ESTE EXEMPLAR CORRESPONDE A REDAÇÃO FINAL DA TESE DEFENDIDA POR Marcos Sampaio Martins E APROVADA PELA COMISSÃO JULGADORA EM 14/06/2010

ORIENTADOR

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA

Marcos Sampaio Martins

Aplicação da DSM no Processo de Análise de Segurança no Desenvolvimento de Aeronaves Comerciais

Campinas, 2010

68/2010

Marcos Sampaio Martins

Aplicação da DSM no Processo de Análise de Segurança no Desenvolvimento de Aeronaves Comerciais

Dissertação apresentada ao Curso de Mestrado da Faculdade de Engenharia Mecânica da Universidade Estadual de Campinas, como requisito para a obtenção do título de Mestre em Engenharia Mecânica.

Área de Concentração: Mecânica dos Sólidos e Projeto Mecânico

Orientador: Prof. Dr. Franco Giuseppe Dedini
Co-orientador:

Campinas
2010

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

Martins, Marcos Sampaio
M366a Aplicação da DSM no processo de análise de segurança no desenvolvimento de aeronaves comerciais / Marcos Sampaio Martins – Campinas, SP: [s.n.], 2010.

Orientador: Franco Giuseppe Dedini
Dissertação de Mestrado - Universidade Estadual de Campinas, Faculdade de Engenharia Mecânica.

1. Aeronaves 2. Projetos 3. Segurança 4. Planejamento 5. Matrizes I. Dedini, Franco Giuseppe. II. Universidade Estadual de Campinas. Faculdade de Engenharia Mecânica. III. Título.

Titulo em Inglês: Application of the design structure matrix in the safety assessment process used in commercial aircraft design.
Palavras-chave em Inglês: Aircraft, Design, Safety, Planning, Matrices.
Área de concentração: Mecânica dos Sólidos e Projeto Mecânico.
Titulação: Mestre em Engenharia Mecânica.
Banca examinadora: Luis Gonzaga Trabasso, Kamal Abdel Radi Ismail.
Data da defesa: 14/06/2010.
Programa de Pós-Graduação: Engenharia Mecânica.

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA MECÂNICA
COMISSÃO DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA
DEPARTAMENTO DE PROJETO MECÂNICO

DISSERTAÇÃO DE MESTRADO ACADEMICO

Aplicação da DSM no Processo de Análise de Segurança no Desenvolvimento de Aeronaves Comerciais

Autor: Marcos Sampaio Martins
Orientador: Prof. Dr. Franco Giuseppe Dedini
Co-orientador:

A Banca Examinadora composta pelos membros abaixo aprovou esta Dissertação:

Prof. Dr. Franco Giuseppe Dedini, Presidente
Universidade Estadual de Campinas - Faculdade de Engenharia Mecânica.

Prof. Dr. Luis Gonzaga Trabasso
Instituto Tecnológico de Aeronáutica - Divisão de Engenharia Mecânica Aeronáutica.

Prof. Dr. Kamal Abdel Radi Ismail
Universidade Estadual de Campinas - Faculdade de Engenharia Mecânica.

Campinas, 14 de junho de 2010

Dedico este trabalho a Deus.

Dedico este trabalho ao meu pai Marcos, minha mãe Maria Helena e meu irmão Marcelo.

Dedico também este trabalho a minha namorada Ligia.

Agradecimentos

Aos meus pais e ao meu irmão pelo apoio e incentivo durante toda a minha vida.

A minha namorada pelo apoio e compreensão nos momentos em que tive que ficar trabalhando na dissertação durante os finais de semana e durante minhas férias.

Ao meu co-orientador técnico e amigo Eng. Jaures Cardoso Junior pelo inestimável apoio nestes quase dois anos de preparação do trabalho. Sua paciência e sua dedicação ao me ensinar foram fundamentais para a execução deste trabalho.

Ao meu orientador Prof. Dr. Franco Giuseppe Dedini que me ofertou o desafio e me auxiliou durante as etapas do trabalho.

A EMBRAER (Empresa Brasileira de Aeronáutica S/A), pelo incentivo e suporte na execução desta dissertação de mestrado.

Ao meu amigo Eng. Antonio Pantalena pelo grande incentivo durante todos estes anos.

Aos engenheiros Guilherme Moreschi, Galvani Lacerda, Arlindo Figueiredo e Pedro Godoy pelo grande apoio durante as reuniões sobre o processo de *Safety Assessment*.

Finalmente, a todos os meus amigos da engenharia de confiabilidade da frota da Embraer.

...Sem saber o que sou e por que estou aqui, a vida é impossível.
Liév Tolstói, Anna Kariênina

Resumo

No desenvolvimento e na integração dos sistemas de uma aeronave comercial, o fator segurança em projeto (*Design for Safety*) é considerado de fundamental importância, uma vez que a certificação do produto pelas autoridades internacionais de aviação civil exige o cumprimento de regulamentações que atestam que um determinado produto (aeronave, componente ou sistema) possui as características mínimas que assegurem seu uso seguro para o tipo de operação pretendida. O processo de projeto que permite a consecução de tais objetivos é conhecido como *Safety Assessment*, que envolve a aplicação de técnicas de análise de risco desde as fases iniciais de projeto até a certificação de tipo do produto. Este processo, devido a sua complexidade, apresenta interdependência entre atividades (ciclo de iteração) e este cenário representa uma barreira para a redução do tempo de ciclo de projeto. Para endereçar esta questão, é proposto um método que tem como foco a aplicação da matriz de estrutura de projetos (*Design Structure Matrix*), que é uma técnica de programação de projeto cujo objetivo é o de otimizar o sequenciamento das atividades do processo através do mapeamento do fluxo de informações e da identificação dos ciclos de iteração.

Palavras Chave

- Aeronaves, Projetos, Segurança, Planejamento, Matrizes.

Abstract

In the commercial aircraft systems development & integration, the safety factor in design (Design for Safety) is considered of fundamental importance since the certification of the product by the international civil aviation authorities requires compliance with regulations that prove that a product (aircraft, component or system) has the minimum requirements to ensure its safe use for the type of intended operation. The design process that allows such purposes is known as "Safety Assessment", which involves the application of hazard analysis techniques since the initial stages of design until the product type certification. This process, due to its complexity, has interdependence between activities (iteration cycles) and this scenario represents a barrier to reducing cycle time on projects. In order to address this issue, a method is proposed focusing on the application of the “Design Structure Matrix", wich is a project management tool whose objective is to optimize activities sequencing through the mapping flow of information and identify the cycles of iteration in the process.

Key Words

- Aircraft, Design, Safety, Planning, Matrices.

Lista de Ilustrações

Lista de Tabelas

Lista de Abreviaturas e Siglas

AEAP – *As Early As Possible.*
AFHA – *Aircraft Functional Hazard Assessment.*
ALAP – *As Late As Possible.*
ANAC – Agência Nacional de Aviação Civil.
ARP – *Aerospace Recommended Practice.*
ATA – *Air Transport Association.*
CCA – *Common Cause Analysis.*
CMA – *Common Mode Analysis.*
CS – *Certification Specifications.*
DAL – *Development Assurance Level.*
DFM – *Design for Manufacturability.*
DFX – *Design for "X".*
DIP – Desenvolvimento Integrado do Produto.
DoD – *Department of Defense.*
DPM – Departamento de Projeto Mecânico.
DSM – *Design Structure Matrix.*
EAP – Estrutura Analitica de Projeto.
EASA – *European Aviation Safety Agency.*
FAA – *Federal Aviation Administration.*
FAR – *Federal Aviation Regulations.*
FHA – *Functional Hazard Analysis.*
FMEA – *Failure Mode and Effects Analysis.*
FTA – *Fault-Tree Analysis.*
HAZOP – *Hazard and Operability Analysis.*
IDEC – Instituto de Defesa do Consumidor.
INCOSE – *International Council on Systems Engineering.*
IPCC – *Intergovernmental Panel on Climate Change.*
LCC – *Life Cycle Cost.*
LCD – *Life Cycle Design.*
MIL-STD – *Military Standard.*
NASA – *National Aeronautics and Space Administration.*
PIB – Produto Interno Bruto.
PRA – *Particular Risk Analysis.*
PSSA – *Preliminary System Safety Assessment.*
SFHA – *System Functional Hazard Assessment.*
SSA – *System Safety Assessment.*
TCDS – *Type Certificate Data Sheet.*
UNICAMP – Universidade Estadual de Campinas.
WBS – *Work Breakdown Structure.*
ZSA – *Zonal Safety Analysis.*

Sumário

1 Introdução

1.1 Justificativa e Importância do Trabalho

Nos últimos 40 anos, o processo de desenvolvimento de aeronaves comerciais experimentou um grande salto evolutivo em função da criação de requisitos de certificação por parte das autoridades mundiais de aviação civil.

A criação destes requisitos, desencadeada pela tendência de estabilização (em um nível não aceitável) da taxa de acidentes com fatalidades na aviação mundial, foi um marco na indústria aeronáutica mundial uma vez que gerou mudanças tanto na forma de como projetar os sistemas quanto na cultura de segurança que passou a ser encorajada pelas organizações (Serra, 2006).

Outro fator que indiretamente estimulou as autoridades de aviação civil a desenvolver requisitos de certificação mais rigorosos, robustos e completos, foi a crescente demanda de vôos e passageiros que vinha se desenhando durante a década de 60 e que era projetada para os anos seguintes.

Segundo a *Air Transport Association* - ATA (2009), entre 1970 e 2008, a quantidade de passageiros oriundos do transporte aéreo cresceu a uma taxa de aproximadamente 5% ao ano. O crescimento do PIB mundial, bem como mudanças na tecnologia empregada e no ambiente regulatório na aviação civil, tiveram importante participação neste crescimento (IPCC, 2009).

Com este aumento da demanda pelo transporte aéreo, os fabricantes passaram a desenvolver aeronaves capazes de transportar muito mais passageiros que as desenvolvidas até então. Esse

cenário demandava uma exigência muito grande com relação aos niveis de segurança a serem buscados para os sistemas e componentes das aeronaves.

Esta exigência era atendida pelo cumprimento do requisito FAR-25.1309, desenvolvido pela autoridade americana de aviação civil (FAA) na década de 70, que posteriormente serviu como base para a criação do requisito europeu e de outras nações fabricantes de aeronaves. O salto tecnológico ocorrido na década de 80, com a utilização cada vez maior de *software* e sistemas altamente integrados, fez que com que fossem desenvolvidos na última década, materiais interpretativos e práticas recomendadas de engenharia de forma a auxiliar os fabricantes na demonstração de atendimento do requisito perante as autoridades.

Estas práticas recomendadas, desenvolvidas em conjunto por fabricantes de aeronaves, autoridades e centros de pesquisa, são utilizadas até os dias de hoje e estão em constante evolução. Tendo como base abordagens, metodologias e sistemáticas de projeto aplicáveis ao nível de complexidade apresentado pelo produto, estas práticas recomendadas englobam os seguintes conceitos:

- Conceito de ciclo de vida do produto.
- Desenvolvimento integrado do produto.
- Projeto voltado à segurança (*Design for Safety*).
- Engenharia de sistemas.
- Técnicas de análise de risco.

Como conseqüência, o processo de desenvolvimento de aeronaves comerciais passou a adotar estes conceitos de forma a garantir cumprimento dos requisitos de segurança exigidos para que o produto seja certificado e possa operar em qualquer parte do mundo.

Ao processo de engenharia responsável por fornecer as ferramentas necessárias para que se possa determinar, de maneira organizada, sistemática e abrangente se os níveis de segurança de um sistema estão sendo atingidos, dá-se o nome de *Safety Assessment*.

Este processo é longo e complexo, uma vez que se inicia durante as fases mais preliminares do projeto e termina ao final da certificação da aeronave. Além disso, apresenta interdependência entre atividades (ciclos de iteração), que é uma característica comum em processos complexos de desenvolvimento de produto.

Em função de sua complexidade, o processo de *Safety Assessment* exige um planejamento estruturado que antecipe gargalos de atividades no decorrer de seu andamento e que forneça um sequenciamento de atividades otimizado visando sua maior eficiência.

Muitas das técnicas tradicionais de programação de projetos, como o Diagrama de Gantt e o PERT/CPM permitem apenas a modelagem de atividades seqüenciais e paralelas, mas não permitem a modelagem de relacionamentos interdependentes.

Para endereçar esta questão, uma técnica de programação de projeto conhecida como matriz de estrutura de projetos ou DSM (*Design Structure Matrix*) foi desenvolvida de forma a permitir a representação do fluxo de informação entre as atividades de um processo visando determinar a sequência mais sensível (ou mais otimizada) para a modelagem das atividades (Yassine, 2004).

A correta aplicação da DSM no gerenciamento de um processo complexo pode representar a minimização do tempo de ciclo e a redução dos custos de desenvolvimento, aumentando a competitividade do produto.

1.2 Objetivos e Estrutura do Trabalho

Este trabalho tem como objetivo a definição de um método visando a aplicação da DSM no gerenciamento do processo de *Safety Assessment* empregado no desenvolvimento de aeronaves comerciais. Este método visa, além de modelar o processo, otimizar o sequenciamento entre as atividades e identificar os ciclos de iteração existentes visando seu gerenciamento.

O trabalho está estruturado em 6 capítulos. O capítulo 1 é composto por esta introdução, contendo a justificativa e a importância do trabalho bem como os objetivos e a estrutura empregada.

O capítulo 2 apresenta a revisão de literatura, constituída do seguinte conteúdo: conceito de desenvolvimento do produto voltado a segurança, a segurança de sistemas aplicado no desenvolvimento de aeronaves comerciais, o gerenciamento do processo de projetos e a matriz de estrutura de projetos (DSM).

O capítulo 3 visa propor um método a ser implementado visando o emprego da matriz de estrutura de projetos (DSM) para otimização das atividades de um processo complexo.

O capítulo 4 apresenta a aplicação do método proposto no capítulo 3 em um processo real de uma empresa, através da modelagem e do gerenciamento de suas atividades visando o mapeamento do fluxo de informação existente, a aplicação da DSM e a construção de uma proposta de programação das atividades através do Diagrama de Gantt. É neste capítulo que se encontram os resultados obtidos.

Finalmente, o capítulo 5 apresenta a discussão dos resultados obtidos na aplicação do método proposto, enquanto que o capítulo 6 apresenta as conclusões e sugestões para próximos trabalhos.

2 Revisão da Literatura

2.1 O Desenvolvimento do Produto Voltado à Segurança

O desenvolvimento do produto voltado à segurança engloba o estudo de alguns conceitos de engenharia importantes, como: conceito de ciclo de vida do produto, as etapas de desenvolvimento, projeto voltado ao ciclo de vida, engenharia simultânea, projeto voltado a maximização de atributos (DFX) e finalmente o conceito de projeto voltado à segurança.

2.1.1 Conceito de Ciclo de Vida do Produto

Segundo Prasad (1996), o ciclo de vida de um produto (*Product Life-Cycle*) representa os estágios nos quais o produto percorre desde a sua concepção até seu descarte. De uma maneira geral, estas etapas podem ser melhor representadas pela figura 2.1, detalhada a seguir:

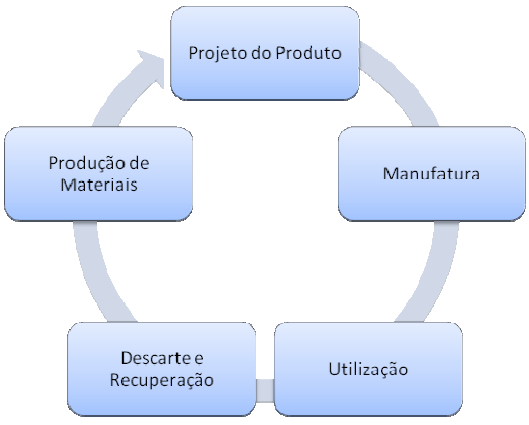


Figura 2.1: Ciclo de Vida do Produto (Fonte: adaptado de http://msl1.mit.edu).

- Projeto e desenvolvimento do produto.
- Manufatura.
- Utilização.
- Descarte / Recuperação (visando a produção de matéria prima primária).

O desenvolvimento do produto é a fase onde as decisões mais importantes de ordem econômica são tomadas, uma vez que o custo de vida do produto (*Life Cycle Cost* ou LCC) tem grande representatividade nesta etapa.

Segundo Bralla (1996), o LCC representa os custos envolvidos em todo ciclo de vida do produto, ou seja, os custos de aquisição (desenvolvimento e manufatura), os custos de operação (manutenção, reparo, atualização, *liability*, etc) e os custos de descarte (armazenamento, aterro, reciclagem etc.).

Ainda segundo Bralla (1996), grande parte do LCC, cerca de 80%, está contida na fase de desenvolvimento. Fica evidente que não é possivel produzir um produto com um baixo LCC caso não se considere este objetivo nesta etapa de seu ciclo de vida.

2.1.2 Etapas de Projeto e Desenvolvimento do Produto

De acordo com o dicionário Merriam-Webster (2009), a palavra *design* (projeto), pode ter as seguintes definições: Como um verbo, pode significar o ato de criar, conceber, executar ou construir de acordo com um plano ou metodologia. Já como um substantivo, a palavra *design* pode significar um objetivo, um desenho, um plano voltado ao cumprimento de algo ou mesmo a arte criativa de executar projetos funcionais ou estéticos.

Segundo Bralla (1996), o projeto do produto é um processo criativo, uma vez que mudanças conceituais e de configuração são inevitáveis durante suas principais etapas.

