é evidenciada, entre outras coisas, pela baixa diversidade de seus componentes, i.e., blocos, colunas e vigas de metal, pedra, madeira ou outro material. As obras eram também executadas somente

com base em conhecimentos práticos, i.e., era sabido que construindo-se as estruturas com determinados materiais e de determinadas dimensões e proporções, estas não colapsariam sob o seu próprio peso. Vitruvius discute a diferença entre teoria e prática, mas para ele teoria estava relacionada com a adequação de estilos para cada tipo de obra e não com as dimensões das estruturas. Somente no século XVII, Galileu usou conceitos da Mecânica para explicar porque os pesos das estruturas aumentavam em proporções diferentes de suas dimensões, fato já conhecido empiricamente por Vitruvius (GALILEI, 1914, p. 2).

Os mecanismos criados pelo homem foram ficando mais complexos e a força, animal ou humana, usada para sua movimentação, foi substituída pela força de correntes de água e vento. O período das grandes navegações exigiu a reunião de grande quantidade de conhecimentos, como construção naval de grande porte, correntes oceânicas e de ventos, Meteorologia, Astronomia, Cartografia, Navegação, Artilharia Naval etc. Os navios de navegação oceânica do século XVI eram os mais complexos sistemas até então construídos, necessitando, para sua construção e operação, de grande quantidade de conhecimentos, planejamento, treinamento, suporte logístico e numerosa tripulação.

A introdução de máquinas a vapor na indústria levou à revolução industrial no século XVIII, provocando o aparecimento da grande indústria e um grande aumento na capacidade de produção e do número de habitantes das cidades, onde seriam introduzidas inovações em transporte público, iluminação a gás, entre outras. Essas inovações, produtos de maiores conhecimentos nas áreas de Engenharia Mecânica e Metalúrgica, no entanto, não necessitaram e não provocaram mudanças na maneira como os projetos eram administrados.

4.2 A Emergência da Engenharia de Sistemas

No transcorrer do século XIX, ocorreu um grande desenvolvimento das ciências naturais. Nas ciências físicas, o estudo dos fenômenos elétricos e magnéticos deu origem à nova disciplina da Engenharia Elétrica. A virada do século XIX para o século XX viu o surgimento de muitas novas tecnologias e inventos

relacionados com a Engenharia Elétrica, e.g., lâmpada incandescente, telégrafo, telefone, rádio, motores elétricos. Essa revolução é chamada por alguns de segunda revolução industrial; e, por outros, de revolução tecnológica. Essa revolução expandiu enormemente a quantidade de conhecimentos e, devido às limitações humanas, levou à especialização dos engenheiros.

A elevada demanda por produtos levou a inovações nos métodos de fabricação, deixando a produção de ser artesanal para ser em massa. A necessidade de aumentar a produção e reduzir os custos originou estudos científicos para racionalizar e aumentar a eficiência do trabalho, trazendo inovações na organização, divisão e administração do trabalho em ambiente industrial.

Muitos projetos de engenharia do início do século XX eram muito complexos e exigiam esforços simultâneos de muitas áreas de engenharia para sua execução. A maior complexidade dos projetos é evidenciada pela maior diversidade dos conhecimentos necessários para poder integrar as muitas partes do projeto na documentação e no produto final. Havia ainda:

- o risco de interferências entre componentes produzidos por diferentes fornecedores;
- a necessidade de decidir qual componente deveria ser alterado para eliminar essas interferências;
- a necessidade de sincronização das diferentes tarefas;
- a presença simultânea de diferentes equipes no sítio de construção,

que não eram de responsabilidade de qualquer dos fornecedores dos componentes do projeto.

Os grandes projetos do século XX exigiam uma quantidade de conhecimentos que uma pessoa somente não era capaz de deter, sendo necessária, para sua consecução, uma equipe responsável pelo projeto. Engenheiros com funções de sistemas surgiram naturalmente em indústrias trabalhando com produtos que exigiam a integração de partes muito distintas, como automóveis e aviões,

e em sistemas extensos, como redes de distribuição de energia elétrica e gás, redes de telégrafos e telefones, redes de transporte ferroviário. (SCHLAGER, 1956).

É geralmente aceito que o primeiro uso do termo “engenharia de sistemas” ocorreu no Bell Telephone Laboratory (Bell Labs) no início dos anos 1940. No entanto, há na literatura duas justificativas complementares para o aparecimento e uso desse termo no Bell Labs. Uma referência explica que os problemas associados com uma rede telefônica de dimensões nacionais apareceram no Bell Labs antes que sistemas complexos fossem comuns em outras indústrias (SCHLAGER, 1956). Outra referência explica que “engenharia de sistemas” era o nome dado à maneira formal como os cientistas do Bell Labs tinham que produzir os documentos contratados pela American Telephone and Telegraph (AT & T) para serem fornecidos para engenheiros e técnicos da Western Electrical, o braço técnico da AT & T (JOHNSON, 2002a, p. 49).

Ainda antes da Segunda Guerra Mundial, o Bell Labs se envolveu, juntamente com o MIT, a Caltech, a Carnegie-Mellon University, a Harvard University e outras instituições, através do National Defense Research Committee (NDRC), na solução do difícil problema de atingir alvos móveis, como navios ou aviões, com projéteis de canhões. Por usar tecnologia ainda emergente, esse problema exigiu, além dos conhecimentos em canhões e balística, já dominados pelos militares, o auxílio de cientistas e engenheiros de várias especializações. Esse problema consistia em determinar a distância, a velocidade e a trajetória dos alvos para, rapidamente, calcular a balística do projétil, movimentar os enormes canhões para um disparo na futura posição do alvo e, antes de disparar o projétil, ajustar seu tempo de detonação para que esta ocorresse nas proximidades do alvo depois de percorrer sua trajetória. Apesar de que servomecanismos já existiam há algum tempo, esse trabalho, conforme informado por Mindell (2000, p. 28), levou à criação de um complexo sistema de defesa antiaérea envolvendo, detecção, acompanhamento, comunicação, computação, operadores e servomecanismos, operando em uma malha fechada e ao desenvolvimento da teoria de servomecanismos. Esses

desenvolvimentos também levaram à constatação de que a comportamento do sistema completo, e.g., velocidade de resposta, estabilidade etc., era uma função complexa da resposta de seus componentes, e à criação da função do integrador de sistemas, responsável pelo sistema completo. Encontra-se também, em Mindell (2000, p. 50), uma citação de 1945 que é relacionada com a definição atual de sistemas nas engenharias:

One must always remember that a fire-control system is more than the sum of component parts. It is an integrated whole with interrelated functioning of all its parts and one is safe in considering parts separately only if one always keeps in mind their relation to the whole. (MINDELL, 2000, p. 50)

4.3 Aquisições pela Força Aérea dos Estados Unidos

Até o ano de 1947, a aviação militar dos Estados Unidos não era uma arma independente, mas parte do exército, primeiramente como Serviço Aéreo do Exército (Army Air Service), depois como Corpo Aéreo do Exército (Army Air Corps) e finalmente como Força Aérea do Exército (Army Air Force). Por ser parte do exército, a aviação não era vista como uma força armada inteiramente nova e com novas capacidades; e os aviões eram vistos pelo exército apenas como um novo tipo de armamento, assim como fuzis, canhões e tanques. Esse braço aéreo tinha inicialmente sua ação limitada apenas ao reconhecimento aéreo do campo de combate e das linhas do inimigo (BENSON, 1997). Essa visão que o exército tinha do avião fez com que a aviação dos Estados Unidos, até o final da Segunda Guerra Mundial, estivesse tecnologicamente atrasada em relação a seus aliados e adversários, sendo que a guerra foi por ele vencida por estar localizado longe do campo de batalha e pela sua muito maior capacidade de produção (BENSON, 1997, p. 19). Quando, em 1947, se tornou uma arma independente, a Força Aérea dos Estados Unidos (United States Air Force-USAF) conseguiu maior independência, mas, diferentemente da marinha e do exército, por não possuir capacidade própria de projetos (*design bureau*) e de produção de armamento (arsenais), a USAF teve de recorrer à indústria de aviação civil para seus projetos de aviões (BENSON, 1997, p. 21).

Devido à maior simplicidade na iniciação de contratos em situações que exigiam urgência, e à dificuldade de prever custos de desenvolvimentos em áreas que envolvessem tecnologias não demonstradas, muitos dos contratos da USAF para aquisição de um novo avião eram do tipo “Cost Plus Fixed Fee” (CPFF). Nesses contratos, o governo reembolsava a contratada por todos os custos e esta recebia um pagamento fixo pelo seu trabalho. Nesse procedimento, também conhecido na USAF como “Fly-Before-You-Buy”, um protótipo modelo-X (experimental), era desenvolvido, testado, voado e modificado repetidas vezes antes que um contrato de produção fosse assinado. Um contrato CPFF não exigia muito planejamento e documentação para iniciar um desenvolvimento, e, desde que os riscos eram somente do governo, não exigia esforço da contratada para reduzir custos e melhorar a qualidade e desempenho do produto (BENSON, 1997). É evidente que contratos CPFF tinham problemas com instabilidade de escopo, aumento de custos e prazos, e baixa efetividade do produto. À indústria aeronáutica interessavam muito mais os lucrativos contratos de produção. Um problema menos evidente do conceito “Fly-before-you-buy” é que, quando se contratava o desenvolvimento de um protótipo, concentrava-se em um resultado técnico imediato, e.g., a velocidade de vôo, esquecendo-se das necessidades futuras durante a vida de um grande lote de aviões em termos de suprimentos, instalações, treinamento, manutenção etc. Os resultados eram, muitas vezes, desastrosos. Muitos projetos eram suspensos porque havia uma dificuldade tecnológica insuperável, mas a maioria dos problemas era simplesmente devido à má administração. Quando o projeto chegava até o final, seus custos e prazos eram grandemente excedidos e o avião não atendia às necessidades. Quando o avião atendia as necessidades, os custos de produção ou manutenção eram tão elevados que o projeto não entrava em produção seriada. Outras vezes, o desenvolvimento era tão lento que a aeronave era obsoleta antes de entrar em produção. Os esforços para melhorar o processo de aquisição se refletiram na tentativa de uso de outros tipos de contratos, e.g., “Cost Plus Incentive Fee”, “Firm Fixed Price”, “Fixed Price Incentives”, “Bundle Bidding”, “Total Package Procurement Concept”, com sucesso muito variável.

A inexperiência da Força Aérea e a falta de metodologias e ferramentas para gerenciar desenvolvimentos tão complexos como aeronaves militares levavam, por exemplo, aos seguintes problemas:

- Dificuldades de comunicação com as contratadas e subcontratadas;
- Incompatibilidade entre os componentes produzidos por diferentes contratadas;
- Falta de controle e de atualização das modificações nos documentos;
- Má sincronização da prontidão de cada uma das partes sendo desenvolvidas.

4.4 A Ciência como Arma de Guerra

A primeira metade do século XX trouxe muitas inovações provocadas pela invenção da válvula termiônica, pelo desenvolvimento da tecnologia do rádio e da Engenharia de Eletrônica. Muitas dessas inovações originaram-se no desenvolvimento de armas e contramedidas militares, e.g., giroscópios, servomecanismos, radares, radionavegação, foguetes, energia nuclear, computadores.

A Segunda Guerra Mundial se caracterizou por ser uma guerra tecnológica principalmente pelo uso de máquinas complexas e pela introdução de tanques, aviões e foguetes. Atributos de superioridade de liderança anteriormente importantes, como tamanho dos exércitos, melhor posicionamento em combate, melhor treinamento e melhor uso das armas, foram sobrepujados por diferenças na capacidade tecnológica. As novas armas eram tão mais complexas que o conceito de arma evoluiu para o conceito de um sistema de armas (weapon system).

As operações militares da Segunda Guerra Mundial também ficaram muito complexas, fazendo com que, tal como havia ocorrido algumas décadas antes quando a ciência foi usada para racionalizar e aumentar a eficiência do trabalho na indústria, cientistas fossem contratados para estudar meios de aumentar a eficiência no uso das armas. Esses estudos levaram ao aparecimento de duas novas disciplinas: a Pesquisa Operacional, preocupada

com problemas táticos e o melhor uso das armas disponíveis, e a Análise de Sistemas, preocupada com problemas estratégicos e desenvolvimento de novas armas. Problemas militares, que durante milênios eram exclusivamente da jurisdição dos militares, passaram a ser também atividade de cientistas (KAHN; MANN, 1957, p. 1).

Para obter apoio científico, a Força Aérea criou o Army Air Force Scientific Advisory Board (AAF-SAB), formado por cientistas de altíssimo nível, e iniciou um projeto de pesquisa e desenvolvimento na Douglas Aircraft Company, chamado RAND (Research And Development). O projeto RAND se tornaria três anos mais tarde na RAND Corporation, uma organização "think tank", sem fins lucrativos que, empregando milhares de cientistas, realizava estudos em Pesquisa Operacional e Análise de Sistemas para a Força Aérea e outros órgãos do governo americano. Já em 1946, com vistas na possibilidade de lançamento de um satélite artificial da Terra, o Projeto RAND produziu para a Força Aérea o documento: Preliminary Design of an Experimental World Circling Spaceship. Esse documento, de quase 400 páginas e produzido em apenas duas semanas, analisa muitos dos problemas inerentes ao projeto de lançadores e de satélites (RAND, 1946).

Ao final da Segunda Guerra Mundial, membros do AAF-SAB visitaram instalações alemãs tomadas pelas forças americanas e produziram o monumental trabalho "Toward New Horizons", composto de 13 volumes e tratando das inúmeras questões relacionadas com o uso de aviões e foguetes com fins militares. Como exemplo do altíssimo nível dos membros do AAF-SAB, notam-se, entre os autores desses volumes, dois futuros laureados com prêmios Nobel e o estudante de doutorado que entrevistou o prisioneiro Wernher von Braun, mais tarde seria o líder do programa espacial chinês. O segundo volume desse trabalho, intitulado "Where We Stand", compara as capacidades técnicas americanas e alemãs na área de aviões a jato e foguetes e, reconhecendo a inferioridade americana, faz os seguintes comentários e recomendações sobre a maneira de desenvolver um projeto altamente tecnológico (VON KARMAN, 1946):

- Sobre a condição alemã:

It is important for us to note that one element in their success was the fact that they had under a single leadership in one organization, experts in aerodynamics, structural design, electronics, servomechanisms, gyros and control devices, and propulsion; in fact, every group required for the development of a complete missile. (VON KARMAN, 1946, p. 13).

- Sobre a condição americana:

The present facilities and organization for research and development of pilotless aircraft appear inadequate. It cannot be expected that such complex problems can be successfully solved by any group which is specialized in only one of the several fields which are involved.

Leadership in the development of these new weapons of the future can be assured only by uniting experts in aerodynamics, structural design, electronics, servomechanisms, gyros, control devices, propulsion, and warhead under one leadership, and providing them with facilities for laboratory and model shop production in their specialties and with facilities for field tests. Such a center must be adequately supported by the highest ranking military and civilian leaders and must be adequately financed, including the support of related work on special aspects of various problems at other laboratories and the support of special industrial developments. It seems to us that this is the lesson to be learned from the activities of the German Peenemünde group. (VON KARMAN, 1946, p.16).

Apesar desse alerta, o período imediatamente após a Segunda Guerra Mundial caracterizou-se por uma desmobilização militar geral e por redução de despesas com armamentos (BENSON, 1997, p. 19). Isso causou uma temporária redução na velocidade e no número de projetos de desenvolvimento de mísseis iniciada durante a guerra, mas que foi logo recuperada e grandemente aumentada com a demonstração da capacidade nuclear soviética em 1947 e a invasão da Coreia, que deram início ao período da Guerra-Fria.

4.5 O Desenvolvimento de Mísseis ICBM

No período do pós-guerra, os Estados Unidos se envolveram no desenvolvimento de vários tipos de mísseis derivados das bombas voadoras alemãs V1 e V2. Havia necessidade de mísseis de curto e médio alcance, para serem lançados de terra e de mar, e de mísseis de longo alcance. Houve grande disputa entre as diferentes Forças Armadas dos Estados Unidos para o desenvolvimento desses mísseis, mas decidiu-se que mísseis de médio alcance para serem lançados do mar seriam de responsabilidade da Marinha; mísseis lançados de terra com alcance até 400 km seriam de responsabilidade do Exército, e mísseis de médio e longo alcance lançados de terra seriam de responsabilidade da Força Aérea (KOHN, 1993, p. 47, nota 53).

O período que se seguiu à demonstração da capacidade nuclear da União Soviética foi caracterizado por uma corrida entre Estados Unidos e União Soviética para obter armas nucleares mais potentes e meios de empregá-las contra o inimigo, além de contramedidas contra ataques com armas nucleares. O desenvolvimento de armas e contramedidas eficazes exigiu um grande esforço de pesquisa e desenvolvimento por parte dos Estados Unidos, envolvendo centenas de milhares de pessoas, milhares de empresas e gastos anuais de dezenas de bilhões de dólares.

Dentro da Força Aérea dos Estados Unidos havia duas correntes de pensamento quanto ao melhor veículo de lançamento para as bombas nucleares: a corrente mais tradicionalista (The Air Force of the Present), defendida por pilotos, baseava-se no uso de enormes aviões bombardeiros de longo alcance; e a corrente mais avançada (The Air force of the Future), defendida por militares-engenheiros, baseava-se no uso de foguetes de longo alcance. Entre uma escolha e outra, a Força Aérea ficou com as duas. Como as bombas nucleares então existentes eram imensamente grandes, impedindo que fossem transportadas adequadamente pelos bombardeiros disponíveis, um enorme avião bombardeiro de longo alcance começou a ser desenvolvido. E, mesmo que a enorme massa das bombas nucleares impossibilitasse seu

lançamento por foguetes em longa distância, a Força Aérea contratou o desenvolvimento de um grande foguete, chamado Atlas, que, possivelmente no futuro, pudesse ser utilizado para transportar uma bomba nuclear. No entanto, os resultados de estudos do Teapot Committee, liderado pelo aclamado matemático John von Neumann, demonstrando que, por volta de 1960, a massa das bombas nucleares seria reduzida para menos de 750 kg, levaram à viabilidade do desenvolvimento de um foguete com capacidade intercontinental armado de ogivas nucleares, o Míssil Balístico Intercontinental (Intercontinental Ballistic Missile- ICBM) (NEUFELD, 1990, p. 98).

4.5.1 Uma Abordagem de Sistemas no Desenvolvimento de Mísseis

Um míssil balístico autoguiado era extremamente mais complexo que um avião, pois tinha muitos subsistemas não existentes num avião, tais como: navegação e controle de voo automáticos, computação, telemetria, além de muitos novos problemas como, por exemplo, controlar seu motor foguete extremamente poderoso e seus propelentes exóticos, aerodinâmica de vôo e reentrada atmosférica em velocidade hipersônica, precisão de trajetória, etc.

O projeto de um míssil era também muito mais sensível a mudanças nos subsistemas, o número de variáveis de projeto era enorme e as margens de projeto eram mínimas, a ponto de exigir um minucioso balanço de empuxo, massa, propelente, energia elétrica, tolerâncias e desvios.

Foi percebido que, devido a interferências entre os componentes no projeto final, a otimização de uma característica importante de um componente do míssil não levava à otimização do míssil. O grande número de subsistemas essenciais impedia voar o protótipo com somente parte dos subsistemas.

Devido ao grande número de contratadas, as frequentes modificações de projeto precisavam ser controladas com muito rigor para se conhecer o estado atual da configuração do protótipo e, como um míssil voa somente uma vez, sendo destruído durante o voo, não pode ser desenvolvido pela abordagem “fly-before-you-buy” e precisa, desde o início, de uma linha de produção de

protótipos. Adicionalmente, um mínimo erro de projeto, ou pequeno desvio de fabricação, ou uma modificação mal comunicada, poderia causar a destruição do protótipo, e às vezes também da plataforma de lançamento, sem que se obtivesse qualquer resultado do teste. Trabalhar com uma enorme quantidade de propelente explosivo e uma carga útil nuclear exigia um nível de segurança muito mais rígido do que o usado com aviões e explosivos químicos. Um rigoroso e eficiente processo de controle de configuração teve de ser criado e empregado na produção dos protótipos para:

- Evitar que engenheiros fizessem modificações sem considerar seu efeito em outras funções;
- Determinar a exata configuração do *hardware* em qualquer momento;
- Determinar e registrar modificações feitas em campo;
- Atualizar manuais;
- Atualizar peças sobressalentes;
- Conhecer a configuração real para permitir diagnóstico da causa de falhas;
- Diminuir incompatibilidades entre componentes do sistema;
- Adaptar o inventário de peças sobressalentes às necessidades da configuração atual.

E, como o desenvolvimento era realizado em ritmo de corrida contra o adversário, todas as partes do míssil e as instalações de suporte (fabricação, testes, lançamento, etc.) tinham de ser desenvolvidas e produzidas em paralelo e sincronizadamente. Como o ICBM precisaria de suporte, manutenção e inspeções durante toda sua vida operacional para assegurar sua prontidão para lançamento, um programa de defesa baseado em mísseis também precisava ter uma visão de longo prazo. O projeto de um míssil exigia uma equipe multidisciplinar de projeto, um grupo de gerenciamento técnico com capacidade multidisciplinar e uma abordagem integral de longo prazo (HALLMAN, 1962, p. 182).

Simon Ramo e Dean Wooldridge, dois engenheiros e cientistas que, depois de obterem Ph.D.s em Física trabalharam, no Bell Labs, na General Electric, e na Hughes Electronics, onde juntos trabalharam no desenvolvimento do AIM-4 Falcon, o primeiro míssil guiado ar-ar, fundaram a RW (Ramo-Wooldridge), que mais tarde se juntou à Thompson Products, formando a TRW, a empresa que, sem dúvida, teve maior influência na introdução da Engenharia de Sistemas no desenvolvimento dos ICBMs. Infelizmente, não foi possível encontrar o “Anderson Report” no qual algumas referências citam ter sido sugerida uma abordagem de sistemas ao desenvolvimento do ICBM e ter sido definido um ciclo de vida dividido nas fases de formulação de conceito, definição de contrato, aquisição e operação (MORRIS, 1997, p. 48).

Devido à necessidade de capacidade de produção em grande escala, esses desenvolvimentos foram contratados a grandes corporações ligadas à indústria automobilística e aeronáutica (Ford, Chrysler, Boeing, General Electric, Douglas, McDonnell, Firestone, Martin etc). A empresa Space Technology Laboratories (STL), uma divisão da TRW, tinha a responsabilidade de suporte em ciência e tecnologia e de integradora de engenharia de sistemas. A posição privilegiada do STL foi questionada pelas outras empresas, caso a TRW viesse a concorrer para um contrato de produção de mísseis. Isto forçou que o STL fosse transformado em nova organização "think tank", The Aerospace Corporation. Essa perda não prejudicou a TRW, pois lhe permitiu participar dos muito mais lucrativos contratos de produção.

4.6 O Impacto do Lançamento do Sputnik

Entre fins do século XIX e meados do século XX, a possibilidade de lançamento em órbita de um satélite artificial já havia sido várias vezes considerada, tanto por autores de ficção quanto por cientistas. (SATELLITE, 2012).

Em 29 de julho de 1955, o presidente dos Estados Unidos anunciou sua intenção de lançar um satélite artificial da Terra durante o Ano Geofísico Internacional definido como o período entre 1^o de julho de 1957 e 31 de

dezembro de 1958. Entre os muitos órgãos americanos capazes de desenvolver e lançar um satélite, a escolha do projeto Vanguard, do Naval Research Laboratory (NRL), foi determinada mais por razões de estratégia política do que por razões científicas. O Vanguard, baseado em um pequeno foguete de sondagens, vinha sob os auspícios da National Science Foundation (NSF) - um órgão civil – e, ao mesmo tempo em que demonstrava que os Estados Unidos tinham intenção de fazer uso científico e pacífico do espaço, em consonância com sua política de “open skies” - uma proposta americana de tratado de observação aérea mútua - escondia sua já considerável capacidade militar espacial.

O impacto causado nos Estados Unidos pelo lançamento do Sputnik pela União Soviética, e pela falha do lançamento do Vanguard, apenas um mês depois, transmitida ao vivo pela televisão, foram tão sentidos no prestígio nacional quanto na sua fragilidade como alvo de uma arma para a qual não tinham contramedidas, pois as que estavam sendo desenvolvidas pressupunham bombas nucleares transportadas por aviões. O foguete soviético capaz de colocar em órbita um satélite com a massa do Sputnik poderia colocar uma bomba nuclear em qualquer parte dos Estados Unidos em cerca de 30 minutos. A capacidade de equilibrar as forças com uma retaliação contra um ataque soviético ainda não existia, pois, a despeito dos enormes gastos durante mais de uma década, os foguetes Atlas e Titan - um projeto redundante no caso de insucesso nas soluções empregadas no Atlas - apresentavam problemas de confiabilidade e ainda não estavam operacionais. A liderança da União Soviética na capacidade de lançar mísseis balísticos intercontinentais (ICBM - Intercontinental Ballistic Missile), explicada pela simplicidade de seus foguetes em comparação com os americanos, conhecida como "Missile Gap", causou grande distúrbio militar e político, mas provocou uma aceleração ainda maior nos esforços dos Estados Unidos para aperfeiçoar seus foguetes Atlas e Titan.

4.7 Os Documentos das Séries AFSCM 375 e AFR 375

Infelizmente não foi possível rastrear em detalhes como ocorreu a formalização dos processos de engenharia de sistemas durante o desenvolvimento dos ICBMs. Provavelmente, os documentos que formalizaram os procedimentos usados pela USAF na aquisição de sistemas complexos se basearam em documentos *best-practices* e *lessons learned*, preexistentes nas muitas contratadas e na própria USAF, que foram consolidados pelo sistema de aquisição (GRADY, 2006, p. 13). Uma possível razão de não se conseguir rastrear os detalhes da consolidação desses documentos é que os documentos que lhes serviram de referência eram secretos ou de propriedade das contratadas. Johnson (2002b, p. 198) informa que o trabalho de consolidar os procedimentos foi realizado por iniciativa de um oficial da USAF. O fato é que, a partir de 1960, depois de um grande esforço para uniformizar os processos, métodos de gerenciamento e documentação de projetos, a USAF emitiu os documentos conhecidos como série 375. Houve de fato duas séries com o mesmo número: primeiramente uma série somente do Comando de Sistemas da USAF, o Air Force Systems Command Manual (AFSCM-375) e depois uma série de regulamentos de toda a USAF, a Air Force Regulations (AFR-375).

As referências consultadas não são precisas ao informar os títulos e datas de emissão dos documentos das séries 375. Conforme Johnson (2002a, p. 246), os primeiros três documentos da série AFSCM foram emitidos em 31 de agosto de 1960; e a série revista e expandida consistia dos seguintes documentos:

- AFSCM 375-1 Configuration Management during Definition and Acquisition Phases;
- AFSCM 375-3 System Program Office Manual;
- AFSCM 375-4 System Program Management Manual;
- AFSCM 375-5 Systems Engineering Management Procedures;
- AFSCM 375-6 Development Engineering;
- AFSCM 310-1 Management of Contractor Data and Report.

Apesar da numeração estranha, o documento AFSCM 375-4 System Program Management é o de hierarquia mais alta na série AFSCM. Ele estabelecia procedimentos e políticas a serem seguidas no estabelecimento dos requisitos, e para o projeto, desenvolvimento, testes, operação e manutenção de futuros sistemas. Seu propósito era servir como uma ferramenta de gerenciamento que fornecesse grande visibilidade ao cliente e ao gerente de engenharia de sistemas (MORRIS, 1997, p. 48).

Provavelmente, é nessa série que ocorre a introdução do importante conceito de "system life-cycle", pois esse termo não é encontrado nos livros de Engenharia de Sistemas contemporâneos, como Hall (1962). A USAF foi a primeira organização a publicar um documento abrangente, descrevendo em detalhes os processos da Engenharia de Sistemas (BRILL, 1998). O AFSCM 375-5 prescrevia um ciclo de vida do sistema dividido nas fases Conceitual, Definição, Aquisição e Operacional; definindo "baselines" para: requisitos de programa, requisitos de projeto e configuração de produto (AFSCM 375-5, 1966).

Estranhamente, considerando a sua importância, apenas o AFSCM 375-5 Systems Engineering Management Procedures foi encontrado para consulta no website do DTIC (Defense Technical Information Center), depois de uma demorada pesquisa na Internet, em uma versão eletrônica escaneada de uma fotocópia pouco legível (AFSC, 1966). Uma tentativa de conseguir mais informações sobre as séries através de e-mails enviados a um professor da Defense Acquisition University e a um consultor que trabalhou na NASA nos anos 1960/1970 e que é membro fundador da INCOSE, demonstrou que, possivelmente, os documentos da série AFSCM 375 sejam pouco conhecidos atualmente, mesmo por profissionais da Engenharia de Sistemas.

O gerenciamento de configuração, nascido da experiência da Boeing e trazido para o desenvolvimento do ICBM, cristalizou-se no documento AFSCM 375-1 Configuration Management during Definition and Aquisition Phases (JOHNSON, 2002b, p. 96; 2001, p. 694). Esse documento é o precursor de

uma longa série de normas de gerenciamento de configuração, conforme mostrado na Figura 4.1. (ESA, 2000)

4.7.1 A Atenção aos Requisitos no Documento AFSCM 375-5

Foi feita uma pesquisa no AFSCM 375-5, com a intenção de encontrar referências sobre qualidade de requisitos e sobre rastreamento de requisitos.

Foram encontradas as seguintes referências a requisitos no referido documento:

These requirements shall sufficiently define engineering information, utilizing numerical values with associated tolerances, to provide criteria for the detail design, development, and test of the contract end item (AFSC, 1966, p. 24)...

In translating functions into design requirements, put emphasis on stating requirements in quantified or well-qualified terms (AFSC, 1966, p. 31)...

The initial design requirements will, where feasible, be in terms of (1) the purpose of the function; (2) the parameter of design (i.e., input and output values and allowable tolerances); (3) requirements which constrain design such as frequency allocation, power, physical, interface, time, environment, use of standard parts; and (4) requirements for system effectiveness, reliability, human performance, safety, security, maintainability, and transportability (AFSC, 1966, p. 31)...

The design requirements will be quantified on well qualified terms expressing the (1) purpose of the function; (2) the parameters of design (e.g., input and output performance values and allowable tolerance) (3) requirements which constrain design such as power, physical, interface, environment; and (4) requirements for effectiveness, reliability, human performance, safety, operability, maintainability, transportability, survivability/vulnerability (AFSC, 1966, p. 33).

E, foram encontradas as seguintes referências ao rastreamento de requisitos:

3.3 Comparison Matrix of Design Approaches.

Prepare a matrix to compare the design characteristics for each design approach to determine the degree to which the design approaches satisfy the functional and technical design requirements. The objective is to facilitate rapid comparison and evaluation of potential design approaches and to allow preliminary screening out of those design approaches that are inconsistent with the functional and technical design requirements (AFSC, 1966, p. 94)...

Schematic diagrams shall have the following characteristics in common (the diagrams and significant elements within the diagrams shall be uniquely identified to provide the basis):

a) Generating a family of lower level of detail diagrams traceable from top down or from the bottom up (AFSC, 1966, p. 95)...

Hardware designators established in first- and second-level detail schematics shall be used against the logic elements to depict interfaces with facilities and equipment and maintain a traceable relationship to the other schematic diagrams (AFSC, 1966, p. 96).

4.7.2 Críticas à Série AFSCM 375

Havia problemas com a série AFSCM 375, pois ela regia de forma extremamente rigorosa, minuciosa e ditatorial todos os detalhes de gerenciamento de sistemas e obrigava um enorme esforço burocrático de documentação por parte das contratadas. Por esse motivo, ela não era popular na indústria (MORRIS, 1997, p. 48). Os documentos da série AFSCM-375 eram também enormes: Grady (2006, p. 13) informa que o AFSCM-375-5 tinha 310 páginas e Morrison (1969, p. 87) diz que o AFSCM-310-1 tinha 1200 páginas.

De acordo com as referências, a opinião da indústria sobre a série AFSCM 375 é sempre desfavorável e crítica. Morrison (1969, p. 80) informa que as indústrias questionaram a ingerência da USAF nos seus procedimentos internos, a grande centralização do poder e a dificuldade de se determinar a conformidade com a série. No entanto, a opinião dos militares e do departamento de defesa sobre a posição das indústrias era outra, pois Johnson (2002a, p. 71), citando outro autor:

“If they wanted the job, they had to conform”. (JOHNSON, 2002a, p. 71)

Os próprios dirigentes da USAF, responsáveis pela criação da série AFSCM 375, viriam a criticar a burocratização e rigidez que ocorreu com a institucionalização dos procedimentos anos após a sua emissão, provavelmente quando as normas foram aplicadas mecanicamente e sem entendê-las, pela geração seguinte àquela que as produziu, conforme pode ser visto pelo trecho de entrevista abaixo (KOHN, 1993, p. 65):

But as to management procedures, there is a very interesting historical paradox here. The BMD (Ballistic Missile Division) developed just as we've said; it made leaps and bounds advances in management techniques of weapon systems acquisition. However, after the fact, people attempted to document all of that, to put it down into procedural volume, and that grew into a thing called the -375 series of manuals. Then the Systems Command Headquarters, of course, had to institutionalize that. So they expanded upon it and staffed it ad nauseam, and expanded it in great detail. Then the Air Staff picked up on it and further institutionalized and expanded it in a series of Air Force -375 regulations. Then the Department of Defense thought it was so good they further embellished it and institutionalized it into a whole series of DOD directives regarding how one might manage a program.