Giudice, La Rosa e Risitano (2006) afirmam que o projeto do produto é entendido como uma atividade que aplica técnicas científicas ou princípios visando a transformação das informações que caracterizam e descrevem a demanda pelo produto, nos sistemas capazes de satisfazer esta demanda.

Segundo Ulrich e Eppinger (2008), o desenvolvimento de um produto representa um conjunto de atividades que se inicia com a percepção de uma oportunidade de mercado e termina com a produção, venda e entrega do produto. Ainda segundo eles, três funções são centrais para o desenvolvimento do produto:

- *Marketing*
    - Identificação de oportunidades de produtos e segmentos de mercado.
    - Prospecção das necessidades dos clientes.
    - Promoção do produto.
- *Design* (Projeto)
    - Definição da forma física do produto.
    - Projeto de engenharia (elétrico, mecânico, *software* etc).
    - Projeto industrial (estético, ergonômico, interface com usuário).
    - Atendimento das expectativas dos clientes.
- Manufatura
    - Definição do sistema de produção.
    - Atividades da cadeia de suprimentos (*Supply-Chain Activities*).

Ainda segundo Ulrich e Eppinger (2008), o processo genérico que rege o projeto e desenvolvimento do produto baseia-se em uma seqüência de seis fases que uma empresa emprega para conceber, projetar e comercializar um produto. Estas fases são:

- Planejamento.
    - Missão do projeto, objetivos de negócio, *target* de mercado.

- Desenvolvimento do Conceito.
    - Definição das especificações, funções e características do produto.
- Projeto em nível de sistema.
    - Arquitetura do produto e decomposição (sistemas, subsistemas etc.)
- Projeto detalhado.
    - Especificação de materiais, geometrias, tolerâncias etc.
- Teste e refinamento.
    - Desenvolvimento de protótipo, ferramental e avaliação pré-produção.
- *Ramp-up* de Produção
    - Manufatura do produto usando o sistema de produção planejado.

A figura 2.2 representa as fases descritas acima:

Figura 2.2: Processo genérico de desenvolvimento do produto

(Fonte: Adaptado de Ulrich e Eppinger 2008).

Segundo Dedini (2007), um marco importante para o desenvolvimento de produtos foi a proposição e a difusão das metodologias de projeto que buscam encontrar uma seqüência de etapas e atividades considerada mais adequada para se desenvolver um produto. Algumas das principais metodologias de projeto existentes são:

- Metodologia segundo Asimow (1968)
    - Trabalho pioneiro no desenvolvimento de metodologias de projeto que apresenta um processo que se desenvolve por meio de uma série de fases (estudo de exeqüibilidade, projeto preliminar, projeto detalhado, planejamento para produção, etc.) onde uma nova fase não começará antes que a anterior tenha sido completada.

- Metodologia segundo Blanchard e Fabrycky (1981)
    - Salienta a necessidade do desenvolvimento de produtos com um enfoque no ciclo de vida do produto e utilizando uma visão de sistemas (engenharia de sistemas). Nesta metodologia, o projeto se divide em seis fases: projeto conceitual, projeto preliminar, detalhamento dos sistemas, produção, suporte do ciclo de vida e retirada do sistema.

- Metodologia segundo Pahl & Beitz (1988)
    - Introduz o conceito de projeto sistemático, onde a identificação das necessidades, o projeto conceitual, o projeto preliminar e o projeto detalhado, representam as fases principais do processo. É uma metodologia que tem uma abordagem voltada tanto para o desenvolvimento do produto em si quanto nas atividades de projeto.

As diferenças entre as metodologias de projeto se apresentam principalmente em função da abordagem utilizada e do nível de detalhamento das atividades que as compõem.

2.1.3 O Projeto Voltado ao Ciclo de Vida do Produto

Segundo Knopf, Gupta e Lambert (2007), quando o projeto do produto é executado como um projeto voltado ao ciclo de vida (LCD ou *Life Cycle Design*), ele leva em consideração não apenas os aspectos ligados a manufatura e etapas anteriores, mas também os aspectos das etapas sub-sequentes ao ciclo de vida do produto.

Segundo Giudice, La Rosa e Risitano (2006), o LCD é uma intervenção de projeto que considera todas as fases do ciclo de vida do produto dentro do contexto do processo de desenvolvimento do produto, visando atender as expectativas do cliente e da sociedade. Além disso, é predominantemente orientada na direção da otimização do desempenho do produto em seu ciclo de vida.

Knopf, Gupta e Lambert (2007), afirmam que a avaliação dos potenciais impactos causados pelos diferentes aspectos do ciclo de vida do produto por todo o processo de projeto e desenvolvimento requer uma intervenção sistemática, integrada e simultânea e, neste sentido, duas abordagens metodológicas para a melhoria do projeto do produto podem ser aplicadas:

- Engenharia Simultânea
- Projeto Voltado à Maximização de Atributos ou "*Design for X*" (DFX)

Estas abordagens de projeto, apesar de surgirem de diferentes premissas, levam em consideração o conceito do projeto voltado ao ciclo de vida do produto.

2.1.4 Engenharia Simultânea

Prasad (1996) salienta que antes da metade da década de 1980, o processo de desenvolvimento e projeto do produto tinha uma abordagem seqüencial, onde as tarefas eram funcionalmente distribuídas e executadas por grupos departamentais independentes.

Segundo Singh (1996), na abordagem seqüencial (Figura 2.3), o fluxo de informação flui de forma sucessiva fase a fase. Neste tipo de configuração, as especificações de projeto são muitas vezes revisitadas em função de mudanças ao longo do processo de desenvolvimento do produto. Isto causa, basicamente, as seguintes conseqüências:

- Baixo desempenho do processo de desenvolvimento do produto.
- Aumento do *lead time* (tempo requerido para execução) do projeto.
- Perda de competitividade do produto em função do elevado *time-to-market* (tempo de apresentação do produto ao mercado).
- Aumento nos custos de desenvolvimento e, consequentemente, do LCC do produto.

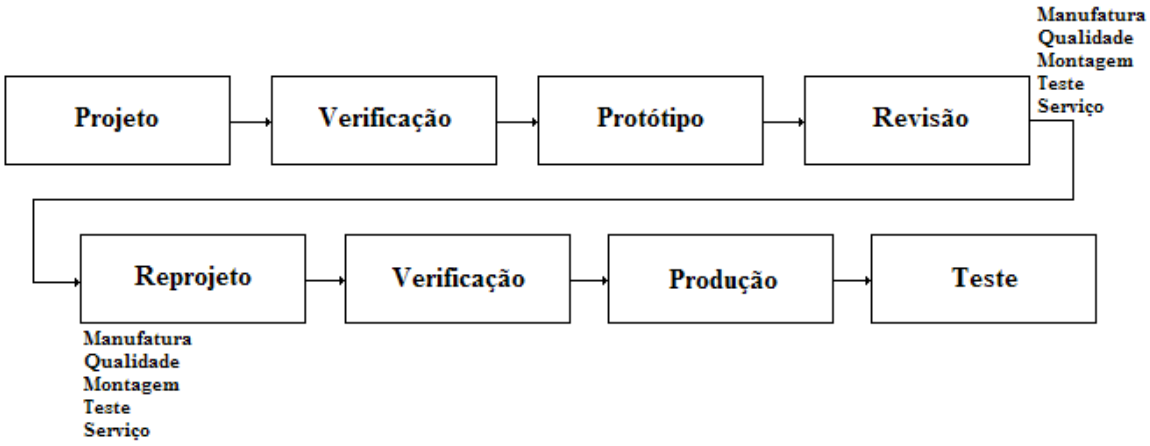


Figura 2.3: Abordagem seqüencial no desenvolvimento do produto.

Ericson (2005) denomina este modelo seqüencial como sendo um "Modelo de Desenvolvimento de Engenharia" onde cada fase deve estar finalizada com sucesso antes de outra fase ser iniciada. Três grandes revisões de projeto são conduzidas em cada mudança de fase, sendo uma ao final da fase de concepção do produto (*System Design Review*), outra ao final do projeto preliminar (*Preliminary Design Review*) e a última ao final do projeto detalhado (*Critical Design Review*).

Segundo Prasad (1996), a partir do final da década de 1980, a indústria passou a analisar sua situação mais criticamente uma vez que com a crescente complexibilidade dos produtos, dos requisitos de projeto e da competitividade, as mesmas tarefas que costumavam ser executadas individualmente por cada departamento não poderiam mais ser executadas de forma independente.

Sendo assim, o modelo seqüencial de projeto e desenvolvimento do produto passou a ser ineficiente uma vez que possuindo uma taxa muito baixa de transferência de informação e um controle muito elevado, resultava em tempos de desenvolvimento elevados para o mercado. Neste sentido, é criado o conceito da engenharia simultânea.

Segundo Singh (1996), a engenharia simultânea é um esforço combinado de integração do processo de desenvolvimento do produto onde a redução de custos, a melhoria da qualidade e da produtividade, e a redução do tempo de desenvolvimento são alcançadas.

Prasad (1996) define a engenharia simultânea como sendo uma abordagem sistemática para o desenvolvimento integrado do projeto de um produto onde vários elementos de seu ciclo de vida são considerados como parte do processo de desenvolvimento (manufatura, custo, prazo, qualidade, entre outros). Estes elementos não servem somente para se atingir as funcionalidades básicas do produto, mas para definir um produto que atenda todas as necessidades dos clientes.

Uma das principais motivações desta abordagem de desenvolvimento do produto reside na busca pela redução do *time-to-market*, uma vez que além de resultar em vantagem competitiva, leva em consideração, de forma simultânea, todos os aspectos do projeto que afetam o ciclo de vida do produto onde times multifuncionais podem atuar.

As figuras 2.4 e 2.5 representam o conceito da engenharia simultânea e do projeto integrado do produto, respectivamente:

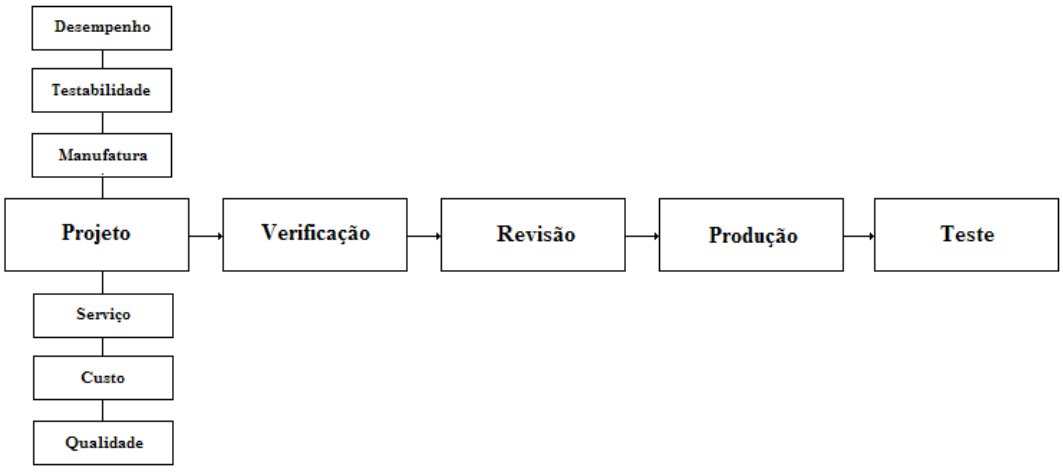


Figura 2.4: Abordagem simultânea no desenvolvimento do produto
(Fonte: Adaptado de Singh 1996).

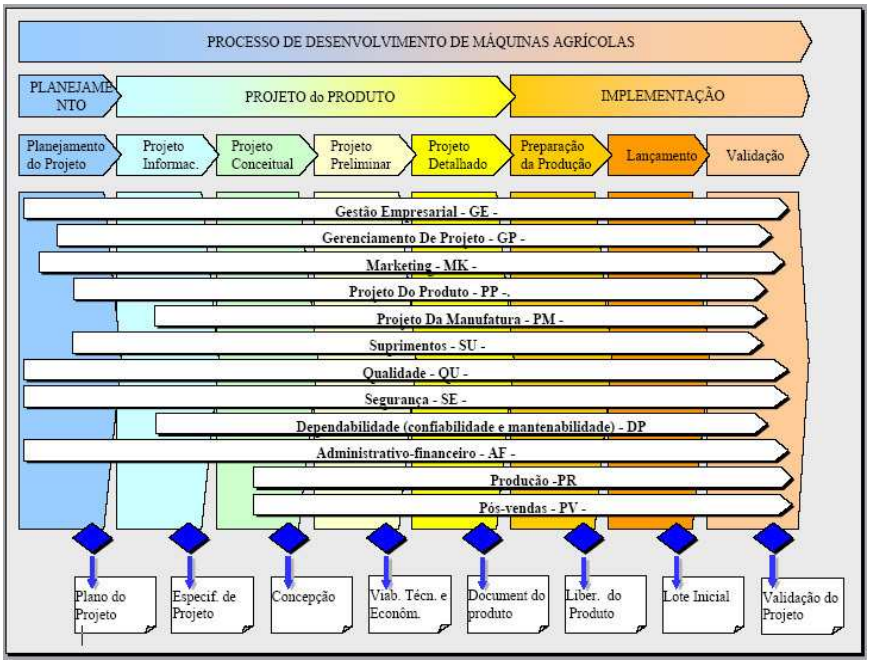


Figura 2.5: Exemplo: Modelo de Desenvolvimento Integrado do Produto no processo de desenvolvimento de máquinas agrícolas. (Fonte: Back 2007).

Segundo Back (2007), o termo engenharia simultânea também é usado para expressar o desenvolvimento integrado do produto (DIP), que considera todo o processo de transformação e geração de informações necessárias na identificação da demanda, na produção e no uso do produto.

Segundo Giudice, La Rosa e Risitano (2006), três fatores demonstram a afinidade entre o LCD e a engenharia simultânea: a integralidade do processo de desenvolvimento do produto, a condução de intervenções de projeto nas fases preliminares e a simultaneidade na análise e síntese de problemas de projeto.

Como apresentado antes, o LCD considera todas as fases do ciclo de vida do produto dentro do contexto do processo de desenvolvimento. Dependendo do atributo que se deseja considerar (confiabilidade, meio ambiente, mantenabilidade, segurança etc), pode ser necessária a priorização de um atributo em detrimento de outro. Neste caso, quando se deseja balancear os atributos, utiliza-se o conceito do "*Design for X*" ou DFX.

2.1.5 Projeto Voltado à Maximização de Atributos (DFX)

Segundo Bralla (1996), o conceito de DFX (*Design for X*) surgiu no final da década de 1980 na AT&T Bell Laboratories, onde foi reconhecida a necessidade de satisfazer os objetivos de cada atributo desejável de projeto de produto. O DFX evoluiu do conceito de DFM (*Design for Manufacturability*), cujo conceito vinha sendo estudado desde a década de 60 pela GE.

O DFM é uma técnica analítica e com base no conhecimento (experiência e julgamento de engenharia) que invoca uma série de princípios, *guidelines*, recomendações ou regras para o projeto de um produto de maneira a facilitar a sua manufatura e minimizar seu custo.

Bralla (1996) afirma que historicamente o processo de desenvolvimento do produto subestima várias outras características em detrimento de três fatores principais: desempenho, aparência e funcionabilidade. Para ele, alguns outros atributos desejáveis para o projeto de um produto podem ser listados. São eles:

- Confiabilidade.
- Conformidade com as especificações ou requisitos aplicados ao produto.
- Durabilidade.
- Mantenabilidade.
- Qualidade percebida.
- Seguranca (*Safety*).
- Facilidade de manufatura (*Manufacturability*).
- Meio ambiente (*Environmental Friendliness*).
- Ergonomia e Facilidade de uso (*User-Friendliness*).
- Facilidade de modificação do produto (*Upgradability*).

Huang (1996), define o DFX como sendo a mais efetiva abordagem na implementação da engenharia simultânea, uma vez que enfatisa em vários atributos que são trabalhados de forma cooperativa nos times multi-funcionais onde todos os problemas de interação no projeto do produto são considerados.

Para Bralla (1996), o DFX pode ser definido como uma abordagem com base no conhecimento que visa maximizar todos os atributos desejáveis no projeto do produto, ao mesmo tempo minimizando o seu custo de vida (*Life Cycle Cost*). O atingimento destes objetivos constitui a excelência do projeto do produto (*Design for Excellence*).

Existem dois aspectos básicos no DFX:

- Aspecto técnico
    - Como a engenharia do produto incorpora os atributos desejáveis de projeto.
- Aspecto gerencial
    - Como uma empresa ou organização deve gerenciar este processo de projeto.

Bralla (1996) afirma que do ponto de vista gerencial, o sucesso da aplicação do DFX depende dos seguintes fatores:

- Comprometimento gerencial
    - Flexibilidade a mudanças e a participação ativa dos altos níveis gerenciais.

- Redirecionamento dos esforços de projeto
    - Um DFX efetivo requer a utilização de engenharia simultânea, atuação de times de projeto multifuncionais.

- Trabalho em equipe
    - Cooperação entre a engenharia de desenvolvimento do produto e outras áreas da empresa.
    - Comunicação aberta, baixo nível de conflito, objetivos claros e bem definidos, e liderança efetiva.

- Liderança técnica
    - Promoção motivacional ao time, medindo o progresso do projeto, promovendo reuniões periódicas, premiando os participantes do time, realizando *brainstorming* e executando *benchmarking*.

O aspecto técnico lida com a forma com que é modificado o atual processo de projeto do produto visando a incorporação dos atributos técnicos de projeto que se deseja maximizar.

Quando a ênfase está na busca do aumento dos níveis segurança do produto em seu ciclo de vida (principalmente na fase de operação e utilização) através da utilização de metodologias de projeto que garantam o atingimento destes objetivos, a dimensão utilizada do DFX é conhecida como *Design for Safety*.

2.1.6 Projeto Voltado à Seguranca (*Design for Safety*)

O estudo do conceito de projeto voltado a segurança considera os seguintes aspectos:

a) A Segurança como um Atributo de Projeto

Segundo Dedini (2007), entre todas as regras que asseguram o desenvolvimento de um bom projeto destacam-se as três seguntes: um projeto tem que ser simples, seguro e inequívoco. Estas regras por sua vez estão relacionadas com três metas gerais de um projeto:

- Satisfação da função técnica.
- Viabilidade Econômica.
- Segurança para o homem e o meio ambiente.

Ainda segundo Dedini (2007), num projeto seguro alguns requisitos de segurança podem gerar grande complexidade ao processo de desenvolvimento e, consequentemente, um elevado custo do produto. Neste sentido, a relação entre a segurança e custo deve assumir um padrão ótimo que é dado pela melhor relação entre complexidade e necessidades de segurança. Num projeto, o atributo segurança distingue-se em quatro áreas distintas:

- Segurança construtiva
    - Segurança voltada à prevenção de falhas de um sistema.
- Segurança funcional
    - Segurança voltada à garantia da realização de uma função desejada.
- Segurança operacional (Fatores Humanos)
    - Segurança voltada ao homem durante o processo de operação.
- Segurança ambiental
    - Segurança do processo em relação às pessoas e em relação ao meio ambiente.

Bralla (1996) afirma que a segurança é um atributo de projeto muito importante e que deve ser considerado desde as fases mais iniciais de projeto do produto. Tanto a qualidade quanto a confiabilidade do produto estão intrinsecamente relacionadas com a segurança, onde deficiências relacionadas a estes dois atributos podem gerar condições de risco. Sendo assim, estes atributos devem ser adequadamente considerados e controlados ao projetar um produto voltado à maximização da segurança.

De uma maneira geral, o projeto de um produto envolve inicialmente o projeto de cada um de seus sistemas e alguns destes sistemas podem ser mais críticos com relação à segurança do que outros. Estes sistemas são conhecidos como de segurança crítica (*safety-critical*).

Segundo Park J.Y e Park Y.W (2004), sistemas de segurança crítica são aqueles cuja falha ou mau-funcionamento pode ameçar a vida humana e podem causar perda ou dano severo a equipamentos ou ao meio-ambiente.

De acordo com Bralla (1996), a maior qualidade do produto é atingida com a minimização do custo de vida do produto (LCC), onde os custos resultantes de eventos geradores de risco à segurança estão incluídos. Estes custos podem ser repassados ao fabricante do produto através de um artifício legal conhecido como responsabilidade civil sobre o produto ou *product liability*.

b) A Responsabilidade Civil Sobre o Produto

A responsabilidade civil sobre o produto (*product liability*) descreve uma ação (processo judicial) no qual a parte demandante procura recuperar do acusado (fabricante, por exemplo), os danos causados por perda ou prejuízos físicos, quando a causa alegada é a falha do produto.

Segundo o IDEC – Instituto Brasileiro de Defesa do Consumidor (2009), o Código de Defesa do Consumidor (Lei nº 8.078, de 11 de setembro de 1990) em seu capítulo IV (artigos de 8 a 18) fornece os meios legais da responsabilidade civil do produto no qual fabricantes, distribuidores, fornecedores estão sujeitos.

Nos EUA, país pioneiro na utilização desta postura, as principais alegações normalmente associadas com a responsabilidade civil sobre o produto são a negligência (omissão, falha ao alertar etc), a responsabilidade no produto (defeito de fabricação, defeito de projeto etc) e a violação de garantia (*warranty claim*). Lá, a maioria das leis de responsabilidade sobre o produto é determinada no nível estadual e podem variar bastante de estado para estado.

Bralla (1996) afirma que as implicações do *product liability* a serem consideradas sobre o processo de desenvolvimento do produto são:

- Incorporação de todos os dispositivos técnicos de segurança no projeto do produto.
- Assumir, durante o processo de projeto, a possibilidade de possíveis litígios futuros.
- Antecipar a utilização incorreta do produto por parte do usuário e considerar estas implicações no projeto do produto.
- Prever as formas de utilização do produto sobre diferentes óticas (manutenção, transporte, estocagem, reparo, etc.) de forma a minimizar os riscos associados.
- Documentar de forma clara e honesta todas as etapas de projeto do produto.
- Manter banco de dados referente ao desenvolvimento do produto (testes, ensaios, etc.) e a operação (problemas de campo, reclamações de clientes, ações corretivas efetuadas, eficácia das soluções apresentadas, etc.) visando a indicação do forte comprometimento com a segurança.

- Desenvolver meios de alertar o usuário em caso de condições de operação insegura.
- Desenvolver, em ambiente de engenharia simultânea, um processo de análise de risco através da atuação de um grupo focado nesta atividade.
- Desenvolver um produto não somente para minimizar as chances de falha, mas também minimizar a possibilidade de ferimentos, caso uma falha ocorra.
- Desenvolver manuais ou treinamentos de operação e manutenção contendo todas as informações necessárias para promover uma correta e segura operação do produto.

Segundo Serra (2006), no projeto e desenvolvimento de um avião, os aspectos legais ligados ao *product liability* nos EUA são aplicáveis independentemente do atendimento, por parte do fabricante, dos requisitos de certificação exigidos pelas autoridades de aviação civil.

c) Principais Aplicações do *Design for Safety*

Uma metodologia de projeto voltado à segurança pode ter inúmeras aplicações , conforme apresentado pela figura 2.6:

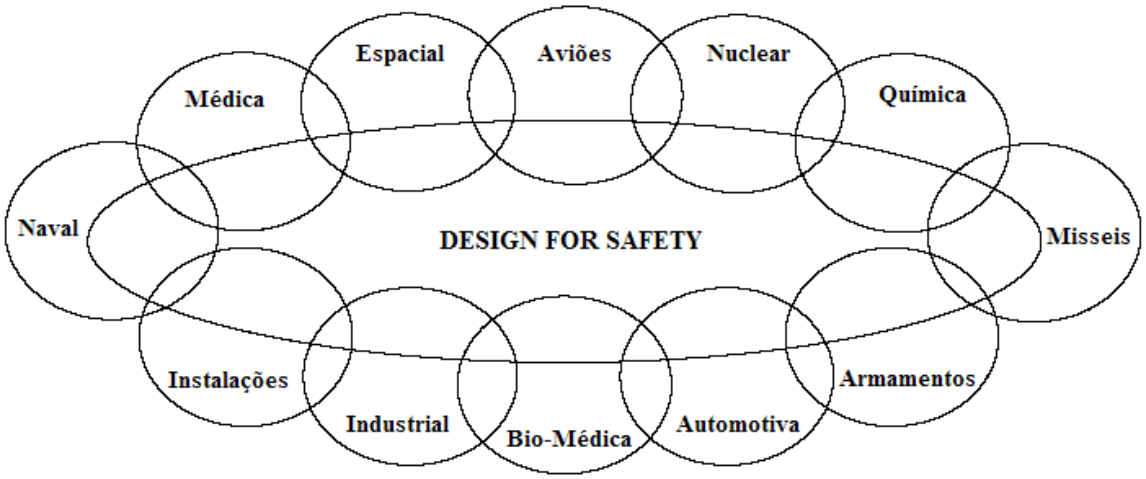


Figura 2.6: "Design for Safety" e suas aplicações (Fonte: Salzmann e Van der Tempel 2006).

Segundo Salzmann e Van der Tempel (2006), o primeiro ramo industrial que requeriu a utilização desta abordagem de projeto foi a indústria médica. Muitos dos equipamentos atualmente utilizados possuem alta complexidade e são desenvolvidos com foco na operação segura (por exemplo: equipamentos de diálise, monitoramento de sinais vitais, marca-passos etc).

Outro ramo importante que vem se desenvolvendo no *Design for safety* é a indústria automotiva que, na trilha da indústria aeronáutica, vem aplicando cada vez mais sistemas eletrônicos e *software* nos automóveis. Salzmann e Van der Tempel (2006) afirmam que esta tendência se iniciou no final da década de 1970 com a introdução dos sistemas digitais de controle nos sistemas de injeção dos motores de combustão e também nos freios, pelo sistema ABS (*Anti-Blocking System*).

Atualmente a aplicação tem se tornado cada vez abrangente, com o desenvolvimento dos sistemas conhecidos como *X-by-Wire*, que consistem basicamente de unidades eletrônicas cujo sinal elétrico de saída é processado por micro-controladores que gerenciam a atividade de comando do motorista por atuadores elétricos. O objetivo principal desta tecnologia é a de aumentar a segurança liberando o motorista das tarefas de rotina e também auxiliá-lo em condições críticas. A figura 2.7 representa um exemplo de sistema automotivo desenvolvido sob a abordagem do *Design for safety*.

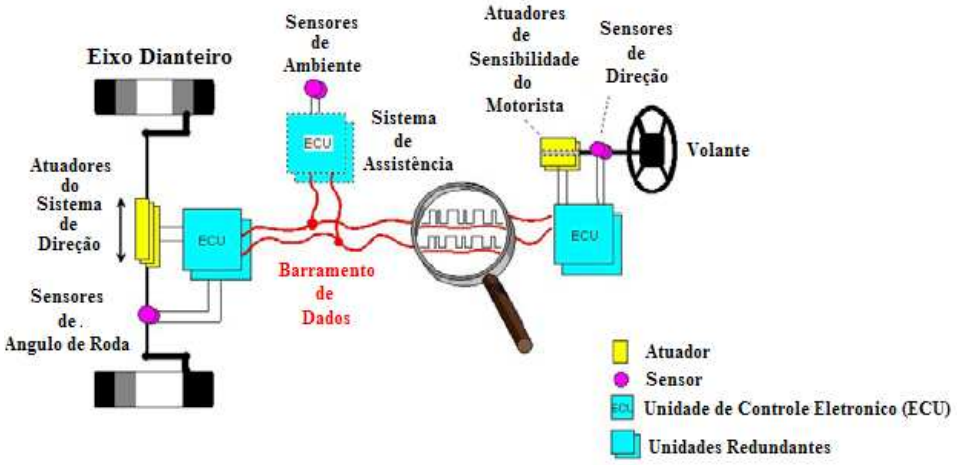


Figura 2.7: Sistema de Direção *Steer-by-Wire* (Fonte: Salzmann e Van der Tempel, 2006).

Apesar da crescente utilização do conceito do *Design for Safety* em vários setores da indústria, o segmento que provavelmente teve a maior evolução e que vem desenvolvendo os mais complexos e sistemáticos processos de desenvolvimento voltados à garantia de elevados níveis de segurança na operação do produto é a indústria aeronáutica e aeroespacial.

Dentre os diversos fatores que contribuíram para a sua grande evolução, destacam-se:

- Busca constante pela redução das taxas de acidentes visando assegurar ao avião o status de um dos meios de transporte mais seguros do mundo.
- Pioneirismo na utilização de técnicas de análise de risco como o FMEA, FTA, e também na aplicação do conceito de fatores humanos no projeto.
- Criação de requisitos de projeto por parte de autoridades (aviação civil ou defesa) visando garantir o cumprimento, por parte dos fabricantes, de níveis de segurança aceitáveis para a certificação do produto.
- Desenvolvimento de práticas recomendadas de engenharia visando o auxílio na condução do processo de *Safety Assessment* por parte dos fabricantes.
- Desenvolvimento de práticas recomendadas de engenharia contendo considerações para o projeto e certificação de sistemas complexos e altamente integrados.
- Criação de comitês técnicos internacionais (formados pelos fabricantes, autoridades e empresas aéreas) voltados à melhoria contínua das práticas recomendadas de engenharia em função de modificações ou revisões de requisito, implantação de novas tecnologias, entre outros.

d) Gerenciamento do Processo de *Design for Safety*

Segundo Bralla (1996), alguns princípios necessários para o bom gerenciamento de um processo de *Design for Safety* são:

- Definição clara por parte da companhia dos níveis de segurança a serem buscados pelo produto.
- Participação ativa de um grupo de especialistas na área de segurança durante as etapas de desenvolvimento do produto.
- Condução de testes e/ou ensaios para validação do projeto dos sistemas visando evidenciar todos os tipos de falha que podem levar a acidentes.
- Desenvolvimento de uma estrutura de engenharia com foco na análise de problemas ou defeitos em campo e com atuação direcionada a rápida disponibilização de ações corretivas.
- Conhecimento profundo dos requisitos de projeto, das práticas recomendadas de engenharia e de *guidelines* por parte do corpo técnico de engenharia.
- Consideração por parte da engenharia de desenvolvimento de todas as possíveis formas de utilização do produto de maneira a antecipar possíveis utilizações incorretas.
- Utilização de técnicas de análise de riscos visando a identificação de modos de falha críticos que podem gerar situações de risco a operação do produto.
- Modificação imediata do produto (*recalls*), caso fique evidenciado problemas de segurança envolvendo a operação do produto em campo.

Para Storey (1999), alguns pontos importantes devem ser cuidadosamente levados em consideração no *Design for Safety*:

I. Gerenciamento de riscos no projeto.
II. Considerações sobre fatores humanos no projeto do produto.
III. Encorajamento da cultura de segurança na organização (*Safety Culture*).

I. Gerenciamento de Riscos no Projeto

Segundo Storey (1999) o gerenciamento de riscos durante o desenvolvimento do produto deve buscar meios para:

- Evitar os defeitos (*Fault Avoidance*).
    - Exemplo: Utilização de técnicas qualitativas ou quantitativas de análise de risco na etapa de projeto.
- Detectar os defeitos (*Fault Detection*)
    - Exemplo: Prover meios de verificação de funcionalidade ou de consistência do sistema durante a operação (Exemplo: *Built-in-Test - BITE*)
- Remover os defeitos (*Fault Removal*)
    - Exemplo: Testes ou ensaios usados na etapa de qualificação do produto.
- Suportar os defeitos (*Fault Tolerance*)
    - Exemplo: Utilização de redundâncias ou barreiras no projeto dos sistemas.

II. Considerações sobre Fatores Humanos no Projeto do Produto

Com relação aos fatores humanos, Storey (1996) afirma que a simplicidade é fator chave no projeto de sistemas *safety-critical*, uma vez que o componente humano, apesar de ter a vantagem da flexibilidade e da adaptabilidade, é imprevisível e não confiável.

Segundo Park J.Y e Park Y.W (2004), os fatores humanos são especialmente importantes para o projeto de sistemas e devem ser integrados no processo de projeto desde as fases iniciais. O ser humano possui considerável influência na segurança e confiabilidade de sistemas.

Cacciabue (1997) afirma que o projeto de um sistema homem-máquina precisa considerar primariamente o processo cognitivo do operador de maneira a prover as interfaces indispensáveis, os meios de tomada de decisão e os procedimentos necessários para o gerenciamento seguro da operação em condições normais ou de emergência.

Ainda segundo Cacciabue (1997), no estudo de fatores humanos, tanto o projeto dos sistemas quanto o processo de *Safety Assessment* têm abordagem prospectiva, ou seja, objetivam a determinação da contribuição de possíveis erros humanos em todo processo de desenvolvimento do produto.

Três requisitos principais devem ser considerados no estudo da interface homem-máquina:

- Modelos de cognição
    - Estudo do comportamento humano utilizando modelos de cognição, ou seja, dos processos mentais que envolvem a percepção, memória, raciocínio, etc.
- Taxonomia de comportamento errôneo
    - Classificação dos possiveis erros humanos e seus efeitos (interpretação, comunicação, procedimento etc).
- Base de dados de observações de campo
    - Voltado a representar a correlação entre os modelos que representam o sistema homem-máquina e o ambiente real de trabalho.

III. Encorajamento da Cultura de Segurança na Organização

Em relação à cultura de segurança, Storey (1999) afirma que a segurança não é atingida simplesmente pela utilização de métodos de desenvolvimento apropriados. Este atributo deve ser planejado e construído através da sua consideração em todas as fases do ciclo de vida do produto. Por causa de sua importância, é essencial que uma cultura de segurança seja encorajada pela organização de maneira que todo o processo de desenvolvimento do produto seja realizado de forma robusta e eficaz.

e) Principais Técnicas de Projeto Utilizadas no *Design for Safety*

Para Storey (1999), os requisitos de segurança caminham lado a lado com os requisitos funcionais de forma a definir os niveis de satistação por todo ciclo de desenvolvimento do sistema. A habilidade de um sistema satisfazer tanto os requisitos funcionais quanto os requisitos de segurança é limitada pela presenca de defeitos, ou seja, condições anormais que podem levar a uma falha. Estes defeitos podem ser:

- Aleatórios.
    - Podem ser investigados estatisticamente.
    - Falha de um componente de sistema (por exemplo: a falha de uma válvula, bomba, eixo, conector etc).
- Sistemáticos.
    - Não podem ser investigados estatisticamente.
    - Exemplo: Comportamento anômalo de software (*bug*).
- Erros de especificação.
    - Normalmente introduzidos da etapa de desenvolvimento.
    - Exemplo: Falha de um interruptor causado por sobretensão.

Para a mitigação ou a eliminação de defeitos é necessária a utilização de algumas técnicas de projeto que são conduzidas por algumas abordagens de projeto.