The first thing you know, it just collapsed of its own weight. In other words, people tried to grasp the good essentials of how programs ought to be managed, and they overkilled it! The bureaucracy overkilled it. In fact, it became then the very bane of the process and had to be killed because all the flexibility had been taken out. So much effort was out to capture the good techniques that had been developed that they then became regulatory and there was no flexibility for the program manager to adapt. (General Marsh)...

I can understand that. The -375 series regulations – you remember – you were at Wright Field and I put out directives that any program director could come in and get any waivers that he wanted from the -375 regulations. That could have worked with ARDC, but what happened is just as you say. A good thing was picked up by the Air Staff, picked up by DOD, and the first thing you know they were managing us without our having any waivers. Unfortunately, I think the way people operate and the way systems operate, they institutionalize almost anything that come along. I have felt that you are luck if any real good management approach in the bureaucracy lasts for five years. You are exceptionally lucky if it lasts for seven then you ought to throw it all away and start all over again. (General Schriever)...

Essa afirmação do General Schriever é coerente com seu prefácio à AFSCM 375-5:

The 375 series of AFSC manuals was designed to cover foreseen management needs of systems programs. Consequently, no program will find all provisions of all manuals necessary to its particular needs. These manuals are to be employed selectively only to the extent that they serve the needs of individual programs. It is the responsibility of each program director to determine when departures are warranted and request necessary waivers under the provisions of AFSCR 375-2. (AFSC, 1966, p. ii):

Escrevendo de uma perspectiva histórica mais distante, Grady (2010, p. 40), que trabalhou em projetos de defesa na década de 1960 e é da geração que usou a série AFSCM- 375, justifica a dificuldade com a burocracia nos anos 1960 pela falta de ferramentas computacionais:

The 375 series attempted to capture information for which a computer was really needed but we humans had not yet figured out how to take full advantage of in association with this work. In the early 1960s, when the 375 series was published, the computer apparatus was not available. The principal mainframe computer input/output device was stacks of punched cards. There was no easy way for a team of people to interact with the contents of the mainframe through terminals. Early attempts to mate the process to distributed computing were also flawed requiring the people to deal with precisely the same forms on the computer screen that were used in the paper implementation. (GRADY, 2010, p. 40)

O Departamento de Defesa, dirigido por Robert McNamara, que tinha trabalhado com Pesquisa Operacional durante a Segunda Guerra Mundial e era um defensor dos métodos da Análise de Sistemas, publicou em 1969 a norma temporária Mil-Std-499 Systems Engineering Management, muito menos rígida e detalhada que os manuais e regulamentos da série AFSCM 375. Apesar da diferença entre os títulos Systems Management, Systems Engineering Management e Systems Engineering, a Mil-Std-499 é considerada a primeira norma geral de Engenharia de Sistemas já emitida. A Mil-Std-499 foi substituída em 1974 pela versão a Mil-Std-499A, e sua versão seguinte, com a denominação Mil-Std-499B Systems Engineering, foi emitida em 1994 apenas como “draft”, e nunca foi aprovada. A Mil-Std-499A foi cancelada em 1995, numa reforma do sistema de aquisição promovida pelo Secretário de Defesa, William Perry. A proibição de uso de normas militares em aquisições pela Defesa, para reduzir gastos depois do fim da Guerra Fria, e a inexistência de normas equivalentes, levaram ao surgimento das normas comerciais ANSI/EIA 632 (1994) e IEEE 1220 (1994), baseadas na Mil-Std-499B. Em seguida, houve uma grande proliferação de diferentes normas comerciais, fazendo com que a organização que pela primeira vez codificou a disciplina de Engenharia de

Sistemas se viu sem um documento próprio (REDSHAW, 2010). A Figura 4.2 apresenta a genealogia das normas da Engenharia de Sistemas:

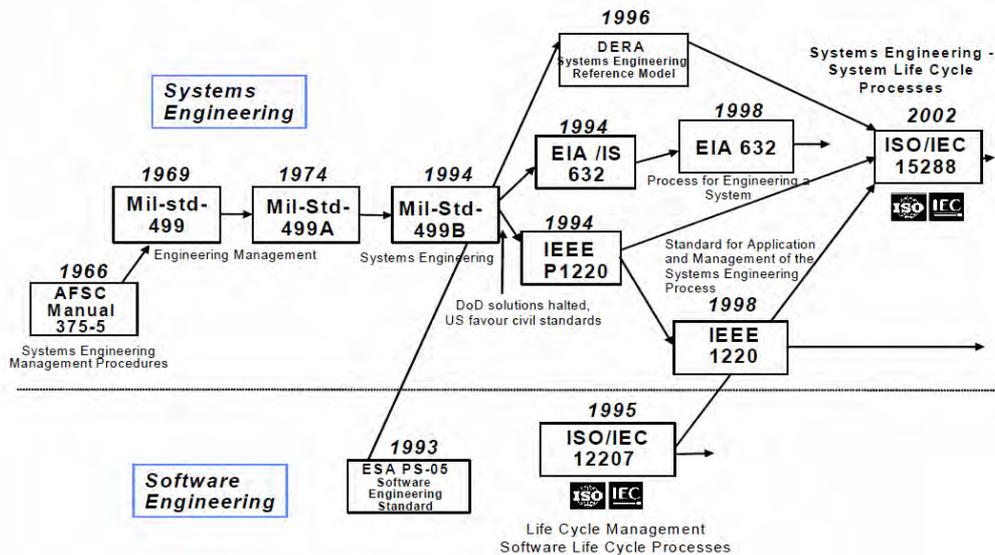


Figura 4.2 - Genealogia das Normas da Engenharia de Sistemas.
Fonte: Arnold (2008).

4.8A Introdução da Engenharia de Sistemas na NASA

A NASA (National Aeronautics and Space Administration) foi criada em 29 de julho de 1958 em substituição à NACA (National Advisory Committee on Aeronautics), seis meses após o lançamento do Explorer, o primeiro satélite lançado com sucesso pelos Estados Unidos.

As primeiras missões tripuladas da NASA, as cápsulas Mercury e Gemini, usaram para seu lançamento os foguetes desenvolvidos como mísseis balísticos pela USAF e pelo exército. E, apesar de terem sido um sucesso, tiveram seu custo da ordem de dez vezes o inicialmente estimado. A NASA, apesar de ter competência técnica demonstrada e reconhecida, ainda não tinha experiência em gerenciamento de grandes projetos. E, no seu próximo projeto que já estava em andamento, o Apollo, além de ter que desenvolver um novo e enorme foguete, também precisaria desenvolver todas as partes do conjunto Apollo, para a viagem a Lua.

Em meados de 1963, ainda durante o programa Gemini, muitas das atividades do programa Apollo já estavam atrasadas, e Holmes, o diretor de missões tripuladas, achava que somente com mais recursos poderia conseguir atingir a meta proposta pelo presidente Kennedy. No entanto, James Webb, o administrador da NASA, não queria enfrentar outra vez o congresso para pedir mais recursos. Subvertendo a hierarquia, Holmes recorre diretamente a Kennedy, mas este, apoiando o seu administrador provoca a demissão de Holmes. George Mueller, que tinha trabalhado no programa de ICBMs da USAF pelo Space Technology Laboratory, é então designado diretor de missões tripuladas da NASA em substituição a Holmes, em 1963, ainda durante o programa Gemini.

Mueller, que estava acostumado com as rigorosas metodologias de gerenciamento da Engenharia de Sistemas empregadas na USAF, consegue autorização de James Webb para uma reestruturação da NASA; e consegue que o General Bernard Schriever, diretor do programa de desenvolvimento de mísseis da USAF, autorizasse a transferência do General Samuel Phillips, líder do desenvolvimento do míssil ICBM Minuteman, para o cargo de Diretor do Programa Apollo. O General Samuel Phillips ainda consegue a transferência para a NASA de mais de 50 militares da USAF já treinados nas metodologias da Engenharia de Sistemas.

A introdução de processos da Engenharia de Sistemas na NASA seria questionada, principalmente por Wernher von Braun e sua equipe, que criticaram a grande centralização das decisões e o rigoroso gerenciamento da configuração (JOHNSON, 2001, p. 701). Mas a posição do General Phillips prevaleceu sobre a posição de von Braun. Provavelmente, nesse período, a experiência da USAF no desenvolvimento de sistemas de foguetes era igual ou superior à da equipe de von Braun, que até então tinha trabalhado para o exército.

A semelhança dos conceitos e dos problemas técnicos do desenvolvimento de mísseis e do programa espacial civil fez com que as práticas da Engenharia de Sistemas fossem tomadas e aplicadas com vigor pela NASA onde, no pico do

desenvolvimento do programa Apollo, havia quase 500.000 pessoas e 20.000 subcontratadas.

O curtíssimo cronograma da Apollo fez com que decisões arriscadas e corajosas, mas com muito sentido prático, tivessem que ser tomadas por Mueller, como por exemplo, testar desde o início todo o sistema integrado do Saturno V e Apollo (all-up testing), ao contrário de uma abordagem de integrações progressivamente mais completas, ao estilo da equipe de von Braun. No entanto, essa abordagem permitiu reduzir grandemente os custos e os prazos, por diminuir o número de diferentes configurações de *hardware* e por avançar muito mais rapidamente para a configuração final.

Hoban e Lawbaugh (1993. p. i) informam, no entanto, que mesmo que processos da Engenharia de Sistemas tenham sido usados no QG e nos vários centros da NASA durante o programa Apollo, não havia uma direção, padrão e um léxico comum para toda a NASA em Engenharia de Sistemas. Em alguns locais a aplicação dos processos da Engenharia de Sistemas era formal, mas em outras era um “back-of-envelope application”. Isso fez com que, após o final do programa Apollo, os processos da Engenharia de Sistemas fossem aplicados com menor rigor e entusiasmo na NASA.

A formalização dos processos da Engenharia de Sistemas na NASA somente aconteceu em 1995 com a publicação da primeira edição do SP-601S NASA Systems Engineering Handbook. Essa publicação foi uma consequência da recomendação de reforçar a Engenharia de Sistemas na NASA proposta pela comissão da National Academy of Public Administration, comandada pelo ex-líder do projeto Apollo, o General Samuel Philips, comissionada para analisar o acidente do Space Shuttle Challenger (HOBAN; LAWBAUGH, 1993, p. i).

4.8.1 A Automatização dos Processos da Engenharia de Sistemas na NASA

É surpreendente notar o pioneirismo da NASA no tratamento dado aos requisitos e ao uso de ferramentas computacionais de automatização de

gerenciamento de informações e requisitos, como pode ser observado nos trechos abaixo, de um documento da NASA de 1967:

As programs become larger in scope and more complex, the need for innovations and discipline in the information sensing, assimilation, processing and dissemination becomes greater.....

A number of Management system improvements are currently being investigated. Improved Data Management techniques are being pursued to streamline and reduce costs of program documentation. Configuration Management and Logistics Management are typical areas where improved data gathering and processing is particularly needed for effective management control....

The mass of information and data generated on programs such as Apollo is fast becoming program critical. More and more, it is becoming necessary to rigorously control information. The only answer to such problems is to turn to automated techniques for data handling and display...

Apollo management has developed many information search and retrieval systems, the most sophisticated of which have been: the Apollo Documentation Management Information System (ADMIS), the Requirements Information System (REINS), the UNIFILE and the Advanced Information Systems (ADVISE). (NASA, 1967, p. 5-1 e p. 5-2)

E, no Apêndice G do mesmo documento:

The Automated Data Management Information System (ADMIS) is a system which is the heart of formal documentation control on Apollo. The principal tools of the System are designed to define the information required, justify its need, acquire minimum data defined, and control data after it is generated. Other Data Management tools are used to indicate progress in the development of data, provide information for distribution of data, establish initial distribution requirements, permit organizations or individuals to order documents they need, and make use of previously prepared applicable DRD's. (NASA, 1967)

4.9 Frustração da Aplicação da Engenharia de Sistemas em Problemas Sociais

A Engenharia de Sistemas esteve em grande evidência durante os anos 1960. Essa evidência é demonstrada pelo fato que, enquanto no ano 1964 não havia

qualquer referência no *Engineering Index* sobre “systems engineering”, em 1969 já havia oito páginas (HUGHES; HUGHES, 2000, p. 1).

O sucesso da Engenharia de Sistemas numa tarefa tão complexa como uma missão tripulada à Lua, fez com que a Engenharia de Sistemas fosse proposta para ser usada em vários países na solução de problemas sociais que haviam resistido a outros métodos de organização, como projetos educacionais, habitacionais, de saúde pública, redução da pobreza e desemprego. Essa fé na ciência é capturada pela frase de Hubert Humphrey, o vice-presidente dos Estados Unidos, de 1968:

The techniques that are going to put a man on the Moon are going to be exactly the techniques that we are going to need to clean up our cities: the management techniques that are involved, the coordination of government and business, of scientist and engineer... (HUGHES; HUGHES, 2000, p. 16)

A fé na ciência também causou a ascensão da classe de administradores cientificamente treinados que, um pouco mais tarde, foi chamada, um tanto pejorativamente, de tecnocratas. Esse momento da história é percebido pelo grande número de publicações nas décadas de 1960/1970 relacionando a Engenharia de Sistemas a projetos educacionais e, como será visto no capítulo 6, muito afetou as atividades do INPE no final da década de 1960 e início da década de 1970, principalmente o Projeto SACI, como pode ser visto no trecho abaixo:

There is increasing interest and promise in applying space management and system development methods (systems approach) to social and economic problems. The former governor of California, for example, has called for systems approach studies and action to confront California's problems of air pollution, crime, traffic and unemployment. Greece contracted with Litton Industries for a system approach to help Greece achieve dramatic economic development goals. In the coming United Nations Conference (Vienna 14-27 August 1968) one session is devoted exclusively to non space applications of space technology. Among the 18 presentations one will describe "Applications of Space Management Techniques to Non-space Problems" and several will discuss government, university and industry roles in technology utilization programs. (CNAE, 1969, p. 17)

Porém, o insucesso da aplicação da Engenharia de Sistemas em projetos sociais, o envolvimento de alguns de seus defensores na guerra do Vietnam, e o movimento hippie e de contracultura fizeram com que se frustrasse o envolvimento da Engenharia de Sistemas no que Hughes e Hughes (2000, p. 1 e p. 24) chamam, referindo-se a sistemas sociais, de "*messy complexity*". Isso fez com que, durante várias décadas, as aplicações da Engenharia de Sistemas ficassem restritas aos grandes e complexos projetos militares e espaciais.

4.10 Comentários Finais

As descobertas científicas do século XIX provocaram um grande crescimento tecnológico na primeira metade do século XX e causaram o surgimento de muitas novas disciplinas da Engenharia. A inexperiência no gerenciamento de projetos complexos que envolviam muitas novas tecnologias e disciplinas no desenvolvimento de armas avançadas era marcado por muitos insucessos. A necessidade de conhecimentos avançados levou à aproximação de cientistas e militares.

A Força Aérea dos Estados Unidos, surgida da introdução de novas tecnologias do século XX, ficou responsável pelo desenvolvimento dos mísseis balísticos intercontinentais. O desenvolvimento de um míssil ICBM era extremamente complexo e teve que ser feito em ritmo de corrida contra o inimigo. O desenvolvimento de mísseis ICBMs exigiu o uso de rigorosas metodologias de gerenciamento que foram formalizadas na série de documentos AFSCM 375. A aplicação de metodologias similares às da AFSCM 375 na NASA tornou possível o cumprimento das extremamente difíceis metas do programa Apollo.

A AFSCM 375-5 exercitava a rastreabilidade de requisitos através do uso de matrizes de verificação para demonstração da completude de projeto. O programa Apollo exigiu a automatização dos processos de gerenciamento de informações e requisitos.

Houve críticas ao extremo rigor de documentação exigido pela série AFSCM 375, por parte da indústria. Grady (2010, p. 40) afirma que o rigor exigido pela série AFSCM-375, somente poderia ser satisfatoriamente atendido com uso de computadores; mas que, naquela época, estes eram ainda primitivos e o trabalho manual feito com sua ajuda seria tão volumoso quanto o feito em papel.

A forma indiscriminada como a série AFSCM 375 foi aplicada pela geração seguinte àquela que a havia criado foi criticada pelos responsáveis por sua criação. Um comitê que analisou as causas do acidente do Space Shuttle Challenger criticou a diminuição do rigor da Engenharia de Sistemas na NASA após o final do programa Apollo.

O sucesso da aplicação da Engenharia de Sistemas no programa espacial civil levou a muitas tentativas, geralmente frustradas, de sua aplicação na solução de problemas sociais.

Foi apresentado que as aplicações da Engenharia de Sistemas passaram por fases de maior e menor sucesso, e de maior e menor rigor, quando aplicadas em sistemas técnicos e sistemas sociais.

5 A ENGENHARIA DE REQUISITOS

Desde que a Engenharia de Requisitos emergiu devido a dificuldades com requisitos de *software*, foi necessário estudar a origem do *software* e da Engenharia de *Software* para melhor entender a Engenharia de Requisitos.

5.1 A Origem do *Software*

O primeiro computador eletrônico digital operacional, o ENIAC (Electronic Numerical Integrator and Computer), apresentado em 1946 pelo U. S. Army's Ballistic Research Laboratory, apesar de revolucionário quando comparado a outras máquinas de cálculo existentes na época, tinha pouca semelhança com os computadores modernos, pois tinha sido projetado para fazer repetidas vezes o mesmo cálculo e não possuía memória para programas, e, mesmo podendo ser reprogramado para fazer outros cálculos, sua capacidade de programação era muito limitada, pois era feita no *hardware*, pela posição de cabos em seus painéis (ENIAC, 2012).

Ainda durante a Segunda Guerra Mundial, a U. S. Navy havia contratado o desenvolvimento de um simulador de voo para treinamento de pilotos que, apesar de ter sido inicialmente planejado para usar um computador analógico, acabou levando ao desenvolvimento do Whirlwind, o primeiro computador eletrônico digital com memória para programas, processamento em tempo real e *display* de tubo de raios catódicos, e que entrou em operação em 1951 (WHIRLWIND, 2012). Apesar de a U. S. Navy ter se desinteressado pelo projeto, o Whirlwind foi mais tarde usado como base dos computadores do sistema SAGE (Semi-automatic Ground Environment), um enorme e complexo sistema de defesa aérea, composto por radares e computadores interligados por uma rede de comunicação digital de enormes dimensões.

Até o final da década de 1950, os sistemas de computação eletrônica digital eram raros, primitivos e, operando com dezenas de milhares de válvulas termiônicas, eram sujeitos a falhas frequentes, consumiam muita potência elétrica e não comportavam grandes programas. No entanto, no início dos anos

1960, com a introdução dos dispositivos eletrônicos semicondutores, os computadores deixaram de ser privilégio de projetos militares, com muitos recursos, e experimentos de universidades com nomes específicos, como ORDVAC, MANIAC, ILLIAC, e passam a ser usados em inúmeras aplicações comerciais como contabilidade bancária, folhas de pagamento, controle de reservas de hotéis e controle de tráfego aéreo. Com o rápido crescimento da capacidade dos computadores eletrônicos digitais, aparecem também as primeiras linguagens de programação de alto nível, tornando a tarefa de programação muito mais fácil, rápida e eficiente e permitindo a criação de programas maiores e mais complexos.

Juntamente com o aumento da complexidade dos programas, começam a aparecer dificuldades no seu desenvolvimento, pois:

- A programação era uma atividade nova e para a qual ainda não havia metodologias apropriadas de desenvolvimento;
- A atividade de programação parecia não exigir muitos conhecimentos, pois bastava conhecer uma linguagem de programação para ser capaz de criar um programa;
- Usando-se a abordagem comum da engenharia, dividia-se um grande programa em partes que eram designadas a diferentes programadores;
- A inexistência de mão de obra treinada, causada por um mercado em rápida expansão, obrigava o uso, em projetos de grandes programas, um exército de programadores quase sem nenhum treinamento.

Essa foi a fórmula do fracasso para muitos dos primeiros projetos de *software* de grande porte.

5.2 Metodologias de Engenharia de Sistemas Aplicadas a *Software*

Até meados dos anos 1960, apesar do custo do *software* já ser da ordem de 3 a 5 vezes o custo do *hardware*, o desconhecimento sobre *software* era ainda tão grande a tal ponto que o *software* não era tratado como um item separado na contratação de sistemas pela Electronic Systems Division (ESD) da Air Force Systems Command (AFSC), mas era tratado ou como *hardware* ou como

documentação necessária para o funcionamento do *hardware* (THAYER, 1975).

Como a USAF já estava envolvida com desenvolvimentos de *software* desde meados dos anos 1950, quando trabalhou com o sistema de defesa aérea SAGE, e foi responsável pela formalização da Engenharia de Sistemas, ocorreu que, antes que metodologias específicas de desenvolvimento de *software* tivessem sido criadas, processos de engenharia de sistemas da série AFSCM-375 tenham sido usados tentativamente pela USAF e pela NASA, na aquisição de *software*. Os trabalhos que relatam esses estudos discutem as fases do ciclo de vida de desenvolvimento do *software*, levantam problemas com indefinição e controle de requisitos, controle de configuração, e demonstram uma grande sensibilidade de seus autores (LIEBOWITZ, 1967), (RATYNSKI, 1967), (SEARLE; NEIL, 1967), (SEARLE; ROSOVE; SIDOW 1969), (PILIGIAN; POKORNEY, 1967), (PILIGIAN; BASHAW; POKORNEY, 1968) e (POKORNEY; MITCHELL, 1967).

É muito instrutivo ler a crítica de Pokorney e Mitchell (1967, p. 13) sobre um problema ocorrido na aquisição de *software* pela USAF.

In the past the Air Force has had some painful experience in the acquisition of computer based systems. In the acquisition of one recent large computer based system, problems generated in the development of the information processing segment proved so great that they prevented the system from becoming operational. One of the factors that contributed to the failure of the project was the lack of a system specification. As a result no explicitly defined nor commonly understood set of system objectives was established. Consequently, specific system requirements could not be allocated to individual computer programs. The design documents for the computer programs were generally not definitive enough and not subject to controls. The net effect was that a design requirements baseline was never established. The contractor was, to a certain degree, left to design and develop the computer programs on his own without detailed guidance and feedback from the Air Force resulting in computer programs that never did satisfy the users requirements even after repeated redesigns. The cost of the computer programs grew to five times the original estimates during the design and development. When the decision was made to delete the system from the inventory approximately \$27,000,000 had been spent on the system, of which about

55% represented computer program costs, and additional funds were needed to meet the system requirements. Even those elements of the system that operated properly suffered from inadequate and inaccurate documentation and thus could not be used to maximum effectiveness. It cannot be claimed that the use of a specific technique or group of techniques would have solved all of the problems associated with this system. It is clear, however, that the use of a systems approach that included the computer programs would have greatly increased the probability of achieving the design goals. (POKORNEY; MITCHELL, 1967).

5.3 A Emergência da Engenharia de Software

Por volta de 1967, depois de um crescimento vertiginoso de suas capacidades e aplicações, os computadores eletrônicos digitais já eram bastante comuns, sendo que, somente na Europa, já havia mais de 10.000 computadores instalados e esse número crescia à espantosa taxa de 25 a 50% por ano (NAUR; RANDELL, 1969, p. 9).

Um exemplo do espantoso crescimento das dimensões e complexidade dos sistemas operacionais dos computadores é apresentado na Figura 5.1 que mostra um crescimento de um fator de quase 1000 vezes em um intervalo de 10 anos, no número de linhas de código de sistemas operacionais identificados na Figura 5.1 por números e nomes. Não é de se estranhar que ocorressem problemas com o desenvolvimento de sistemas tão grandes quanto cinco milhões de linhas de código, em uma área muito jovem, empregando milhares de programadores com pouco treinamento e sem metodologias específicas.

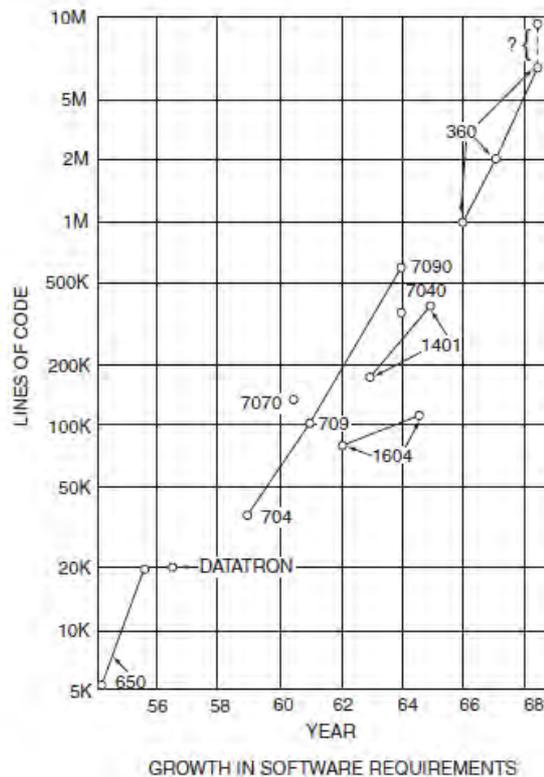


Figura 5.1 - Crescimento da Complexidade de Sistemas Operacionais.
 Fonte: Naur e Randell (1969, p. 38).

Observa-se, em meados dos anos 1960, a existência de três tipos principais de interessados em *software* com preocupações bastante diferentes: os acadêmicos, os desenvolvedores de *softwares* comerciais, e os militares. Os acadêmicos trabalhavam com programas pequenos e estavam mais preocupados com questões teóricas da Ciência da Computação, como demonstrabilidades, adequação de construção de determinadas estruturas (comandos GOTO). Os desenvolvedores de *softwares* comerciais trabalhavam com programação em larga escala, com resultados práticos, com cumprimento de metas de custos e prazos mesmo que resolvidos pela força bruta, i.e., com um exército de programadores. Os militares também trabalhavam com *softwares* enormes e também tinham preocupação com prazos e custos, mas estavam também interessados em *softwares* em tempo real e *softwares* que funcionassem de forma confiável.

Não causa surpresa que a preocupação dos militares com a situação dos desenvolvimentos de *software* tenha feito com que, em 1968, a Organização do

Tratado do Atlântico Norte (OTAN) – a maior organização militar dos países do Ocidente - convocasse uma conferência reunindo acadêmicos e desenvolvedores para discutir a crise por que passava o *software*. O que ficou conhecido na história como “crise do *software*” se manifestava pela incapacidade de cumprir metas de custos e prazos e pela baixa qualidade dos *softwares* de grande porte. Decidiu-se chamar, provocativamente, a conferência de “Engenharia de *Software*”, para demonstrar que a produção de *software* deve ser baseada em fundamentos teóricos e práticos, da mesma forma como as outras áreas da Engenharia (NAUR; RANDELL, 1969, p. 8).

Um fato discutido na conferência é que programas de pequeno porte, como folhas de pagamento, e aqueles que podiam ser produzidos por poucas pessoas, não apresentavam problemas. Observou-se que quando se usava a abordagem da engenharia para resolver grandes problemas e se dividia o *software* em componentes menores, apareciam grandes dificuldades com a descrição das partes e com o gerenciamento das interfaces entre as frações do problema (NAUR; RANDELL, 1969, p. 9).

Nos anos 1970, enquanto os mais otimistas sonhavam que no futuro não haveria necessidade de programação devido à “programação automática” (TEICHROEW, 1972), outros tinham uma perspectiva bastante cética sobre o futuro do *software*. Essa situação é demonstrada pela contundente frase de Edsger Dijkstra, um acadêmico participante da conferência da OTAN, comentando sobre o crescimento da capacidade dos computadores e seu impacto no desenvolvimento de *software*:

The major cause [of the software crisis] is that the machines have become several orders of magnitude more powerful! To put it quite bluntly: as long as there were no machines, programming was no problem at all; when we had a few weak computers, programming became a mild problem, and now we have gigantic computers, programming has become an equally gigantic problem. In this sense the electronic industry has not solved a single problem, it has only created them, it has created the problem of using its products. (DIJKSTRA, 1972)

Publicações do início dos anos 1970 demonstravam que:

- A complexidade dos *softwares* vinha crescendo a taxas muito altas (BOEHM, 1972);
- *Software* era “*Big Business*”, constituindo mais do que 1% do Produto Interno Bruto dos USA (BOEHM, 1972, p. 4);
- O custo do *software* tinha se tornado a maior fração do custo total dos sistemas de grande porte, como demonstrado na Figura 5.2 (BOEHM, 1976);
- A complexidade e dimensão dos sistemas de *software* se tornaram tão grandes que as técnicas empregadas para seu gerenciamento tinham se tornado inapropriadas (BELL; THAYER, 1976);
- Havia a necessidade de automatização dos processos de desenvolvimento de *software* (BELL; BIXLER; DYER, 1977);
-

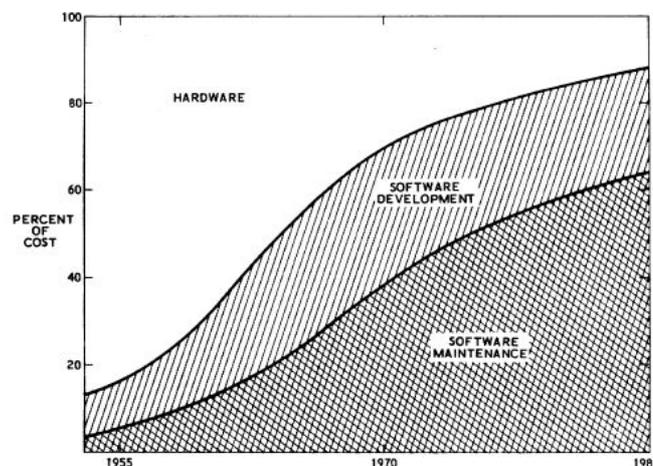


Figura 5.2 - Crescimento do Custo da Fração de *Software*
Fonte: Boehm (1976), p. 1226

A partir da sua emergência, a Engenharia de *Software* evoluiu rapidamente com o contínuo surgimento de novos processos e metodologias criadas para resolver os problemas originados pelas sempre crescentes dimensões e complexidade do *software* provocadas pela sempre crescente capacidade dos computadores. Como ilustração da rápida e ininterrupta evolução da

Engenharia de *Software* e das metodologias surgidas, a Figura 5.3 apresenta um sumário das inúmeras metodologias surgidas, seus protagonistas e o período em que foram introduzidas e utilizadas.

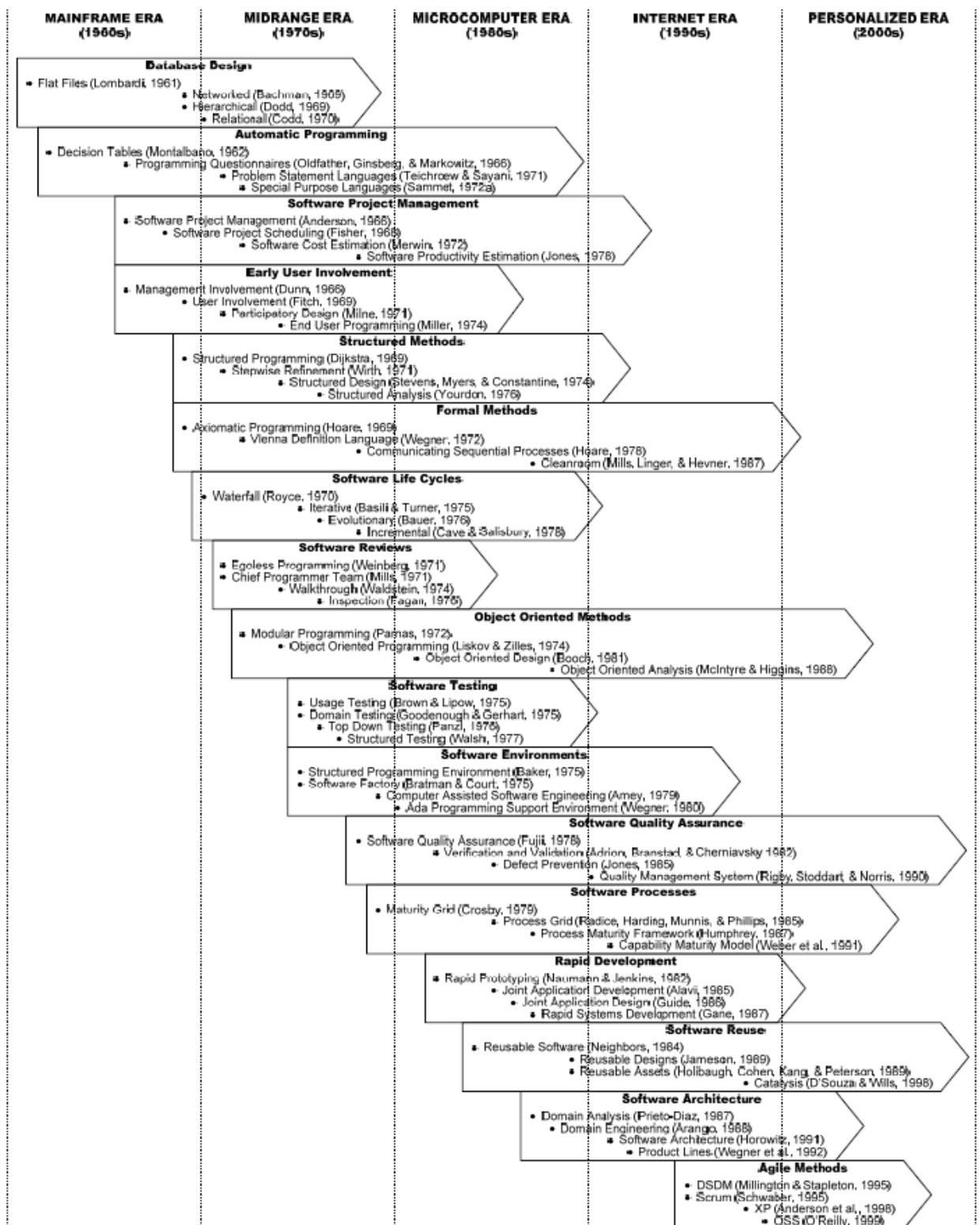


Figura 5.3 - Evolução das Metodologias de Desenvolvimento de Software.
 Fonte: Rico, Sayani e Field (2008).