Segundo Dedini (2007), as técnicas de projeto voltadas à segurança podem ser classificadas da seguinte forma:

- Técnicas Diretas.
    - Exemplo: Abordagens como o projeto para falha segura (*Fail Safe*) e o projeto para vida segura (*Safe-Life*).
- Técnicas Indiretas.
    - Exemplo: Utilização de sistemas de proteção ou barreiras físicas ou lógicas visando à contenção de falhas quando a segurança não pode ser totalmente obtida do projeto.
- Técnicas Indicativas.
    - Exemplo: Utilização de sistemas para advertência de riscos (alertas sonoros, alertas luminosos, indicações em monitores).
- Técnicas de Monitoramento e Diagnose.
    - Exemplo: Sistemas de monitoramento e diagnóstico para identificação e previsão de situações de risco de um equipamento.

A abordagem *Safe Life* é utilizada quando se deseja que um componente ou sistema não falhe num determinado período de tempo (por exemplo: bombas d'água, alguns componentes do trem de pouso de aviões etc). Neste caso, testes e análises intensivas (ensaios de fadiga, por exemplo) são utilizados para estimar a vida esperada. Ao final do período de vida estimado, o componente é descartado.

Musgrave, Larsen e Sgobba (2009) definem um projeto *Fail-Safe* como aquele que garante que falhas não afetarão um determinado sistema e que as mesmas não se converterão num estado onde ferimentos ou danos no sistema possam acontecer. Esta abordagem visa incorporar várias técnicas de mitigação de riscos e é aplicada principalmente a funções de sistema que não são essenciais. Três tipos diferentes de arranjos podem ser apresentados para um projeto *Fail-Safe*:

- *Fail-Passive*
    - O sistema é automaticamente desenergizado e tem a operação cessada até que ações corretivas sejam tomadas.
    - Exemplo: *Circuit-breakers*, fusíveis, sistema de *autopilot* dos aviões.
- *Fail-Active*
    - O sistema fica energizado e redundâncias o mantém num modo de operação seguro até que uma ação corretiva seja tomada.
    - Exemplo: Bateria do sistema de detecção de fumaça de um avião.
- *Fail-Operational*
    - A falha faz o equipamento reverter a um modo de operação seguro, entretanto as funcionalidades que poderiam estar apresentando uma condição insegura são perdidas.
    - Exemplo: Elevadores.

Musgrave, Larsen e Sgobba (2009) ainda afirmam que um projeto *Fail-Safe* não mantém ou garante segurança pelo aumento da confiabilidade do sistema. Um projeto nesta abordagem pode ser não confiável, mas ao mesmo tempo fornece segurança. Por este motivo, o termo *Fail-Safe* não deve ser usado como um sinônimo de redundância.

2.2 O *Design for Safety* Aplicado a Sistemas Complexos

Visando explicar a aplicação do *Design for Safety* em sistemas complexos, serão considerados os seguintes pontos: conceito de sistema, o projeto e desenvolvimento de sistemas complexos, engenharia de sistemas, segurança de sistemas e técnicas de análise de risco.

2.2.1 Conceito de Sistema

Na engenharia, o conceito de sistema é muito bem definido e bastante utilizado em diferentes abordagens, como por exemplo: projeto, processo, planejamento, produção etc.

De acordo com a NASA (1995), sistema é um conjunto de componentes inter-relacionados que interage um com o outro em uma estrutura organizada, na direção de um propósito comum. Esta definição pode ser exemplificada através de uma terminologia hierárquica de sistemas, como sendo: Sistema, Segmento, Elemento, Sub-Sistema, Montagem, Sub-montagem e Componente.

Hitchins (2003) considera o termo interação, definindo sistema como sendo um conjunto de elementos interativos e complementares com propriedades e comportamentos que emergem tanto de cada elemento quanto das interações. Por causa das interações, um sistema pode ser complexo e ser capaz de desempenhar funções que nenhum de seus elementos pode realizar sozinho.

O INCOSE (1998) por sua vez, define sistema como sendo um grupo de elementos integrados que tem por objetivo cumprir um determinado objetivo. Estes elementos podem ser *hardware*, *software*, pessoas, processos, informação, locais, equipamentos etc.

Valeriano (1998) afirma que sistema é um conjunto de elementos inter-relacionados (sub-sistemas e/ou processos) que visa a obtenção de determinados objetivos situados no meio exterior em que está inserido.

De uma forma geral, a definição de sistema está sempre relacionada com a integração, interação ou inter-relação de elementos.

2.2.2 O Projeto e Desenvolvimento de Sistemas Complexos

Segundo Ulrich e Eppinger (2008), uma das variantes de um processo genérico de desenvolvimento do produto é aquela voltada a sistemas complexos.

De acordo com Honour (2006), um sistema complexo é aquele formado por vários elementos interativos de natureza reflexiva, ou seja, onde a ação de cada elemento impacta os elementos a sua volta. Além disso, um sistema complexo apresenta determinados comportamentos que não são percebidos a partir de cada elemento individualmente.

Produtos de larga escala como os automóveis e os aviões são sistemas complexos compostos de inúmeras interações em seus níveis inferiores. Este tipo de variante apresenta as seguintes características:

- A fase de concepção considera a arquitetura de todo o sistema e múltiplas arquiteturas podem ser consideradas.
- Durante a fase de projeto em nível de sistema, os sistemas são decompostos em sub-sistemas e estes em componentes. Times são formados para o desenvolvimento de cada componente e outos times são formados com a responsabilidade de integrar componentes em sub-sistemas e estes no sistema completo.
- O projeto detalhado é considerado um processo altamente paralelo, nos quais muitos times trabalham ao mesmo tempo e quase sempre de forma independente.
- A fase de testes e refinamentos inclui não somente a integração de sistemas, mas também testes extensivos e validação em todos os níveis.

A figura 2.8 apresenta o processo genérico de projeto e desenvolvimento de sistemas complexos.

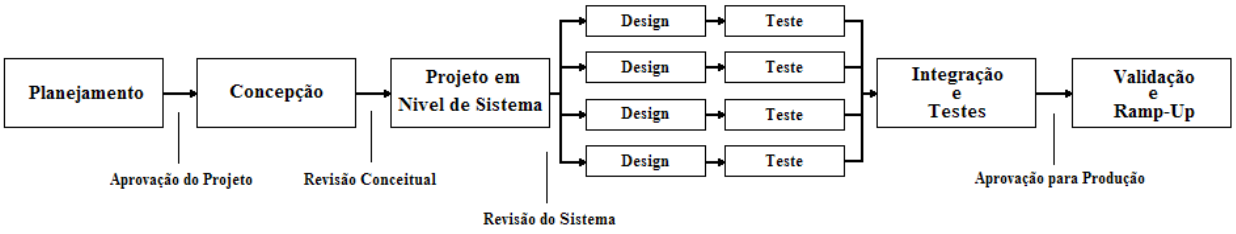


Figura 2.8: Processo genérico de projeto e desenvolvimento de sistemas complexos (Adaptado de Ulrich e Eppinger 2008)

Segundo Prasad (1996), a decomposição do produto é uma das abordagens utilizadas no projeto de sistemas complexos. Ela é largamente utilizada como uma base para a exploração da simultaneidade no projeto e desenvolvimento integrado do produto. Os principais tipos de decomposição do produto são:

- Decomposição holística do produto
  - Executada através de blocos hierárquicos ou classes fisicas.
  - Exemplo: Projeto de avião (sistemas, subsistemas, componentes etc)
- Decomposição funcional
  - Executada através de hierarquia funcional.
  - Exemplo: Projeto de um *software* (arquitetura, módulos, processadores)
- Decomposição por atividade
  - Executada através de atividades.
  - Exemplo: Especificações a partir de requisitos de clientes.

Segundo Park J.Y e Park Y.W (2004), uma metodologia essencial para o projeto de sistemas *safety-critical* de maior complexidade consiste numa abordagem interdisciplinar para redução e eliminação de potenciais riscos ao sistema. Esta abordagem é conhecida como engenharia de sistemas.

2.2.3 Engenharia de Sistemas

O termo Engenharia de Sistemas surgiu na decada de 40 no Bell Telephone Laboratories como conseqüência do crescente aumento da complexidade de projetos e sistemas. Ele teve por objetivo inicial a identificação e manipulação de propriedades do sistema como um todo que, em sistemas complexos, podem ser bem diferentes quando comparadas a soma das propriedades de cada elemento do sistema. As primeiras aplicações desta disciplina foram no desenvolvimento de sistemas de mísseis pelo DoD na década de 1950 (INCOSE, 2009).

Segundo a NASA (1995), a engenharia de sistemas é uma abordagem robusta para o projeto, criação e operação de sistemas. De uma forma simples, esta abordagem consiste na identificação e quantificação dos objetivos do sistema, criação de conceitos alternativos de projeto de sistemas, seleção e implantação do melhor projeto, integração de sistemas e avaliação da pós-implementação de forma a garantir o cumprimento de seus objetivos.

Por sua vez o INCOSE (2009), define a engenharia de sistemas como sendo uma abordagem interdisciplinar, cujo objetivo está nas necessidades do cliente no ciclo de desenvolvimento de projeto, e em todo ciclo de vida do sistema. A engenharia de sistemas integra todas as disciplinas e grupos de especialidade em um processo de desenvolvimento estruturado que flui desde o projeto até o fim da vida útil do sistema.

Prasad (1996) afirma que a engenharia de sistemas consiste basicamente na concatenação entre diversas boas práticas e conceitos de engenharia, como por exemplo: times de trabalho cooperativos, reengenharia de processo e o gerenciamento do ciclo de vida do produto. Além disso, enfatiza que o processo de realização do produto tenha uma visão centrada no sistema em oposição à visão centrada nos componentes.

Blanchard (2003) reitera que o processo de engenharia de sistemas é aplicável em todas as fases do ciclo de vida, onde na fase inicial o objetivo é o de entender as reais necessidades do cliente e desenvolver os requisitos para o sistema.

A figura 2.9 representa de uma forma geral este conceito:

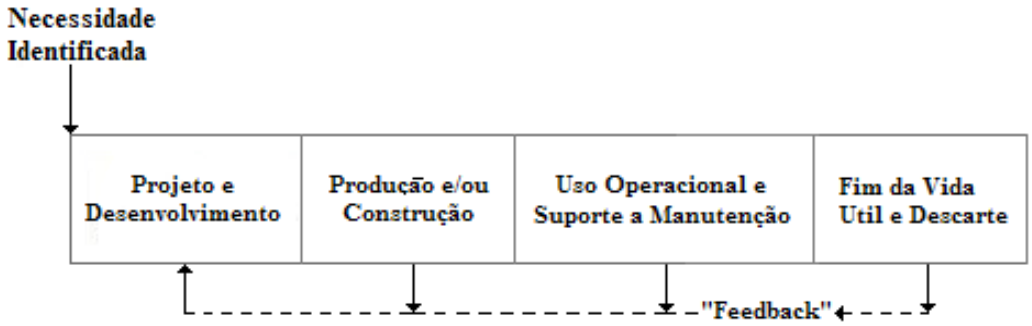


Figura 2.9: Ciclo de vida de um sistema (Fonte: Adaptado de Blanchard 2003).

Como mostrado na figura 2.9, o ciclo de vida de um sistema inicia-se com a identificação das necessidades que por sua vez são estendidas por várias fases até o final da sua vida útil. Tão logo as atividades em cada fase se interagem com as atividades das outras fases, é essencial considerar o ciclo de vida total ao endereçar problemas que envolvem a avaliação dos riscos associados ao processo de tomada de decisão.

Segundo Blanchard (2003) as atividades chave da engenharia de sistemas são:

- Análise de Requisitos.
- Análise e Alocação Funcional.
- Sintese do Projeto (Arquitetura dos Sistemas).
- Controle e Análise do Sistema.

Estas atividades podem ser mais bem definidas segundo o DoD (2001), que considera as seguintes etapas no processo da engenharia de sistemas:

- Entrada do Processo
    - Necessidades e objetivos dos clientes.
    - Requisitos técnicos e ambientais, normas e especificações.
    - Experiência de projetos anteriores.
    - Base tecnológica disponível.

- Análise de requisitos
    - Análise dos requisitos de entrada do processo.
    - Identificação dos requisitos funcionais.
    - Definição e refinamento dos requisitos restritivos de projeto e desempenho.

- Análise e Alocação Funcional
    - Decomposição das funções para níveis inferiores.
    - Alocação dos requisitos funcionais para todos os níveis subseqüentes.
    - Definição e refinamento das interfaces funcionais (internas e externas).
    - Definição, refinamento e integração da arquitetura funcional.

- Síntese do Projeto
    - Transformação das arquiteturas funcionais em físicas.
    - Definição dos sistemas alternativos, itens de configuração e elementos de sistema.
    - Seleção das melhores soluções de processo.
    - Definição e refinamento das interfaces físicas (internas e externas).

- Controle e Análise do Sistema
    - Análises de custo beneficio e efetividade em custos.
    - Gerenciamento de riscos.
    - Gerenciamento de configuração.
    - Gerenciamento de dados.
    - Medição de desempenho do processo.

- *Loop* de Requisitos e de Projeto
    - Processos iterativos que visam revisitar condições inicialmente assumidas (requisitos *versus* arquitetura de sistemas) de forma a adequar a novos cenários.

- Saída do Processo
    - Arquitetura e configuração dos sistemas.
    - Especificações e *baselines*.

Ainda segundo o DoD (2001) , o processo que rege a engenharia de sistemas, além de abrangente e iterativo, possui uma abordagem *top-down*, uma vez que a análise do sistema como um todo é essencial para entender toda a integração existente. A figura 2.10 exemplifica este processo:

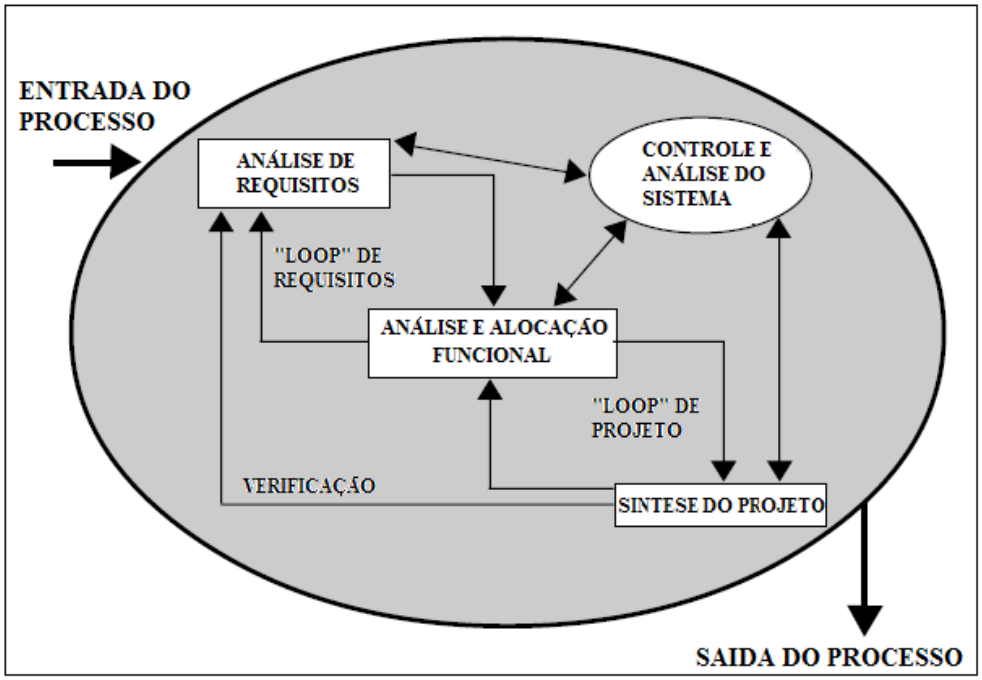


Figura 2.10: Processo de Engenharia de Sistemas (Fonte: Adaptado de DoD, 2001)

A análise de requisitos (também conhecida como engenharia de requisitos) é considerada a primeira etapa dentro do processo de engenharia de sistemas e tem por objetivo desenvolver os requisitos funcionais e de desempenho do sistema através da tradução dos requisitos e das necessidades dos clientes em um grupo de requisitos que irão definir as caracteristicas do sistema em desenvolvimento.

Segundo Ulrich e Eppinger (2008) o principal objetivo da engenharia de sistemas é o da transferência (*flow-down*) de requisitos ou especificações em sistemas complexos.

Segundo Hull, Jackson e Dick (2005), os requisitos são a base de todo projeto, pois definem o que os *stakeholders* (clientes, usuários, desenvolvedores, fornecedores, autoridades) necessitam de um novo sistema e também o que o sistema deve fazer para satisfazer tal necessidade.

O DoD (2001) categoriza os requisitos em 6 tipos principais relacionados ao gerenciamento técnico do processo de engenharia de sistemas:

- Requisitos de Clientes.
- Requisitos Funcionais.
- Requisitos de Desempenho.
- Requisitos de Projeto.
- Requisitos Derivados.
- Requisitos Alocados.

Ainda segundo Hull, Jackson e Dick (2005), a engenharia de requisitos possui papel vital em cada estágio do desenvolvimento de um sistema uma vez que fornece condições para a sua qualificação.

Segundo Buede (1999), qualificação é o processo de validação e verificação do projeto de um sistema visando sua aceitação por parte do *stakeholder*. Ainda segundo ele, a validação consiste em se determinar se o processo de engenharia de sistemas produziu aquilo que era esperado pelo *stakeholder*, ou seja, garantir que os requisitos para o produto estão suficientemente corretos e completos. Já a verificação consiste em avaliar a implementação dos requisitos do produto para determinar se eles foram atingidos.

Para apresentar de uma forma mais clara a relação entre o conceito da engenharia de requisitos e o processo de qualificação do produto, utiliza-se o modelo "V&V" (*V model*), que é uma representação gráfica do ciclo de vida do desenvolvimento de sistemas (figura 2.11).

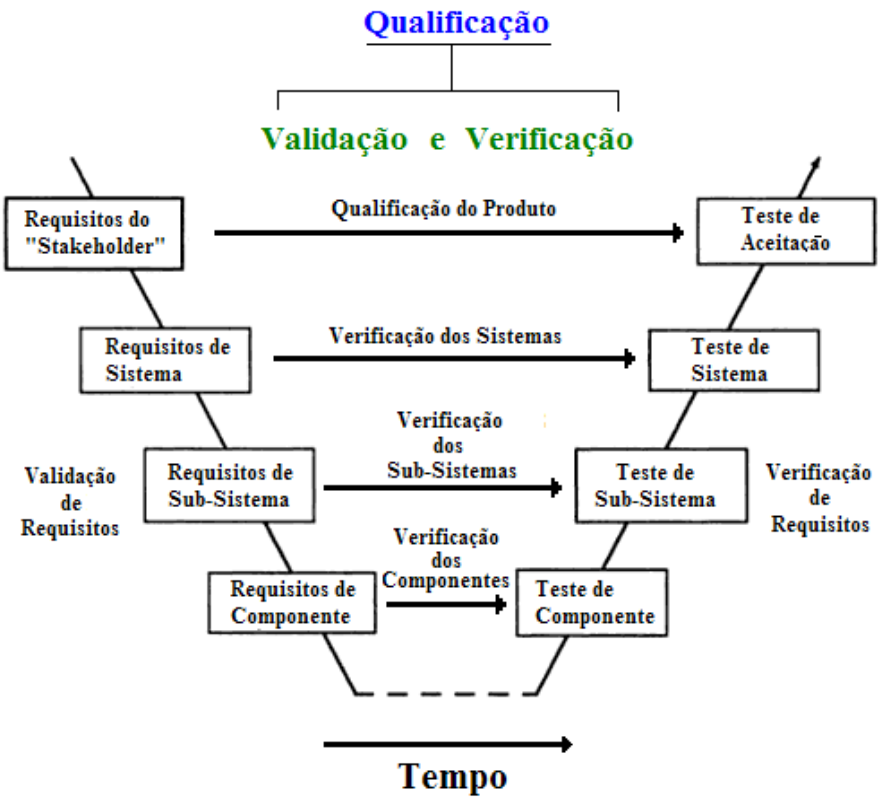


Figura 2.11: Modelo "V&V" da Engenharia de Sistemas

(Fonte: Adaptado de Hull, Jackson e Dick, 2005)

Segundo Blanchard (2003), o modelo "V&V" foi introduzido na década de 90 e reflete uma abordagem *top-down* e *bottom-up* do processo de engenharia de sistemas. O lado esquerdo do modelo representa a alocação e validação dos requisitos nos níveis de desenvolvimento do produto e o lado direito representa a integração e verificação que irão qualificar o projeto.

2.2.4 Segurança de Sistemas

A MIL-STD-882D (2000) afirma que historicamente a falta de atenção adequada aos riscos no projeto e na operação, particularmente em sistemas complexos, ocasionou a introdução do conceito de seguranca de sistemas (*System Safety*).

Roland e Moriarty (1990) definem a segurança de sistemas como sendo a aplicação de habilidades técnicas e gerenciais para a identificação e controle sistemático de riscos por todo o ciclo de vida de um projeto, programa ou atividade.

McCollum e Hugues (2005) definem segurança de sistemas como uma disciplina que envolve a aplicação de principios gerenciais e de engenharia com o intuito de identificar e minimizar riscos dentro de restrições de custo, planejamento e requisitos de projeto em todas as fases de ciclo de vida de um sistema.

Segundo Ericson (2005), a segurança de sistemas é o processo de gerenciamento de riscos de acidentes em sistemas, pessoas, meio ambiente e saúde durante a fase de desenvolvimento do produto. Ainda segundo ele, é um processo empregado desde a conceituação inicial de projeto até o final de sua vida útil.

Ericson (2005) ainda afirma que o principal objetivo do processo de segurança de sistemas é o de eliminar os riscos que podem resultar em morte, ferimentos, perda de sistema e dano ao meio ambiente. Quando a eliminação dos riscos não é possível, o próximo objetivo é o de reduzir o risco de um acidente através de medidas de controle de projeto, ou seja, através da redução da probabilidade e/ou severidade do acidente.

De uma forma geral, o principal foco deste processo é o de gerenciar o risco de acidentes através da identificação dos riscos e da utilização de técnicas voltadas a sua mitigação. Um projeto seguro é um pré-requisito para operações seguras e, no caso da aviação comercial, é um fator importantíssimo para a melhoria contínua dos níveis de segurança das aeronaves.

Segundo Ericson (2005), um processo simplificado de segurança de sistemas pode ser mostrado conforme a figura 2.12:

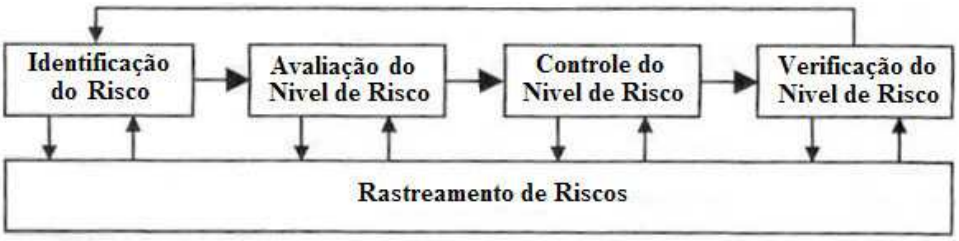


Figura 2.12: Processo simplificado de seguranca de sistemas (Fonte: Ericson 2005)

A figura acima representa um processo simplificado (de ciclo fechado) de rastreamento de risco onde a segurança é atingida pela identificação, avaliação, controle e verificação dos riscos intrínsecos.

Com o intuito de identificar os efeitos e seus fatores causadores visando a determinação dos niveis de risco de um sistema, análises de risco (*Hazard Analysis*) são efetuadas de forma a permitir que medidas de segurança possam ser estabelecidas para eliminação ou mitigação de riscos.

2.2.5 Principais Técnicas de Análise de Risco

Segundo Ericson (2005) existem sete tipos de análise de risco em segurança de sistemas e sua utilização é crítica para a identificação e mitigação de riscos visando a minimização dos riscos residuais:

- Análise de risco no projeto conceitual (CD-HAT).
- Análise de risco no projeto preliminar (PD-HAT).
- Análise de risco no projeto detalhado (DD-HAT).
- Análise de risco voltado aos sistemas (SD-HAT).
- Análise de risco operacional (OD-HAT).
- Análise de risco voltada à saúde humana (HD-HAT).
- Análise de risco voltado ao projeto de requisitos (RD-HAT).

É importante ressaltar que todos os tipos de análise de risco citados acima representam de certa forma as fases do ciclo de vida no desenvolvimento do produto, ou seja, desde a concepção até o descarte. A figura 2.13 apresenta os tipos de análise representados como filtros de risco por fase do ciclo de vida do produto.

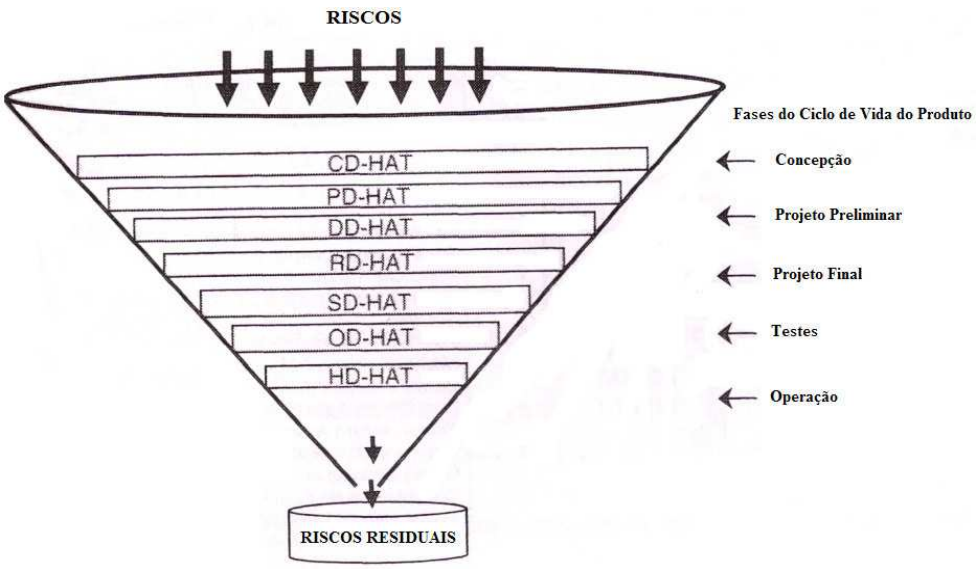


Figura 2.13: Filtros de risco (Fonte: Adaptado de Ericson 2005)

Ericson (2005) classifica as técnicas de análise de risco nos seguintes tipos:

- Indutivas ou Dedutivas.
- Qualitativas ou Quantitativas.

Os termos indutivo e dedutivo se referem à forma de lógica. A forma indutiva é um processo lógico na qual a conclusão é proposta a partir de experiência ou observação. Já na forma dedutiva, o processo lógico é baseado em premissas.

Com relação às análises qualitativas em quantitativas, é possível descrevê-las a partir de suas vantagens e desvantagens.

Uma análise qualitativa é considerada mais subjetiva, entretanto ela permite maior generalização, sendo neste caso menos restritiva. Tipicamente este tipo de análise utiliza categorias (limitadas por faixas) para separar diferentes parâmetros.

Já uma análise quantitativa envolve a utilização de parâmetros numéricos e é mais objetiva e precisa que uma análise qualitativa. Por outro lado, este tipo de análise apresenta custos elevados e níveis de complexidade muito superiores as análises qualitativas. Além disso, os tempos requeridos para este tipo de análise são também muito elevados.

Ericson (2005) afirma que durante uma avaliação de riscos a precisão numérica não é sempre necessária uma vez que os riscos de acidentes não são facilmente estimados usando probabilidades e estatísticas quando os fatores causais não estão bem entendidos. Dentro deste contexto, é recomendada inicialmente uma avaliação dos riscos sob uma ótica qualitativa para, em caso de riscos de conseqüências mais severas, aplicar uma análise quantitativa para obter maior precisão.

Dentre as principais técnicas de análise de riscos aplicadas na segurança de sistemas, destacam-se:

- *Fault-Tree Analysis* (FTA) – Análise de Arvore de Falhas.
- *Failure Mode and EffectsAnalysis* (FMEA) – Análise de Modo e Efeito de Falha.
- *Functional Hazard Assessment* (FHA) – Avaliação de Riscos Funcionais.
- *Operating and Support Hazard Analysis* (O&SHA) – Análise de Riscos Operacionais.
- *Hazard and Operability Analysis* (HAZOP) – Análise de Riscos e Operabilidade.
- *Sneak Circuit Analysis* – Análise de Circuitos Ocultos.
- *Common Cause Analysis* (CCA) – Análise de Causa Comum.
- *Software Safety Assessment.* – Avaliação de Seguranca de *Software*.

A tabela 2.1 apresenta cada análise de risco classificada em função de seus tipos e de suas características principais:

Tabela 2.1: Técnicas de análise de risco e suas características (Fonte: Adaptado de Ericson 2005).

Técnica de Análise de Risco	Tipo	Qualitativa ou Quantitativa	Indutiva ou Dedutiva
Functional Hazard Assessment (FHA)	PD-HAT e DD-HAT	Qualitativa	Indutiva
Fault-Tree Analysis (FTA)	PD-HAT e DD-HAT	Qualitativa e Quantitativa	Dedutiva
Failure Mode and Effects Analysis (FMEA)	DD-HAT	Qualitativa e Quantitativa	Indutiva
Operating and Support Hazard Analysis (O&SHA)	OD-HAT	Qualitativa	Indutiva e Dedutiva
Hazard and Operability Analysis (HAZOP)	SD-HAT e DD-HAT	Qualitativa	Indutiva
Sneak Circuit Analysis.	SD-HAT e DD-HAT	Qualitativa	Dedutiva
Common Cause Analysis (CCA)	PD-HAT e DD-HAT	Qualitativa	Dedutiva
Software Safety Assessment	SD-HAT e DD-HAT	Qualitativa	N/A

No projeto e desenvolvimento de sistemas em aeronaves comerciais, a utilização de algumas destas técnicas de análise de risco tem fundamental importância e compõe um processo de engenharia conhecido como *Safety Assessment*.

2.3 A Segurança de Sistemas Aplicada no Desenvolvimento de Aeronaves Comerciais

Neste tópico serão apresentados os conceitos de projeto seguro na aviação, o processo de *Safety Assessement* e suas respectivas perspectivas (Funcional, Instalativa e Operacional).

2.3.1 O Conceito de Projeto Seguro

Segundo a Serra (2006), o conceito de projeto seguro na aviação comercial é aquele que considera os efeitos das falhas e as combinações destes efeitos em seu processo, através da utilização dos seguintes princípios de projeto:

- Integridade projetada através da utilização de componentes com vida limite.
- Sistemas críticos redundantes ou de *back-up* (não necessariamente idênticos).
- Independência nas arquiteturas redundantes.
- Confiabilidade de componentes comprovada.
- Meios de detecção de falha através de indicações (visuais, sonoras etc).
- Procedimentos de tripulação para condições inseguras.
- Meios de contenção de falhas e tolerância ao erro para eventos não previstos.

Serra (2006) define um projeto seguro como o resultado de uma metodologia organizada, sistemática e abrangente, que considera alguns pontos de fundamental importância ao processo de *Safety Assessment*, são eles:

- Tecnologia Empregada
  - Relação Seguranca x Custo (viabilidade econômica e efetividade em custos)
  - Na indústria aeronáutica, a segurança deve ser tratada como o valor central em um projeto, entretanto não deve ser absoluta. Os valores de ordem econômica também são importantes, pois são eles que tornam o avião um meio de transporte economicamente viável.
- Metodologia Utilizada
  - Elaboração do plano de certificação para demonstração de concordância dos requisitos aplicáveis.
- Lições Aprendidas
  - Aprendizado com incidentes e acidentes do passado.
- Requisitos de Certificação
  - Conhecimento dos requisitos de certificação aplicáveis ao projeto.
- Fatores Humanos
  - Contribuidor primário em mais de 70% dos acidentes aéreos em aviões comerciais (Boeing, 2009).
- Administração dos Riscos
  - Reconhecer a não existência do risco zero.

2.3.2 O Processo de *Safety Assessment*

Serra (2006) define o termo *Safety Assessment*, comumente traduzido como Análise de Confiabilidade e Seguranca de Sistemas, como um processo de engenharia que fornece as ferramentas necessárias para que se possa determinar, de maneira organizada, sistemática e abrangente se os níveis de segurança de um sistema estão sendo atingidos.

O termo inglês "*assessment*" representa uma avaliação baseada em julgamento de engenharia onde múltiplos aspectos de projeto (quantitativos e principalmente qualitativos) são considerados.

No que se refere à abordagem da engenharia de sistemas, o processo de *Safety Assessment* é um processo baseado em requisitos que atua de forma paralela, simultânea e integrado com o desenvolvimento de engenharia do produto, desde as fases mais preliminares passando pelo projeto detalhado dos sistemas, instalação (produção e montagem), ensaios, até a entrada em serviço.

A etapa da emissão do certificado de tipo é apenas um marco que finaliza o processo de certificação da aeronave, processo este que se inicia antes da fase de concepção do produto através da definição da base de certificação (verificação das últimas revisões e *ammendments* dos requisitos), e caminha até o encerramento da fase de integração e testes.

O *Safety Assessment* é apenas uma das partes que compõe todo o processo de certificação. Após a entrada em serviço e ao longo de praticamente toda a vida operacional de uma aeronave, inicia-se o processo de aeronavegabilidade continuada, que visa garantir cumprimento dos niveis de segurança que serviram de base para a certificação.

A tabela 2.2 representa os processos de *Safety Assessment* e de Aeronavegabilidade Continuada, inseridos no ciclo de desenvolvimento de engenharia e no ciclo de vida do produto.

Tabela 2.2: O *Safety Assessment* e a Aeronavegabilidade Continuada.

CICLO DE DESENVOLVIMENTO DE ENGENHARIA NO PROJETO DE AERONAVES COMERCIAIS								
Concepcao	Arquitetura	Ante-Projeto	Projeto Detalhado	Instalação	Ensaios	Emissao do Certificado de Tipo	Entrada em Serviço	Vida Operacional
PROCESSO DE "SAFETY ASSESSMENT"							***AERONAVEGABILIDADE CONTINUADA***	

O processo de *Safety Assessment* consiste em uma avaliação dos riscos associados às falhas ou combinação de falhas, erro humano ou eventos que possam impedir a aeronavegabilidade continuada de uma aeronave.

O requisito FAR 25.1309, criado na década de 70 pela autoridade de aviação civil dos EUA, o FAA (*Federal Aviation Administration*), apresenta as regras de certificação do projeto de sistemas voltadas a garantir níveis de segurança aceitáveis, através da administração dos riscos inerentes para cada condição de falha identificada.