5.4 A Origem da Engenharia de Requisitos

As origens da Engenharia de Requisitos estão associadas a dificuldades de descrição dos requisitos de *software*. Estas foram muito cedo observadas conforme mostra o texto abaixo escrito sob uma visão de Engenharia de Sistemas da série AFSCM-375 e antes que a palavra “*software*” fosse definitivamente usada na USAF para designar “computer program”:

In computer programming we are faced with one other factor—the flexibility of the programmable digital computer. A somewhat exaggerated view of this capability had led unwary managers to the belief that, since changes to a finished program can be easily made, the time and costs to define requirements, and the need to exercise control once they have been released, can be avoided in computer programming. Another difficulty is caused by the use of computers to provide flexibility in systems which operate in a changing environment. These systems are designed so that computer programs will absorb the major portion of expected changes. This allows firm definition of hardware requirements early in the development cycle of the system but makes it difficult to define the computer program requirements at the same time.”

“The impossibility under realistic conditions of establishing complete and definitive requirements in the programming process should not, however, be interpreted as a hopeless impasse to management control. On the contrary, if we accept as the general rule that the definition of requirements is an evolving process, we must conclude that a major function of management control is to insure that the development of the part 1 spec is an orderly one. This can be accomplished by providing mechanisms for the development of the spec and then for controlling changes to the spec as the requirements become better defined. This is actually what configuration management is all about as discussed in the paper by Searle and Neil. (LIEBOWITZ, 1967)

Ainda com relação às dificuldades apresentadas pelos requisitos de *software*, a Figura 5.4 apresenta o resultado de um estudo publicado em 1975, que demonstra que, apesar de o problema ser conhecido dos militares há mais de uma década, muito ainda restava a fazer para solucioná-lo. A Figura 5.4 mostra que 64% do total de erros encontrados durante o desenvolvimento de um *software* são devido ao projeto e 36% devido à codificação; e, considerando somente os erros encontrados após a aceitação do *software*, 45% são devido ao projeto e 9% à codificação. A figura 5.4 demonstra que os erros de projeto

representam a maior parte dos erros totais e a maior parte dos erros encontrados na fase final do projeto.

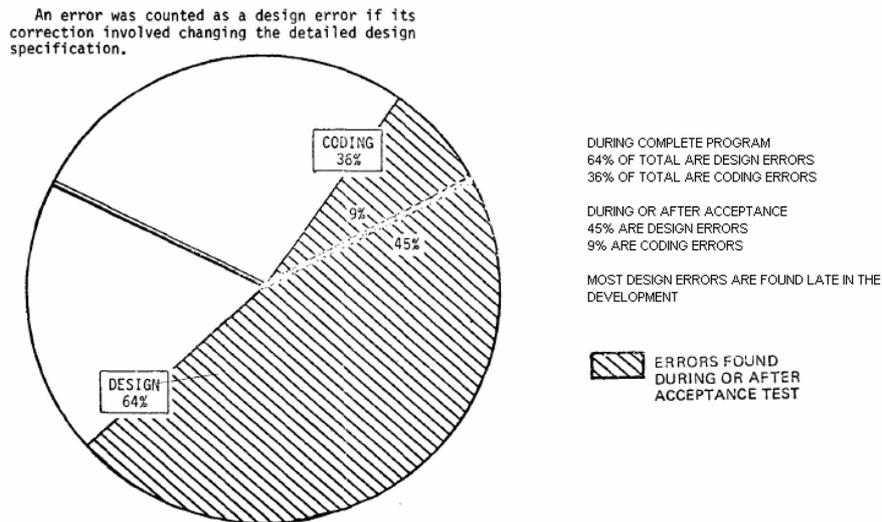


Figura 5.4 - Frações de Tipos de Erros de *Software*.
Fonte: Boehm, McClean e Ulfrig(1975).

Um outro estudo, cujo resultado é mostrado na Figura 5.5, apresenta o custo relativo para corrigir um erro de *software* como função da fase do seu ciclo de vida e demonstra a dificuldade de corrigir erros nas fases finais de projeto.

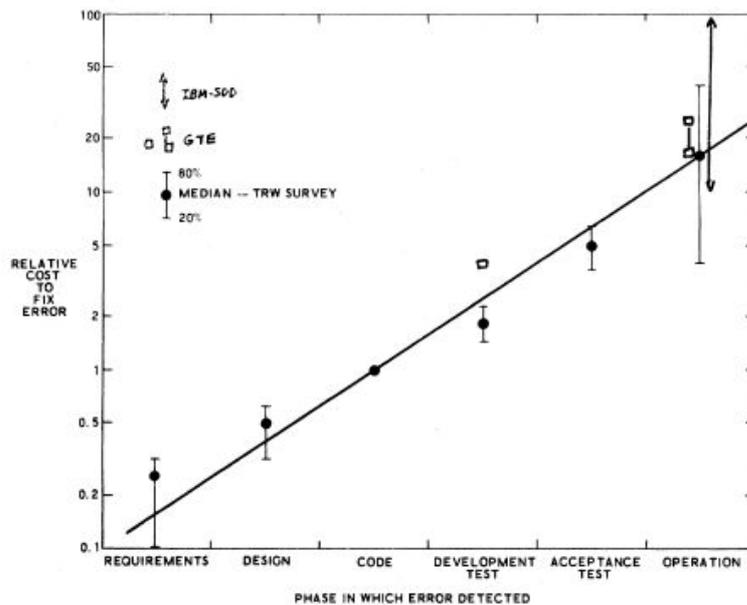


Figura 5.5 - Custo para Corrigir um Problema de *Software*.
Fonte: Boehm (1976).

Com relação ao impacto dos requisitos sobre o desenvolvimento de *software*, foi observado que:

- A maior parte dos problemas encontrados durante o ciclo de vida de um *software* não era devido a erros de projeto ou de codificação, mas devido à má qualidade dos requisitos iniciais (BOEHM, 1976);
- Problemas devido à má qualidade dos requisitos somente eram descobertos na fase de aceitação dos sistemas e eram extremamente difíceis de corrigir (BOEHM, 1976);
- Requisitos não aparecem naturalmente de uma necessidade óbvia, mas precisam ser descobertos por esforço diligente (BELL; THAYER, 1976);
- A linguagem natural não era apropriada para representar requisitos (BELL; THAYER, 1976);
- Havia a necessidade de se desenvolver metodologias para tratamento de requisitos (ALFORD, 1977).

Talvez a primeira definição de Engenharia de Requisitos apareça no artigo “Software Engineering”, em que Boehm (1976) sumariza os problemas que dominavam o desenvolvimento de *software*.

Software requirements engineering is the discipline for developing a complete, consistent, unambiguous specification – which can serve as a basis for common agreement among all parties concerned – describing what the software product will do (but not how it will do it; this is to be done in the design specification).

The extreme importance of such a specification is only now becoming generally recognized. Its importance derives from two main characteristics: 1) it is easy to delay or avoid doing thoroughly; and 2) deficiencies in it are very difficult and expensive to correct later. (BOEHM, 1976)

Devido ao alto nível e ao grande número de artigos pioneiros, publicados no início dos anos 1970, é possível afirmar que a emergência da Engenharia de Requisitos ocorreu como fruto do trabalho de pesquisadores da TRW envolvidos em projetos do Ballistic Missile Defense Advanced Technology Center (BMDATC).

A TRW, que esteve envolvida na formalização da Engenharia de Sistemas no início dos anos 1960, também é responsável por muito dos trabalhos pioneiros que deram origem à Engenharia de Requisitos nos anos 1970. Esses trabalhos descrevem a metodologia Software Requirements Engineering Methodology (SREM) em que já eram empregados o rastreamento de requisitos (ALFORD, 1977, DAVIS; VICK, 1977), ferramentas de automação aplicadas ao desenvolvimento de *software* (BOEHM; MCLEAN; ULFRIG, 1975) e ferramentas de automação do rastreamento de requisitos (DAVIS; VICK, 1977) e (BELFORD et al., 1976).

Foi somente em 1986 que Fred Brooks, que tinha sido o gerente de desenvolvimento do sistema operacional e computadores da série IBM/360 – um dos maiores e mais importantes empreendimentos de *software* do seu tempo, mas também um enorme fracasso de custos e prazos – escreveu o pequeno ensaio “No Silver Bullet” (republicado como o Capítulo 17 de Brooks, Jr. 1995) contestando o contínuo aparecimento de novas metodologias de desenvolvimento de *software* alardeando serem a solução para todos os problemas. Nesse texto ele afirma que o *software* se distingue das outras criações do homem; e que para *software* não há “bala mágica” e somente muito trabalho, esforço, muita organização e disciplina produzem resultados. Nesse texto ele também expressou sua conhecida frase a respeito da dificuldade de trabalhar requisitos de *software*:

The hardest single part of building a software system is deciding precisely what to build. No other part of the conceptual work is as difficult as establishing the detailed technical requirements, including all the interfaces to people, to machines, and to other software systems. No other part of the work so cripples the resulting system if done wrong. No other part is more difficult to rectify later. (BROOKS JR., 1995, p. 199)

5.5A Introdução da Automatização dos Processos da Engenharia de Requisitos

Desde que os problemas com desenvolvimento de *software* eram mais aparentes em projetos de grande porte, e estes geralmente eram projetos militares, é de se esperar que as soluções para os problemas também surgissem em projetos militares. De acordo com o relatório produzido sob contrato do Secretário de Defesa, a TRW, trabalhando para o Centro de Tecnologias Avançadas de Defesa contra Mísseis Balísticos (Ballistic Missile Defense Advanced Technology Center - BMDATC), havia desenvolvido:

an array of software tools, procedures, design methodology, management controls, and verification methods. The software aids address a broad spectrum covering requirements, design, implementation, test and integration, documentation, and software management....

....TRW believes that many technical, cost and schedule problems relating to software can be traced to inadequately defined requirements. They emphasize the importance of establishing traceability between requirements, specifications, and design. To this end, special requirements languages and automated tools to translate requirements into detailed specifications are being developed..... (KOSSIAKOFF et al., 1975)

A TRW desenvolveu uma metodologia própria para tratamento de requisitos, a SREM (Software Requirements Engineering Methodology) e uma linguagem de descrição de requisitos de *software*, a RSL (Requirements Statement Language). Como parte desse trabalho, a TRW desenvolveu uma das primeiras ferramentas de automatização de rastreamento de requisitos de *software* com características similares às das ferramentas atuais (DAVIS; VICK, 1977). A necessidade de ferramentas de automatização é óbvia quando se está trabalhando com um sistema com 8248 requisitos e um documento de especificação de 2500 páginas, como reportado por Bell, Bixler e Dyer (1977).

Não se pode deixar de citar que também houve estudos e desenvolvimentos realizados por universidades e outras indústrias para automatizar o desenvolvimento de *software*, sendo relevante citar o ARTS (Automated Requirements Traceability System) da Lockheed Missile and Space Company

(Dorfman e Flynn, 1984). Nesses trabalhos são encontrados muitos dos termos e conceitos hoje usados em Engenharia de Requisitos, como *requirements engineering*, *trace*, *tracing*, *requirements traceability* e *requirements management*. Essas primeiras ferramentas experimentais CASE (Computer-Aided Software Engineering) de rastreamento de requisitos eram comumente usadas por empresas da área aeroespacial como um tipo de arma secreta para obterem vantagens comerciais em relação aos seus concorrentes. (FLYNN; DORFMAN, 1990, p. 424)

5.6 Comentários Finais

As dificuldades apresentadas pelo *software* foram cedo notadas durante o desenvolvimento de *software* para os programas militares e espaciais quando, devido à inexistência de metodologias específicas para *software*, foram usadas metodologias de Engenharia de Sistemas.

A Engenharia de *Software* evoluiu associada ao aumento da capacidade dos computadores e ao aumento do custo da fração de *software* no custo total dos sistemas.

A Engenharia de Requisitos emergiu das dificuldades com requisitos e seus efeitos no custo, prazo e qualidade dos desenvolvimentos.

A necessidade da automatização dos processos de Engenharia de *Software* e de Engenharia de Requisitos levou ao desenvolvimento de muitas metodologias e das primeiras ferramentas CASE experimentais.

6 A ENGENHARIA DE SISTEMAS NO INPE

Muitos aspectos da história das atividades espaciais no Brasil já foram estudados por pesquisadores como Pereira (2008), Costa Filho (2000; 2006), Escada (2005), Leite Junior (2010), Moreira (2009), Vasconcellos (2008) e Souza (2008). Este capítulo se destina somente a apresentar um estudo da introdução e evolução da Engenharia de Sistemas no INPE e, sua intenção é permitir um melhor entendimento do estado atual da Engenharia de Sistemas do INPE e das forças que atuaram durante sua evolução. É sugerido neste capítulo que a Engenharia de Sistemas no INPE passou por três fases de maior atividade e evolução, sendo que as duas primeiras estão diretamente relacionadas com projetos de desenvolvimento de satélites, e a terceira e atual fase, é aquela na qual se inserem este e outros estudos recentes que estão sendo realizados no INPE.

6.1 A Primeira Fase da Engenharia de Sistemas no INPE

6.1.1 O Início das Atividades Espaciais no Brasil

Talvez se possa afirmar que as atividades espaciais no Brasil se iniciaram quando Julio Alberto de Moraes Coutinho e Fernando de Mendonça - que viria a ser primeiro diretor do INPE - ainda como alunos no Instituto Tecnológico de Aeronáutica (ITA), solicitaram ao Naval Research Laboratory (NRL), órgão responsável pelo desenvolvimento do Vanguard - o presumido primeiro satélite dos Estados Unidos - para instalar em São José dos Campos uma estação Minitrack, a ser usada no rastreamento do Vanguard (HARVEY; SMID; PIRARD, 2010, p. 311). Essa estação foi de fato instalada, mas devido à falha no lançamento do Vanguard foi usada, depois de algumas modificações, no rastreamento dos satélites Sputnik e Explorer (ESCADA, 2005, p. 48).

Após concluir o curso no ITA, Fernando de Mendonça foi para os Estados Unidos, onde estudou na Stanford University para obtenção do título de doutor em ciências radioionosféricas, e onde foi representante do Conselho Nacional de Pesquisas (CNPq) perante a NASA. Foi nesse contato com a NASA que se iniciaram as boas relações do INPE com a NASA e que Fernando de

Mendonça foi introduzido à Análise de Sistemas, da qual se tornou um adepto determinado a transplantá-la para o Brasil (MADDOX, 1974a). Retornando ao Brasil, Fernando de Mendonça em pouco tempo assumiu a direção do recentemente criado Grupo de Organização da Comissão Nacional de Atividades Espaciais (GOCNAE), mais conhecida apenas como CNAE (ESCADA, 2005, p. 50, nota 28).

Como a especialidade de Fernando de Mendonça era em ciências radio-ionosféricas, é natural que as primeiras atividades espaciais da CNAE fossem voltadas para as áreas de ciências espaciais e atmosféricas e, devido à escassez de recursos, alguns dos primeiros laboratórios da CNAE foram montados com equipamentos emprestados ou doados pela NASA e pela Stanford University. Muitas dessas atividades de pesquisa envolviam experimentos de sondagens em alta atmosfera realizadas por sensores instalados em balões e foguetes. O projeto Sondagem Aeronômica com Foguetes (SAFO), por exemplo, consistia na determinação das características do ambiente espacial e era parte de estudos de segurança das tripulações do programa Apollo (INPE, 1971). Esse projeto também envolvia o Ministério da Aeronáutica e os lançamentos eram feitos do Centro de Lançamento da Barreira do Inferno por membros da Força Aérea Brasileira (FAB) treinados nos Estados Unidos em meteorologia, segurança, manipulação, operação, além de outros conhecimentos necessários para uso dos foguetes (ESCADA, 2005, p. 51).

6.1.2O Projeto SACI e suas Consequências

A iniciativa da CNAE para rapidamente aumentar sua capacidade de pesquisa, foi convencer pesquisadores estrangeiros a montar laboratórios no Brasil e, para formar a próxima geração de pesquisadores brasileiros, o projeto PORVIR enviava os melhores estudantes brasileiros, selecionados das melhores escolas, para realizar estudos de doutorado em instituições dos Estados Unidos e Europa (INPE, 1971; ESCADA, 2005, p. 52).

Uma oportunidade para a CNAE se envolver com o desenvolvimento de um satélite surgiu em 1967 quando estudantes da CNAE estando na Stanford University, envolveram-se no estudo Advanced System for Communication and Education in National Development (ASCEND). O estudo ASCEND era parte de um curso de engenharia de sistemas e além de fazer uma análise comparativa de dois sistemas de telecomunicações para os países Índia, Indonésia e Brasil, um sistema baseado em um satélite geostacionário e outro baseado em uma rede terrestre, apresentava uma concepção preliminar para um satélite geoestacionário de telecomunicações (STANFORD UNIVERSITY, 1967).

O relatório do estudo ASCEND gerou na CNAE um relatório muito similar com o título “Projeto SACI” (SACI - acrônimo de Satélite Avançado de Comunicações Interdisciplinares), apresentando um estudo conceitual dos vários segmentos da missão e dos subsistemas de um satélite geoestacionário de telecomunicações dirigido para aplicações educacionais (CNAE, 1968).

O objetivo final do Projeto SACI era bastante ambicioso, pois planejava a operação pela CNAE de um satélite doméstico de telecomunicações, com lançamento proposto para 1976 (CNAE, 1970, p. 24). Como primeiro passo na direção de obter o satélite SACI e, devido à necessidade de demonstrar sua utilidade para convencer os órgãos financiadores, a CNAE iniciou um projeto homônimo (Projeto SACI) que envolvia os testes de um avançadíssimo satélite experimental da NASA, o satélite ATS-F (Applications Technology Satellite) - depois do lançamento chamado ATS-6 - em um experimento de educação à distância com transmissão direta de sinais de televisão a receptores instalados em escolas de áreas remotas do estado do Rio Grande do Norte (MADDOX 1974a). Experimento similar com o satélite ATS-F estava inicialmente planejado para ser realizado com a Índia (VON BRAUN, 1970), mas graças às excelentes relações do INPE com a NASA, o Brasil teve precedência na sua realização (MADDOX, 1974b).

6.1.3A Introdução da Engenharia de Sistemas na CNAE

Como a CNAE era voltada para pesquisas em ciência espacial e o desenvolvimento de um satélite como o SACI exigia uma capacidade não existente na CNAE, houve a necessidade de obter competência em gerenciar projetos espaciais de alta complexidade. Apesar de a CNAE já ter sido introduzida nos métodos da Análise de Sistemas e já ter criado o Núcleo de Análise de Sistemas (NAS), não havia ainda na CNAE competência em Engenharia de Sistemas necessária ao planejamento e execução do projeto de um satélite.

Não foram encontrados registros na biblioteca do INPE que indicam com precisão como ocorreu a introdução da Engenharia de Sistemas na CNAE. O documento Projeto SACI Relatório Nº III (CNAE, 1970a, p. 2) cita que a CNAE havia contratado consultoria de uma empresa americana e uma inglesa, sem citar os nomes das empresas; e que estas tinham colocado à disposição da CNAE um total de 30 engenheiros experientes trabalhando no exterior e no Brasil. Não foram encontrados registros que demonstram como a General Electric Space Division foi envolvida na tarefa de treinar os funcionários da CNAE em Engenharia de Sistemas. Na proposta submetida à NASA, solicitando uso dos recursos do satélite ATS-F no experimento do Projeto SACI, essa informação está da seguinte forma:

Conscious of both the incomplete development of aerospace systems capability and the time dependent value of improve education in Brazil, but recognizing the exhaustively demonstrated competence of US firms in space systems implementation, CNAE has contracted for systems management consulting services with a major private American space systems organization in defining the SACI Program in detail including the steps necessary to carry out the proposed ATS-F Experiment. (CNAE, 1970b, p. 1.2)

Esse mesmo documento, mais à frente, se refere diretamente à General Electric Company Space Systems Organization, que teria 20 engenheiros treinados em Engenharia Espacial e engenharia de sistemas, e à British Aircraft Company, que teria 2 engenheiros trabalhando na CNAE e outros 6

trabalhando em Bristol, na Inglaterra. (CNAE, 1970b, p. 7.10). Outros documentos encontrados são: o relatório do Seminário de Engenharia de Sistemas apresentado no INPE para um grupo de funcionários do Instituto Nacional de Estudos Pedagógicos (INEP), do qual participam funcionários da CNAE e três consultores da General Electric Space Division (CNAE, 1971a, p. 43); e dois volumes usados pela General Electric Space Division no treinamento de seus funcionários na preparação de propostas de acordo com as rigorosas normas da série AFSCM-375 (GENERAL ELECTRIC, 1972a, 1972b). No entanto, há uma discordância entre as datas desses documentos, pois os documentos da General Electric são de 1972, e o seminário para o INEP ocorreu em 1970. Uma possível explicação para esse desacordo de datas é que houve mais de um seminário da General Electric, inclusive alguns possivelmente anteriores a 1970, e somente os volumes de 1972 foram arquivados na biblioteca.

No relatório do seminário, nas razões para o uso de uma abordagem de sistemas no desenvolvimento do Projeto SACI, são citados os problemas normalmente no desenvolvimento de projetos:

Efetividade Limitada: normalmente os projetos, por falta de controle de qualidade técnica, bem como de planejamento integrado, chegavam ao fim com sistemas de qualidade bastante pobre e aquém dos requisitos exigidos.

Resultados Frequentemente não Relacionados com as Necessidades Reais tendo em vista o fato que os participantes do projeto estavam muito preocupados com a solução do problema do ponto de vista técnico, muitas vezes o sistema resultante estava desvinculado dos objetivos que se queriam atingir, isto porque faltava definição clara, do enunciado do problema, bem como controle e avaliação conveniente.

Atrasos Sérios de Cronograma - em virtude da falta de sistema conveniente de controle de progresso, os projetos atrasavam e se alongavam, muitas vezes tornando-se obsoletos.

Custos Excessivos - a falta de estrutura adequada de estimativa e controle de custo levava os projetos a superarem, em muito, seus orçamentos iniciais.

Má Direção: com a inexistência de sistema de informações convenientes, o gerente do projeto não dispunha dos dados

*necessários às ações corretivas no andamento dos programas.
(CNAE, 1971a, p. 22 - 23)*

A CNAE também se esforçou no sentido de automatizar o gerenciamento de informações com a criação de um banco de dados descrito num estudo preliminar que seria criado devido:

Os motivos que levaram a CNAE ao estabelecimento do presente trabalho foram: de um lado a criação de um modelo de automatização que simplifique as atuais rotinas de busca à informação, e que possa ser copiado ou adaptado por instituições similares; e de outro que este modelo, através do progresso de integração com órgãos similares venha a contribuir na solução global do problema.

Os fatores mais importantes que influenciaram no equacionamento do problema são:

- 1) O crescimento da quantidade de informações disponível.*
- 2) A necessidade de obtenção rápida destas informações.*
- 3) Os espaços cada vez maiores que vêm sendo ocupados pelo armazenamento das informações em arquivos clássicos.*
- 4) A diversificação dos meios pelos quais a informação é atualmente adquirida e a falta de padronização.*
- 5) A disponibilidade atual de técnicas de processamento extremamente rápidas. (CNAE, 1971c).*

6.1.3.1A Disseminação da Engenharia de Sistemas pela CNAE

Foi tão grande o entusiasmo com as possibilidades oferecidas pela Análise de Sistemas e pela Engenharia de Sistemas que a CNAE se propôs a ser um disseminador desses conhecimentos através da tradução e adaptação do livro "Introduction to Systems Analysis" (Moraes, 1970), que foi publicado somente como relatório interno da CNAE, e pela implantação de um Curso de Mestrado em Análise de Sistemas em suas Aplicações em 1968. No planejamento do currículo desse curso aparecem nomes de autores dos hoje clássicos livros da Engenharia de Sistema, havendo referências que alguns desses autores teriam sido contatados para participarem do curso (WALTER, 1971).

Foi criado o Núcleo de Análise de Sistemas da CNAE, que congregava especialistas em Pesquisa Operacional, Administração por Sistemas,

Simulação, Economia, para auxiliar os trabalhos da CNAE e também para servir de apoio a órgãos governamentais (MENDONÇA, 1972, p. 8).

Devido à necessidade de ser conhecida para conseguir apoio de outros órgãos do governo para o projeto SACI, a CNAE organizou, a partir do início da década de 1970, dezenas de seminários dirigidos a órgãos do governo que pudessem fazer uso das técnicas de Análise de Sistemas e Engenharia de Sistemas, tais como, ministérios do planejamento, da saúde, da educação, das comunicações e transportes, e também para indústrias privadas, como Johnson e Johnson.

Percebendo o potencial de aplicações da Engenharia de Sistemas em outras áreas, a direção do INPE chegou a convencer o Ministério do Planejamento da oportunidade de criação de um centro nacional de análises de sistemas, nos moldes da RAND Corporation e, para instalação desse centro, chegou a ser desapropriada uma área no município paulista de Cachoeira Paulista. A criação desse centro não se concretizou, mas a área desapropriada foi transferida para o patrimônio do INPE (CNAE, 1971b, p. 42-44).

O Núcleo de Análise de Sistemas do INPE foi muito ativo durante as décadas de 1970 e 1980, como pode ser verificado número de publicações relacionadas à aplicação de Engenharia de Sistemas no planejamento do projeto Sensores Remotos de Recursos Naturais, projeto do banco de dados do INPE, no planejamento do Curso de Mestrado em Análise de Sistemas e suas Aplicações em 1968, e na área educacional.

6.1.3.2 Os Manuais de Engenharia de Sistemas do INPE

Outro resultado do esforço de treinamento em Engenharia de Sistemas foi a dissertação de mestrado “Engenharia de Sistemas: Planejamento e Controle de Projetos”, a primeira publicação original em língua portuguesa nessa área (PULCHÉRIO et al., 1971). Publicada em 1971, essa dissertação foi posteriormente publicada como livro, com o mesmo título, com edições nos anos 1972, 1973 e 1976 (MENDONÇA, 1972). Essa dissertação e livro

apresentam uma introdução com as origens da Engenharia de Sistemas e é seguida de um exemplo da abordagem de Engenharia de Sistemas aplicada ao desenvolvimento do projeto SACI. Na sua bibliografia aparecem títulos de autores de clássicos da Engenharia de Sistemas (Goode, Machol, Hall, Wymore, entre outros) e referências à então recentemente publicada, série AFSCM-375 da USAF e aos documentos da General Electric Space Division. Estranhamente, mesmo sendo as relações com a NASA citadas no prefácio do livro como um fator que contribuiu para a experiência do INPE em Engenharia de Sistemas, não há referência ou citações a qualquer documento da NASA.

Uma pesquisa realizada pela Internet no catálogo da Biblioteca Nacional, e no sistema Dédalus, que congrega o catálogo de 43 bibliotecas da Universidade de São Paulo, e no site de venda de livros usados Estante Virtual, que lista o acervo de mais de 2000 lojas e contém milhões de livros em estoque, determinou que o livro “Engenharia de Sistemas: Planejamento e Controle de Projetos” é o primeiro livro sobre Engenharia de Sistemas publicado no Brasil e que há somente mais um livro publicado no Brasil sobre Engenharia de Sistemas, com o título “Manual de Engenharia de Sistemas e Projetos: Uma Abordagem Prática”, também publicado por pesquisadores do INPE (REIS et al., 1980).

6.1.3.3Os Requisitos no Manual de Engenharia de Sistemas do INPE

É importante comentar alguns parágrafos que aparecem no primeiro livro de Engenharia de Sistemas do INPE (MENDONÇA, 1972) devido à sua similaridade com as definições e critérios atuais da Engenharia de Requisitos. Nesses parágrafos são apresentadas definições e objetivos das especificações e os critérios de qualidade do texto das especificações, que, por razões de espaço, são apresentadas aqui de forma resumida.

Segundo esse livro, o texto das especificações deve ter (MENDONÇA, 1972, p. 82):

- Linguagem concisa;
- Objetividade;

- Suficiência;
- Clareza;
- Exaustividade;
- Compatibilidade;

e ausência de:

- Exigências desnecessárias;
- Dúvidas;
- Incertezas;
- Opções;

e ser:

- Completa;
- Não-ambígua;
- Não-explicativa;
- Inequívoca;
- Num formato padrão que favoreça a referência.

O manual apresenta vários exemplos de especificações, entre elas as especificações técnicas de equipamentos do projeto SACI, das quais se reproduz um pequeno trecho para demonstrar suas características e seu formato (MENDONÇA, 1972, p. 87):

4. Características de Entrada do Componente

4.1 – Frequência de operação: o receptor de UHF deverá operar numa frequência central nominal de 850 MHz.

4.2 – Impedância de entrada: o receptor de UHF deverá apresentar ao pré-amplificador que o precede uma impedância de 50 Ω .

4.3 – VSWR: a VSWR entre o pré-amplificador de UHF e o receptor de UHF não deve exceder 1.6:1, o que corresponde a uma reflexão de potência não maior que 5,5%.

4.4 – Nível de sinal na entrada do receptor: -59.7 dBm.

4.5 – Conectores: o conector de entrada será do tipo BNC.

Com relação ao problema de nomenclatura e terminologia que conforme discutido no Capítulo 2 ainda ocorre atualmente, é interessante notar o esforço na distinção de termos apresentados no manual, entre documento de requisitos e especificação:

Vale destacar a diferença entre especificações e documento de requisitos. Documento de requisitos é um documento anterior às especificações, contendo as características mais salientes do sistema. Já o documento de especificações é mais completo, pois contém todos os requisitos que um sistema deve satisfazer, além de outros detalhes, como introdução e documentos aplicáveis. Dentro do processo de engenharia de sistemas, a especificação é o produto final da fase de planejamento. (MENDONÇA, 1972, p. 83)

6.1.3.4A Documentação Conforme o Manual de Engenharia de Sistemas do INPE

É interessante também notar a atenção dada à documentação e arquivamento no manual de engenharia de sistemas do INPE, assunto tratado nos últimos parágrafos do livro. Nesses parágrafos são descritos o sistema de documentação e arquivo e as suas finalidades, destacando-se as características de simplicidade, flexibilidade, economia, utilidade, segurança e acessibilidade. É também informado que o projeto SACI tinha uma biblioteca com serviço automatizado por computador, possibilitando buscas por número de tomo, autor, editores ou entidades, e por palavra-chave pelo sistema KWIC (Key-Word In Context) (MENDONÇA, 1972, p. 268).

Há uma interessante frase no último parágrafo do livro que informa que, por uma alteração na definição de “documentação” pela hoje extinta Federação Internacional de Documentação, o foco de documentação tinha deixado de ser o “documento” e passado a ser a “informação”. Ocorre também nesse parágrafo o termo “informática”, provavelmente se referindo ao termo introduzido pelo uso de computadores. A leitura desse parágrafo dá indicações que o próximo passo no gerenciamento de documentos no INPE seria um controle ao nível das informações contidas nos documentos.

6.1.3.5 Ferramentas de Engenharia de Sistemas no Projeto SACI

A necessidade de aumentar a eficiência e automatizar os processos da Engenharia de Sistemas no projeto SACI levou o INPE ao uso de ferramentas computacionais. As informações relativas ao uso de computadores na automatização do trabalho estão dispersas em vários documentos, mas comprovam que o INPE desenvolveu e usou computadores e *softwares* para:

- Análise PERT (Project Evaluation and Review Technique) usando *software* PMS (Project Management System) da IBM e Mark III da Boyan Associates (CNAE, 1970a, p. 11; 1970b, p. 6.8; 1971b, p. 47);
- Controle de custos do projeto e plotagens de gráficos de custos (MENDONÇA, 1972, p. 254);
- Cronogramas em gráficos de barra e para plotagens em 3-D (CNAE, 1971b, p. 47);
- Linguagem de simulação GASP-II (CNAE, 1971b, p. 48);
- Sistema KWIC – Key-Word-In-Context - para automação da pesquisa bibliográfica (MENDONÇA, 1972, 270).

6.1.4 O Fim do Satélite SACI

A autonomia do INPE em determinar o caminho das atividades espaciais no Brasil foi perdida, em 1971, com a criação da Comissão Brasileira de Atividades Espaciais (COBAE) que passou a ter a responsabilidade de gerir as atividades espaciais, civis e militares, no Brasil. Diferentemente do INPE, a COBAE era vinculada ao Conselho de Segurança Nacional, tinha em sua formação vários ministros militares, era subordinada ao Estado Maior das Forças Armadas (EMFA) e presidida pelo ministro chefe do EMFA (PEREIRA, 2008, p. 39; COSTA FILHO, 2000, p. 94).

Apesar do sucesso das demonstrações do projeto SACI com o satélite ATS-F, realizadas já em meio a disputas por recursos e prestígio com os ministros das comunicações, educação e militares dentro da COBAE – o ministro da educação achava que se o satélite era voltado para educação devia estar sob seu ministério; o mesmo argumento era usado pelo ministro das comunicações

por ser um satélite de comunicações; os ministros militares não gostavam da maneira autônoma como o diretor do INPE dirigia o projeto – o projeto SACI foi interrompido em 1976 e seu maior defensor substituído na direção do INPE (PEREIRA, 2008, p. 47).

Não muito sobrou para o INPE da experiência de Engenharia de Sistemas do projeto SACI já que parte do grupo deixou o Instituto em seguida ao encerramento do projeto; muitos deles tendo ido trabalhar ainda com satélites, mas na Telebrás (COSTA FILHO, 2000, p. 105).

Os objetivos do projeto SACI se realizariam 10 anos mais tarde, não por um satélite do INPE, mas por um satélite do Sistema Brasileiro de Televisão por Satélite (SBTS) da Telebrás, lançado em 1985, e operado pela Embratel - o primeiro satélite da série Brasilsat.

6.2 A Segunda Fase da Engenharia de Sistemas no INPE

6.2.1 A Missão Espacial Completa Brasileira

A COBAE tinha a função de coordenar todas as atividades espaciais no Brasil, função que anteriormente, no contexto civil e por quase uma década, eram prerrogativas do INPE. O caráter militar da COBAE e a saída do diretor do INPE fizeram com que as relações com a NASA deixassem de ser tão estreitas quanto tinham sido. Isso fez com que o INPE intensificasse suas relações com a França, sendo que o Centre National d'Études Spatiales (CNES) passou a ser seu maior interlocutor (PEREIRA, 2008 p. 46).

Devido à falta de recursos e à dificuldade de se compatibilizar interesses civis e militares, as discussões sobre uma missão espacial com a França arrastaram-se por vários anos no âmbito da COBAE. Foram tão diversas e durante tanto tempo discutidas as várias propostas de concepção para a missão que engenheiros brasileiros, e a própria COBAE, passaram a vislumbrar a possibilidade de realização de uma missão completa exclusivamente brasileira. Foi uma decisão política aprovar, em 1980, a Missão Espacial Completa Brasileira (MECB) envolvendo o Centro Técnico Aeroespacial (CTA) e INPE,

com as tarefas de desenvolver e construir um veículo lançador e uma base de lançamento, 4 satélites, um centro de missão e controle, e algumas estações de rastreamento (PEREIRA, 2008, p. 73).

Entre 1980 e 1982 a MECB lutou contra a falta de recursos, mas isso não impediu que engenheiros do INPE fizessem estágios no CNES, Aerospatiale (França), General Electric (EUA), e SPAR (Canadá) - onde estavam sendo fabricados os satélites Brasilsat - e cursos do Projeto de Preparação de Especialistas da Área Espacial (PROPESA), do CTA, ministrados por especialistas trazidos do exterior. Muitos estudos e trabalhos foram produzidos durante esse período, com vistas na definição da concepção da MECB, mas a maioria dos trabalhos era voltada para estudos de subsistemas de satélites, não havendo qualquer trabalho voltado para a Engenharia de Sistemas (FERNANDES, 1987). De 1983 a 1985, a MECB decolou graças aos recursos substanciais aportados pelo governo do presidente João Batista Figueiredo.

6.2.2 A Reintrodução da Engenharia de Sistemas no INPE

Com o fim do regime militar, em 1985, e a reestruturação dos ministérios do governo, o INPE passou a ser subordinado ao recém-criado Ministério da Ciência e Tecnologia no governo do presidente José Sarney. Por orientação deste ao ministro Renato Archer, e deste ao novo diretor geral do INPE, que tinha excelentes relações com o ministro daquela pasta, o INPE recuperou seu prestígio e autonomia e continuou a ter suficiente aporte financeiro para o desenvolvimento de sua parte da MECB, mesmo esta estando ainda subordinada à COBAE (COSTA FILHO, 2006, p. 107).

Entre dezembro de 1984 e março de 1985, um engenheiro do INPE é enviado ao CNES para um estágio na equipe de gerenciamento do desenvolvimento do satélite de sensoriamento remoto francês SPOT. Esse estágio é completado por uma dissertação de mestrado, defendida no INPE em abril de 1987 (FERNANDES, 1987). Diferentemente dos livros sobre Engenharia de Sistemas, publicados pelo INPE, essa dissertação não é um manual geral de Engenharia de Sistemas, mas apresenta de forma prática e objetiva, o “estado

da arte” de meados dos anos 1980 da Engenharia de Sistemas Espaciais empregada no desenvolvimento do satélite SPOT e adaptada ao programa MECB. A dissertação contém, nos anexos, esqueletos de documentos a serem gerados para a MECB e textos integrais de documentos da MECB. Além de fazer referência a documentos do projeto SPOT, sua bibliografia também faz referência aos dois livros anteriormente publicados pelo INPE sobre Engenharia de Sistemas e aos dois volumes de treinamento da General Electric Space Division em Engenharia de Sistemas existentes na biblioteca do INPE. Apesar de realizado mais de dez anos após o encerramento do projeto SACI, esse trabalho não é desvinculado dele, pois ainda havia remanescentes desse projeto no INPE.

A organização da Engenharia de Sistemas descrita nessa dissertação é implantada no INPE com a criação, em 1986, da Gerência e Grupo de Sistemas da MECB do INPE responsável pelo desenvolvimento dos Satélites de Coleta de Dados (SCD) (PEREIRA, 2008, p. 98).

A aplicação de modernas metodologias de Engenharia de Sistemas, com recursos financeiros suficientes, mão-de-obra treinada e motivada, um recém-contratado gerente de garantia de produto com experiência de ter trabalhado no programa Brasilsat, fizeram com que o desenvolvimento do satélite SCD-1 ocorresse bastante rapidamente. Todo o trabalho de planejamento e concepção da missão dos satélites SCD e a maior parte do trabalho de desenvolvimento, fabricação e testes foi realizado internamente pelo próprio INPE, tendo sido contratadas empresas brasileiras ou estrangeiras apenas para a fabricação de alguns componentes.

Como o gerenciamento de sistemas dos satélites SCDs foi feito pelo INPE, poder-se-ia se dizer que no período da MECB o conhecimento em Engenharia de Sistemas Espaciais ficou restrito ao Instituto. Mas, de fato, o INPE foi novamente disseminador de técnicas de Engenharia de Sistemas quando exigiu que as empresas brasileiras contratadas para suprir os componentes dos satélites usassem metodologias similares às suas na organização do trabalho sendo realizado para o INPE.

6.2.3A Importância dos Requisitos no Programa MECB

A referida dissertação de mestrado na qual se baseiam os documentos da MECB dá grande atenção aos requisitos, define e dá suas características principais, afirma que o processo de sua elaboração é complexo e precisa ser sistemático, como se vê no trecho abaixo:

4.2.1 ESPECIFICAÇÃO DE REQUISITOS

A definição das características de um projeto (satélite) começa a ser identificada durante a fase B do projeto. Esta identificação é feita inicialmente em termos de requisito, ou seja, não se identifica o produto e suas partes pela solução a ser adotada, mas sim pelos requisitos que ela deve atender. Com isso consegue-se delimitar “o contorno dentro do qual a solução deve estar” e, ao mesmo tempo, garante-se a flexibilidade necessária para a concepção industrial do produto.

O processo de elaboração das especificações é complexo e por isso deve ser sistemático. Na medida em que se estabelece requisitos para o sistema (satélite) são gerados requisito para os subsistemas...O estabelecimento de um requisito para um subsistema pode gerar outros requisitos para o próprio subsistema ou para outro subsistema...(Fernandes, 1987, p. 34).

Fernandes (1987) distingue tipos e dá as características importantes dos requisitos e das especificações de requisitos (porém não na forma de lista como apresentada abaixo):

Completa:

Uma especificação de qualquer parte da árvore (de especificações) deve ser completa, isto é, a especificação de um produto deve descrever diretamente ou por referência todos os requisitos que se deseja especificar para esse produto (Fernandes, 1987, p. 38).

Não-ambígua:

Não se deve deixar implícito (sem fazer referência) que as especificações de nível mais alto na árvore sejam também aplicáveis (Fernandes, 1987, p. 38).

Verificável:

Descritos estes requisitos é necessário, então, que se descreva como eles serão verificados. Isso é feito prescrevendo-se para

cada um deles qual o método de verificação a ser implementado (Fernandes, 1987, p. 36).

Concisa:

Deve-se evitar "florear" o seu texto. Lembrando que uma especificação não é feita para leigos no assunto (Fernandes, 1987, p. 38).

Restrições:

Existem outros requisitos que são estabelecidos devido a outras experiências espaciais (por exemplo, a proibição de se utilizar certos materiais ou tecnologias que não deram certo em outros projetos espaciais) ou às condições espaciais (por exemplo, certas partes do satélite devem funcionar suportando uma aceleração de X acelerações da gravidade) (Fernandes, 1987, p. 35).

Requisitos Funcionais:

Requisitos são oriundos das funções que o produto deve desempenhar. A definição desses requisitos deve levar em conta todas as fases da vida do produto, desde o começo da fabricação até o fim de sua utilização (Fernandes, 1987, p. A1).

Requisitos Derivados:

O estabelecimento de um requisito para um subsistema pode gerar outros requisitos para o próprio subsistema ou para um outro subsistema (por exemplo: para se ter uma rigidez Y na estrutura é necessário que o controle térmico garanta uma faixa de temperatura de A-B °C) (Fernandes, 1987, p. 35).

Preciso e verificável:

Exceto quando são estritamente descritivos, os requisitos devem ser definidos quantitativamente (valor nominal e tolerância) e devem ser verificáveis (Fernandes, 1987, p. A1).

Não-prescritivos:

Os requisitos devem ser expressos em termos de necessidades. Quando houver soluções impostas deve-se descrevê-la na seção apropriada (Fernandes, 1987, p. A2).

E, no oitavo e último Capítulo – Comentários Gerais – o autor sugere para continuidade do seu trabalho:

Como exemplo de trabalhos complementares tem-se: estabelecer um processo sistemático para a elaboração das especificações de requisitos... (Fernandes, 1987, p. 86).

6.2.4 O Programa CBERS

Assinado com a República Popular da China em 1988, o acordo do Programa Sino-Brasileiro de Satélites de Recursos Terrestres – CBERS – (China-Brazil Earth Resources Satellite), previa o desenvolvimento de dois satélites de sensoriamento remoto e, a despeito da maior capacitação da China em tecnologia espacial quando comparada ao Brasil, não havia no acordo qualquer cláusula relativa à transferência de tecnologia entre os países (SANTANA; COELHO, 1999, 208).

O envolvimento da indústria brasileira no desenvolvimento dos satélites CBERS seguiria uma política diferente da seguida na MECB. No programa CBERS, devido ao muito maior esforço de projeto e fabricação e consoante com a política de progressivo envolvimento da indústria nacional, contrataram-se o desenvolvimento e a fabricação dos subsistemas da indústria brasileira.

O programa CBERS sofreu grande escassez de recursos durante seus primeiros anos, fazendo com que muitos dos contratos com a indústria fossem somente assinados a partir de 1993. Mesmo que muitas das empresas contratadas já tivessem experiência no desenvolvimento de sistemas aeronáuticos e de defesa, o INPE foi novamente disseminador de metodologias da Engenharia de Sistemas Espaciais já que as obrigou a usar metodologias similares às suas no trabalho realizado para a Instituição.

A CAST (Chinese Academy of Spacecraft Technology), parceira chinesa do INPE no desenvolvimento do satélite CBERS, um órgão civil entre muitos dos organismos civis e militares chineses envolvidos com tecnologia espacial, já tinha bastante experiência no desenvolvimento de sistemas espaciais. No entanto, devido à diferente origem de suas metodologias e ao longo período em que a China esteve isolada do resto do mundo, as suas metodologias de desenvolvimento eram muito diferentes das empregadas pelo INPE. Isso causou alguns desentendimentos no começo dos trabalhos. Mas, percebendo que as metodologias do INPE eram superiores às suas, a CAST passou a

adotar as metodologias empregadas pelo INPE, conforme observou o gerente do lado brasileiro do programa:

...beneficiaram-se sobremaneira, os chineses, dos conhecimentos das metodologias de gerenciamento modernas dominadas pelos parceiros brasileiros. (SANTANA E COELHO, 1999, p. 209)

Apesar dos atrasos devido à falta de recursos ocorridos no início do programa, os trabalhos transcorreram com bastante rapidez uma vez assinados os contratos com a indústria (SANTANA; COELHO, 1999, 207).

Três satélites da série CBERS foram lançados e operados com sucesso em 1999, 2003 e 2007, demonstrando que dois países em desenvolvimento, e extremamente diferentes do ponto de vista cultural, poderiam cooperar com sucesso em uma área tão complexa quanto a Engenharia de Sistemas Espaciais (COSTA FILHO, 2000).

O sucesso do programa CBERS fez com que a China se dispusesse a assinar um novo acordo com o Brasil para o desenvolvimento dos satélites CBERS-3 e CBERS-4. Nesse novo programa, a participação brasileira nos gastos aumentou devido à responsabilidade sobre dois instrumentos da carga útil, mas a divisão de responsabilidades no desenvolvimento de subsistemas é praticamente idêntica a do projeto anterior, assim como a ausência de qualquer acordo relativo à transferência de tecnologias.

6.2.5 Outros Programas Internacionais

Houve vários projetos e programas internacionais de pequeno, médio e grande porte em que o INPE poderia ter absorvido e atualizado conhecimento de Engenharia de Sistemas de seus parceiros. No entanto, esses projetos e programas ou tiveram curta duração, ou envolveram poucas pessoas, ou eram somente contratos de fabricação no Brasil com quase nenhum envolvimento do INPE, e causaram pouca alteração das metodologias de Engenharia de Sistemas empregadas pelo Instituto. É necessário mencionar a inovação introduzida por dois documentos, ocorrida devido a alguns desses projetos: o

relatório Design Justification File (DJF) com justificativas de projeto, uma exigência introduzida pelo projeto French-Brazilian Microsatellite (FBM) e o documento Review Item Discrepancy (RID) como parte do processo de revisão, uma exigência do projeto da International Space Station (ISS).

6.3 Falta de Treinamento em Engenharia de Sistemas no INPE

A Engenharia de Sstemas evoluiu muito nas duas últimas décadas, principalmente pela incorporação de processos, métodos e ferramentas criadas pela Engenharia de *Software*. No entanto essa evolução não se refletiu no INPE.

Atualmente, nenhum membro do grupo de sistemas da ETE tem formação em Engenharia de Sistemas, mas sim em uma área específica da Engenharia (Elétrica, Eletrônica ou Mecânica) ou Computação. E, por geralmente serem oriundos de um grupo de subsistemas, têm conseqüentemente uma visão de subsistemas. Não há treinamento em Engenharia de Sistemas para os novos funcionários ou aperfeiçoamento dos funcionários antigos. Em 1999 houve um esforço de treinamento em Tecnologia Espacial para a ETE, mas esse treinamento foi geral e em nível de subsistemas. Como, até recentemente, não havia um Curso de Engenharia de Sistemas no Brasil, e devido ao reduzido contingente de pessoal, não há atualmente na ETE um funcionário com treinamento formal em Engenharia de Sistemas e com responsabilidade de manter atualizado o conhecimento em Engenharia de Sistemas. Isso é ainda agravado pelo conhecido conservadorismo da Engenharia de Sistemas Espaciais.

Em consequência disso, apesar de há muito tempo serem usadas várias ferramentas CAD, CAE, EDA e CAM em atividades de nível de subsistemas, exemplificadas pelas marcas NASTRAN, ANSYS, AUTOCAD, STK, SINDA, SOLID-WORKS, CATIA, ORCAD, ANALOG WORKBENCH, entre inúmeras outras, nenhuma ferramenta específica de Engenharia de Sistemas é usada.

Os processos e ferramentas de Engenharia de Sistemas do INPE quase nada evoluíram desde que foram introduzidos nas décadas de 1960 e 1980.

6.4 A Terceira Fase da Engenharia de Sistemas no INPE

Nas últimas duas décadas, estimulada pelas exigências dos desenvolvimentos dos sistemas atuais e pela crescente disponibilidade de recursos computacionais, a Engenharia de Sistemas passou e passa por uma nova fase evolucionária que é caracterizada pela incorporação de processos e ferramentas criadas pela Engenharia de *Software* e pela Engenharia de Requisitos (RAMO; ST. CLAIR, 1998). Essa fase evolucionária atual da Engenharia de Sistemas, similarmente ao que também ocorre com a Engenharia de Requisitos, e discutido no Capítulo 2, não é provocada só por eventos e forças da área militar, mas é devida à grande complexidade e integração dos sistemas comerciais atuais como, por exemplo, produtos eletrônicos de entretenimento, telecomunicação, computação pessoal, e indústria automobilística e aeronáutica. Essas forças evolucionárias podem ser resumidas em:

- Onipresença de sistemas comerciais grandes, complexos e integrados;
- Enorme competição por mercados forçando desenvolvimentos com custo e tempo reduzidos;
- Chegar ao mercado antes, pode fazer a diferença entre o sucesso ou fracasso comercial de um produto;
- Desenvolvimentos globalizados, com partes desenvolvidas por diferentes empresas em diferentes partes do mundo, e que exigem rigoroso gerenciamento;
- Rápida evolução tecnológica causando rápida obsolescência e substituição de produtos.

Esse atual interesse em Engenharia de Sistemas se reflete na necessidade de cursos de formação geral e treinamentos específicos em Engenharia de Sistemas. Isso fez com que, no Curso de Engenharia e Tecnologia Espaciais do INPE (ETE), que é usuário de Engenharia de Sistemas, tenha havido nos

últimos anos um movimento voltado para a formação de pessoal em nível de mestrado e doutorado, em Engenharia de Sistemas, similar ao que ocorreu na CNAE, nos anos 1970, e discutido no início deste capítulo.

Os sinais mais importantes desta fase da Engenharia de Sistemas e da Análise de Sistemas no INPE são:

- A sucessiva introdução, desde 1997, de várias disciplinas relacionadas à Engenharia de Sistemas, na Opção Mecânica Espacial e Controle (CMC) do Curso de Engenharia e Tecnologia Espaciais (ETE), incluindo:
 - CMC-219-3 Introdução à Tecnologia de Satélites;
 - CMC-426-4 Introdução à Engenharia de Sistemas;
 - CMC-220-3 Verificação e Validação de Sistemas Reativos;
 - CMC-217-4 Engenharia de Requisitos;
 - CMC-215-4 Modelagem e Simulação em Tempos Virtual e Real;
 - CMC-318-4 Simulação e Controle em Tempos Virtual e Real;
 -
 - CMC-211-4 Modelagem e Tolerância a Falhas em Tempos Virtual e Real;
 - CMC-320-4 Tolerância a Falhas e Controle em Tempos Virtual e Real;
 - CMC-424-4 Controle e Integração/Testes (VVA) em Tempos Virtual ou Real.
- A concomitante criação, desde 1997, do Laboratório de Sistemas, Ambientes, e Controle de Atitude e de Órbita (LabSystems) na Divisão de Mecânica Espacial e Controle (DMC) para P&D, ensino, extensão, demos.
- A criação, em 2008, da Opção Engenharia e Gerenciamento de Sistemas Espaciais (CSE) dentro do Programa de Pós-Graduação em Engenharia e Tecnologia Espaciais, incluindo, adaptando e renomeando aquelas 9 disciplinas da Opção CMC como:
 - CSE-200-4 Introdução à Tecnologia de Satélites
 - CSE-201-4 Introdução à Engenharia de Sistemas Espaciais;

- CSE-207-4 Verificação e Validação de Sistemas Espaciais;
 - CSE-204-4 Engenharia de Requisitos;
 - CSE-202-4 Modelagem de Sistemas Espaciais;
 - CSE-304-4 Simulação de Sistemas Espaciais;
 - CSE-203-4 Confiabilidade e Prevenção de Falhas em Sistemas Espaciais;
 - CSE-305-4 Redundância e Tolerância à Falhas em Sistemas Espaciais;
 - CSE-332-4 Integração, Verificação, Validação e Aceitação de Sistemas Espaciais em Tempos Virtual ou Real.
- A apresentação, em 2010 e 2011, dos trabalhos dos estudantes orientados pelos professores do Curso ETE, no 1º. E 2º. Workshops de Engenharia e Tecnologia Espaciais.
 - A criação, em 2011, do Laboratório de Engenharia de Sistemas (Lab Sys) do LIT, para dar suporte às atividades de montagem, integração e testes do Laboratório de Integração e Testes (LIT) que incluem o desenvolvimento de equipamentos de suporte mecânico e elétrico (Mechanical Ground Support Equipment - MGSE e Electrical Ground Support Equipment - EGSE). O laboratório pretende integrar várias modernas ferramentas de análise, tais como: ferramentas de análise de *stakeholders*, ferramentas de Engenharia de Requisitos, ferramentas de modelagem, entre outras.
 - A realização, de 12 a 16 dezembro de 2011, da 1ª. Semana da Engenharia de Sistemas.
 - Os esforços, desde 2010, para criar uma seção brasileira do International Council on Systems Engineering (INCOSE).
 - A criação do Curso Ciência do Sistema Terrestre - CST em 2009.

6.5 Comentários Finais

A Engenharia de Sistemas no INPE passou por fases de maior e menor importância. A primeira e segunda fases estavam fortemente relacionadas com o desenvolvimento de satélites, respectivamente o satélite SACI e o satélite SCD-1 da MECB. A fase atual está relacionada com a criação da Opção Engenharia e Gerenciamento de Sistemas Espaciais no Programa de Pós-Graduação em Engenharia e Tecnologia Espaciais e do Curso de Ciência do Sistema Terrestre, a utilização de modernas ferramentas de automatização de processos.

O INPE foi introduzido em Engenharia de Sistemas pela General Electric Space Division como parte da preparação para o desenvolvimento do satélite SACI. Essa introdução ocorreu por volta do ano de 1968, somente poucos anos após a Engenharia de Sistemas ter sido formalizada na USAF e introduzida na NASA. A General Electric Space Division era uma importante contratada da USAF e da NASA em desenvolvimentos espaciais civis e militares e, conseqüentemente era obrigada a seguir metodologias rigorosas de Engenharia de Sistemas nesses desenvolvimentos. O treinamento dado ao INPE pela General Electric Space Division foi baseado em um documento de treinamento usado pela própria General Electric no treinamento de seu pessoal. A NASA deve ter servido de ponte entre o INPE e a General Electric Space Division.

Um documento do INPE de 1971 ((PULCHÉRIO et al., 1971)) já alertava para a necessidade de atenção às especificações e a qualidade dos requisitos, apresentava o uso de computadores no planejamento e controle de projeto com o uso de ferramentas PERT e gráficos de vários tipos, e dava indicações do uso de ferramentas computacionais no gerenciamento de informações. Um documento do INPE de 1971 (CNAE, 1971c) apresentava um estudo preliminar de um banco de dados para automatização do gerenciamento de informações. Não foram encontradas referências sobre rastreabilidade de requisitos em qualquer desses documentos.

A dissertação de mestrado de Fernandes (1987), na qual se baseiam os documentos de nível de sistemas da MECB, chamava a atenção para a qualidade das especificações e características apropriadas dos requisitos e sugeria a sistematização do processo de preparação de especificações de requisitos. Não foram encontradas referências sobre rastreabilidade de requisitos nessa dissertação apesar de que, como será visto no próximo Capítulo, a rastreabilidade de requisitos de *software* era exigida no documento de referência do plano de garantia do produto da MECB.

Em várias oportunidades o INPE foi disseminador de conhecimentos de Engenharia de Sistemas para órgãos do governo, empresas e parceiros nacionais e internacionais em projetos de satélite, mas houve pouca evolução nos processos e ferramentas de Engenharia de Sistemas usadas no desenvolvimento de satélites do INPE a partir do final da década de 1980.

7 A EVOLUÇÃO DAS EXIGÊNCIAS DE RASTREABILIDADE NO DESENVOLVIMENTO DE SATÉLITES DO INPE

Este Capítulo analisa alguns documentos dos programas MECB, CBERS 1 & 2 e CBERS 3 & 4 com o objetivo de estudar a evolução dos processos de preparação e armazenamento de requisitos e das exigências de rastreabilidade de requisitos nesses programas.

7.1 As Origens do Sistema de Documentação dos Programas de Desenvolvimento de Satélite do INPE

Os documentos dos programas de desenvolvimento de satélite do INPE geralmente apresentam, nas páginas iniciais, uma lista indicando os seus documentos de referência. Como essa regra não é seguida rigorosamente, em alguns casos tem de ser feito um trabalho de comparação entre documentos para determinar sua origem.

São visíveis os sinais da herança dos programas de ICBM da USAF da série AFSCM-375 na documentação do Programa CBERS 3 & 4. Essa herança é percebida pelas semelhanças na estrutura, nos tópicos, no conteúdo e na terminologia dos documentos de especificação do Programa CBERS 3 & 4 com o documento AFSCM-375-5 da USAF. Essa herança se explica pelo fato de a documentação do programa CBERS derivar de documentos dos programas MECB, SPOT e ESA, todos influenciados pela NASA e, através desta, pelos programas de ICBM da USAF, conforme descrito por Johnson (2002a). Além da herança de programas existentes nos anos 1980, há alguns documentos recentes do Programa CBERS 3 & 4 que derivam diretamente de documentos recentes do sistema de normas ECSS (European Cooperation for Space Standardization) da European Space Agency (ESA).

Para permitir uma comparação, a Tabela 7.1 mostra uma parte do índice de conteúdo de três documentos: do AFSCM 375-5, da USAF, do documento A-ERC-0001 do Programa MECB e do documento RB-DCS-0001 do Programa CBERS 3 & 4. É importante esclarecer que o documento AFSCM 375-5 é o

único disponível e não é uma especificação de Design and Construction, mas um documento de gerenciamento.

Tabela 7.1 - Comparação de Documentos da USAF, MECB e CBERS 3 & 4

AFSCM 375-5 USAF		A-ERC-0001 MECB		RB-DCS-0001 CBERS 3 & 4	
3.3	DESIGN AND CONSTRUCTION	3	GENERAL SPECIFICATIONS	3	GENERAL SPECIFICATIONS
3.3.1	GENERAL DESIGN FEATURES	3.1	ENGINEERING STANDARDS	3.1	ENGINEERING STANDARDS
3.3.2	SELECTION OF SPECIFICATIONS AND STANDARDS	3.2	INTERFACE CONTROL	3.2	INTERFACE CONTROL DATA
3.3.3	MATERIAL, PARTS AND PROCESSES	3.3	MATERIALS AND COMPONENTS	3.3	INTERCHANGEABILITY
3.3.4	STANDARDS AND COMMERCIAL PARTS	3.4	PRODUCTIONS REQUIREMENTS HARDWARE	3.4	RELIABILITY
3.3.5	MOISTURE AND FUNGUS RESISTANCE	3.5	INTERCHANGEABILITY	3.5	REDUNDANCY
3.3.6	CORROSION OF METAL PARTS	3.6	RELIABILITY	3.6	SAFETY
3.3.7	INTERCHANGEABILITY AND REPLACEABILITY	3.7	REDUNDANCY	3.7	MAINTAINABILITY
3.3.8	WORKMANSHIP	3.8	MAINTAINABILITY	3.8	TEST POINTS
3.3.9	ELECTROMAGNETIC INTERFERENCE	3.9	TEST POINTS		
3.3.10	IDENTIFICATION AND MARKING	3.10	SAFETY		
3.3.11	STORAGE	3.11	WORKMANSHIP		
		3.12	IDENTIFICATION OF PRODUCT		
		3.13	ENVIRONMENTAL CONDITIONS		
		3.14	PREPARATION FOR DELIVERY		

7.2 O Processo de Documentação nos Programas Espaciais do INPE

Em meados da década de 1980, quando o INPE iniciou o Programa MECB, estava em curso no Brasil a revolução provocada pelo uso de microcomputadores na automatização de tarefas comuns de escritório. Isso fez com que uma parte da documentação da MECB tenha sido manualmente datilografada em formulários de papel e uma outra parte tenha sido digitada por dos editores de texto existentes na época, como “WordStar”, “WordPerfect”, e depois datilografada eletronicamente em formulários de papel. Em ambos os casos, os originais dos documentos consistiam de documentos de papel.

O Programa CBERS 1 & 2 foi quase contemporâneo com a MECB e usou a estrutura e os processos de preparação de documentação criados pela MECB, com a diferença de ter havido uma maior fração de documentos editados eletronicamente. Os originais dos documentos do Programa CBERS 1 & 2 também eram documentos de papel.

O uso generalizado de microcomputadores no Programa CBERS 3 & 4 fez com que a documentação do Programa CBERS 3 & 4 tenha sido preparada em editores de texto avançados e sem a necessidade de impressão de originais em formulários. Os originais dos documentos passaram a ser os arquivos gerados pelos editores de texto. Cópias são feitas do arquivo eletrônico, sem haver necessidade de impressão de documentos. Exceção a esse procedimento são os relatórios emitidos em campo cujos originais são manuscritos em formulários de papel que, para arquivamento, são digitalizados em um original eletrônico.

Conforme descrito no documento R-MNG-1023 - Procedimento para Aprovação e Modificação de Documentos do Programa CBERS, versão 0, de 9 de julho de 2005 (BARBOSA; RABAY, 2005), que descreve o procedimento usado para tratamento e modificação nos documentos, todos os documentos eletrônicos do Programa CBERS 3 & 4 são arquivados no sistema SEGI (Sistema Eletrônico de Gerenciamento de Informações).

Algumas observações devem ser destacadas:

- 1) Os editores de texto eletrônicos têm reduzidos recursos de rastreamento de informações;
- 2) Não foi encontrado no arquivo do Programa CBERS 3 & 4 um plano de gestão de configuração equivalente ao A-CFG-0001 Plano de Gestão da Configuração da MECB ((INPE, 1986), 1986). O documento do Programa CBERS 3 & 4 que mais se aproxima de um plano de gestão da configuração é o R-MNG-1023 Procedimento para Aprovação e Modificação de Documentos do Programa CBERS (BARBOSA; RABAY, 2005).
- 3) Apesar do nome, não é possível usar o sistema SEGI no Programa CBERS 3 & 4 no gerenciamento de informações porque as informações do programa CBERS 3 & 4 estão na forma de documentos eletrônicos. O SEGI somente pode ser usado como sistema de arquivamento de documentos de forma similar à feita com documentos de papel.

7.3 A Evolução da Rastreabilidade nos Programas de Desenvolvimento de Satélites do INPE

Foi feito um estudo para investigar a origem e a evolução dos processos de rastreamento nos Programas MECB, CBERS-1 & 2 e CBERS 3 & 4. Esse estudo compreendeu as seguintes etapas:

- Um levantamento da ocorrência do termo “traceability” nos documentos desses programas;
- Um levantamento da origem dos documentos onde ocorre o termo “traceability”;
- Uma comparação entre os diferentes tipos de “traceability” descritos nos documentos desses programas.

a) Levantamento da ocorrência do termo “traceability” nos documentos

Pesquisando o termo “traceability” nos documentos dos Programas MECB, CBERS 1 & 2 e CBERS 3 & 4 encontrou-se sua ocorrência nos seguintes programas e documentos:

- No Programa MECB, no documento:
 - A-GQL-0006 MECB Product Assurance Plan, versão 2, de 29 de abril de 1992 (PASCHOTTO; RABAY, 1990).
- No Programa CBERS 1 & 2, no documento:
 - C-PAD-003 Basic Requirements for Contractor’s Product Assurance Plan of CBERS Spacecraft and Associated Documents, versão 4, de 19 de abril de 1990 (FERNANDES et al., 1989).
- No Programa CBERS 3 & 4, nos documentos:
 - RB-PAD-0002 CBERS 3 & 4 Product Assurance Requirements, versão 2, de 5 de novembro de 2005 (INPE, 2004c) e,

- RB-MNG-1022 CBERS-3 & 4 Verification Guidelines, versão 0, de 2 de junho de 2005 (BRANCO, 2005).

b) Levantamento da origem dos documentos onde ocorre o termo “traceability”

Analisando-se a origem dos documentos do item anterior constata-se:

- No documento A-GQL-0006 MECB Product Assurance Plan (versão 2 de 29 de abril de 1992) do Programa MECB

Apesar desse documento não apresentar uma lista de documentos de referência, há claras indicações que ele é baseado no documento ESA–PSS-01-0 (ESA, 1981), Basic Requirements for Product Assurance for ESA Spacecraft and Associated Equipment, publicado em 1981 e existente no arquivo da biblioteca do INPE (ESA, 1981). Comparando-se os dois documentos pode-se dizer que o documento da MECB é uma simplificação do documento da ESA.

Observa-se também que, apesar de o Programa MECB ter se iniciado vários anos antes do Programa CBERS 1 & 2, este documento foi publicado somente em 26 de outubro de 1990, aproximadamente dois anos após o documento equivalente do Programa CBERS 1 & 2, publicado em 28 de outubro de 1988.

- No documento C-PAD-003 Basic Requirements for Contractor’s Product Assurance Plan of CBERS Spacecraft and Associated Documents, versão 4, de 19 de abril de 1990, do Programa CBERS 1 & 2 (FERNANDES et al., 1989)

Este documento também deriva do documento ESA–PSS-01-0, Basic Requirements for Product Assurance for ESA Spacecraft and Associated Equipment (ESA, 1981), citado na lista de documentos de referência. Há grande similaridade entre esses documentos, percebida claramente pela semelhança entre seus títulos. Conforme comentado acima, apesar do

Programa CBERS 1 & 2 ser posterior ao Programa MECB, este documento foi publicado dois anos antes do documento equivalente do Programa MECB.