Este requisito serviu de base para a criação de outros requisitos similares, como por exemplo o requisito do autoridade européia, a EASA (*European Aviation Safety Agency*), conhecido como CS 25.1309 e o requisito da autoridade brasileira, a ANAC (Agência Nacional de Aviação Civil), conhecido atualmente como RBAC 25.1309.

De acordo com a ANAC (2009), a certificação (ou homologação) de aeronaves é, especialmente na aviação civil, uma atividade necessária à segurança do transporte aéreo. Consiste em avaliar e atestar se um determinado produto (aeronave, componente ou sistema) possui as características mínimas que assegurem seu uso seguro para o tipo de operação pretendida.

Durante muitos anos, principalmente após o surgimento do requisito FAR 25.1309, nenhum padrão foi estabelecido pelos fabricantes ou autoridades no que se refere ao modelo ideal de um processo de *Safety Assessment*. Sendo assim, práticas recomendadas foram desenvolvidas na tentativa de se padronizar o processo.

A prática recomendada mais completa e detalhada sobre o processo de Safety Assessment é a ARP-4761 (*Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems*), elaborada pela SAE (*Society of Automotive Engineers*) em 1996, em conjunto com os fabricantes de aviões. Ela é recomendada pelas autoridades de forma a servir de base para demonstrar o cumprimento de requisito.

Para auxiliar a certificação de sistemas complexos e altamente integrados, a SAE emitiu também em 1996 a ARP-4754 (*Certification Considerations for Highly-Integrated or Complex Aircraft Systems*).

Segundo esta prática recomendada, um sistema complexo é aquele cuja segurança não pode ser demonstrada unicamente por testes e cuja lógica é de difícil compreensão sem a ajuda de ferramentas analíticas. Já um sistema altamente integrado é aquele que contribui para múltiplas funções no avião. Estes tipos de sistema apresentam grande susceptividade a erros de desenvolvimento uma vez que são constituídos basicamente de *hardware* e *software*.

Sendo assim, a ARP-4754 relaciona-se com a ARP-4761 no que se refere a garantir que os níveis de segurança estão sendo adequadamente substanciados no processo de desenvolvimento dos sistemas através do Nível de Garantia de Desenvolvimento ou DAL (vide tabela 2.3) que é definido ao longo do processo de *Safety Assessment*.

Para cada DAL se tem um nível de exigência no desenvolvimento e na certificação de um *software* ou *hardware* (tipo de arquitetura, dissimilaridade, independência, parcionamento de projeto para limitar os efeitos funcionais cruzados de eventuais erros de projeto em partes separadas do sistema etc). Um software categoria A, por exemplo, apresenta um nível de exigência elevado, uma vez que a sua inoperância pode resultar numa condição de falha catastrófica (ARP-4754, 1996).

Tabela 2.3: *Development Assurance Levels* (Fonte: Adaptado da ARP-4754, 1996)

Severidade da Condição de Falha	DAL
Catastrophic	A
Hazardous / Severe Major	B
Major	C
Minor	D
No Safety Effect	E

De uma forma geral, segundo McIntyre (2002), a ARP-4754 define o "o que fazer" enquanto que a ARP-4761 define o "como fazer", num processo de *Safety Assessment*.

A figura 2.14 representa a relação entre os processos da ARP-4761 e ARP-4754.

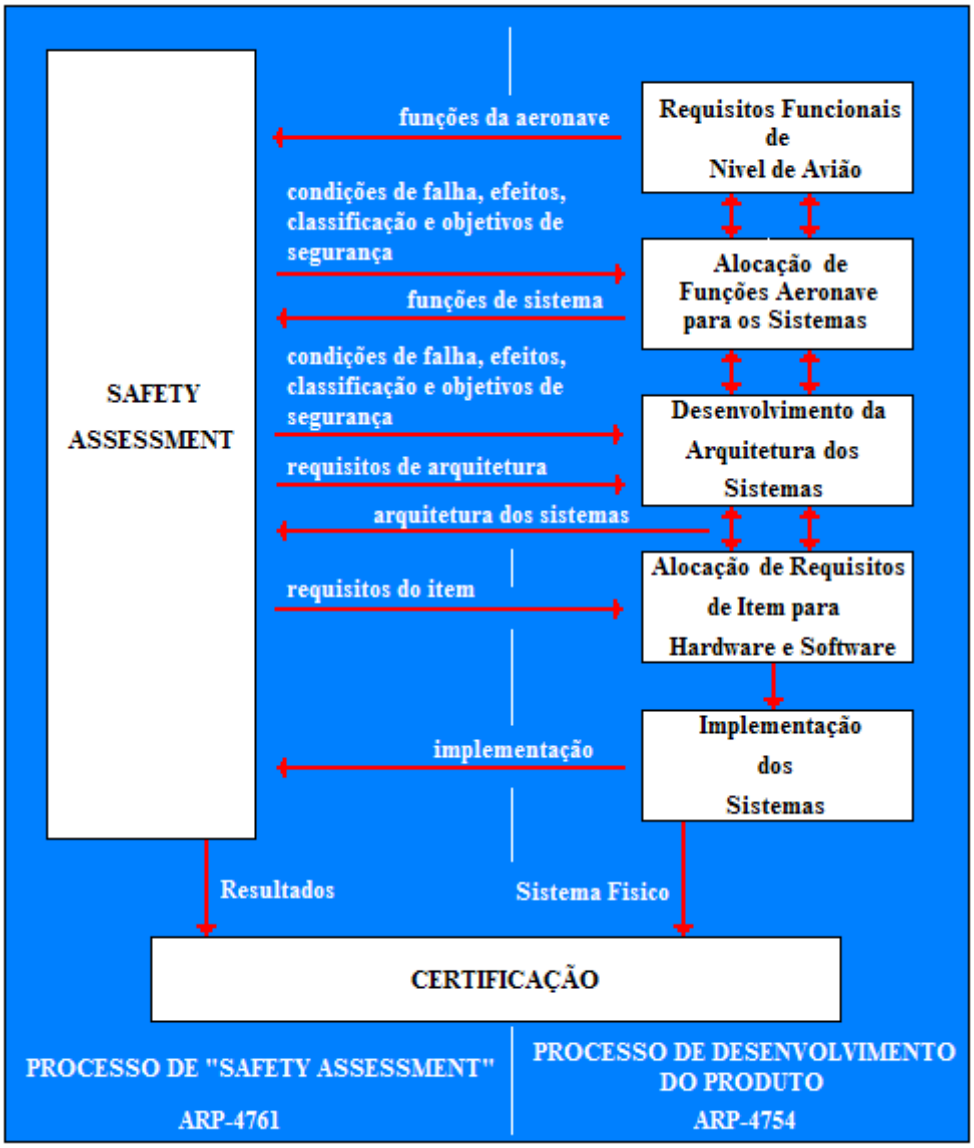


Figura 2.14: Relação entre os processos da ARP-4761 e ARP-4754 (Fonte: ARP-4754, 1996)

Segundo Hasson e Crotty (1997), a execução do processo de *Safety Assessment* pode ser vista sob três perspectivas:

- Funcional.
- Instalativa ou Física.
- Operacional.

O *Safety Assessment* funcional é considerado o principal e mais extenso de todo o processo e visa demonstrar o quão bem os sistemas ou equipamentos desempenham determinadas funções, onde o foco está nas arquiteturas funcionais. Pelo seu papel central no processo, possui interface com as outras duas perspectivas, recebendo dados de entrada de cada uma delas.

O *Safety Assessment* físico ou instalativo visa demonstrar como os sistemas e equipamentos são instalados no avião e foca nas questões de *lay-out* físico e problemas de configuração. Esta avaliação é extremamente importante, pois é responsável por validar as redundâncias e as premissas de independência assumidas durante o a avaliação funcional.

Já o *Safety Assessment* operacional deve ser realizado de forma a analisar o impacto de determinados cenários operacionais não usuais durante a fase de desenvolvimento da aeronave.

2.3.3 A Perspectiva Funcional no *Safety Assessment*

Segundo a ARP-4761 (1996), a perspectiva funcional é constituida de três fases principais:

- Análise funcional.
- PSSA (*Preliminary System Safety Assessment)*.
- SSA (*System Safety Assessment*).

A análise funcional corresponde à etapa onde são determinadas as condições de falha de cada função (seja ela de alto nível ou de sistema), os efeitos destas condições de falha, as severidades correspondentes e onde serão definidos, validados e desdobrados os requisitos de segurança (alto nível ou em nível de sistema). A principal técnica de análise de risco utilizada nesta fase é o FHA.

O PSSA corresponde à etapa onde são definidos e validados os requisitos de independência funcional (inter-sistemas e intra-sistemas), as metas de probabilidade dos subsistemas, os níveis de desenvolvimento de *software* e *hardware* (DALs), e onde será validado se a arquitetura preliminar proposta atende aos requisitos de segurança definidos na análise funcional. Nesta fase, a principal técnica de análise de risco utilizada é a FTA (qualitativa).

Já o SSA corresponde à etapa onde serão verificadas as metas de probabilidade definidas no PSSA, a integração e testes em diversos níveis (sistema, multi-sistemas e avião), e onde serão geradas considerações para a elaboração de manuais de manutenção, operação e de despacho da aeronave. A FMEA e a FTA (quantitativa) são as principais técnicas utilizadas nesta fase.

Com relação aos níveis de desenvolvimento (avião, sistema e item), enquanto a Análise Funcional atua tanto em nível avião como no nível de sistema, o PSSA atua somente no nível de sistema e o SSA atua em nível de sistema e item.

2.3.4 A Perspectiva Instalativa no *Safety Assessment*

A perspectiva física ou instalativa dentro do processo de *Safety Assessment* visa validar as redundâncias e as premissas de independência assumidas durante a avaliação funcional.

Segundo Hasson e Crotty (1997), os requisitos físicos ou instalativos de segurança são derivados de ameaças físicas a aeronave. Estas ameaças físicas podem ter uma origem externa ou interna aos sistemas.

Segundo Ericson (2005) uma Análise de Causa Comum (*Common Cause Analysis*) se constitui de uma técnica para a identificação de causas comuns de eventos múltiplos de falha. Uma falha de causa comum é uma falha pontual que destrói a independência entre redundâncias (exemplo: falha de dois computadores redundantes e independentes para gerenciamento de funções de comando de vôo devido à falha de um *circuit breaker* comum ao sistema, o qual fornece energia elétrica).

Segundo a ARP-4761 (1996) uma Análise de Causa Comum (CCA) pode ser subdividida em três diferentes tipos de análise:

- Análise de Segurança Zonal (*Zonal Safety Analysis* – ZSA)
  - Relacionada ao estudo de influências internas ao sistema.
- Análise de Riscos Particulares (*Particular Risk Analysis* – PRA)
  - Relacionada ao estudo de influências externas ao sistema.
- Análise de Modo Comum (*Common Mode Analysis* – CMA)
  - Relacionada à qualificação e instalação de equipamentos.

Segundo a ARP-4761 (1996), a análise de segurança zonal é uma abordagem qualitativa (efetuada em cada zona da aeronave) que permite a consideração de aspectos de instalação de sistemas ou componentes e a influência mútua com respeito à proximidade entre eles.

Hasson e Crotty (1997), que denominam as influências internas como sendo ameaças instalativas (*installation threats*), definem uma análise zonal como sendo aquela que utiliza a experiência de campo para validar e gerar um conjunto de diretrizes de *design*. Estas diretrizes definem, por exemplo, requisitos de separação e orientação entre várias formas de configuração (por exemplo: fiações elétricas, linhas de combustível, cabos de comando, linhas hidráulicas, linhas de oxigênio, dutos pneumáticos, tubos de torque, etc.).

Similarmente à análise de segurança zonal, a análise de riscos particulares representa uma série de análises qualitativas que são utilizadas para a avaliação da instalação de sistemas ou componentes dentro de uma determinada zona considerando influências externas ao sistema.

Segundo a ARP-4761 (1996) um risco particular é definido como sendo um evento ou influências os quais estão fora do sistema, mas que podem violar os requisitos de independência.

Diferentemente de uma análise zonal (que é restrita a uma zona especifica), em uma PRA os riscos particulares podem influenciar várias zonas ao mesmo tempo. Alguns dos principais riscos particulares analisados no desenvolvimento de uma aeronave comercial são:

- Fogo, neve, gelo, granizo.
- Impacto de pássaro (*Bird Strike*).
- Despalhetamento de rotores do motor (*Rotor Burst* ou *Rotor Non-Containment*)
- Impacto de raios (*Lightning Strike*).
- Análise de estouro de pneus (*Tire Burst*).
- Análise de vazamento e escoamento de fluidos.
- Explosão de cilindros pressurizados (oxigênio, extinção de fogo).
- *High Intense Radiated Fields* (HIRF).

Segundo a ARP-4761 (1996), a Análise de Modo Comum (*Common Mode Analysis*) é uma ferramenta analítica que visa determinar a qualidade de um projeto. Esta análise contribui para a verificação dos princípios de independência que foram aplicados no projeto dos sistemas. Um exemplo prático seria considerar itens com o mesmo *hardware* ou *software* que podem ser susceptíveis a falhas genéricas as quais podem causar avarias em outros itens.

A CMA é aplicável em ambas as instâncias de desenvolvimento de uma aeronave (functional e instalativa). Durante o desenvolvimento functional, a análise de modo comum é focada, por exemplo, na segregação de canais de sistemas que desempenham funções redundantes ou complementares, com o intuito de impossibilitar uma falha que possa afetar canais redundantes ao mesmo tempo.

A figura 2.15 apresenta o ciclo típico de *Design for Safety* de uma aeronave comercial e um sumário dos relacionamentos entre os níveis de desenvolvimento, as etapas de projeto, a perspectiva funcional e instalativa do processo de *Safety Assessment* e as técnicas de análise de risco aplicadas.

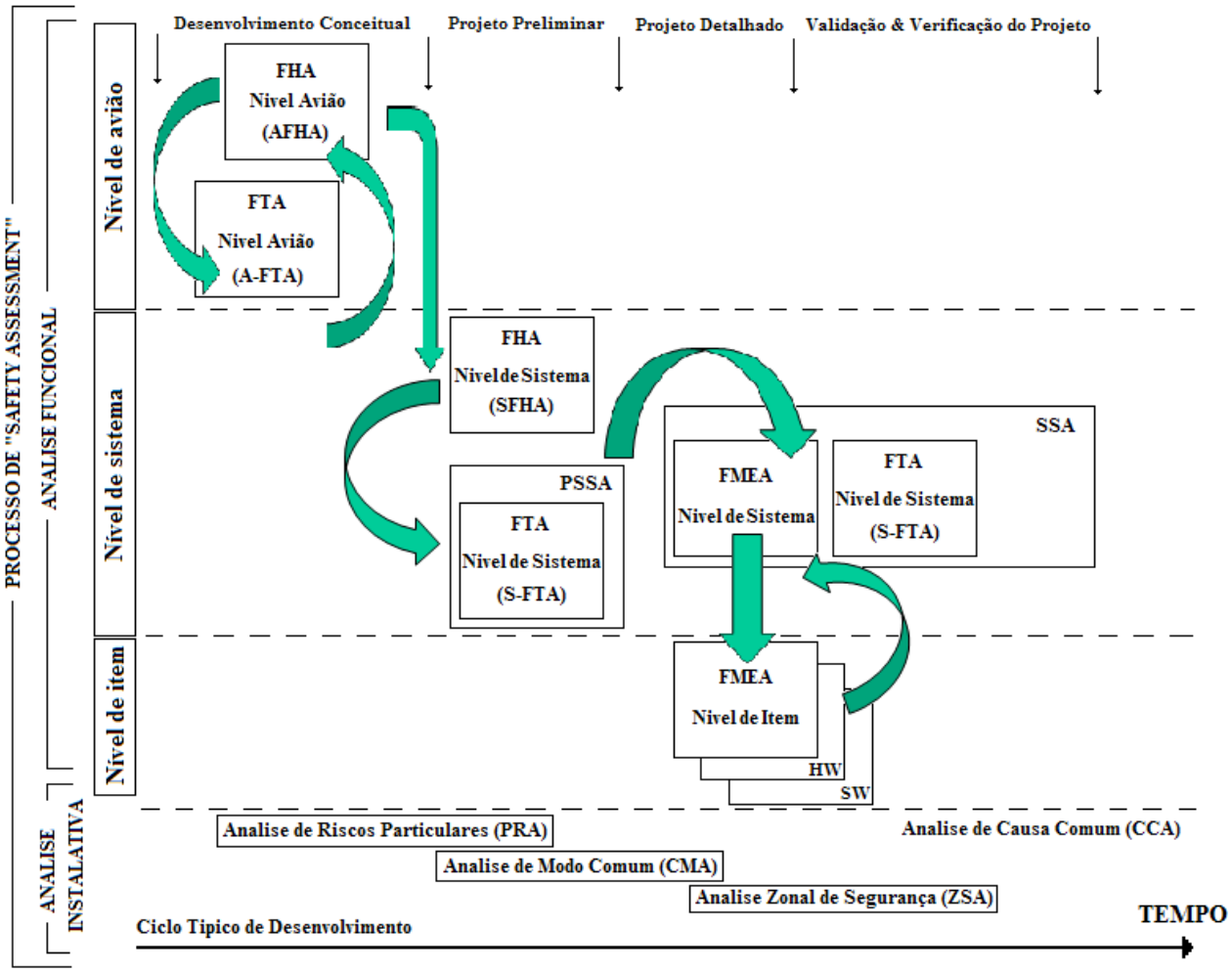


Figura 2.15: Ciclo típico de *Design for Safety* de uma aeronave comercial.
(Adaptado da ARP-4761, 1996).