- No documento RB-PAD-0002 CBERS 3 & 4 Product Assurance Requirements, versão 2, de 5 de novembro de 2005 (INPE, 2004c) do Programa CBERS 3 & 4

Esse documento, a despeito da diferença de título, é muito similar ao documento equivalente do Programa CBERS 1 & 2 e, conseqüentemente também deriva do documento ESA-PSS-01-0 Basic Requirements for Product Assurance for ESA Spacecraft and Associated Equipment (ESA, 1981).

- No documento RB-MNG-1022 CBERS-3 & 4 Verification Guidelines, versão 0 de 2 de junho de 2005 (BRANCO, 2005).

A lista de documentos de referência informa que esse documento é baseado no documento ECSS-E-ST-10-02 Verification, do sistema de normas ECSS da ESA (ESA, 2009). Há grande similaridade entre esses documentos.

c) Comparação entre os diferentes tipos de “traceability”

A Figura 7.1 apresenta um sumário dos parágrafos onde ocorre o termo “traceability” nos documentos estudados. Comparando-se a ocorrência desse termo nos documentos, pode-se afirmar que:

- O documento A-GQL-0006 da MECB tem o termo “traceability” somente relacionado com informações de fabricação e teste de *hardware*;
- O documento C-PAD-003 tem o termo “traceability” relacionado com informações relativas à qualidade de fabricação e teste de *hardware* e com qualidade de *software*;
- Como o documento A-GQL-0006 tem a mesma origem do documento C-PAD-003, era de se esperar no documento A-GQL-0006 a ocorrência do termo “traceability” relacionado com qualidade de *software*. No entanto, essa exigência deve ter sido excluída

propositalmente por não haver *software* com funções consideradas essenciais nos satélites da MECB;

- As ocorrências do termo "traceability" no documento RB-PAD-0002 são quase idênticas às do documento C-PAD-003;
- O documento RB-MNG-1022 (BRANCO, 2005) tem o termo "traceability" somente relacionado com verificações.

Conclui-se que houve muita evolução nas exigências de rastreabilidade entre os programas MECB, CBERS 1 & 2 e CBERS 3 & 4, pois:

- No Programa MECB havia somente exigências de rastreabilidade para fabricação de *hardware*;
- No Programa CBERS 1 & 2 havia exigências de rastreabilidade para fabricação de *hardware* e qualidade de *software*;
- No Programa CBERS 3 & 4 há exigências de rastreabilidade para fabricação de *hardware*, qualidade de *software* e relacionada com verificações.

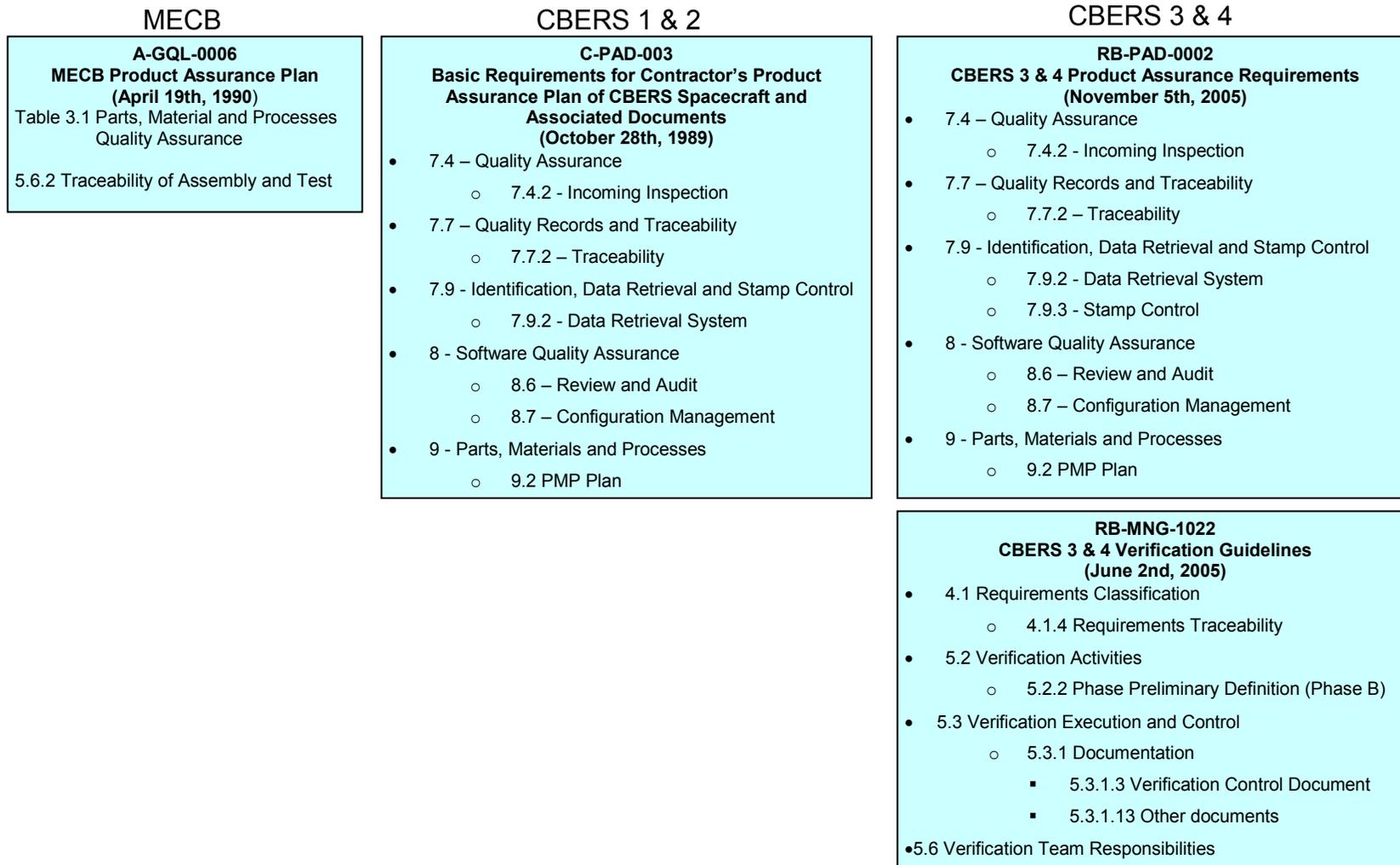


Figura 7.1 - Sumário das Ocorrências do Termo “Traceability” nos Programas MECB, CBERS 1 & 2 e CBERS 3 & 4.

Analisando-se com mais detalhes as ocorrências do termo “traceability” no parágrafo 8 do documento RB-PAD-0002, apresentado resumidamente, por questões de espaço, abaixo:

- 8 - Software Quality Assurance
 - 8.6 – Review and Audit
 - Rastreabilidade de requisitos de *software* através dos métodos de testes, planos de testes, procedimento de testes e resultados de testes;
 - 8.7 – Configuration Management
 - Controle de configuração de *software*;

e analisando-se as ocorrências do termo “traceability” nos parágrafos 4 e 5 do documento RB-MNG-1022, apresentados de forma resumida, por questões de espaço, abaixo:

- 4.1 Requirements Classification
 - 4.1.4 Requirements Traceability

Referindo-se ao uso de rastreabilidade de requisitos para melhorar a completude e a coerência da estratégia de verificação selecionada.
- 5.2 Verification Activities
 - 5.2.2 Phase Preliminary Definition (Phase B)

Referindo-se ao uso de verificação como apoio à alocação e rastreabilidade de requisitos.
- 5.3 Verification Execution and Control
 - 5.3.1 Documentation
 - 5.3.1.3 Verification Control Document (VCD)

Referindo-se à aplicação do VCD (Verification Control Document) para fornecer rastreabilidade do processo de verificação.

- 5.3.1.13 Other documents

Citando outros documentos que podem ser parte do processo de verificação e fornecer rastreabilidade de registros de eventos.

- 5.6 Verification Team Responsibilities

Referindo-se à responsabilidade da equipe de verificação no apoio a alocação e rastreamento de requisitos.

observou-se que há várias instâncias com exigências de rastreabilidade de requisitos em documentos do Programa CBERS 3 & 4.

7.4 Comentários Finais

O Programa CBERS tem heranças no Programa MECB, que foi influenciado pelo programa espacial europeu. Este último, por sua vez, teve influência da NASA, que foi influenciada pelo programa de mísseis ICBM da USAF. A herança do programa de mísseis da USAF é percebida pela similaridade de terminologia e estrutura dos documentos da MECB e CBERS com um documento da série AFSCM 375 da USAF. Houve pouca influência externa no INPE nas últimas duas e meia décadas, com exceção da influência do sistema de normas ECSS da ESA.

Não houve muita evolução na forma de registrar informações nos programas de desenvolvimento de satélites do INPE. Houve somente a substituição de documentos de papel por documentos eletrônicos; e de arquivos em pastas por arquivos eletrônicos. Documentos não são o meio mais eficiente de manter informações pois dificultam pesquisas e relacionamento entre as informações.

Verificou-se que houve aumento nas exigências de rastreabilidade de informações, pois:

- No Programa MECB havia somente necessidade de rastreabilidade de informações de fabricação de *hardware*;
- No Programa CBERS 1 & 2 havia necessidade de rastreabilidade de informações de fabricação de *hardware* e de rastreabilidade de requisitos relacionada com a qualidade de *software* e;
- No Programa CBERS 3 & 4 há necessidade de rastreabilidade de informações de fabricação de *hardware*, de rastreabilidade de requisitos relacionada com a qualidade de *software* e de rastreabilidade de requisitos como parte do processo de verificação.

Processos e ferramentas manuais de rastreamento de fabricação de *hardware* foram criados pelo Programa MECB e foram posteriormente usados nos desenvolvimentos dos satélites CBERS 1 & 2 e CBERS 3 & 4.

Não foram criados processos e ferramentas para atender às exigências de rastreabilidade de requisitos do Programa CBERS. Injustificadamente, o Programa CBERS não mantém rastreabilidade de requisitos exigida em documentos de requisitos de garantia do produto e no guia de verificações.

8 ESTUDO DO SUBSISTEMA DE ENERGIA ELÉTRICA DO SATÉLITE CBERS 3 & 4

Este Capítulo apresenta um exercício de aplicação de uma ferramenta de automatização do gerenciamento de requisitos aplicada à documentação do subsistema de energia elétrica do satélite CBERS 3 & 4.

O Programa CBERS-3 & 4 foi escolhido por ser o maior e mais bem documentado programa de desenvolvimento de satélites do INPE. O Programa CBERS 3 & 4 gerou uma quantidade tão grande de informações que se tornou difícil gerenciá-lo sem o uso de técnicas e ferramentas apropriadas. Um levantamento realizado em agosto de 2010 determinou que o arquivo eletrônico do lado brasileiro do programa é maior do que 10,7 GB, sendo constituído por 1007 pastas e um total de mais de 9582 documentos. A Figura 8.1 apresenta de forma resumida e parcial a grande variedade de tipos de documentos existentes nos vários níveis hierárquicos do projeto.

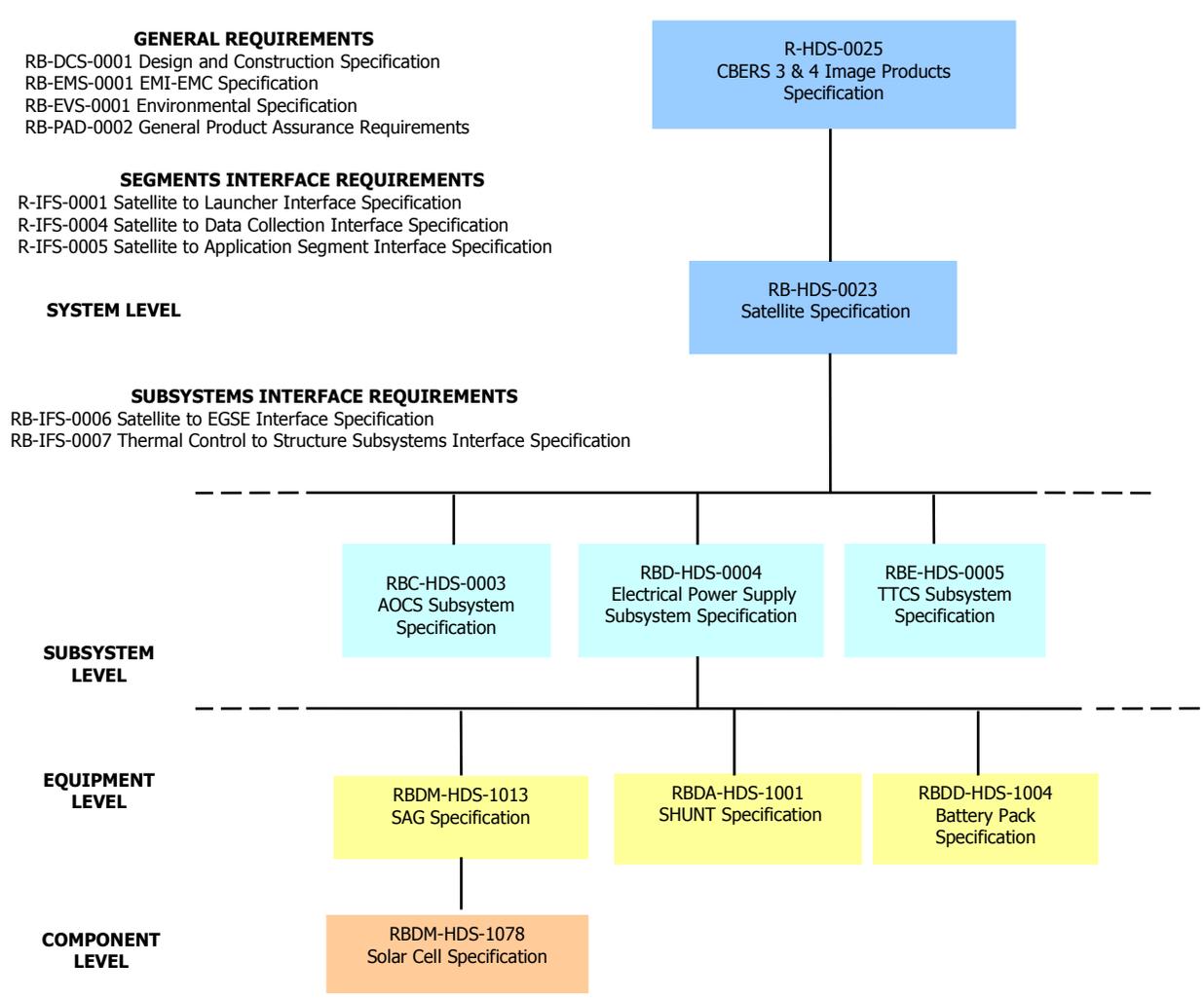
Por ter sua documentação organizada em níveis hierárquicos, o Programa CBERS 3 & 4 é o melhor exemplo disponível para aplicação de uma ferramenta de automatização do gerenciamento de requisitos. No entanto, a aplicação não é um caso real, mas apenas um exercício, pois os documentos já existiam.

Desde meados da década de 1980, quando se iniciaram os desenvolvimentos de satélites no INPE, o grupo de energia elétrica do INPE se capacita e acumula conhecimentos no desenvolvimento desse subsistema de satélites. Tendo sido responsável pelo desenvolvimento desse subsistema nos satélites SCD-1, SCD-2, SCD-2A, SACI-1, SACI-2, SATEC, CBERS-1, CBERS-2 e CBERS-2B, é natural que o subsistema de energia elétrica dos satélites CBERS 3 & 4 também fosse de responsabilidade do INPE.

O subsistema de energia elétrica foi escolhido para o estudo por ser de responsabilidade do INPE e, portanto, haver no INPE toda sua documentação e competência técnica para interpretá-la. Adicionalmente, o autor desta

dissertação gerenciou-o durante 18 anos, tendo sido responsável pela primeira versão de muitos dos documentos analisados.

Somente a parte da documentação relativa aos documentos de especificação que afetam o subsistema de energia elétrica, e que contém exclusivamente requisitos, foi usada no estudo.



- SYSTEMS PLANS**
 R-MNG-0003 Document Coding and Format
 R-MNG-0004 Program Work Breakdown Structure
 R-MNG-1023 Proc. para Aprov. e Mod. de Documentos
 RB-MNG-0002 Development and Test Plan
 RB-MNG-0007 Rules for Marking CBERS Products
 RB-MNG-0009 Master Schedule
 RB-MNG-0010 Product Matrix
- SYSTEMS BUDGETS**
 RB-IFD-0001 Telemetry List
 RB-IFD-0002 Telecommand List
 RB-IFD-0003 Mass Properties
 RB-IFD-0005 Link Budget
 RB-IFD-0006 Alignment Budget
 RB-IFD-0008 Propellant Budget
 RB-IFD-0009 Reliability Budget
 RB-IFD-0015 Mass Budget
 RB-IFD Power Budget
- SYSTEMS REPORTS**
 R-TRP-1010 Payload Operation Modes
 R-TRP-1014 Attitude and Image Correlation
 RB-TRP-0001 CBERS 3 Orbit Design
 RB-TRP-0002 CBERS 3 FMEA
 RB-REV-0019 Satellite PDR Data Package
 RB-REV-0022 PDR Summary Report
- SUBSYSTEM REPORTS**
 RBD-REV-0004 EPSS PDR Data Package
 RBD-REV-1060 CDR do EPSS
 RBD-REV-1061 Relatório da CDR do EPSS
- CONTRACT DOCUMENTS**
 RBD-SOW-1001 DDT Contrato EPSS
 RBDD-SOW-1018 SOW Battery Pack
 RBDM-SOW-1010 DDT Contrato SAG
- INTERFACE CONTROL DOCUMENTS**
 RBDA-IDS-0024 SHUNT Interface Datasheet
 RBDB-IDS-0025 BDR Interface Datasheet
 RBDC-IDS-0026 BHC Interface Datasheet
 RBDD-IDS-0027 Battery Interface Datasheet
 RBDE-IDS-0028 DCDC 1 Interface Datasheet
 RBDM-IDS-0036 SAG Interface Datasheet
- DESIGN DRAWINGS**
TEST PROCEDURES
TEST REPORTS
INSPECTION REPORTS
NONCONFORMANCE REPORTS
FAILURE REPORTS
DEVIATION
WAIVERS

Figura 8.1 - Documentos do Programa CBERS 3 & 4

8.1 O Subsistema de Energia Elétrica dos Satélites CBERS 3 & 4

De acordo com os documentos RB-MNG-0002 CBERS 3 & 4 Satellite Development and Test Plan (INPE, 2004b) e RB-MNG-0010 CBERS 3 & 4 Satellite Product Matrix (BUENO et al., 2008), o INPE é responsável por fornecer, para integração nos satélites CBERS 3 & 4, os seguintes equipamentos do subsistema de energia elétrica:

- **SAG** - Gerador Solar, cuja função é gerar e fornecer a energia elétrica primária para o satélite;
- **BAT** - Baterias, cuja função é armazenar e suprir energia para o satélite;
- **SHUNT** - Regulador *Shunt*, cuja função principal é estabilizar a tensão primária fornecida ao satélite durante o período de iluminação solar;
- **BDR** - Regulador de descarga da bateria, cuja função principal é estabilizar a tensão primária fornecida ao satélite durante os períodos não iluminados e durante qualquer período em que o consumo do satélite exceda a capacidade do SHUNT;
- **BCHC** – Controlador de carga e aquecimento da bateria, cujas funções principais são gerenciar o estado de carga e a temperatura das baterias;
- **DCDC** – 8 Reguladores de tensão de múltiplas saídas, cuja função principal é estabilizar as tensões secundárias fornecidas ao satélite.

Equipamentos com funções de distribuição e proteção, que geralmente são parte do subsistema de energia elétrica, nos satélites CBERS 3 & 4, por razões históricas, são parte do subsistema de Circuitos de Sistemas (Systems Circuitry), que é de responsabilidade da CAST.

O subsistema de energia elétrica dos satélites CBERS 3 & 4 é de baixa complexidade e com reduzido número de funções. No entanto, seu projeto apresenta algumas dificuldades pelas seguintes razões:

- Executa funções essenciais para o satélite;
- Deve ser isento de pontos de falha simples;

- Deve ter alta confiabilidade, pois sua falha pode provocar cascata de falhas e perda de funções do satélite;
- Não é desligado durante operação em vôo;
- Tem interface com todos os equipamentos elétricos do satélite;
- Tem interfaces com sistemas de solo para alimentar, ligar e desligar o satélite durante testes de integração e nas operações pré-lançamento;
- Tem equipamentos não-eletrônicos;
- Tem equipamentos com elevada dissipação térmica;
- Tem equipamentos de elevada massa;
- Tem equipamento sensível a temperatura;
- Tem equipamento que deve suportar um grande número de ciclos térmicos de grande amplitude;
- Tem equipamento com grande área;
- Tem equipamento que recebe elevada dose de radiação;
- Tem equipamento perecível e que geralmente limita a vida do satélite;
- Tem massa total elevada causando impacto no projeto da estrutura do satélite;
- Tem consumo de potência elevado causando impacto no projeto elétrico do satélite.

O desenvolvimento do subsistema de energia elétrica do CBERS 3 & 4 é um desenvolvimento de baixo risco devido à similaridade deste satélite com os satélites da série anterior, onde o subsistema de energia elétrica operou com sucesso. Além das diferenças no projeto detalhado dos equipamentos, as principais diferenças entre o subsistema de energia elétrica dos satélites CBERS 3 & 4 e os satélites CBERS 1 & 2 se resumem a:

- Separação das funções de gerenciamento de estado de carga e temperatura das baterias no BCHC, que nos satélites anteriores ficavam na caixa do BDR com simplificação da cablagem e divisão da massa em dois equipamentos;
- Grande redução de volume e massa dos equipamentos eletrônicos, possibilitada pela evolução na tecnologia de encapsulamento e

montagem de componentes eletrônicos, de montagem PTH (Plated Through Holes) para SMD (Surface Mounted Devices);

- Modificações importantes no circuito de comutação de redundâncias dos conversores DC/DC, de chaveamento de redundâncias por relés para chaveamento através de MOSFET;
- Substituição de capacitores por nova geração de capacitores plásticos;
- Substituição do circuito integrado do controle de chaveamento;
- Modificações em circuitos que processam telemetrias.

O desenvolvimento dos equipamentos do subsistema de energia elétrica dos satélites CBERS 3 & 4 foram contratados da indústria brasileira, sendo que os equipamentos com funções eletrônicas (SHUNT, BDR, BCHC e conversores DC/DC) foram reunidos em um contrato, e os equipamentos BAT e SAG são objetos de dois outros contratos.

O documento que descreve o trabalho a ser realizado pela contratada para o fornecimento dos equipamentos do subsistema de energia elétrica é o RBD-SOW-1001/01 - Descrição Detalhada do Trabalho de Desenvolvimento, Fabricação e Testes de Equipamentos do Subsistema de Suprimento de Energia Elétrica dos Satélites CBERS 3 e 4 de 14 de setembro de 2004 (INPE, 2004a).

8.2A Documentação do Subsistema EPS do Satélite CBERS 3 & 4

Os requisitos do subsistema de energia elétrica e de seus equipamentos são registrados em documentos de especificações. A Figura 8.2 apresenta de forma parcial a árvore de especificações do satélite CBERS 3 & 4 com destaque para o subsistema de energia elétrica. Por limitações de espaço, não estão representadas na referida figura as especificações dos subsistemas da carga útil e, para o subsistema de energia elétrica, as especificações de nível inferior a equipamento.

Conforme descrito no documento R-MNG-0004 CBERS 3 & 4 Work Breakdown Structure (KONO; BUENO, 2004), que mostra a estrutura de divisão do

trabalho do Programa CBERS 3 & 4, o pacote de arquitetura elétrica RB00E110 é de responsabilidade compartilhada entre Brasil e China e, o pacote RBD0M100 do subsistema EPS é de responsabilidade do Brasil. Portanto, a especificação do subsistema de energia elétrica é de responsabilidade compartilhada entre os arquitetos elétricos, chinês e brasileiro, e as especificações para os equipamentos do subsistema de energia elétrica são de responsabilidade somente do grupo de energia elétrica do INPE. A especificação do subsistema de energia elétrica atende requisitos do satélite e impõe requisitos para os equipamentos que compõem o subsistema de energia elétrica.

Os documentos de nível igual e superior a subsistemas são necessariamente redigidos em Inglês para poderem ser entendidos pelos lados chinês e brasileiro do projeto. Os documentos de nível inferior a subsistemas poderiam ser redigidos em Português, mas foram redigidos em Inglês pelos motivos:

- a) Manter compatibilidade com a terminologia dos documentos de nível superior (obrigatoriamente redigidos em Inglês);
- b) Haver dificuldade de traduzir termos técnicos;
- c) Ser a língua inglesa mais precisa que a portuguesa para textos técnicos.

As técnicas usadas na preparação das especificações do subsistema de energia elétrica do satélite CBERS 3 & 4 são as mesmas usadas desde os anos 1980, i.e., um formato padrão (template) de documento, que tem raízes nos documentos da MECB. Os documentos foram redigidos em linguagem natural, mas devido à sua importância como parte dos contratos, foi dada grande atenção à qualidade do texto, ao entendimento e à eliminação de erros.

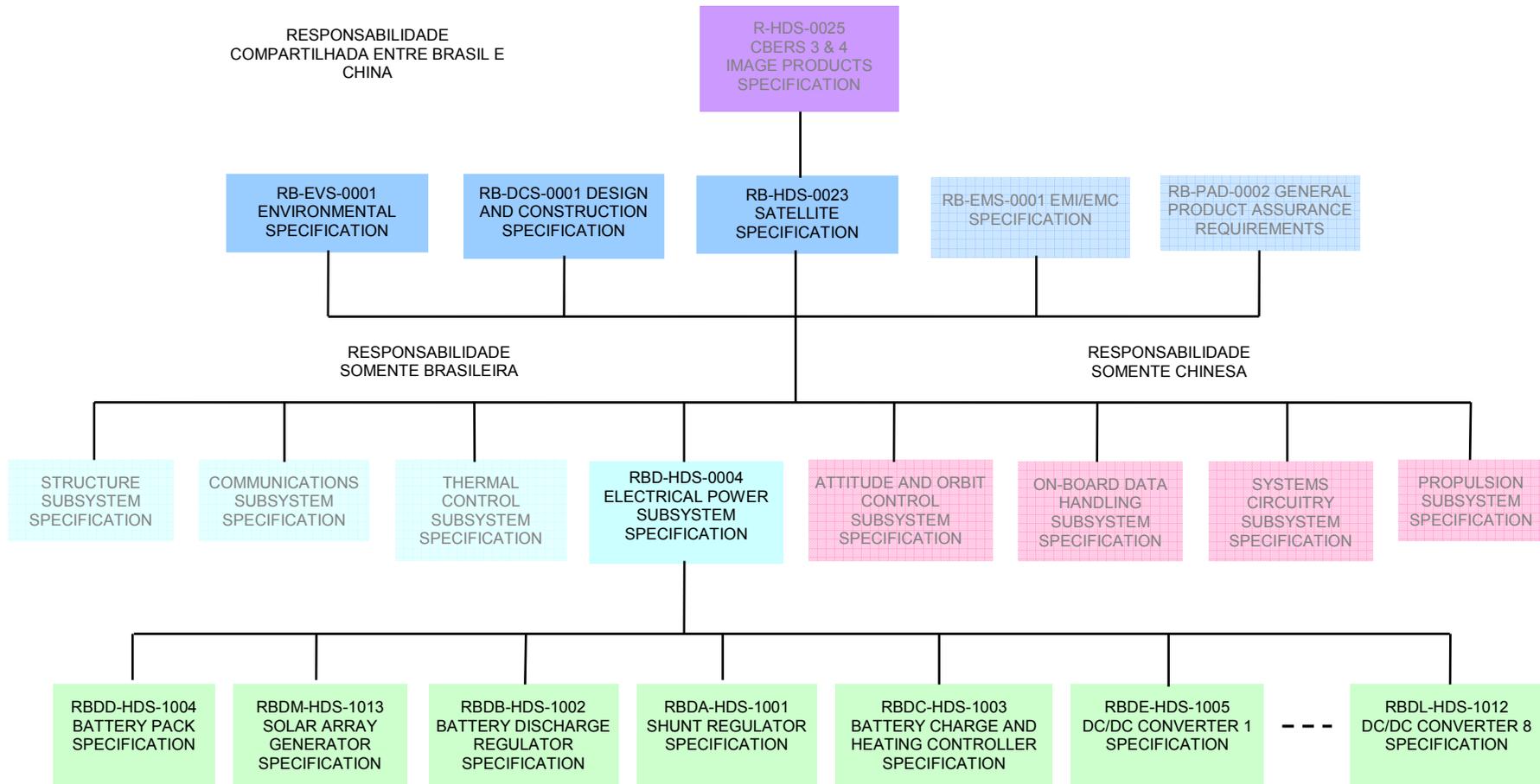


Figura 8.2 - Parte da Árvore de Especificações do Satélite CBERS 3 & 4.

8.3 Limitações do Estudo

Conforme já citado no Capítulo 3, este estudo tem as seguintes limitações:

- a) Uma ferramenta de automatização de rastreamento de requisito foi usada sem que se tenha feito um treinamento formal no seu uso. Portanto, parte de seus recursos não foram utilizados por serem desconhecidos.
- b) O estudo se baseia em documentos preexistentes e preparados sem preocupação com a rastreabilidade de requisitos. Portanto, os documentos tiveram de ser alterados manualmente, antes de serem introduzidos na ferramenta de automação, de forma a tornar os requisitos:
 - Unicamente identificáveis - A ferramenta apontará erro se houver múltiplos requisitos com uma mesma identificação;
 - Indivisíveis - Não há sentido em rastrear requisitos se houver mais que um requisito em mesmo identificador.
 - Rastreáveis – São incluídas referências de cobertura que informam o rastreamento.
- c) Como todas as ferramentas conhecidas, a ferramenta usada no estudo não é automática, e todas as informações de rastreamento tiveram que ser incluídas manualmente durante a preparação dos documentos;
- d) Erros existentes nos documentos não foram corrigidos e foram usados como exemplo de problemas que podem ser detectados e destacados pelo uso de ferramenta;
- e) Alguns documentos não puderam ser usados no estudo porque somente existiam em formatos incompatíveis com a ferramenta ou em formatos

que não podiam ser alterados, e.g., documentos digitalizados em *scanners*.

- f) O arquivo de documentos do Programa CBERS 3 & 4 armazenado no SEGI não tem informações de rastreabilidade de requisitos. Portanto estas foram criadas para poderem ser usadas no estudo.

Dentre os muitos processos de rastreamento de requisitos foi escolhido um processo que atende aos requisitos de rastreabilidade de requisitos do parágrafo 8.6 do documento RB-PAD-0002 CBERS 3 & 4 Product Assurance Requirements (INPE, 2004c) e dos parágrafos 4.1.1, 4.1.2 e 4.1.4 do documento RB-MNG-1022 CBERS 3 & 4 Verifications Guidelines (BRANCO, 2005) e permite o uso de uma ferramenta comercial de gerenciamento de requisitos.

8.4 A Ferramenta Requirements Gateway

A ferramenta de gerenciamento de requisitos usada no estudo é a Requirements Gateway, comercializada pela National Instruments Corporation.

A Requirements Gateway é resultado de um acordo entre a National Instruments Corporation e a Geensoft, proprietária da ferramenta Reqtify. A Requirements Gateway é uma modificação da ferramenta Reqtify que a tornou compatível com os produtos da National Instruments Corporation citados acima. Desde junho de 2010 a Geensoft é parte da Dassault Systèmes, também proprietária das conhecidas marcas CATIA, SolidWorks, entre outras.

A Requirements Gateway é uma ferramenta capaz de trabalhar com requisitos em documentos e em banco de dados; e é capaz de rastrear requisitos em uma grande variedade de formatos de arquivos de documentos, em programas escritos em linguagem de programação visual LabVIEW, LabWindows/CVI, e

em rotinas de testes automáticos TestStand e MATRIXx, produzidos e comercializados pela National Instruments Corporation, conforme mostrado na Figura 8.3.

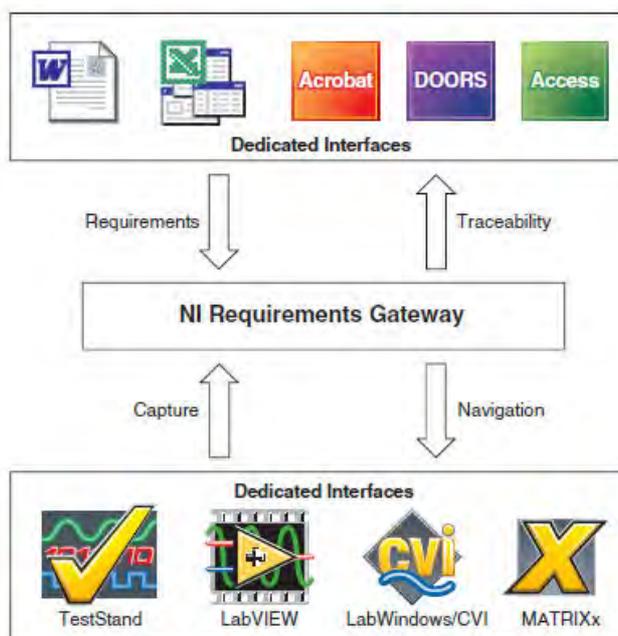


Figura 8.3 - Compatibilidade da Requirements Gateway.

Fonte: Material publicitário da National Instruments.

A versão da ferramenta Requirements Gateway usada no estudo é uma instalação de demonstração da versão 1.1 (build 586) que foi obtida por *download* do site da National Instruments Corporation. A instalação de demonstração da Requirements Gateway tem, pelo período de 30 dias, todas as funcionalidades de uma ferramenta adquirida. Como não foi possível fazer um treinamento formal no uso da ferramenta, o seu uso foi aprendido apenas dos tutoriais, dos manuais disponibilizados pelo fabricante e pela função de “ajuda”. Portanto, há recursos que a ferramenta disponibiliza, mas que não puderam ser usados.

A Requirements Gateway não é uma ferramenta “high-end”, e não tem capacidade de fazer o gerenciamento de requisitos de forma abrangente. No

entanto, é uma ferramenta de baixo custo, fácil de usar, com recursos visuais, e que permite fazer o rastreamento de requisitos em documentos, sendo apropriada para organizações que trabalham com documentos de requisitos em oposição a banco de dados de requisitos.

Obviamente não seria possível apresentar aqui detalhes da ferramenta Requirements Gateway. Portanto somente são apresentadas as funcionalidades importantes para o entendimento do trabalho realizado:

a) Compatibilidade com Formatos de Arquivos de Documentos

Como especificações em formato de documentos de texto são uma forma comum de registrar requisitos, a ferramenta Requirements Gateway foi desenvolvida para ser compatível com arquivos de documentos, e.g., arquivos tipos .txt, .doc, .docx, .pdf entre outros tipos. Sem que o usuário tenha consciência, a Requirements Gateway cria um banco de dados de requisitos e relações, a partir de informações contidas nos documentos. Dentro da Requirements Gateway, os requisitos passam a ser objetos de um banco de dados. Cada objeto desse banco de dados pode ter atributos e informações associadas que ficam somente arquivados na ferramenta. Dessa forma, mantém-se a compatibilidade com a forma mais usual de registrar requisitos (documentos de texto) e as informações adicionais ficam registradas na ferramenta.

b) Arquivo Geral de Documentos do Projeto

Os documentos analisados não precisam ser importados para o computador onde a ferramenta Requirements Gateway está instalada. Os arquivos com os documentos sendo analisados podem estar localizados em qualquer lugar de qualquer computador de uma rede. A ferramenta Requirements Gateway localiza e captura os arquivos a serem analisados, bastando que se informe o caminho de sua localização. Os documentos de projeto ficam arquivados em

um computador de uma rede e, qualquer pessoa, em qualquer outro computador da rede, realiza as análises. Os usuários da ferramenta e dos documentos não possuem cópias dos documentos e acessam diretamente o arquivo dos documentos originais somente para leitura.

c) Requisitos Não-Derivados e Requisitos Derivados

A ferramenta Requirements Gateway define e distingue requisitos não-derivados e requisitos derivados:

- Requisito não-derivado é um requisito diretamente ligado à cobertura de um requisito de um documento de hierarquia superior;
- Requisito derivado é um requisito em determinado nível de hierarquia, que não cobre qualquer requisito em documentos de nível de hierarquia superior.

Essa nomenclatura não é universal e é um pouco confusa porque o que tem a palavra "não" é aquele que "tem associação" com um outro requisito superior. O nome requisito derivado tem sua origem no fato de o requisito poder ser derivado de outros requisitos, similar à definição de requisito derivado dado pela GSFC na Tabela 2.4.

d) Atributos

A Requirements Gateway tem campos para armazenamento de atributos de cada requisito, tais como *assumptions* e *rationales*. Esses atributos podem ser filtrados/escondidos, por exemplo, quando for necessário para produzir cópias impressas das especificações.

e) Tipos de Análises

O Requirements Gateway faz quatro tipos de análises:

- **Management View:** mostra uma visão geral do status do projeto com sumários de número total de requisitos, porcentagem de requisitos cobertos, número de requisitos derivados, número de requisitos não-derivados, etc;
- **Coverage Analysis:** mostra uma visão da cobertura dos requisitos de um nível hierárquico pelos requisitos de nível hierárquico inferior;
- **Impact Analysis:** mostra, simultaneamente, todos os requisitos relacionados com um requisito selecionado, em um nível imediatamente superior e um nível imediatamente inferior em relação ao nível do requisito selecionado. Mostra o texto dos requisitos nos três níveis hierárquicos, simultaneamente e em uma mesma tela, facilitando a verificação do impacto de mudanças.
- **Graphical View:** mostra graficamente as relações entre todos os requisitos de todos os documentos sendo analisados.

f) Configuração de um Projeto:

Há duas maneiras para definir coberturas quando se configura um projeto na Requirements Gateway, conforme representado na Figura 8.4:

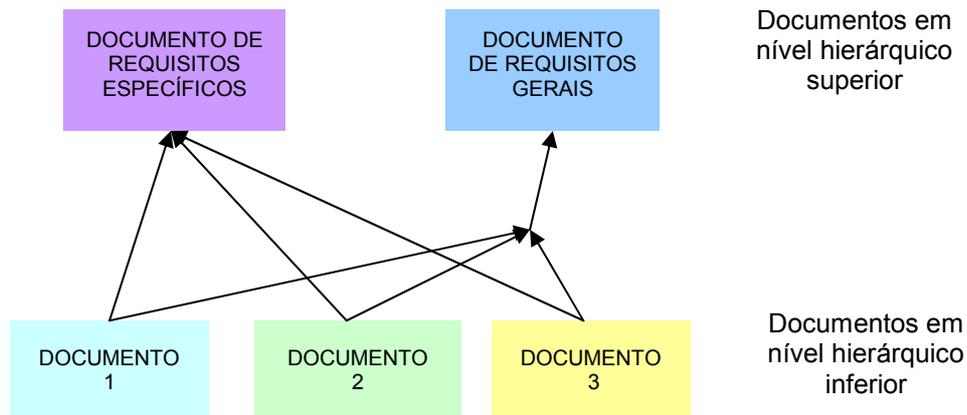


Figura 8.4 - Configuração de um Projeto na Ferramenta Requirements Gateway.

A Figura 8.4 representa a configuração de um projeto com o seguinte significado: no nível superior, tem-se um documento com requisitos específicos e um documento com requisitos gerais; no nível inferior documentos que têm requisitos específicos e requisitos gerais. Os requisitos do documento de hierarquia superior com requisitos específicos devem ser cobertos por pelo menos um requisito em qualquer um dos documentos de hierarquia inferior. Os requisitos no documento hierarquia superior com requisitos gerais devem ser cobertos por pelo menos um requisito em cada um dos documentos de hierarquia inferior.

A cobertura é verificada da seguinte forma:

a) Um requisito no documento de hierarquia superior com requisitos específicos é considerado coberto quando é relacionado com um ou mais requisitos em qualquer documento de hierarquia inferior, como mostrado na Figura 8.5. A porcentagem de cobertura é calculada como a fração coberta dos requisitos de hierarquia superior.

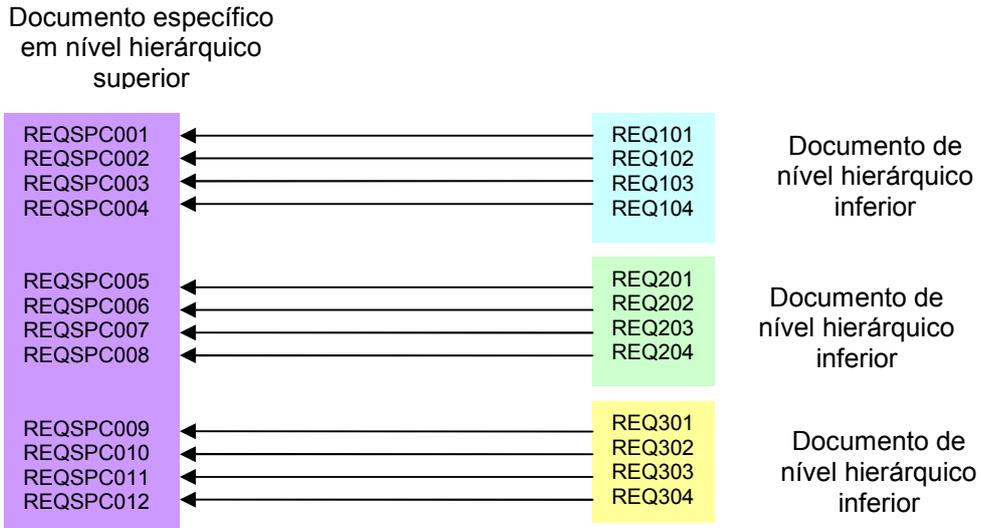


Figura 8.5 - Cobertura para Requisitos Específicos.

b) Um requisito no documento de hierarquia superior com requisitos gerais é considerado coberto quando é relacionado com um ou mais requisitos em cada um dos documentos de hierarquia inferior, como mostrado na Figura 8.6. A porcentagem de cobertura é calculada como a fração coberta de requisitos de hierarquia superior.

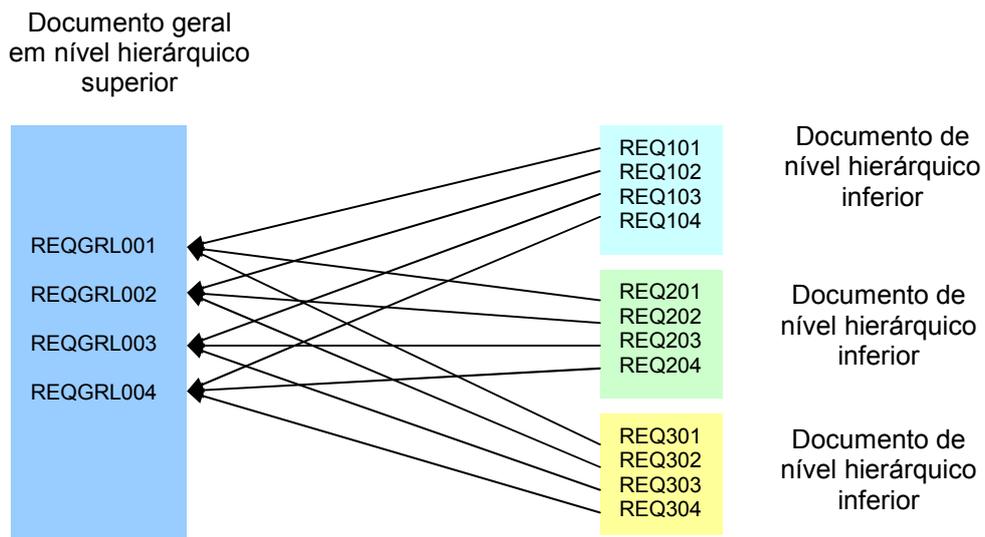


Figura 8.6 - Cobertura para Requisitos Gerais.

8.5 Requisitos nas Especificações dos Satélites CBERS 3 & 4

As especificações do satélite CBERS 3 & 4 não foram redigidas seguindo metodologias da Engenharia de Requisitos. Os requisitos foram redigidos em linguagem natural em língua inglesa e não têm muitas das propriedades que um requisito deve possuir. Em particular, muitos não têm as propriedades necessárias à rastreabilidade, i.e., serem unicamente identificáveis, indivisíveis, e rastreáveis.

Seguem abaixo alguns exemplos de problemas de requisitos nos documentos dos satélites CBERS 3 & 4 e do subsistema EPS que impedem o uso de ferramentas de gerenciamento de requisitos, e comentários críticos sobre sua redação e organização quando observados sob critérios da Engenharia de Requisitos:

- Problema: Requisitos em nível hierárquico inadequado.

Documento: RB-HDS-0023 CBERS 3 & 4 Satellite Specification (BUENO et al., 2005).

Paragraph 3.5.3 Electrical Power Supply Subsystem (EPSS)

- d) SAG power output (EOL) > 2300 W*
- e) Main bus voltage range 28.0 V +/-0.6V*
- f) DC/DC converters efficiency > 73%*
- g) DC/DC voltage regulation +/-1%*
- h) Battery capacity > 90 Ah*
- i) Battery DOD < 20%*

Comentário: Não é correto colocar requisitos de desempenho de nível de equipamento num documento de nível de satélite, pois, além de não ter qualquer sentido, em geral não é possível verificar esses requisitos quando o sistema está integrado. No nível de sistema devem ser colocados apenas requisitos para o sistema, mesmo porque se isto não for feito, limitar-se-á desnecessariamente o subsistema antes da análise de seus requisitos.

- Problema: Requisito não-aplicável

Documento: RBD-HDS-0004 Electrical Power Supply Subsystem Specification (FREIRE et al., 2008).

Paragraph 3.1.4.3.3 Battery Pack Thermal Gradient

The Thermal subsystem shall provide means so that when the whole Batteries (four packs) are installed on the satellite the maximum thermal gradient between packs shall be lower than 5 °C.

Comentário: Esse requisito obviamente não é para o subsistema de energia elétrica, mas para o subsistema de controle térmico.

- Problema: Requisitos não-individualizados e não identificados.

Documento: RB-HDS-0023 CBERS 3 & 4 Satellite Specification (BUENO et al., 2005).

Paragraph 3.5.3 - Electrical Power Supply Subsystem (EPSS)

The EPSS shall convert the sun light energy into electric energy using a Solar Array Generator (SAG), condition the electric energy and to store this energy in batteries.

Comentário: Há três requisitos funcionais para o EPS nesse requisito: gerar a energia elétrica primária usando gerador solar; condicionar a energia elétrica e armazenar a energia elétrica em baterias. Esse requisito não é conciso, não é único e impõe uma solução. Esse tipo de construção impossibilita o rastreamento e dificulta a verificação.

- Problema: Redundância de texto.

Documento: RBDA-HDS-1001 Shunt Specification (FREIRE et al., 2006).

Paragraph 3.2.7 - Failure Modes

The SHUNT shall have no Single Point Failure (SPF) and no single failure shall result in a permanent short circuit of the Main Bus, of the Solar Array Generator and between the Solar Array Generator and the Main Bus .

No single failure of the SHUNT shall cause loss of regulation of the Main Bus.

Comentário: Claramente existe redundância de requisitos no texto.

- Problema: Condições de validade não-especificadas.

Documento: RBD-HDS-0004 Electrical Power Supply Subsystem Specification (FREIRE et al., 2008).

Paragraph 3.2.3 Battery Packs

i) Battery Thermal Gradient

The Battery Pack design shall be such that the maximum thermal gradient between cells of a pack shall be lower than 2 °C.

Comentário: Não estão definidas as condições de validade desse requisito já que o gradiente térmico é dependente de muitas variáveis.

- Problema: Requisito referindo-se a requisito em outro documento.

Documento RBDA-HDS 1001 Shunt Specification (FREIRE et al., 2006).

Paragraph 3.3.6.4 Cabling and Connectors

The SHUNT shall comply with the requirements of AD1) section 6.4 (cabling and shielding) and section 6.5 (connectors).

Comentário: O artifício de não repetir o texto de uma referência era usado para evitar alterações em cascata em vários documentos quando os documentos eram impressos em papel. O uso de documentos no formato eletrônico e ferramentas de automação de rastreamento de requisitos facilitam determinar os impactos de mudanças e fazer as mudanças em todos os documentos que se referem a uma referência alterada.

- Problema: Requisitos não-cobertos por outros requisitos.

Documento RBD-HDS-0004 Electrical Power Supply Subsystem Specification (FREIRE et al., 2008).

Paragraph 3.1.4.6 Electrical Ground Support Equipments (EGSE) Interface

Ground operations of the EPSS during satellite integration and at the launching site shall be performed through the Separation Connector.

The EPSS shall provide the necessary inputs to power the spacecraft and to recharge the batteries via the Electrical Ground Support Equipment.

The interface power lines between EGSE and EPSS shall have protections, situated within the EPSS, to preclude undesirable reverse current paths.

Comentário: Por um erro não detectado, mesmo após muitas revisões, esse requisito de subsistema não está coberto por qualquer outro requisito de equipamento. Esse erro poderia ter causado uma grave não-conformidade nos equipamentos pela ausência de uma importante função de proteção.

8.6 Preparação dos Documentos para Uso da Ferramenta

O sistema SEGI onde estão armazenados os documentos do Programa CBERS é protegido e foi necessária a designação de usuário e senha para poder acessar os documentos. Para agilizar o uso decidiu-se transferir todos os documentos existentes no SEGI relativos ao Programa CBERS 3 & 4 para o computador onde seriam preparados e analisados pela ferramenta, um arquivo de mais de 10 GB. Os documentos foram obtidos via transferência FTP diretamente do sistema SEGI, na única versão disponível de cada documento (geralmente a última versão). Os documentos arquivados no SEGI normalmente estão disponíveis em dois formatos: no formato original em que foi gerado, e.g., .txt. ou .doc, e no formato .pdf.

Devido ao grande número de documentos e a quantidade de trabalho necessário, a preparação dos documentos foi feita por etapas, conforme descrito a seguir:

a) Transformação em arquivo de texto simples

Tentou-se inicialmente usar um documento do arquivo CBERS 3 & 4 no formato .doc. No entanto, quando esse documento foi introduzido na ferramenta Requirements Gateway e analisado, ocorreram vários alertas de erros na ferramenta. Suspeitando que a causa dos erros fosse a existência de

resíduos de formatação dos arquivos antigos recuperados das versões anteriores de editores de texto (Word-3, Word-4, etc), decidiu-se eliminar toda a formatação do arquivo pela transformação em arquivo de texto simples (.txt). Essa decisão não trouxe qualquer prejuízo para o trabalho e, conforme se verificou, trouxe a vantagem de os arquivos no formato de texto simples serem muito mais rapidamente manipulados pela ferramenta (abertos, salvos, fechados etc) do que os arquivos .doc.

b) Isolação de requisitos individuais no texto dos documentos

Um dos problemas que impedem o rastreamento de requisitos nos documentos analisados é a presença de múltiplos requisitos em um único parágrafo identificador. Conseqüentemente, a maior parte das modificações no texto dos documentos consistiu da separação e identificação dos requisitos em requisitos individuais.

Separar os requisitos em requisitos individuais não é uma tarefa fácil, sendo quase tão difícil quanto escrever requisitos. Abaixo é dado um exemplo bastante simples de como um requisito no documento original foi dividido em requisitos individuais para ser introduzido na ferramenta.

- Requisito original (em formato MS-Word):

The EPSS shall provide primary and secondary power to the spacecraft power buses.

- Requisitos preparados para introdução na ferramenta (em formato .txt):

EPS_EPS_REQ11: PRIMARY BUS

<<The EPSS shall provide primary power to the spacecraft power bus.>>

EPS_EPS_REQ12: SECONDARY BUS

<<The EPSS shall provide secondary power to the spacecraft power buses.>>

c) Inclusão de Identificadores:

A Requirements Gateway é programada para automaticamente identificar os requisitos em um texto. A programação de como a Requirements Gateway identifica os requisitos é chamada "type". Essa programação pode ser feita por sinais gráficos especiais ou por seqüência reservada de caracteres. No caso *default* para texto simples a seqüência reservada de caracteres é a seqüência "REQ". Um rótulo identificador do requisito deve anteceder a seqüência reservada e esta deve ser seguida por um número e o sinal gráfico dois pontos (:). O título do requisito deve seguir o sinal gráfico dois pontos (:). O texto do requisito deve seguir imediatamente a identificação e estar limitada pelo sinal gráfico colchetes bicudos duplos << >>, como no exemplo abaixo:

EPS_EPS_REQ1: SOLAR POWER GENERATION FUNCTION

<<a) The EPS shall convert the incident solar power into electrical power through a solar array generator.>>

A Requirements Gateway não interpreta qualquer texto, i.e., ela apenas cria um banco de dados com as informações que ela consegue identificar. Todo o texto não identificável, i.e., que não é requisito, é ignorado.

d) Inclusão de Coberturas

A indicação de cobertura de requisitos também é feita pelo uso de uma seqüência reservada de caracteres que pode ser programada. A seqüência reservada *default* é "Covers:" seguida do identificador do requisito coberto, limitados pelos sinais gráficos colchetes quadrados [], conforme exemplo abaixo:

[Covers: SAT_EPS_REQ1][Covers: SAT_EPS_REQ2]

8.7 Uso da Ferramenta Requirements Gateway com a Documentação do EPS

Os parágrafos seguintes descrevem a utilização da ferramenta Requirements Gateway para rastrear requisitos dos documentos do EPS dos satélites CBERS 3 & 4. A descrição apresentada obviamente só apresenta as características mais importantes da ferramenta e do seu uso.

8.7.1 Configuração do Projeto

A configuração do projeto, tela Configuration, consiste em indicar para a ferramenta o nome e a localização dos arquivos dos documentos do projeto e as relações entre esses documentos. A tela de configuração de projeto, apresentada na Figura 8.7, indica que:

- O documento de requisitos específicos no primeiro nível hierárquico é o Satellite Specification;
- No segundo nível hierárquico está o documento EPSS Specification - em um estudo mais abrangente, nesse nível apareceriam todos os outros documentos de nível de subsistema;
- No terceiro nível hierárquico aparecem todos os documentos de especificação dos equipamentos que compõem o subsistema EPS - devido à grande similaridade entre os conversores DC/DC, somente um dos oito documentos de especificação de conversores DC/DC foi incluído. Até este nível todos os documentos foram produzidos pelo INPE;
- No quarto nível hierárquico aparecem os documentos que cobrem os requisitos específicos do documento SHUNT Specification. Nesse caso são documentos de descrição de projeto detalhado, resultados de análises e procedimento de testes. Esses documentos foram produzidos pela empresa contratada para desenvolvimentos do equipamento

SHUNT. Na data em que foi feita a transferência de arquivos eram esses os documentos disponíveis no SEGI para o equipamento SHUNT;

- Em uma hierarquia separada aparecem no primeiro nível os documentos Design and Construction Specification e Environmental Specification que têm requisitos gerais para todos os equipamentos do subsistema EPS. Nesse nível poderiam aparecer outros documentos de requisitos gerais para equipamentos, como, por exemplo, o EMI/EMC Requirements Specification etc;
- As indicações de relações hierárquicas são feitas pelas setas interligando os documentos. Observa-se que é diferente a indicação de relação para documentos com requisitos específicos e para documentos com requisitos gerais;
- O lado direito da tela apresenta a estrutura dos arquivos do projeto;
- A parte inferior da tela apresenta detalhes dos documentos e a localização do seu arquivo eletrônico.

8.7.2 Descrição das Telas e das Análises da Requirements Gateway

Management View

Na Figura 8.8 é mostrada a tela Management View. Nessa tela temos um resumo do status do projeto com as seguintes informações:

- Número total de documentos no projeto;
- Número total de requisitos no projeto;
- Número total de requisitos derivados;
- Número de requisitos não-cobertos;
- Número de requisitos não-definidos;
- Porcentagem de coberturas para cada documento;
- Informações detalhadas sobre um documento selecionado (número de requisitos, número de requisitos não-cobertos, número de requisitos que são referidos no documento, tipo de arquivos, localização do arquivo etc).

Coverage Analysis

Na Figura 8.9 é mostrada a tela Coverage Analysis. Nessa tela tem-se um resumo das coberturas dos requisitos dos documentos com as seguintes informações:

- A estrutura dos documentos aparece no campo central superior da tela. Pode-se expandir essa estrutura até os requisitos de qualquer documento;
- Ao selecionar um requisito de um documento no campo central superior aparecem, no campo esquerdo, o documento e o requisito que é coberto pelo requisito selecionado; e, no campo direito, todos os documentos e requisitos que estão relacionados com o requisito selecionado. No caso apresentado na Figura 8.9, um requisito de subsistema no campo central cobre um requisito de nível de satélite no campo esquerdo e é coberto por vários requisitos em vários documentos no campo direito.
- Os campos inferiores mostram, simultaneamente, os requisitos em três níveis hierárquicos (*upstream*, *selection*, *downstream*), permitindo comparar os textos dos requisitos nos três documentos;
- A fração de cobertura de cada documento é apresentada na hierarquia inferior sobre o documento da seleção;
- Os atributos dos requisitos podem ser adicionados e examinados nos cantos inferiores da tela;
- Muitas outras informações sobre os requisitos são apresentadas, conforme mostrado na Figura 8.9.

Navigation

O recurso de navegação (Navigation) é apresentada na Figura 8.10. A navegação consiste no recurso de, estando dentro da ferramenta, poder abrir uma janela da aplicação do documento sendo analisado (no caso de documentos .txt, uma janela Notepad) que permite rápida alteração do texto no documento. Ao salvar a alteração e fechar a janela da aplicação, a ferramenta percebe a alteração no arquivo e pergunta se se deseja reanalisar o projeto. Esse recurso auxilia no período de preparação do projeto, mas não pode ser usado por qualquer usuário e quando o documento está sob controle de configuração.

Impact Analysis:

A Figura 8.11 mostra a tela Impact Analysis (Análise de Impacto). Nessa tela têm-se as seguintes informações:

- A estrutura dos documentos aparece no campo central superior da tela. Pode-se expandir essa estrutura até os requisitos de qualquer documento;
- Ao se selecionar um requisito em qualquer documento no campo central superior, aparecem nos campos superiores esquerdo e direito segmentos de árvore de documentos com todos os requisitos no nível hierárquico superior (*upstream*) e inferior (*downstream*) que têm alguma relação com o requisito selecionado. Clicando-se em qualquer dos requisitos nesses segmentos de árvore de documentos, o requisito selecionado é mostrado no campo inferior. Podem ser visualizados simultaneamente os textos de um requisito em cada um dos três níveis (*selection*, *upstream* e *downstream*);
- Os atributos dos requisitos podem ser adicionados ou examinados nos cantos inferiores;
- Essa tela permite uma rápida visualização e comparação dos textos.

Graphical View

A tela gráfica (Graphical View) apresenta as relações entre todos os requisitos de todos os documentos, graficamente, ligando-os por linhas. Na tela gráfica é possível ver mais do que três níveis de hierarquia simultaneamente, sendo este número somente limitado pela resolução do texto na tela, mas se tem detalhes de apenas um requisito destacado. Ao destacar um requisito, destacam-se todos os requisitos a ele relacionados. Como pode se observar pela Figura 8.12, essa tela fica congestionada para projetos grandes; nesse caso é possível filtrar as informações na tela para apresentar somente os requisitos relacionados com o requisito selecionado, eliminando-se os outros requisitos irrelevantes para a análise. As funções *zoom* e *panning* permitem visualizar toda a imagem ou qualquer fração dela.

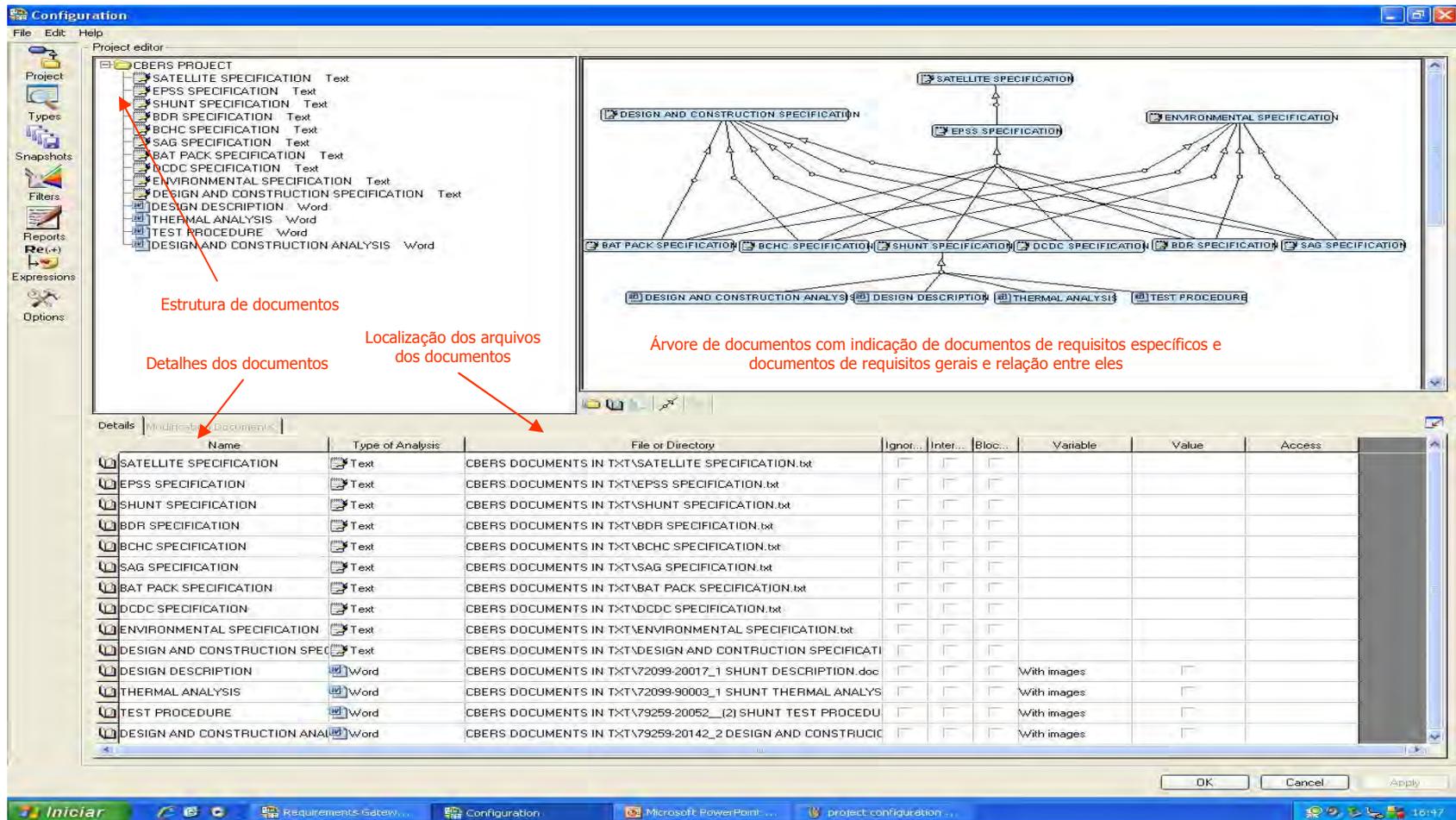


Figura 8.7 - Tela de Configuração do Projeto na Requirements Gateway.

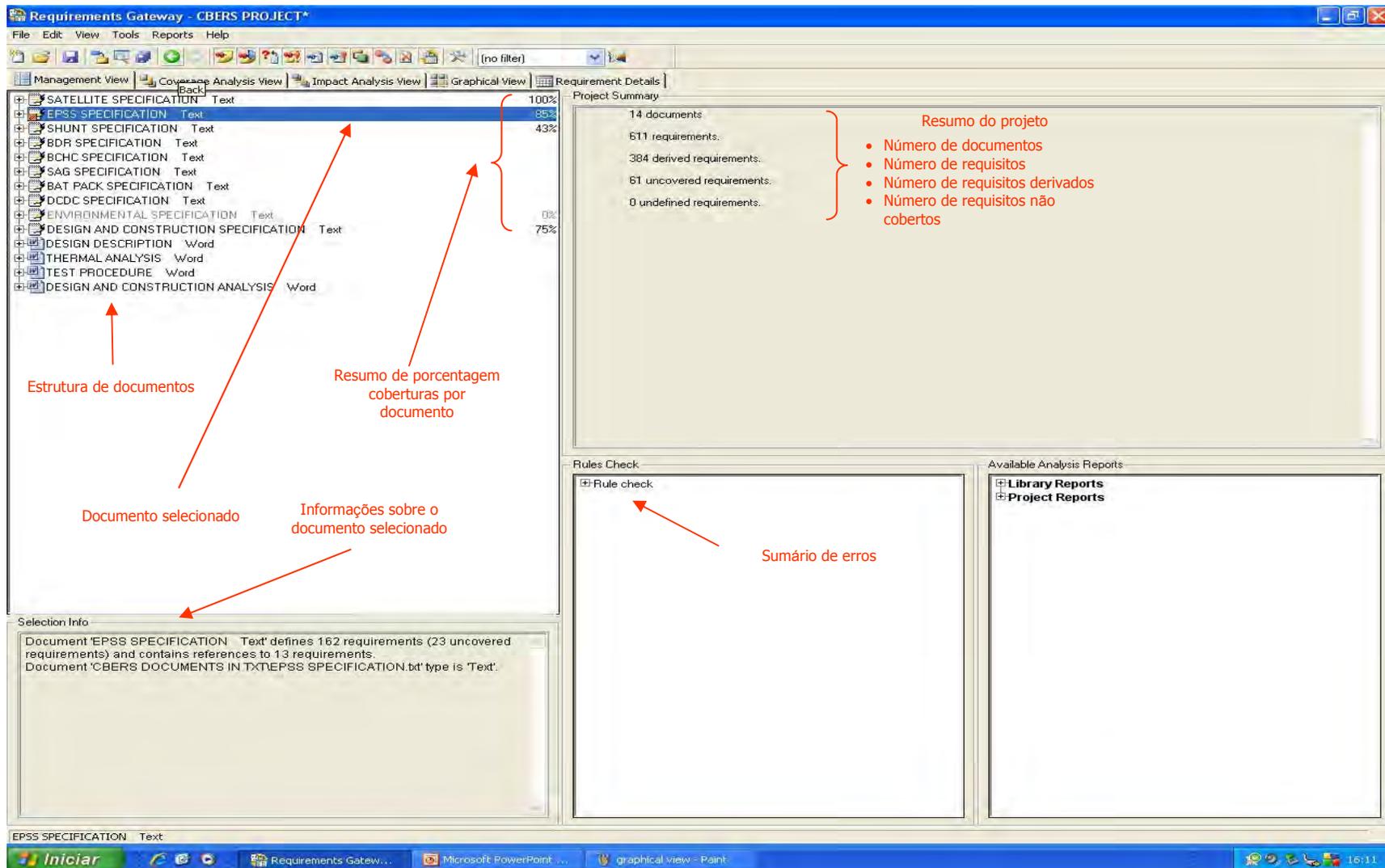


Figura 8.8 - Tela de Visão Gerencial da Requirements Gateway.