2.3.5 A Perspectiva Operacional no *Safety Assessment*

Segundo Hasson e Crotty (1997), a experiência tem mostrado que cenários operacionais incomuns podem levar a acidentes. Para resolver esta preocupação uma analise especifica é realizada com o intuito de avaliar a segurança operacional dentro do projeto de uma aeronave.

Neste sentido, o *Safety Assessment* operacional deve ser realizado de forma a analisar o impacto de determinados cenários operacionais não usuais durante a fase de desenvolvimento da aeronave, como por exemplo:

- *Extended Operations* (ETOPS).
    - Segundo o FAA (2009) representa um tipo de operação onde a aeronave está a mais de 60 (em configuração de 2 motores) ou 180 (em configuração de 3 ou 4 motores) minutos de vôo de um aeroporto adequado em uma condição aprovada de operação monomotor.
    - Este tipo de operação normalmente ocorre em vôos trans-oceânicos ou em vôos sob grandes regiões desérticas (Exemplo: vôo São Paulo – Paris).
    - Exige níveis de confiabilidade e segurança elevados dos motores e de outros componentes.

- Pouso em Condição *Steep Approach*.
    - Representa a operação de pouso de uma aeronave num ângulo de aproximação maior que o usual. Requisito normalmente utilizado para a certificação da operação de aeronaves em aeroportos restringidos por obstáculos ou devido a regulamentações de nível de ruído (Exemplo: Aeroporto London City - LCY).

- Operação sob Presença de Cinzas Vulcânicas (*Volcanic Ashes*).
    - Representa a operação de aeronaves em regiões onde há a presença de cinzas vulcânicas no ar. Este tipo de condição pode acarretar falhas nos motores e desgaste nos parabrisas do *cockpit* e nas superfícies e comandos de vôo.

- Operação sob Presença de Cristais de Gelo (*Ice Crystals*).
    - Representa a operação de uma aeronave na presença de cristais de gelo na atmosfera em altitudes elevadas, causada por condições climáticas adversas (por exemplo, tempestades). Este tipo de condição insegura, que não é detectada pelos radares meteorológicos, pode acarretar a falha de motores e de outros sistemas na aeronave.

Como visto, o processo que envolve a aplicação das três perspectivas no desenvolvimento de aeronaves comerciais é de grande complexidade e necessita de um planejamento detalhado e rigoroso de suas atividades visando a correta validação e verificação dos requisitos de segurança.

Segundo Cho(2001) o correto gerenciamento deste processo complexo envolve o conhecimento detalhado de cada atividade (entradas, saídas, duração), do fluxo de informações, das possíveis iterações que possam existir e das técnicas de programação de projetos visando a definição do planejamento a ser efetuado. Este planejamento tem como objetivo o mapeamento das atividades visando a otimização de seu seqüenciamento de forma a garantir o cumprimento dos prazos estabelecidos.

Neste sentido, o próximo tópico apresentará as principais características do gerenciamento de processos complexos em projetos.

2.4 Gerenciamento do Processo de Projetos

O estudo do gerenciamento de projetos complexos deve levar em consideração alguns tópicos importantes como o conceito de iteração, gerenciamento do tempo de projetos, decomposição hierárquica de atividades, fluxo de informações e as técnicas de programação.

2.4.1 Conceitos de Iteração em Projetos Complexos

Os processos de gerenciamento de projetos são realizados por uma equipe do projeto e geralmente se enquadram em duas categorias principais que estão em constante interação:

- Processo relacionado à descrição, organização e conclusão do trabalho, cujo objetivo é o de cumprir as fases do ciclo de vida de um projeto: iniciar, planejar, executar, controlar, e encerrar.

- Processo orientado ao produto, onde são especificados e criados os produtos. Este processo e normalmente relacionado às fases do ciclo de vida do projeto e produto.

A figura 2.16 representa a relação entre o ciclo de vida do produto e de um projeto.

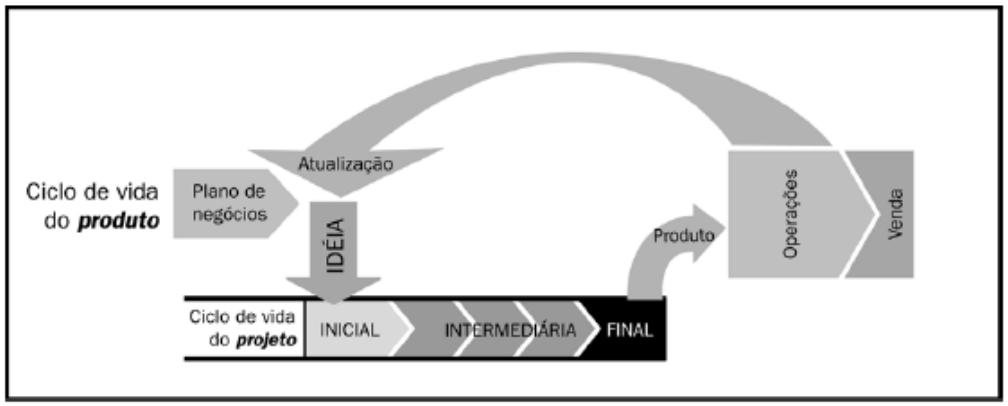


Figura 2.16: Ciclo de vida do produto versus o ciclo de vida do projeto (Fonte: PMBOK, 2004)

Segundo o PMBOK (2004), um gerenciamento de projetos bem sucedido inclui o gerenciamento ativo das interações entre estes processos, de forma a atender as necessidades das partes interessadas.

Eppinger e Salminen (2001) afirmam que o desenvolvimento de produtos complexos é um processo altamente interativo e geralmente complexo, pois envolve a troca de informações entre inúmeras tarefas.

Yassine e Braha (2003) afirmam que alguns problemas fundamentais da engenharia simultânea devem ser solucionados pelo gerenciamento do processo de projetos complexos. Estes aspectos são:

- Iteração
    - Alterações de projeto no decorrer do processo que geram retrabalho.
- Paralelismo (*Overllaping*)
    - Superposição de atividades do processo visando redução do tempo de desenvolvimento.

- Decomposição e Integração
    - Decomposição de um sistema complexo em subsistemas que possam ser gerenciados de forma independente.

- Convergência
    - O processo de desenvolvimento de um produto é considerado estável se o número total de problemas de projeto sendo solucionados é maior que o número total de problemas sendo criados.

Yassine e Braha (2003) salientam que estes problemas fundamentais (com excessão da convergência) são considerados não-temporais ou estáticos (exemplo: formação de times ou da arquitetura de produtos). Já o problema da convergência é considerado temporal ou dinâmico, uma vez que envolve o entendimento do comportamento complexo das tarefas de desenvolvimento do produto ao longo do tempo.

Eppinger (2002) afirma que o desafio essencial num ambiente de engenharia simultânea é o de integrar peças separadas em uma solução para o sistema e que a necessidade de integração depende da interação entre as atividades voltadas a resolução de um determinado problema.

Em um ambiente de engenharia simultânea, equipes multi-disciplinares atuam com o objetivo de resolver questões que envolvem etapas diferentes do processo de forma simultânea (exemplo: projeto e produção, projeto e qualidade, etc) garantindo assim a redução do tempo de ciclo.

De uma maneira geral, o estudo das iterações em projetos complexos envolve o conhecimento de alguns tópicos importantes como o estudo do tempo de ciclo em projetos, configurações que caracterizam os relacionamentos entre elementos de um sistema e os tipos de iteração existentes.

a) Tempo de Ciclo em Projetos

Browning (1998) afirma que a redução do tempo de ciclo é um importante aspecto do desenvolvimento integrado do produto. Assim, é possível obter vantagem competitiva e redução de custos.

Ainda segundo Browning (1998) os principais desafios na redução do tempo de ciclo em um projeto estão inseridos nos seguintes aspectos:

- Distribuição ineficiente de pessoas e recursos (ferramentas, informações, etc).
- Requisitos instáveis de produto causam indecisão e geram reprojeto (algumas decisões devem ser realizadas o mais cedo possível).
- Algumas atividades de curta duração necessitam de atividades de longa duração para produzir os resultados esperados (retrabalho).
- Integração e coordenação das atividades e dos resultados.
- Atividades altamente interdependentes aumentam as chances de iteração ou retrabalho.
- A redução de tempo de ciclo não deve ser feita simplesmente através da redução dos tempos planejados, uma vez que eleva os riscos de não cumprimento das atividades.
- Aparentemente, muitas soluções para a redução do tempo de ciclo possuem efeitos colaterais que podem reduzir a efetividade antecipada. Desta feita, a redução efetiva do tempo de ciclo exige uma perspectiva sistêmica, levando em conta os *feedbacks* que tenderão a diminuir a eficácia geral do processo.

Browning e Eppinger (2000) afirmam que o problema da redução do tempo de ciclo não pode ser minimizado apenas através da transformação de atividades executadas em série, em atividades executadas em paralelo (*overlapping*). Agindo-se desta maneira sem considerar as interdependências que possam existir entre as atividades, pode-se gerar desperdício de recursos sem a redução desejada no tempo de ciclo, ou seja, causando retrabalho e iteração.

Segundo Yassine e Braha (2003), a boa prática da engenharia simultânea requer a habilidade de uma organização em compartilhar informações (no tempo requerido) entre membros de times multi-funcionais de desenvolvimento do produto.

b) Configurações que Caracterizam um Sistema

Yassine (2004) afirma que existem três tipos básicos de configurações que caracterizam o relacionamento entre os elementos de um sistema: paralelo (simultâneo), seqüencial (dependente) e iterativo (interdependente). A figura 2.17 representa estas configurações através da utilização de grafos direcionados.

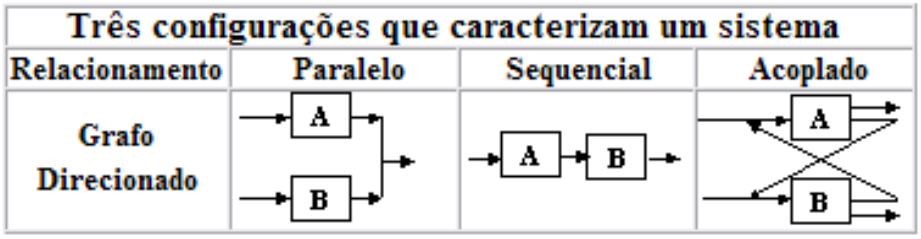


Figura 2.17: Tipos de relacionamento entre elementos de sistema (Fonte: www.dsmweb.org)

Na configuracao paralela, os elementos de sistema não interagem uns com os outros e o entendimento do comportamento individual de cada elemento permite o completo entendimento do comportamento do sistema. O elemento A não depende das informações do elemento B e vice-versa.

Na configuração seqüencial um elemento influencia o comportamento do outro de forma unidirecional, ou seja, o elemento B depende das informações do elemento A.

Já na configuração acoplada ou iterativa ocorre a interdependência entre os elementos de sistema e o fluxo de informação e entrelaçado, uma vez que A influencia B e B influencia A. Neste caso é caracterizada uma dependência cíclica ou circuito (Yassine, 2004).

c) Definição de Iteração

Smith e Eppinger (1993) definem iteração como sendo o processo de repetição de tarefas que é comum a todos os tipos de projetos de engenharia sendo, segundo Browning (1998), uma barreira para a redução do tempo de ciclo.

Browning e Eppinger (2000) afirmam que o estudo da iteração se tornou ainda mais importante com a ênfase dada ao conceito de engenharia simultânea, onde atividades que eram anteriormente consideradas distintas e seqüenciais são agora sobrepostas, resultando numa necessidade maior de coordenação das interações e retroalimentações (*feedbacks*).

Safoutin e Smith (1996) afirmam que existem variedades da iteração que são visíveis e que denotam de ciclos de modificações em um projeto em evolução. Cho (2001) descreve dois tipos principais de iteração em projetos com relação ao fluxo de informação:

- Iteração com sobreposição.
- Iteração sequencial.

A iteração de sobreposição pode ser causada pelo recebimento de uma nova informação de atividades sobrepostas, após início com uma entrada preliminar. Já a iteração seqüencial pode ser causada por uma mudança de uma entrada quando outras tarefas são retrabalhadas ou quando ocorre falha em atender a algum critério previamente estabelecido.

De uma forma geral, a iteração sequencial em projetos implica no retrabalho ou refinamento, retornando para atividades previamente trabalhadas.

Segundo Smith e Eppinger (1997), a iteração em projetos pode ser causada por vários fatores, principalmente:

- Atualização de informações devido a outras atividades.
- Mudanças no objetivo do projeto.

Eppinger (1995) afirma que uma atualização de informações pode ser originada nos seguintes cenários:

- Atividades posteriores causando mudancas na entrada de dados de atividades anteriores (por exemplo: mudança de requisitos ou parâmetros de projeto, correção de premissas utilizadas, etc). Neste caso é destacado o fenômeno do *feedback* de informação.

- Atividades anteriores causando mudança na entrada de dados de atividades posteriores devido a erros ou incompatibilidades (por exemplo: etapas de verificação e validação). Neste caso é apresentado o fenômeno do retrabalho.

- Atividades acopladas causando mudanças na entrada de dados de uma determinada atividade, freqüentemente em função de mudanças no compartilhamento de informações.

Browning (1998) afirma que a falha em se convergir para certas especificações de projeto podem requerer retrabalho de atividades precedentes. Este tipo de deficiência no projeto é mais provável quando os requisitos e objetivos são muito susceptíveis a modificações.

Ainda segundo Browning (1998), a redução do tempo de ciclo no desenvolvimento do produto pode ser realizada através dos seguintes passos:

- Minimização das iterações não intencionais (ou não-planejadas), ou seja, que resultam da nova informação que chega com atraso durante o processo (Browning 1998; Cho 2001).
    - Este tipo de iteração pode ser minimizada através das seguintes ações:
        - Seqüenciamento correto das atividades.
        - Redução de erros de projeto.

- Gerenciamento das iterações intencionais (ou planejadas), ou seja, aquelas que são propositalmente executadas em um processo visando convergência a uma solução desejável (Browning 1998; Cho 2001).
    - Este tipo de iteração pode ser minimizada através das seguintes ações:
        - Seqüenciamento correto das atividades.
        - Co-alocação de pessoas que executam atividades altamente interdependentes.
        - Integração de ferramentas de análise de engenharia.
        - Remoção de atividades extras do processo.

Similarmente, Cho (2001) afirma que o correto gerenciamento das iterações (descritas por ele como planejadas e não-planejadas) deve ser realizado de forma que planos apropriados de contigência possam ser criados como parte do planejamento otimizado do processo.

Uma iteração planejada representa uma possível iteração entre atividades sobrepostas (iteração com sobreposição) ou atividades localizadas dentro de blocos acoplados (iteração seqüencial). Já uma iteração não planejada representa a possivel iteração entre fases principais de desenvolvimento que não são levados em consideração no planejamento de um projeto. Este tipo de iteração é normalmente causada por mudanças inesperadas de mercado e inovações tecnológicas no meio do processo.

Goldt (1995) afirma que a redução efetiva do tempo de ciclo requer pensamento de sistema e foco no processo contribuinte. O fluxo de informações no processo deve ser aberto e direto de forma a evitar a formação de silos, ou seja, pacotes de informação gerados por grupos organizacionais que se tornam indisponíveis para outros grupos. Canais de comunicação mais fechados (maior interação) previnem este tipo de fenômeno no processo.

Verifica-se assim que o correto gerenciamento do fluxo de informações é um fator importante na redução de tempo de ciclo de um projeto. O correto seqüenciamento das atividades de um processo tem um papel importante neste sentido, uma vez que além de minimizar as iterações que possam existir, melhora o nível de interação entre as atividades.

2.4.2 Gerenciamento do Tempo em Projetos

A figura 2.18 representa uma visão geral do gerenciamento do tempo de projeto (segundo o PMBOK, 2004), onde os processos de definição e seqüenciamento de atividades são apresentados.

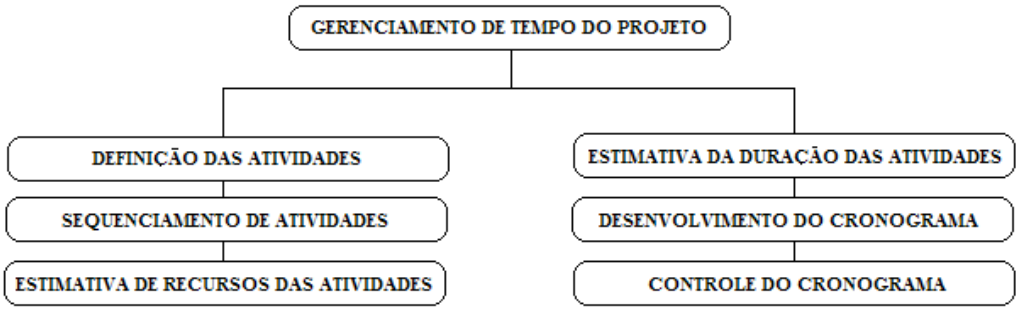


Figura 2.18: Visão geral do gerenciamento do tempo de projeto
(Fonte: Adaptado do PMBOK, 2004)

Segundo Cho (2001), a identificação e a definição das atividades, bem como as informações trocadas entre elas são fundamentais para o entendimento da estrutura de um projeto complexo.