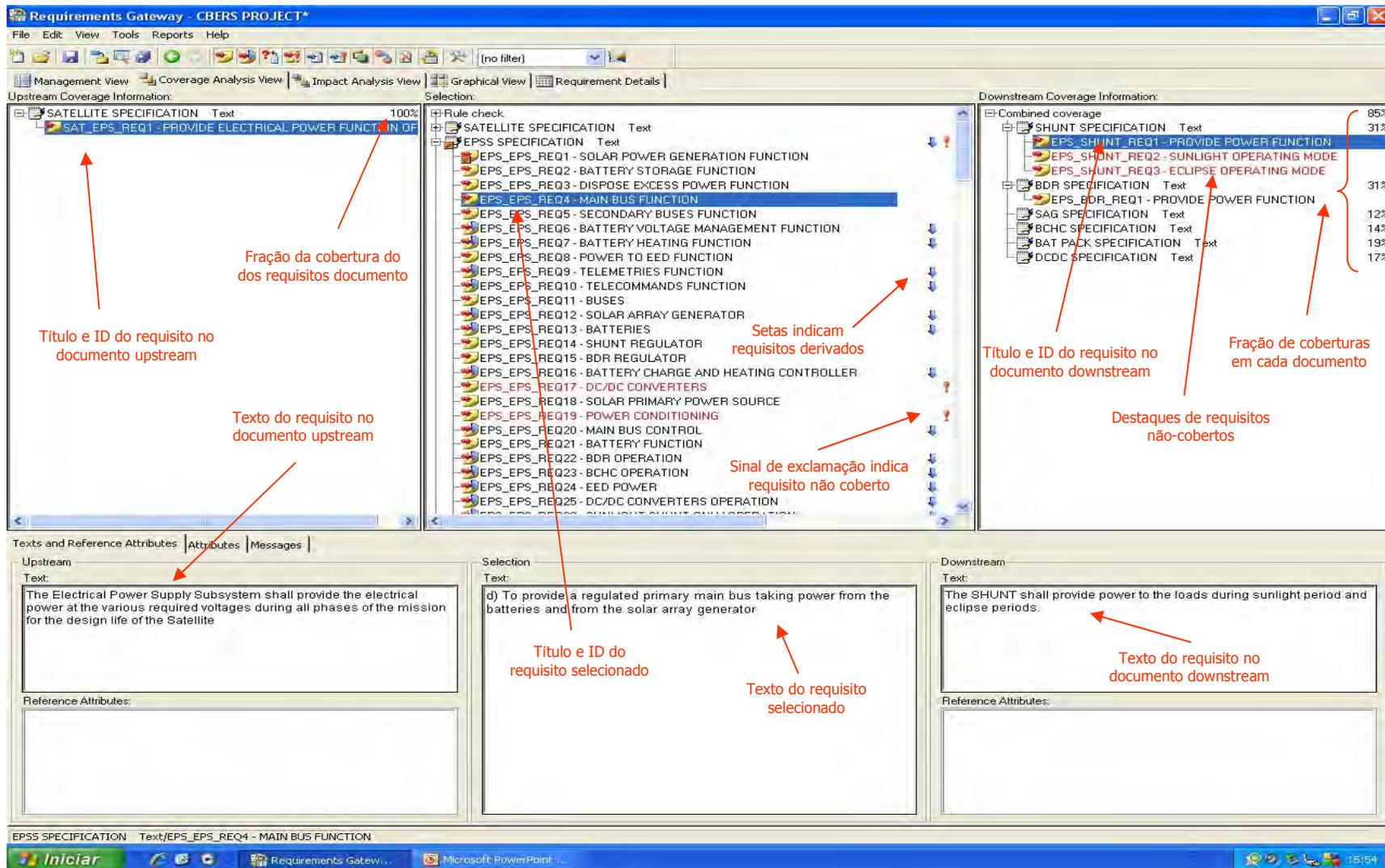


Figura 8.9 - Tela de Análise de Cobertura da Requirements Gateway.

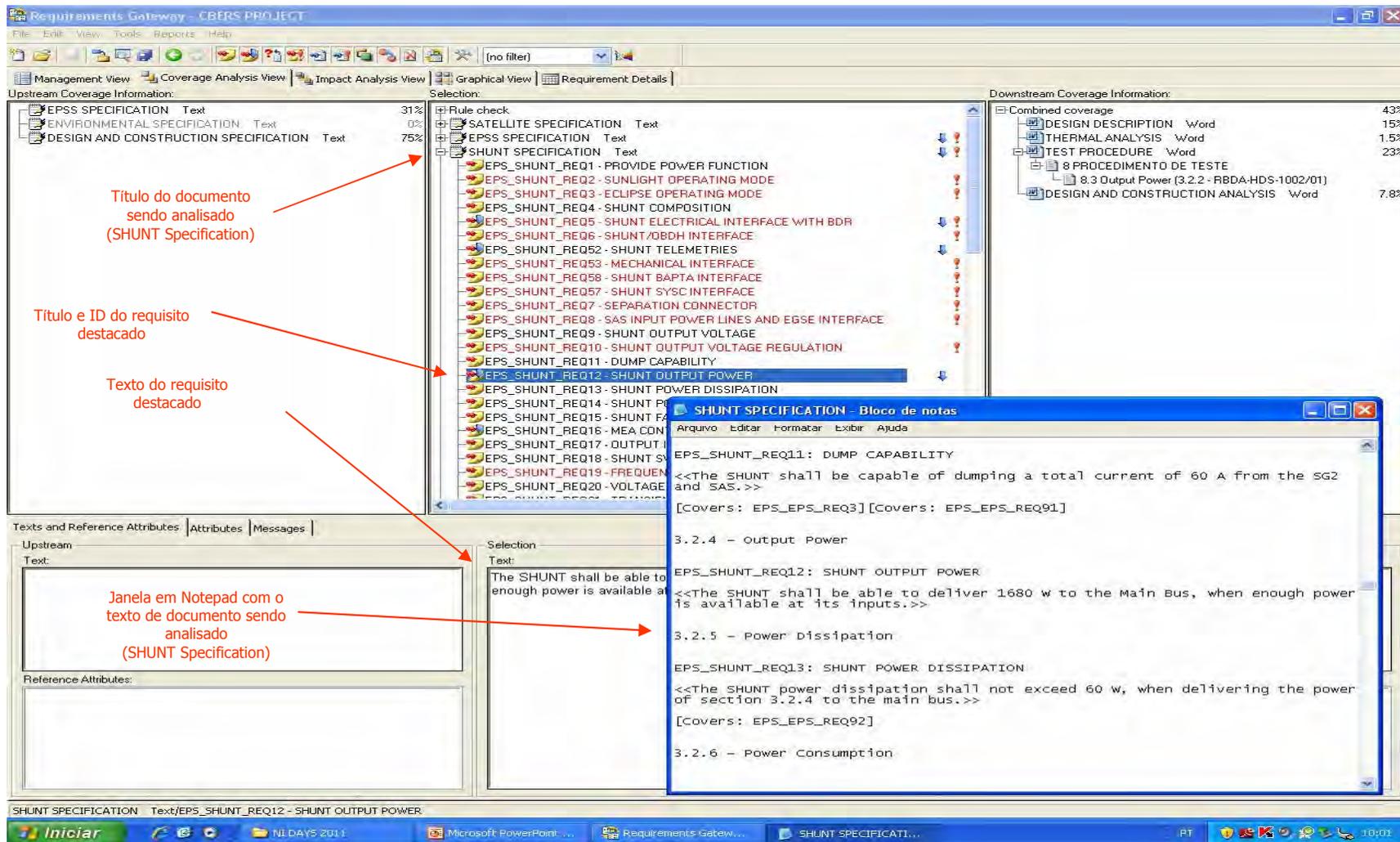


Figura 8.10 - Navegação entre a Ferramenta Requirements Gateway e um Documento.

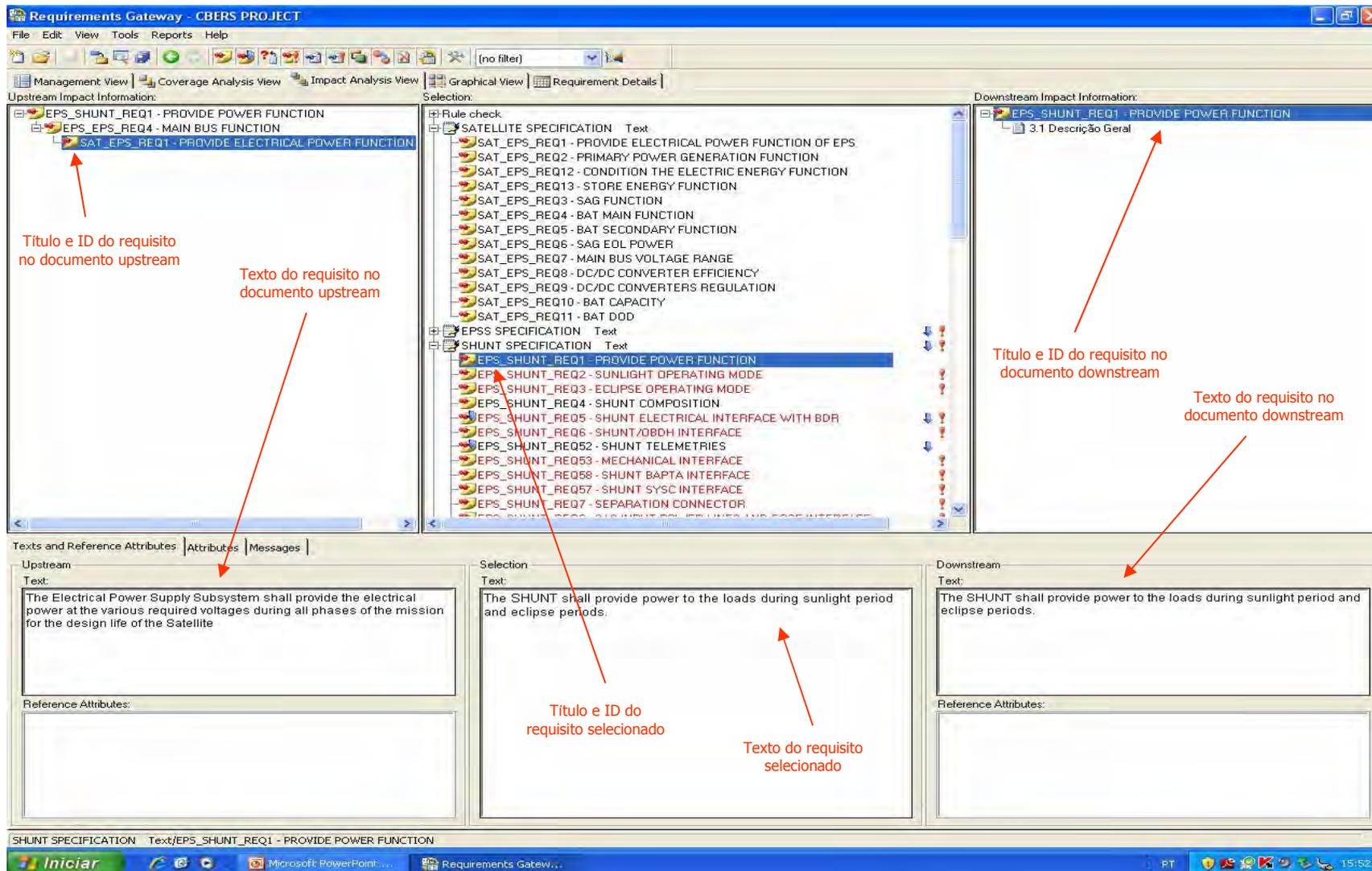


Figura 8.11 - Tela de Análise de Impacto na Requirements Gateway.

The screenshot displays the Requirements Gateway interface for the CBERS PROJECT. It features three main panes showing requirement hierarchies:

- SATELLITE SPECIFICATION:** Lists requirements from SAT_EPS_REQ1 to SAT_EPS_REQ11.
- EPSS SPECIFICATION:** Lists requirements from EPS_EPS_REQ1 to EPS_EPS_REQ36.
- SHUNT SPECIFICATION:** Lists requirements from EPS_SHUNT_REQ1 to EPS_SHUNT_REQ32.

Relationships are shown as lines connecting requirements across these panes. A specific requirement, EPS_EPS_REQ4 (MAIN BUS FUNCTION), is selected. The interface also includes a 'Selection' pane with 'Attributes', a 'Text' pane with the description of the selected requirement, and a 'Messages' pane with a confirmation message.

Annotations in Portuguese explain the elements:

- Título e ID do requisito no documento upstream:** Points to the requirement title and ID in the SATELLITE SPECIFICATION pane.
- Relações entre requisitos:** Points to the lines connecting requirements across the panes.
- Título e ID do requisito selecionado:** Points to the title and ID of the selected requirement in the EPSS SPECIFICATION pane.
- Texto do requisito selecionado:** Points to the description text in the 'Text' pane.
- Títulos e IDs dos requisitos no documento downstream:** Points to the requirement titles and IDs in the SHUNT SPECIFICATION pane.
- Mensagem sobre o requisito:** Points to the confirmation message in the 'Messages' pane.

Figura 8.12 - Tela Gráfica na Requirements Gateway

8.7.3 Observações Durante a Preparação dos Documentos

Durante o trabalho de modificação dos documentos para entrada na ferramenta foram feitas as seguintes observações:

- Não é possível fazer um trabalho perfeito usando documentos preexistentes, pois muitas vezes foi difícil encontrar a relação de rastreabilidade entre os requisitos num texto que não foi escrito com essa preocupação;
- Para fazer o rastreamento corretamente é preciso planejar cada documento, a informação que ele vai conter e como a informação é apresentada;
- É necessário que os requisitos sejam escritos de uma forma rigorosa e clara para permitir avaliar as relações de forma inequívoca;
- É uma tarefa difícil transformar requisitos múltiplos, não-identificados e não-rastreáveis em requisitos simples, rastreáveis, unicamente identificados;
- Incluir informação de rastreabilidade é muito susceptível a erros. No caso estudado, basta escrever um número errado e a informação de rastreabilidade está errada. Deve-se verificar cada informação introduzida na ferramenta;
- Não pode haver erros nas informações de rastreabilidade pois esses erros são difíceis de serem percebidos e corrigidos mais tarde, principalmente se tiver de ser feita por outra pessoa, que não tenha memória do trabalho realizado;
- Incluir informações de rastreabilidade com método, atenção e cuidado;
- São necessários vários dias de trabalho para adaptar um documento para análise pela ferramenta (separar os requisitos, identificá-los e incluir as relações de cobertura). A entrada de informações em um caso real pode ser mais rápida desde que os requisitos tenham sido redigidos adequadamente.

8.7.4 Observações Sobre o Caso Estudado

Algumas observações podem ser feitas sobre o caso estudado.

- Qualidade dos documentos:

Os documentos usados são escritos em linguagem natural, sem indexação de requisitos. Isso dificulta o processo de verificação, pois é difícil identificar o que deve ser verificado. Há erros nos documentos, mesmo depois de submetidos a várias revisões. O número de erros poderia ser diminuído usando técnicas adequadas de preparação de requisitos. No entanto, os documentos contêm uma descrição bastante completa dos equipamentos, pois foi possível construir os equipamentos usando esses documentos como referência.

- Número de documentos usados no estudo:

Foram usados 14 documentos, dos quais 8 são especificações e contém requisitos. Não foram incluídas 7 especificações de conversores DC/DC, por não trazerem nenhuma contribuição para o estudo devido à sua grande similaridade com um documento já incluído. Não é possível calcular a fração de documentos de especificação usados no estudo em relação ao número total existente no sistema SEGI, pois os arquivos nesse sistema não estão organizados de forma a permitir esse cálculo. Estima-se que essa fração seja da ordem de entre 5% a 10% do total de documentos de especificação do Programa CBERS 3 & 4.

- Número de requisitos:

Foram contabilizados 611 requisitos em 8 documentos de especificação. Considerando que somente um documento de especificação de conversores DC/DC foi incluído no estudo, e que esse documento tem 65 requisitos, estima-se que o número total de requisitos do subsistema EPS, se fossem considerados todos os documentos de especificação, deva ser da ordem de 1066 requisitos. O número total de requisitos do

subsistema EPS era totalmente desconhecido antes de iniciar este estudo. Considerando que o satélite tem aproximadamente dez vezes o número de equipamentos que o subsistema EPS tem, e alguns dos quais muito complexos, é possível estimar que o satélite possa ter um número total da ordem de 20.000 requisitos.

8.8 Vantagens de Uso da Ferramenta Requirements Gateway

A ferramenta Requirements Gateway é extremamente fácil de usar. Na verdade, a maior parte do trabalho é feito fora da ferramenta e se resume a preparar e redigir requisitos rastreáveis. Se houver informações de cobertura que sejam óbvias, estas podem ser incluídas diretamente durante a preparação do documento. Uma vez que os documentos estejam na ferramenta, esta mostra vários requisitos de forma visual auxiliando incluir outras coberturas nos documentos. É possível navegar da ferramenta para os documentos para fazer uma correção no documento. A ferramenta automaticamente atualiza os arquivos e reanalisa os documentos.

A Requirements Gateway deve ser a única ferramenta comercial que é compatível com arquivos de programas LabView e rotinas de testes automáticos TestStand. Consequentemente, resultados de testes podem ser usados como coberturas de requisitos verificados por testes automatizados por LabView e TestStand. Isto permite automatizar o rastreamento de requisitos por toda a cadeia hierárquica, desde requisitos de sistema até resultados de testes.

Parte dos equipamentos do subsistema EPS dos satélites CBERS 3 & 4 (os equipamentos eletrônicos) tem alguns de seus testes funcionais automatizados por sistemas de testes que usam programas LabView (mas não TestStand). Consequentemente, listagens de programas LabView poderiam ser usados como cobertura em uma análise pela Requirements Gateway, juntamente com

documentos (relatórios de testes, de análises, de demonstrações, de inspeções).

8.9 Limitações Observadas na Ferramenta Requirements Gateway

Como todas as ferramentas conhecidas, a Requirements Gateway não é totalmente automática. Ela somente registra e exibe simultaneamente vários requisitos e suas relações de forma visual, ajudando o usuário a fazer o rastreamento.

Há alguns comentários sobre algumas limitações observadas das funcionalidades da ferramenta:

- Na tela de Configuração de Projeto:
 - Os recursos de *zoom* e *panning* somente estão disponíveis depois de esconder a parte inferior com os detalhes dos documentos. O botão que esconde os detalhes dos documentos fica num local de difícil localização;
 - Há poucos recursos de *zoom* e *panning*. O *zoom* é relativamente grosseiro e o *panning* é feito com botões laterais da tela;
 - Os caracteres ficam ilegíveis abaixo de certo nível de *zoom*.

- Na tela Graphical View:
 - A Visualização gráfica não tem muita utilidade para grandes projetos porque a imagem fica muito congestionada, impossibilitando ver as linhas que ligam os requisitos e dificultando posicionar os quadros com os documentos, como mostrado na Figura 8.13 para os 14 documentos analisados no estudo. Para melhorar esse recurso visual, a ferramenta tem um filtro que mostra somente aqueles requisitos relacionados com um requisito selecionado.

- No suporte para controle de versões:
 - Não foi encontrada na documentação da ferramenta qualquer referência sobre controle de versões dos documentos. Portanto, outra ferramenta é necessária para manter registros das mudanças, cópias das diferentes versões.

Como observado por Kannenberg e Saiedian (2009, p. 17), que fizeram comentários críticos sobre algumas ferramentas de rastreamento de requisitos, a Requirements Gateway parece ter se originado de uma ferramenta experimental de uso interno de uma empresa, e que posteriormente passou a ser comercializada como um produto.

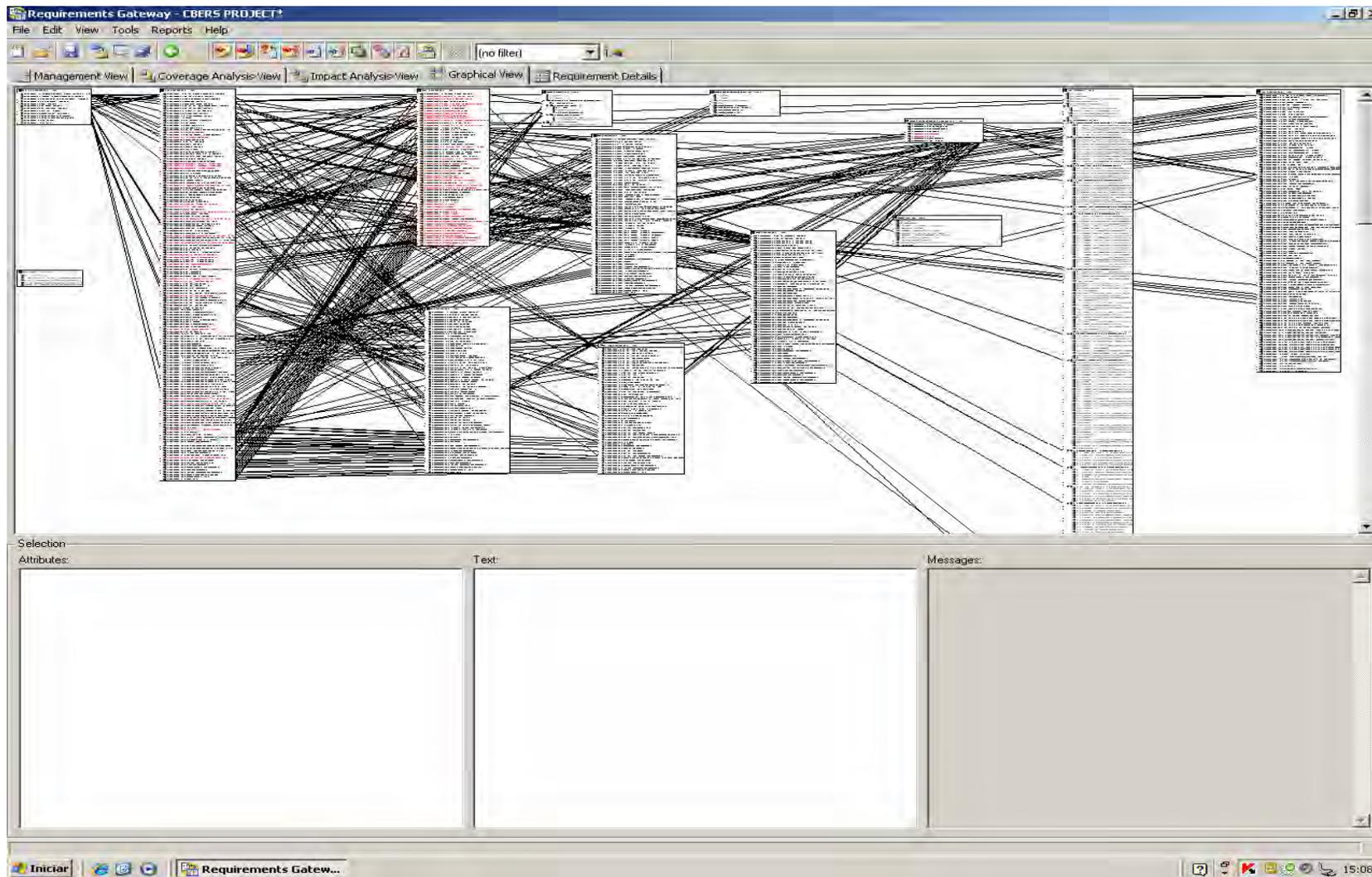


Figura 8.13 - Congestionamento da Tela de Visão Gráfica para Grandes Projetos.

8.10 Comentários Finais

A ferramenta ajudou a encontrar erros em uma documentação que era admitida isenta de erros. Esses erros poderiam ter causado questões legais com empresas contratadas, caso tivessem impacto no custo, pois ocorreram em documentos do INPE. Mesmo sem ter sido preparada com o máximo rigor possível, felizmente a documentação demonstrou-se ser suficientemente correta e completa.

A aplicação *a posteriori* de uma ferramenta de gerenciamento de requisitos sobre documentos preexistentes do programa CBERS é bastante diferente de uma aplicação real. Observam-se as seguintes diferenças:

- A aplicação pode ser feita toda de uma só vez em todos os documentos e é estática, pois os requisitos são imutáveis. Um caso real é bastante dinâmico, porque as informações e documentos são analisados à medida que são produzidos e porque os requisitos e outras informações estão sujeitos a revisões e alterações;
- Em um caso real, os requisitos devem ser redigidos de forma a demonstrar de forma clara e inequívoca suas relações de rastreabilidade com outros requisitos. Neste exercício, muitas vezes foi difícil encontrar as relações de rastreabilidade entre os requisitos;
- O estudo se limitou a documentos de especificação, planos de testes e relatórios de análises. Não havia outros documentos que pudessem conter informações de rastreamento, e.g., relatórios de testes;
- Os erros graves encontrados não foram corrigidos; Não sendo corrigidos geralmente são apontados como erros (requisitos não-cobertos) pela ferramenta;
- Os documentos analisados estavam disponíveis apenas em uma versão (última). Em uma ferramenta com capacidade de controle de configuração várias versões poderiam ser introduzidas na ferramenta.

Ao usar a ferramenta de Requirements Gateway neste estudo, observou-se o seguinte:

- A ferramenta Requirements Gateway é compatível com o modo usado no INPE para registrar informações, i.e., documentos de texto;
- O uso da ferramenta Requirements Gateway exige um formato de requisitos que reduz os erros e melhora a qualidade da documentação;
- O uso da ferramenta Requirements Gateway ajuda a evitar e a encontrar erros;
- O INPE não usa uma metodologia de preparação de requisitos e documentos apropriada ao uso de ferramentas de automatização do rastreamento de requisitos;
- O rastreamento de requisitos ajuda a não "esquecer" de implementar ou de testar os requisitos;
- A preparação dos requisitos, a inclusão das informações de rastreamento e a entrada dos dados na ferramenta são tarefas relativamente burocráticas, mas têm de ser feitas corretamente e por quem entende os detalhes das informações e dos processos;
- Já que requisitos inevitavelmente terão de ser preparados, por sua necessidade nos contratos, no controle de configuração e na engenharia, é vantajoso que sejam preparados de forma a permitir uma maior automatização dos processos;
- Projetos espaciais são de longa duração e, portanto, sujeitos a substituição de pessoal das equipes e esquecimentos. Uma ferramenta de gerenciamento de requisitos armazena as informações de forma que fiquem mais facilmente acessíveis.

9 CONCLUSÕES, RECOMENDAÇÕES E SUGESTÕES PARA CONTINUIDADE DO TRABALHO

9.1 Conclusões

Para entender as necessidades de uso e as causas do não-uso de ferramentas de automatização de processos da Engenharia de Sistemas no INPE, este estudo pesquisou as origens e a formalização da Engenharia de Sistemas e da Engenharia de Requisitos, e a introdução e evolução da Engenharia de Sistemas no INPE.

A seguir apresentam-se resumidamente as conclusões encontradas.

O desenvolvimento das ciências durante o século XIX causou o surgimento de novas tecnologias e novas disciplinas de engenharia. O aumento da quantidade de conhecimentos e as limitações humanas levaram à especialização das engenharias. Engenheiros com funções de sistemas surgiram naturalmente em empresas trabalhando com sistemas de grandes dimensões (redes de telégrafos, energia elétrica, telefones, ferrovias) e em sistemas formados pela integração de partes muito diferentes.

A aquisição de sistemas de armas que integravam tecnologias avançadas era marcada por insucessos devido à inexperiência no desenvolvimento de sistemas de tão alta complexidade. A necessidade de conhecimentos avançados levou à aproximação de militares e cientistas. O projeto de um míssil foguete autoguiado tinha margens mínimas, forte interação entre suas partes e não podia ser feito de baixo para cima a partir de partes projetadas independentemente umas das outras. O desenvolvimento de um míssil foguete exigia um projeto de cima para baixo e com um rigoroso controle da configuração das partes. As rigorosas metodologias de gerenciamento usadas durante o desenvolvimento de mísseis balísticos intercontinentais levaram à formalização da Engenharia de Sistemas. O extremo rigor de documentação

exigido pela Engenharia de Sistema foi muito criticado pela indústria. Esse rigor somente poderia ser completamente atendido com uso de ferramentas computacionais ainda inexistentes na época. Ferramentas computacionais foram rapidamente criadas para auxiliar o gerenciamento de grande quantidade de informações em desenvolvimentos de *software* e de sistemas militares e espaciais.

O aumento da capacidade dos computadores levou a um grande aumento no custo e na complexidade do *software* antes que metodologias apropriadas para seu desenvolvimento tivessem sido criadas. A chamada "crise do *software*" era marcada pelo não-cumprimento de metas de custos, prazos e qualidade nos desenvolvimentos de *softwares* de grande porte. Estudos demonstraram que a maior causa da crise era que a maior fração dos erros no *software*, e também aqueles que causavam maior impacto nos custos, prazos e qualidade, eram causados pela má descrição das partes do *software* quando este era dividido para ser desenvolvido por várias pessoas. Dessa crise emergiram a Engenharia de *Software* e a Engenharia de Requisitos.

As forças que hoje fazem evoluir a Engenharia de Sistemas vêm da área comercial e estão relacionadas ao uso de sistemas complexos em aplicações críticas para a segurança e à grande competição no desenvolvimento de sistemas comerciais complexos.

A Engenharia de Requisitos é a parte mais importante da engenharia de um sistema desenvolvido de cima para baixo, porque é ela que define e descreve o trabalho de forma que possa ser entendido por quem vai executá-lo. Todo o trabalho que vai ser realizado depende da Engenharia de Requisitos: se os requisitos estiverem errados, o trabalho será errado, não importa quão bem ele seja feito. A Engenharia de Requisitos é árdua, trabalhosa e difícil e sua padronização ainda não foi possível por ser uma disciplina nova e ainda em

evolução, e por ser praticada de forma diferente em diferentes locais e contextos.

As necessidades de treinamento da equipe do satélite SACI levaram à introdução da Engenharia de Sistemas, pela primeira vez no Brasil, no INPE, pela General Electric Space Division, no final dos anos 1960, apenas poucos anos após esta ter sido formalizada pela USAF e introduzida na NASA. A Engenharia de Sistema foi reintroduzida no INPE em meados dos anos 1980 durante o desenvolvimento do Programa MECB. Nessas duas ocasiões foi dada grande atenção à qualidade da redação de requisitos, mas não foram introduzidos processos formais de preparação de requisitos.

O rigor da aplicação da Engenharia de Sistemas na USAF, na NASA e no INPE passou por variações que podem estar relacionadas com dificuldades na transferência de conhecimento e treinamento entre as sucessivas gerações das equipes de trabalho.

O rastreamento de requisitos de *software* é exigido no Programa CBERS desde 1988. Sua não-implementação naquela época pode ser justificada pela impossibilidade de fazer o rastreamento de requisitos manualmente e pela inexistência de ferramentas comerciais de rastreamento de requisitos. É difícil justificar a não-implementação do rastreamento de requisitos no Programa CBERS 3 & 4 hoje. No entanto, ele não poderá ser implementado adequadamente sem que, antes, os processos de preparação de requisitos sejam melhorados.

A Engenharia de Sistemas evoluiu nas últimas duas décadas pela incorporação de processos, metodologias e ferramentas da Engenharia de *Software*. O conservadorismo da Engenharia de Sistemas Espaciais, a falta de parceiros mais avançados, a falta de pessoal, a falta de treinamento e aperfeiçoamento fizeram com que, não obstante o grande sucesso de suas missões espaciais, a

evolução da Engenharia de Sistemas ocorrida nas últimas décadas não tenha se refletido nos processos e ferramentas usadas no desenvolvimento de satélites do INPE.

O estudo da Engenharia de Requisitos e de ferramentas de automatização do gerenciamento de requisitos mostrou que:

- A preparação de requisitos é um trabalho difícil;
- O rastreamento e gerenciamento de requisitos têm de ser isentos de erros e não podem ser feitos manualmente, exceto para sistemas com pouquíssimos requisitos;
- Os requisitos inevitavelmente devem que ser preparados e redigidos, por sua necessidade nos contratos, na engenharia, controle de configuração. Portanto, é vantajoso que sejam redigidos e preparados de forma a permitir uma maior automatização dos processos;
- A aquisição e uso de uma ferramenta de gerenciamento de requisitos não elimina a necessidade de criação de processos de requisitos;
- O rastreamento de requisitos é um compromisso que não pode ser interrompido, exige treinamento de pessoal, tem custos e pode exigir mudanças na cultura da empresa;
- Ferramentas de automatização do gerenciamento de requisitos não são panacéias e não substituem mão-de-obra especializada, apenas ajudam a fazer o serviço.

O exercício de aplicação de uma ferramenta de automatização do gerenciamento de requisitos à documentação do subsistema de energia elétrica dos satélites CBERS 3 & 4 demonstrou que uma ferramenta de automatização:

- Permite rastrear requisitos de sistemas com milhares de requisitos;
- Reduz os erros nos requisitos;
- Melhora a qualidade da documentação;

- Melhora a comunicação entre os *stakeholders*;
- Faz melhor uso da mão-de-obra;
- Auxilia o processo de verificação;
- Reduz riscos de não-cumprimento de prazos, custos e qualidade.

A ferramenta ajudou a encontrar erros que poderiam ter causado questões legais com uma contratada devido a seus efeitos no custo e pelos contratos do INPE serem de preço fixo. Assim sendo, poder-se-ia incluir um item muito importante, por ser uma possibilidade bastante real:

- Reduz o risco de problemas legais com as contratadas, causando atrasos, suspensão de atividades etc.

9.2 Recomendações

O INPE já possui os pré-requisitos necessários e se adaptará facilmente ao uso de uma ferramenta de gerenciamento de requisitos, pois já usa metodologias de Engenharia de Sistemas e tem uma cultura acostuada com requisitos, documentos, revisões de projeto, controle de configuração, rastreabilidade. No entanto, antes que uma ferramenta de automatização do rastreamento de requisitos possa ser usada, devem ser melhorados os processos de preparação de requisitos e documentos.

Como há no mercado uma grande variedade de ferramentas comerciais de gerenciamento de requisitos, com grandes diferenças de preços e características, é preciso que se escolha uma ferramenta que melhor se adapte aos processos já existentes no INPE. Algumas ferramentas são grandes pacotes para todos os processos do ciclo de vida e acabam muitas vezes sendo subutilizadas pois têm funcionalidades não necessárias aos processos existentes. Como o INPE tem por tradição trabalhar com documentos, a escolha mais indicada é uma ferramenta do tipo "centrada em documentos". Como o INPE faz o controle de configuração e rastreamento de fabricação de

hardware de forma manual, seria interessante encontrar uma ferramenta que auxiliasse nessas duas tarefas.

O uso da ferramenta tem de ser feito por quem entende perfeitamente o trabalho sendo realizado. Portanto há necessidade de pessoal especializado e de treinamento adequado. O uso de uma ferramenta é um compromisso que não pode ser interrompido, pois pode não ser possível recuperar o *status* se dados deixarem de ser entrados. A entrada de dados tem de ser isenta de erros, pois pode ser impossível corrigir no futuro se tiver de ser feita por outra pessoa e se a única informação existente for a que está na ferramenta. Como dados em arquivos eletrônicos podem ser facilmente corrompidos e apagados, o uso de uma ferramenta computacional exige a preocupação permanente com a criação de *backups* e um rigoroso controle de acesso.

Deve haver preocupação em manutenção e aperfeiçoamento dos conhecimentos em Engenharia de Sistemas Espaciais. A formação de um engenheiro de sistemas espaciais é demorada e complexa: a Engenharia de Sistemas Espaciais exige conhecimentos em áreas muito diversas e para ela não existem cursos. Um projeto espacial é de longa duração e sujeito à saída e substituição de pessoal, mas geralmente, não há preocupação com formação das novas equipes de trabalho.

9.3 Sugestões para a Continuidade do Trabalho

Desde que houve notícia no INPE de que o sistema SEGI está sendo substituído pela ferramenta Windchill, um pacote de ferramentas com funcionalidades para todo o ciclo de vida de um produto, é preciso que se determine a melhor maneira de fazer uso dos recursos dessa nova ferramenta.

Poder-se-ia iniciar-se um trabalho com as seguintes bases:

- É preciso que se criem processos adequados de preparação de requisitos e que o INPE ainda não tem;
- É preciso que sejam definidos um processo de requisitos e um modelo de rastreabilidade para decidir "o que", "como", "porque", e um limite de até onde rastrear e quais atributos registrar. Quanto mais se registrar na ferramenta, mais difícil será o gerenciamento dessas informações;
- É possível que a ferramenta Windchill possua recursos que possam ser usados para automatizar o processo manual de gerenciamento da configuração. É possível que seja necessário adaptar alguns processos aos recursos da ferramenta;
- É preciso decidir sobre como serão armazenados os documentos dos programas anteriores. Esses documentos não têm informação de rastreabilidade, mas têm informações importantes e devem ser arquivados de forma a facilitar seu uso. A ferramenta Windchill pode ser usada para melhorar o arquivamento dessas informações.

REFERÊNCIAS BIBLIOGRÁFICAS

AIR FORCE SYSTEMS COMMAND (AFSC). **Systems engineering management procedures**. Washington, DC: Andrews Air Force Base, 1966. 232p. (AFSCM 375-5).

ALFORD, M. W. A requirements engineering methodology for real-time processing requirement. **IEEE Transactions on Software Engineering**, v. SE-3, n. 1, p. 60 – 69, January 1977.

ARNOLD, S. **Transforming systems engineering principle into integrated project team practice**. 2008. 379p. Tese de Doutorado - Cranfield University, Defense College of Management and Technology, Engineering Systems Department, Cranfield, UK, 2008. Disponível em <<https://dspace.lib.cranfield.ac.uk/bitstream/1826/3033/1/Transforming%20Systems%20Engineering%20Principles%20into%20Integrated%20Project%20Team%20Practice.pdf>>. Acessado em: 11 de outubro de 2011.

STANFORD UNIVERSITY. **ASCEND** : advanced system for communication and education in national development: an interdisciplinary engineering course in space systems engineering at Stanford. Stanford, CA: School of Engineering, Stanford University, 1967.

AZZOLINI, J. **Introduction to systems engineering practice**. Session 1, Requirements. Greenbelt, MD: NASA. GSFC. Disponível em: [http://ses.gsfc.nasa.gov/ses_data_2000/000712_Azzolini.ppt#277,25,Requirements Analysis](http://ses.gsfc.nasa.gov/ses_data_2000/000712_Azzolini.ppt#277,25,Requirements%20Analysis). Acesso em: 27/10/2011.

BAHILL, A. T. ;DEAN, F. F. Discovering system requirements. In: SAGE, A. P.; ROUSE, W. B. (eds.). **Handbook of systems engineering and management**. 2. ed. New Jersey, NJ: John Wiley & Sons, 2009. 1476p.

BARBOSA, J. I. M.; RABAY, S. **Procedimento para aprovação e modificação de documentos do programa CBERS**. Versão 0. 9 de julho de 2005. China Brazil Earth Resources Satellite. São José dos Campos: INPE, 2005. 25p. (R-MNG-1023).

BELFORD, P. C. ;BOND, A. F.; HENDERSON, D. G.; SELLERS, L. S. Specifications – a key to effective software development. In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING, 2., 1976, Los Alamitos, CA, USA. **Proceedings...** San Francisco, CA: IEEE, 1976. p. 71 - 79.

BELL, T. E.; BIXLER, D. C.; DYER, M. E. An extendable approach to computer-aided software requirements engineering. **IEEE Transactions on Software Engineering**, v. SE-3, n. 1, p. 49 – 60, January 1977.

BELL, T. E.; THAYER, T. A. Software requirements: are they really a problem?. In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING, 2., 1976, Los Alamitos, CA, USA. **Proceedings...** San Francisco, CA: IEEE, 1976. p. 61 – 68.

BENSON, L. R. **Acquisition management in the United States Air Force and its predecessors**, Air Force History Support Office, Washington, DC, 1997.

BERENBACH, B; PAULISH, D. J.; KAZMEIER, J.; RUDORFER, A. **Software & systems requirements engineering in practice**. New York, NY: Mc Graw Hill, 2009. 320p.

BOEHM, B. W. **Software and its impact: a quantitative assessment**. Santa Monica: RAND Corporation. P - 4947, 1972.

BOEHM, B. W. Software engineering. **IEEE Transactions on Computers**, v. C-25, n. 12, p. 1226 - 1241, December 1976.

BOEHM, B. W.; McCLEAN, R. K.; ULFRIG, D. B. Some experience with automated aids to the design of large-scale reliable software. In: INTERNATIONAL CONFERENCE ON RELIABLE SOFTWARE, 1975, Los Angeles, CA. **Proceedings...** New York, NY: ACM, 1975. v. 10, issue 6, p. 105 - 113.

BOEHM, B. Some future trends and implications for systems and software engineering processes. **Systems Engineering**, v. 9, Issue 1, p. 1 - 19, Jan. 2006.

BRANCO, M. S. A. **Verification guidelines**. Versão 0. 2 de Junho de 2005. China Brazil Earth Resources Satellite CBERS 3 & 4. São José dos Campos: INPE, 2005. 27p. (RB-MNG-1022).

BRILL, J. H. Systems engineering – a retrospective view. **Systems Engineering**, v. 1, n. 4, p. 258-266, 1998.

BROOKS JR., F. P. **The mythical man-month: essays on software engineering**. 2. ed. Boston, MA: Addison-Wesley Longman, Inc., 1995. 322p.

BUENO, L. A. R.; CHEN QINNAN, et al. **CBERS 3 & 4 Satellite Specification**. versão 1, 9 de novembro de 2005, São José dos Campos: INPE, 2005. 28 p. (RB-HDS-0023).

BUENO, L. A. R.; LI J. D.; CHEN, Q. **CBERS 3 & 4 Satellite product matrix**. China Brazil Earth Resources Satellite CBERS 3 & 4. Versão 4. 16 de maio de 2008. São José dos Campos: INPE, 2008. 12p. (RB-MNG-0010).

COMISSÃO NACIONAL DE ATIVIDADES ESPACIAIS (CNAE). **Projeto SACI: satélite avançado de comunicações interdisciplinares**. São José dos Campos, SP: CNAE, 1968, (CNAE-75-LAFE)

_____. **Project SACI**: report no. II, part b. São José dos Campos, SP: CNAE, 1969 (CNAE-91-LAFE).

_____. **Projeto SACI**: report no. III. São José dos Campos, SP: CNAE, 1970a (CNAE-104-LAFE).

_____. **Brazilian educational radio and TV experiment on ATS-F**: experiment proposal. São José dos Campos, SP: CNAE, 1970b, 101 p. (CNAE-109-LAFE).

_____. **Sugestão para interação CNAE e INEP no projeto SACI**. São José dos Campos, SP: CNAE, 1971a, 110 p. (CNAE-151-LAFE).

_____. **Resumo dos 10 primeiros anos de atividades da CNAE**. São José dos Campos, SP: CNAE, 1971b. 89 p. (CNAE-161-LAFE).

_____. **Banco de dados**: estudo preliminar. São José dos Campos, SP: CNAE, 1971c (CNAE-153-LAFE).

COSTA FILHO, E. J. **A política científica e tecnológica no setor aeroespacial brasileiro**: da institucionalização das atividades ao fim da gestão militar – uma análise do período 1961-1993. 2000. 236p. Dissertação (Mestrado em política científica e tecnológica) - Instituto de Geociências da Universidade Estadual de Campinas, Campinas, SP, 2000.

COSTA FILHO, E. J. **A dinâmica da cooperação espacial sul-sul**: o caso do programa CBERS (China-Brazil Earth Resources Satellite). 2006. 326p. Tese (Doutorado em política científica e tecnológica) - Instituto de Geociências da Universidade Estadual de Campinas, Campinas, SP, 2006.

DAVIS, C. G.; VICK, C. R. The software development system. **IEEE Transactions on Software Engineering**, v. SE-3, n. 1, p. 69 – 84, Jan. 1977.

DIJKSTRA, E. W. The humble programmer. **Communications of the ACM**, v. 15, Issue 10, p. 859 - 866, Oct. 1972.

DORFMAN, M.; FLYNN, R. F. ARTS – An automated requirements traceability system. **Journal of Systems and Software**, v. 4, Issue 1, p. 63 – 74, 1984.

DORFMAN, M. System and software requirements engineering. In THAYER, R. H.; DORFMAN, M. (eds.). **System and software requirements engineering**, p. 4 - 16, Los Alamitos, CA: IEEE Computer Society Press, 1990. 719p.

ELECTRONIC NUMERICAL INTEGRATOR AND COMPUTER (ENIAC). 2012. Disponível em: <http://en.wikipedia.org/wiki/ENIAC>. Acesso em: 24 fev. 2012.

ESCADA, P. A. S. **Construção e usos sociais da pesquisa científica e tecnológica**: um estudo de caso da divisão de processamento de imagem do INPE. 2010. 231p. Tese (Doutorado em ciência política) - Faculdade de

Filosofia, Letras e Ciências Humanas da Universidade de São Paulo, São Paulo, SP, 2010.

ESCADA, P. A. S. **Origem, institucionalização e desenvolvimento das atividades espaciais brasileiras**. 2005. 129p. Dissertação (Mestrado em ciência política) - Departamento de Ciência Política do Instituto de Filosofia e Ciências Humanas da Universidade Estadual de Campinas, Campinas, SP, 2005.

EUROPEAN SPACE AGENCY (ESA). **Verification**. Noordwijk, The Netherlands: ESA Requirements and Standards Division, ESTEC, 2009. 45p. (ECSS-E-ST-10-02).

_____. **Basic requirements for product assurance of ESA spacecraft and associated equipment**. Noordwijk, The Netherlands: ESA Scientific and Technical Publications Branch, ESTEC, 1981. (ESA-PSS-01-0).

_____. **Small and medium enterprise initiative**. Configuration management training, Section 1, 2000. Disponível em: http://esamultimedia.esa.int/docs/industry/SME/Configuration/Section_1-CM.pdf. Acesso em: 26/10/2011.

EVELEENS, J. L.; VERHOEF, C. The rise and fall of the chaos report figures. **IEEE Software**, IEEE Computer Society, v. 27, n. 1, p. 30 – 36, Jan./Feb. 2010 doi:10.1109/MS. 2009.154.

FERNANDES, E. **Uma estruturação lógica para gestão de projetos espaciais**. 1987. 211p. Dissertação (Mestrado em análise de sistemas) - Instituto de Pesquisas Espaciais, São José dos Campos, SP, 1987. PI (INPE-4435-TDL/310) P.I.

FERNANDES, E.; CARVALHO H. C. C.; KONO, J.; BARBOSA, J. I. M.; LIBERATO JR., J.; LIMA, M. G. R.; SILVEIRA, V. B.; YAMAGUTI, W.; XU L., ZHANG S. H. **Basic requirements for contractor's product assurance plan of CBERS spacecraft and associated documents**. São José dos Campos: INPE, Oct. 28, 1989. (C-PAD-003).

FETTERS, C.; HSU, T.; SMEED, B. Requirements tracing. In: **Software quality, validation, and verification**. CIS 841 Class Project. Kansas State University, Fall 1999. Disponível em: <http://people.cis.ksu.edu/~hankley/d841/Fa99/chap2.htm>. Acesso em: 28/10/2011.

FINKELSTEIN, A.; EMMERICH, W. The future of requirements management tools. In: QUIRCHMAYR, G.; WAGNER, R.; WIMMER M. (eds.). **Information systems in public administration and law**. Osterreichische Computer Gesellschaft, 2000.

FLYNN, R. F.; DORFMAN, M. The automated requirements traceability system (ARTS): an experience of eight years. In: THAYER, R. H.; DORFMAN, M. (eds.). **System and software requirements engineering**. Los Alamitos, CA: IEEE Computer Society Press, 1990. 719p.

FREIRE, C. F. S.; TORRES, L. C. G.; MAGALHÃES, R. O. **Shunt Specification**. versão 1, 9 de agosto de 2006. São José dos Campos: INPE, 2006. 14 p. (RBDA-HDS-1001).

FREIRE, C. F. S.; LOURO, A. C.; WANG, J. **Electrical Power Supply Subsystem Specification**, versão 5, 14 de maio de 2008, São José dos Campos: INPE, 2008. 22 p. (RBDD-HDS-1004)

GALILEI, G. **Dialogue concerning two new sciences**. New York, NY, USA: The Macmillan Company, 1914, 328p.

GENERAL ELECTRIC. **Systems engineering management workshop**. v. 1. Schenectady, NY, USA: General electric space division, 1972a.

GENERAL ELECTRIC. **Systems engineering management workshop**. v. 2. Schenectady, NY, USA: General electric space division. 1972b.

GHAZARIAN, A. A matrix-less model for tracing software requirements to source code. **International Journal of Computers**, v. 2, Issue 3, p. 301 – 309, 2008.

GOTEL, O. C. Z.; FINKELSTEIN, A. C. W. An analysis of the requirements traceability problem. In: INTERNATIONAL CONFERENCE ON REQUIREMENTS ENGINEERING, 1., 1994, Colorado Springs, CO. **Proceedings...** Colorado Springs, CO: IEEE, 1994. p. 94 – 101. ISBN 0-8186-5480-5.

GOTEL, O.; MÄDER, P. How to select a requirements management tool: initial steps. In: IEEE INTERNATIONAL REQUIREMENTS ENGINEERING CONFERENCE, 17., 2009, Atlanta, GA. **Proceedings...** Los Alamitos, CA: IEEE, 2009. p. 365 – 367.

GRADY, J. O. **Systems requirements analysis**. Amsterdam: The Netherlands: Academic Press, 2006, 455p.

GRADY, J. O. **Systems synthesis, product and process design**. Boca Raton, FL: CRC Press, 2010, 538p.

HALL, A. D. **Systems engineering**. Princeton, NJ: D. van Nostrand Company, Inc. 1962. 478p.

HALLMAN, L. B. The role of management in the development of an aerospace system. **IRE Transactions on Engineering Management**, New York, NY, USA: The Institute of Radio Engineers, Inc. EM-9 (4), p. 182-185, Dec. 1962.

HARVEY, B.; SMID, H.; PIRARD, T. **Emerging space powers – the new space programs of Asia, The Middle East and South America**. Chichester, UK: Praxis Publishing Ltd, 2010. 624p.

HOBAN, F. T.; LAWBAUGH, W. M. **NASA SP - 6102 Readings in systems engineering**. Washington, D. C: National Aeronautics and Space Administration, 1993.

HOKKANEN, M. **Requirements traceability**. 2001. 84p. Dissertação (Mestrado em engenharia industrial e gerenciamento) - Lappeenranta University of Technology, Lappeenranta, Finlândia 2001.

HUGHES, A; HUGHES, T. P. **Systems, experts, and computers: the systems approach in management and engineering, world war II and after**. Cambridge, MA: The MIT Press, 2000. 513p.

INTERNATIONAL BUSINESS MACHINE (IBM). **Gain control of your requirements process and meet your business goals with IBM Rational DOORS**. 2009. Material publicitário da Rational Doors - Folheto. Disponível em: <http://www-05.ibm.com/de/events/innovate/pdf/infomaterial/RAB14016USEN-1.pdf>. Acesso em: 2/11/2011.

INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS(IEEE). **IEEE Std 830-1998**: Recommended practices for software requirements specification. New York, NY, USA, 1998. 37p.

_____. **IEEE Std 1233-1998**: Guide for developing systems requirements specifications. New York, NY. USA, 1998. 36p.

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS (INPE) **Plano de gestão da configuração da MECB**. Versão 1 (4 de junho de 1986). São José dos Campos: INPE, 1986. (A-CFG-0001)

_____. **Descrição detalhada do trabalho de desenvolvimento, fabricação e testes de equipamentos do subsistema de suprimento de energia elétrica dos satélites CBERS 3 e 4**. Versão 1. 14 de setembro de 2004. São José dos Campos: INPE, 2004. 67p. (RBD-SOW-1001)

_____. **CBERS 3 & 4 Satellite development and test plan**. Rev 03. São José dos Campos: INPE, 2004. 24p. (RB-MNG-0002).

_____. **CBERS 3 & 4 Product assurance requirements**. Versão 2. 5 de Novembro 2005. China Brazil Earth Resources Satellite CBERS 3 & 4. São José dos Campos: INPE, 2004. 51p. (RB-PAD-0002).

INTERNATIONAL COUNCIL ON SYSTEMS ENGINEERING (INCOSSE). INCOSSE requirements management tools survey. Disponível em: <http://www.incose.org/productspubs/products/rmsurvey.aspx>. Acesso em: 2/11/2011.

JOHNSON, S. Samuel Phillips and the taming of Apollo. **Technology and Culture**, v. 42, n. 4, p. 701-702, Oct. 2001.

JOHNSON, S. **The secret of Apollo – systems management in american and european space programs**. Baltimore, MD: The John Hopkins University Press, 2002a. 290p.

JOHNSON, S. **The United States Air Force and the culture of innovation**. Washington, D.C: Air Force History and Museums Program, 2002b. 288p.

KAHN, H.; MANN, I. **Techniques of systems analysis**. Santa Monica, CA: The RAND corporation, 1957. (RM-1829-1-PR).

KANNENBERG, A.; SAIEDIAN, H. Why software requirements traceability remains a challenge. **Crosstalk The journal of Defense Software Engineering**, July/August, 2009.

KOHN, R. **Reflections on research and development in the United States Air Force**. Washington, D. C: Center for Air Force History, 1993.

KONO, J.; BUENO, L. A. R. **CBERS 3 & 4 Program work breakdown structure**. Rev. 0 (19 de novembro de 2004). China Brazil Earth Resources Satellite CBERS 3 & 4. São José dos Campos: INPE, 2004. (R-MNG-0004)

KOSSIAKOFF, A.; PRETTYMAN, E.C.; PARK, J.M.; HAZAN, P.L.; SLEIGHT, T.P. **DOD weapon system software management study**. Laurel, MD: John Hopkins University Applied Physics Laboratory, 1975, 206p.

KOSSIAKOFF, A.; SWEET, W. N.; SEYMOUR, S. J.; BEIMER, S. M. **Systems engineering principles and practice**. 2. ed. New Jersey, NJ: John Wiley and Sons, 2011. 528p.

LEITE JÚNIOR, J. A. P. **O INPE como agente executor das políticas públicas de ciência e tecnologia**: o caso da política espacial brasileira. In: CURSO DE PÓS-GRADUAÇÃO EM GESTÃO ESTRATÉGICA DA CIÊNCIA E TECNOLOGIA EM IPP'S, 1., 2009, São José dos Campos. Coletânea... São José dos Campos: INPE, 2011. On-line. Disponível em: <http://urlib.net/8JMKD3MGP7W/39GA9JB>. Acesso em: 24 fev. 2012.

LIEBOWITZ, B. H. The technical specification – key to management control of computer programming. In: SPRING JOINT COMPUTER CONFERENCE - AFIPS '67 (Spring). 1967, New York. **Proceedings...** New York: ACM, 1967. p. 51 – 59.

MADDOX, B. Mendonça's dream of Brazil in space. **New Scientist**, v. 62, n. 894, p. 114-115, 18 Apr. 1974a.

MADDOX, B. A brazilian satellite for what ?. **New Scientist**, v. 62, n. 895, p. 178-179, 25 Apr. 1974b.

MENDONÇA, F. (organizador) **Engenharia de sistemas: planejamento e controle de projetos**. Petrópolis, RJ: Vozes, 1972. 307p.

MINDELL, D. A. Automation's finest hour: radar and system integration in world war II. In: HUGHES, A; HUGHES, T. P. **Systems, experts, and computers: the systems approach in management and engineering, world war II and after**. Cambridge, MA: The MIT Press, 2000. 513p.

MORAES, G. S. **Introdução à análise de sistemas**. São José dos Campos: INPE, 1970. 68 p. (INPE-100/LAFE). Tradução e adaptação do artigo Introduction to Systems Analysis publicado por PARKER, J. K. no relatório n. 298 do Management information service. International city managers association, Washington, DC, USA, 1970.

MOREIRA, M. L. **Formação de competências em ciência e tecnologia espaciais: uma análise da trajetória da pós-graduação no instituto nacional de pesquisas espaciais**. 2009. 209p. Tese (Doutorado em política científica e tecnológica) - Instituto de Geociências da Universidade Estadual de Campinas, Campinas, SP, 2009.

MORRIS, P. W. G. **The management of projects**. London: Thomas Telford, 1997. 358p.

MORRISON, E. J. Defense systems management: the 375 series. In: CLELAND, D. I.; KING, W. R. (eds.) **Systems, organization, analysis, management: a book of readings**. New York, NY: McGraw-Hill Company Books, 1969. 410p.

NASA. **NASA-APOLLO program management**. Washington, DC: Office of Manned Space Flight, Dec. 1967. v. 1.

NAUR, P.; RANDELL, B. **Software engineering – report on a conference sponsored by the NATO science committee**. Garmisch, Germany, 1969. Edição eletrônica. Disponível em:

<http://homepages.cs.ncl.ac.uk/brian.randell/NATO/nato1968.PDF>. Acesso em: 24 fev. 2012.

NEUFELD, J. **The development of ballistic missile in the United States Air Force 1945-1960**. Washington, DC: Office of Air Force History, 1990.

NATIONAL INSTRUMENTS (NI).. **Customization Guide**. Requirements Gateway. Jan. 2009. (372657A-01). Versão eletrônica. Disponível em: <http://www.ni.com/pdf/manuals/372657a.pdf>. Acesso em: 24 fev. 2012.

_____. **Getting Started with NI Requirements Gateway**. Requirements Gateway. Jan. 2009. (375035A-01). Versão eletrônica. Disponível em: <http://www.ni.com/pdf/manuals/371912a.pdf>. Acesso em: 24 fev. 2012.

PASCHOTTO, M.; RABAY, S. **MECB Product assurance plan**. Versão 2 (29 de abril 1992). São José dos Campos: INPE, INPE, 1990. 52p. (A-GQL-0006)

PEREIRA, G. R. **Política espacial brasileira e a trajetória do INPE (1961-2007)**. 2008. 222p. Tese (Doutorado em política científica e tecnológica) - Instituto de Geociências da Universidade Estadual de Campinas, Campinas, SP, 2008.

PILIGIAN, M. S.; BASHAW, C. J.; POKORNEY, J. L. **Configuration management of computer program contract end items**. Bedford, Massachusetts: Electronic System Division. Air Force Systems Command, USAF, 1968.

PILIGIAN, M. S.; POKORNEY, J. L. Air Force concepts for the technical control and design verification of computer programs. In: SPRING JOINT COMPUTER CONFERENCE -AFIPS '67 (Spring). 1967, New York. **Proceedings...** New York: ACM, 1967. p. 61 – 66.

PINHEIRO, F. A. C. Requirements traceability. In: LEITE, J. C. S. P.; DOORN, J. H. (eds.), **Perspectives in software requirements**. Dordrecht, The Netherlands: Kluwer Academic Publisher , 2004. p. 93 – 113.

POKORNEY, J. L.; MITCHELL, W. E. **A systems approach to computer programs**. Bedford, Massachusetts, USA: Electronic System Division. Air Force Systems Command, USAF, 1967.

POWERS, T.; STUBBS, C. **A study on current practices of requirements traceability in systems development**. 1993. 71 p. Dissertação (Mestrado em gerenciamento de tecnologia da informação) - Naval Postgraduate School, Monterey, CA, 1993.

PROJECT MANAGEMENT INSTITUTE INC. (PMBOK). **A guide to the project management body of knowledge**. Newtown Square, PA, USA: PMI Publishing Division, 1996. 182p.

PULCHÉRIO, A. F. et al. **Engenharia de sistemas: planejamento e controle de projetos**. Dissertação (Mestrado em análise de sistemas) - Instituto de Pesquisas Espaciais, São José dos Campos, SP: CNAE, 1971. (CNAE-170-LAFE)

RAMESH, B.; JARKE, M. Towards reference models for requirements traceability. **IEEE Transactions on Software Engineering**, v. 27, n. 1, p. 58 – 93, 1999.

RAMESH, B.; STUBBS, C.; POWERS, T.; EDWARDS, M. Requirements traceability: theory and practice. **Annals of Software Engineering**, v. 3, p. 397 – 415, 1997.

RAMO, S.; ST. CLAIR, R. K. **The systems approach**: fresh solutions to complex problems through combining science and practical common sense. Anaheim, CA, USA: KNI, Incorporated, 1998. 156p.

RAND CORPORATION. **Preliminary design of an experimental world-circling spacecraft**. Santa Monica, CA, USA: Douglas Aircraft Company, Inc.. 1946. (Report No. SM-11827).

RATYNSKI, M. V. The Air Force computer acquisition concept. SPRING JOINT COMPUTER CONFERENCE -AFIPS '67 (Spring). 1967, New York. **Proceedings...** New York: ACM, 1967. p. 33 – 44.

REDSHAW, M. C. Building on a legacy: renewed focus on systems engineering in defense acquisition. Defense Acquisition University. **Defense Acquisition Review Journal**, n. 53, p. 96-97, Jan. 2010.

REIS, J. R.; MONTEIRO, R. P.; CIMA, S. C. F.; MEDEIROS, J. A. S. **Engenharia de sistemas**: uma abordagem prática. Petrópolis, RJ: Vozes, 1980, 224p.

RICO, D. F.; SAYANI, H. H.; FIELD, R. F. History of computer, electronic commerce and agile methods. **Advances in Computers – Emerging Technologies**, v. 73, p. 1 – 55, 2008.

ROYCE, W. W. Managing the development of large software systems. In: IEEE WESCON 26 (Aug 1970), 1970, [S.I.]. **Proceedings...** [S.I.]: IEEE, 1970. p. 1–9. Disponível em:
<http://www.cs.umd.edu/class/spring2003/cmsc838p/Process/waterfall.pdf>.
Acesso em: 24 fev.2012.

SANTANA, C. E, COELHO, J. R. B.; O projeto CBERS de satélites de observação da Terra. **Parcerias Estratégicas**, n. 7, p. 203 – 210, out. 1999.

SOCIETY OF THE AUTOMOTIVE ENGINEERS (SAE). **SAE ARP 4754**. Certification Considerations for Highly-Integrated or Complex Aircraft Systems. Warrendale, PA, USA, 1996. 88p.

SATELLITE. 2012. Disponível em: http://en.wikipedia.org/wiki/Earth_Satellite. Acesso em: 24 fev. 2012.

SCHLAGER, K. J. Systems engineering: key to modern development. **IRE Transactions on Engineering Management**. New York, NY, USA: The Institute of Radio Engineers, Inc., v. EM-3 (3), p. 64-66, July, 1956.

SEARLE, L. V.; NEIL, G. Configuration management of computer programs by the Air Force: principles and documentation. SPRING JOINT COMPUTER CONFERENCE, 1967, New York, NY. **Proceedings...** New York, NY: ACM, 1967. p. 45–49.

SEARLE, L. V.; ROSOVE, P. E.; SYDOW, E. H. **System management applied to large computer programs in BUIC III; review of experience.** Bedford, Massachusetts: Electronic Systems Division, Air Force Systems Command, USAF, 1969.

SEBOK. **Guide to the software engineering body of knowledge.** Los Alamitos, CA, USA: IEEE Computer Society, 2004. 202p.

SHROFF, V. **Requirements management metrics.** 2001. 115p. Master Thesis (Electrical and Computing Engineering) - University of Calgary, Calgary, Alberta, Canada, 2001.

SOMMERVILLE, I. **Engenharia de software.** 8. ed. São Paulo, SP: Pearson Education do Brasil, 2007. 552p.

SOUZA, A. A. M. **A Especialização do lugar: São José dos Campos como centro da tecnologia aeroespacial no país.** 2008. 191p. Tese (Doutorado em geografia) - Faculdade de Filosofia, Letras e Ciências Humanas da Universidade de São Paulo, São Paulo, SP, 2008.

TEICHROEW, D. A survey of languages for stating requirements of computer-based information systems. AFIPS JOINT COMPUTER CONFERENCE. AFIPS '72, (Fall, part II), Dec. 1972, New York. **Proceedings...** New York, NY: ACM, 1972. p. 1203 - 1224.

THAYER, R. The Air Force software manager. In: ACM NATIONAL CONFERENCE, ACM' 75, 1975, New York. **Proceedings...** New York, NY: ACM, 1975. p. 294 - 297.

VASCONCELLOS, R. R. **Barreiras e facilitadores na transferência de tecnologia para o setor espacial: estudo de caso de programas de parceria das agências espaciais do Brasil (AEB) e dos EUA (NASA).** 2008. 469p. Tese (Doutorado em engenharia) - Escola Politécnica da Universidade de São Paulo, São Paulo, SP, 2008.

VITRUVIUS **The ten books on architecture.** 1. ed. New York, NY: Dover Publications, 1960, 331p. SBN: 486-20645-9.

VON BRAUN, W. World's first TV broadcast satellite. **Popular Science**, v. 196, n. 5, p. 64 – 66, May, 1970.

VON KARMAN, T. **Where we stand.** Dayton, Ohio: Headquarter Air Material Command, 1946.

WALTER, F; et al. **Curriculum de mestrado em análise de sistemas: uma abordagem de sistemas.** São José dos Campos, SP: CNAE, 1971 (CNAE-154-LAFE).

WHIRLWIND. 2012. Disponível em:
[http://en.wikipedia.org/wiki/Whirlwind_\(computer\)](http://en.wikipedia.org/wiki/Whirlwind_(computer)). Acesso em 24 fev. 2012.

WIEGERS, K. E. **Software requirements**. 2. ed. Redmond, WA, USA: Microsoft Press, 2003, 516p.

WIEGERS, K. E. **More about software requirements: thorny issues and practical advice**. Redmond, WA, USA: Microsoft Press, 2006. 201p.

WINKLER, S.; VON PILGRIM, J. A survey of traceability in requirements engineering and model-driven development. **Software systems modeling**, v. 9, n. 4, Springer Verlag, Sept. 2009. Disponível em:
<http://www.springerlink.com/content/u7158k450t7x41ll/fulltext.pdf>. Acess em: 1 set. 2011.

YOUNG, R. R. **The requirements engineering handbook**. Norwood, MA, USA: Artech House, Inc., 2004. 254 p.