Num projeto complexo, o efetivo gerenciamento do tempo de ciclo exige o desdobramento das macro-atividades em atividades mais facilmente gerenciáveis. Este desdobramento pode ser efetuado através das “estruturas analíticas do projeto”.

O processo de seqüenciamento de atividades visa a identificação das dependências ou relacionamentos lógicos entre as atividades do projeto. Basicamente, este processo tem como entradas a lista de atividades do projeto e também seus atributos. Como saída, são identificadas atualizações na lista e nos atributos das atividades, bem como o correto seqüenciamento das atividades que será usado na construção do cronograma do projeto (utilizando estimativas de duração e recursos).

2.4.3 Decomposição Hierárquica (Estrutura Analítica de Projeto)

O PMBOK (2004) define a Estrutura Analítica do Projeto (EAP) ou *Work Breakdown Structure* (WBS) como sendo uma decomposicao hierárquica das atividades de um projeto visando o atingimento de seus objetivos e de seu escopo (resultados, abordagem e conteúdo).

Valeriano (1998) afirma que a EAP consiste na criteriosa decomposição tanto do produto quanto de seus processos e é considerada uma entrada fundamental na definição do planejamento de um projeto.

Segundo Haugan (2001), a EAP foi criada na década de 60 pela NASA e pelo DoD, e sua metodologia foi primeiramente apresentada através da norma MIL-STD-881 (*Work Breakdown Structures for Defense Materiel Items*).

Peralta (2002), que utiliza o termo Estrutura de Desdobramento do Trabalho (EDT), afirma que as atividades podem ser planejadas e controladas com o detalhamento das etapas do processo de projeto.

Chase, Jacobs e Aquilano (2006), que designam a EAP como uma Estrutura Analítica do Processo, afirmam que ela é importante na organização de um projeto uma vez que ela o quebra em partes ou níveis controláveis.

Estes níveis controláveis, também conhecidos como entregas (*deliverables*), são definidos como qualquer resultado mensurável, tangível e verificável que deve ser fornecido de forma a completar um projeto ou parte dele.

De uma forma análoga à decomposição de sistemas, a utilização da EAP visa administrar melhor a complexidade de um projeto uma vez que o gerenciamento de partes menores permite definir os relacionamentos e as interfaces entre eles de uma forma mais simples.

Segundo o PMBOK (2004), as entregas do nível mais baixo de uma EAP são conhecidas como pacotes de trabalho (*work packages*), que podem ser subdivididos em componentes mais gerenciáveis chamados de atividades. Esta quebra e feita visando fornecer uma base para a estimativa de um cronograma.

Uma EAP pode ser representada na forma de organograma ou árvore de decomposição, ou também na forma de tabela. Em alguns casos, a representação seqüencial das fases de um projeto pode ser aplicada para facilitar a visualização. A figura 2.19 e a tabela 2.4 abaixo representam as duas representações de uma EAP:

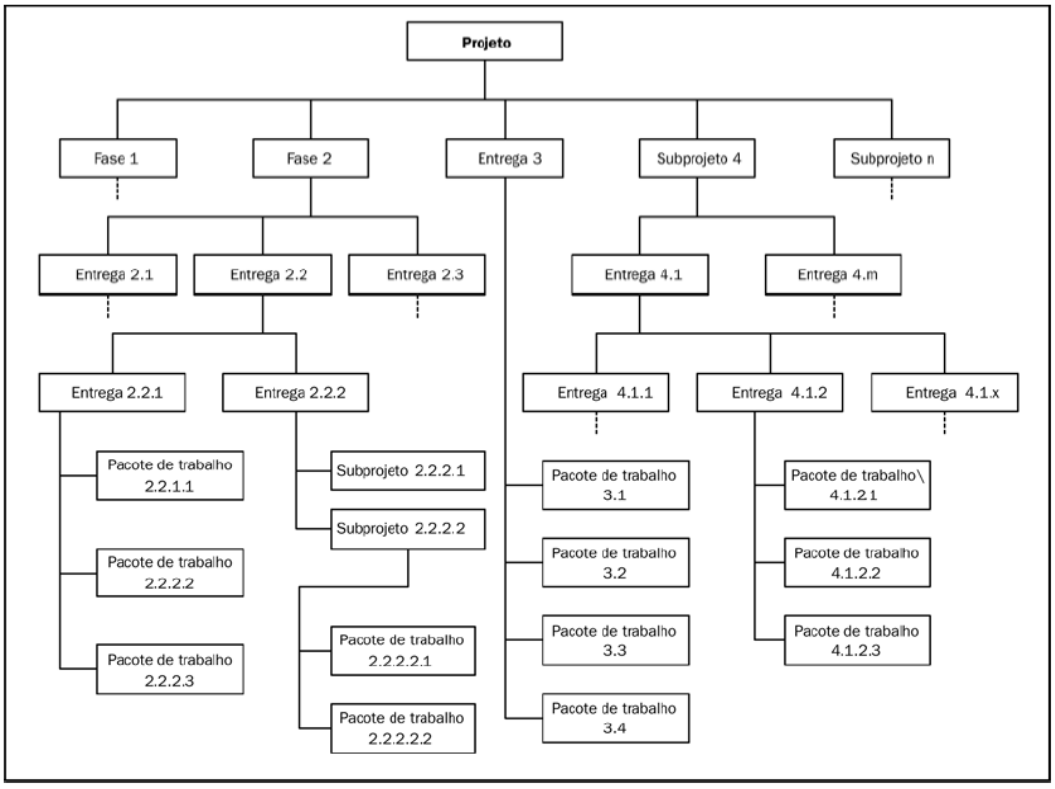


Figura 2.19: EAP na forma de uma árvore de decomposição (Fonte: PMBOK, 2004)

Tabela 2.4: Exemplo de uma EAP na forma de tabela (Adaptado de Peralta, 2002)

TABELA DE ATIVIDADES					
Atividade	**Responsável**	**Duração**	**Atividade Precedente**	**Recursos**	**Interfaces**

Numa EAP, cada etapa ou bloco deve conter uma declaração de trabalho, que é composta basicamente dos seguintes dados:

- Descrição.
- Responsável.
- Entradas.
- Recursos e as interfaces.
- Saidas.
- Duração.

Uma vez obtida as atividades do projeto e suas respectivas informações oriundas da declaração de trabalho da EAP, é possível determinar a programação do projeto, que consiste no correto seqüenciamento das atividades e na construção de um cronograma.

Peralta (2002) afirma que a aplicação da EAP em processos tem por objetivo a compreensão de todas as atividades necessárias para a realização dos objetivos do projeto e não deve refletir a estrutura da organização nem decompor o produto em disciplinas.

Segundo Hoffmeister (2003), existem três formas de se orientar uma EAP para decompor o trabalho inserido no desenvolvimento de um produto. São elas:

- Orientação fornecida pelos componentes do produto a ser desenvolvido.
- Através da estrutura de funções (análise funcional).
- A partir das ferramentas e procedimentos empregados para desenvolvê-lo.

Ainda segundo Hoffmeister (2003), no desenvolvimento de novos produtos é indicada a utilização das EAPs orientadas a partir dos resultados das ferramentas e tarefas dos procedimentos empregados no processo de desenvolvimento de produtos e prescritas em metodologias de projeto, como por exemplo, o QFD (*Quality Function Deployment*).

Caso o produto já esteja desenvolvido e se deseja a otimização ou o reprojeto, as EAPs orientadas pelos componentes do produto ou as de estrutura de funções são as mais indicadas.

2.4.4 Fluxo de Informação em Projetos

Como visto anteriormente, a representação das atividades de um projeto baseada no fluxo de informações é muito importante para o correto seqüenciamento das atividades visando a redução do tempo de ciclo. As interligações entre as atividades neste caso devem ser definidas pela dependência de informações e não pelo seu fluxo de trabalho no tempo.

Segundo Cho (2001) duas atividades pode apresentar dois tipos de fluxo de informação:

- Fluxo onde uma atividade posterior requer os dados de saída finais de uma atividade anterior para iniciar seu trabalho (chamada pelo autor de Tipo "1").
- Fluxo onde uma atividade posterior utiliza dados de saída finais de uma atividade anterior não no início, mas no decorrer de seu processo e/ou se inicia com uma informação preliminar da atividade anterior (chamada pelo autor de Tipo "2").

A figura 2.20 abaixo representa estes dois tipos de fluxo:

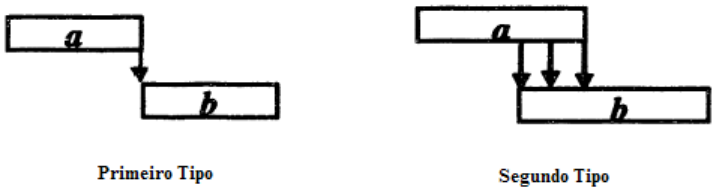


Figura 2.20: Tipos de fluxo de informação entre atividades (Fonte: Cho, 2001)

Além do tipo de fluxo, as dependências de informação também podem ser categorizadas de acordo com seu tipo.

Segundo Cho (2001) as dependências de informação entre as atividades de um projeto podem ser vinculativas (*binding*) ou não-vinculativas (*non-binding*). A vinculativa representa uma dependência que é considerada restritiva entre duas atividades dependentes, onde o atraso na transferência de informação de uma atividade anterior pode afetar diretamente a posterior. Já em uma dependência não-vinculativa não ocorre nenhum efeito restritivo entre duas atividades dependentes, uma vez que o atraso na transferência de informação não impacta diretamente a programação, mesmo havendo fluxo de informação entre as atividades.

A figura 2.21 exemplifica dois tipos de dependência de informação entre atividades:

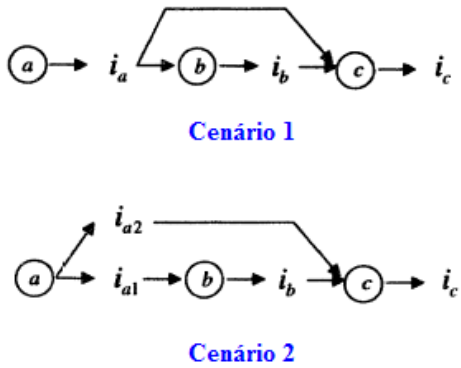


Figura 2.21: Tipos de dependência de informação entre duas atividades (Fonte: Cho, 2001)

Sendo, por exemplo, i_a, i_b e i_c dados dimensionais das atividades “a”, “b” e “c”, no primeiro cenário, a atividade “c” necessita das informações i_a e i_b para se iniciar. Neste caso, a informação i_a é recebida por ambas as atividades “b” e “c”. Já no segundo caso, a atividade “c” necessita de informações da atividade “b” (no caso i_b), e também de uma outra informação adicional que provem de “a” (no caso i_{a1}). Neste caso, a informação i_{a1} é somente recebida pela atividade “c”.

Em ambos os cenários (1 ou 2), apesar da atividade “c” necessitar de informações da atividade “a”, ela não fornece uma precedência restritiva pois a atividade “a” deve estar completada somente quando a atividade “c” começar (após a atividade “b” estar finalizada). Portanto, a dependência de informação entre as atividades “a” e “c” é não-vinculativa, enquanto as outras duas são vinculativas.

Vale ressaltar que tanto o fluxo tipo “1” quanto o fluxo tipo “2” (conforme figura 2.21) podem caracterizar dependências vinculativas ou não-vinculativas.

Segundo Prasad (1996) existem inúmeras formas de representação gráfica do fluxo de informação entre as atividades de um processo. De uma maneira básica, esta representação se baseia na identificação das entradas e das saídas nas atividades.

Conforme representado na figura 2.18, os fluxos de informação podem ter uma configuração sequencial, paralela e interdependente (acoplado). Estas configurações podem ser representadas graficamente e alguns modelos podem ser utilizados.

Uma forma gráfica de representação destas atividades pelo mapeamento do fluxo de informações se dá através dos diagramas IDEF0 (*Integration Definition for Function Modeling*) que geram uma visão da dependência de informações entre as atividades.

Prasad (1996) afirma que o diagrama IDEF0 é a forma gráfica mais utilizada para a representação de atividades durante uma modelagem conceitual.

Segundo Austin et al (1999), os diagramas IDEF0 foram criados na década de 70 para a indústria aeroespacial americana e tiveram como base a metodologia SADT (*Structured Analysis and Design Technique*).

A visão hierárquica baseada no fluxo de informações denota a identificação dos *inputs* e *outputs* de cada atividade bem como os recursos e controles necessários. A figura 2.22 representa um diagrama IDEF0.

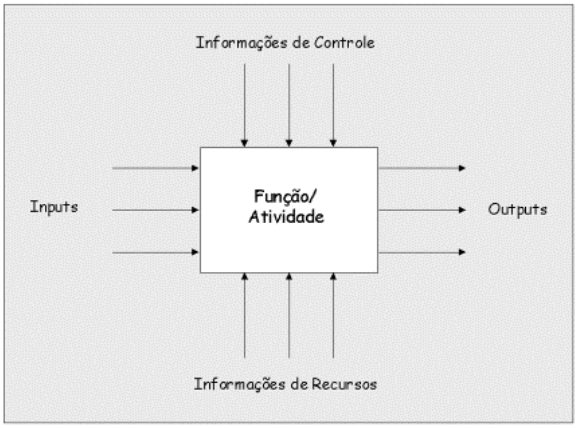


Figura 2.22: Elementos básicos de um Diagrama IDEF0 (Fonte: www.numa.org.br)

Cada atividade ou função é conceitualmente representada por uma caixa retangular contendo quatro setas que representam os componentes que compõe o modelo. Estas setas representam as interfaces entre as atividades e cada atividade pode ser decomposta em diversos níveis, seguindo a mesma convenção. Sendo assim, o modelo completo de um diagrama IDEF0 é uma representação hierárquica de um processo composto de atividades ou funções em vários níveis.

Prasad (1996) salienta que a utilização do IDEF0 na representação de modelos de engenharia ou manufatura tem encontrado dificuldades uma vez que ela não permite a modelagem de iterações entre níveis do processo.

Bosilj-Vuksic, Giaglis e Hlupic (2001) e Mykolayczky e Tortato Jr. (2006) são autores que também descrevem com detalhes o IDEF0 como ferramente de representação de fluxos em processos.

Outra forma de representação do fluxo de informação em um processo é através das redes de Petri. Segundo Prasad (1996) as redes de Petri fornecem uma representação matemática do sistema. Assim como no diagrama IDEF0, as redes de Petri são compostas de quatro elementos principais: um conjunto de posições, um conjunto de transições, uma função de entrada e uma função de saída. A figura 2.23 apresenta um exemplo da utilização de uma rede de Petri.

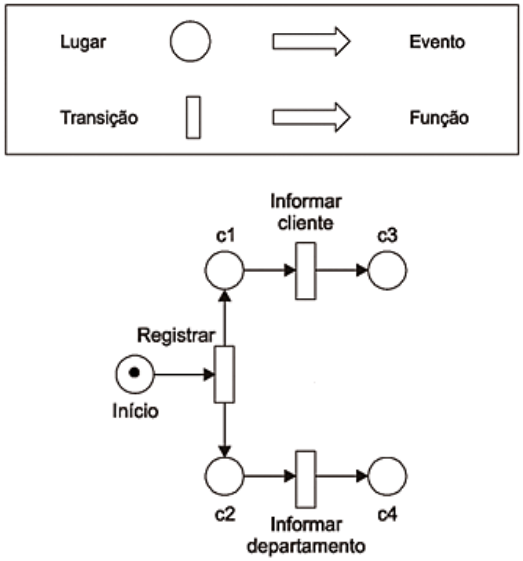


Figura 2.23: Elementos básicos de uma rede de Petri (Fonte: www.numa.org.br)

As redes de Petri permitem um modelamento gráfico do comportamento do sistema, simultaneamente permitindo a introdução de regras matemáticas para a definição do modelo.

Segundo Bosilj-Vuksic, Giaglis e Hlupic (2001), as redes de Petri permitem a representação da dinâmica do sistema através da dinâmica de chegada de entidades, disponibilidade de recursos, interdependência de recursos, tempo de fila etc.

Para Prasad (1996), as redes de Petri não capturam os fluxos de informação como feito nos diagramas IDEF0, uma vez que são mais rigorosas do que o IDEF0, mas não são tão úteis para a análise de processos de produto visto que possuem uma representação bem mais complexa.

Do ponto de vista da melhoria de um processo, o estudo do fluxo de informação entre atividades tem grande importância uma vez que permite a transformação de um modelo "as is", ou seja, aquele que representa o processo atual (incluindo aspectos de tempo), em um modelo "to be", que representa o processo melhorado.

Para tal, a utilização de técnicas de programação de projetos tem fundamental importância, uma vez que podem atuar na otimização do seqüenciamento das atividades, seja através do fluxo de atividades, seja através do fluxo de informação.

2.4.5 Técnicas de Programação de Projetos

Após o mapeamento das atividades de um projeto, seja através do seu fluxo de trabalho ou através do fluxo de informações, é necessário definir a programação do projeto, ou seja, montar um seqüenciamento temporal de maneira a dispô-las na melhor ordem possível para o projeto.

Segundo Peralta (2002), a programação do projeto tem por objetivo arranjar sistematicamente as tarefas visando à realização de um objetivo. Além disso, a programação exige um alto grau de detalhamento uma vez que a montagem do cronograma do projeto se realiza nesta etapa. Sob o ponto de vista do seqüenciamento de atividades, as principais formas de representação das atividades de um projeto são:

- Diagramas de Gantt.
- Diagramas de Precedência de Rede.
- Matriz de Estrutura de Projeto (*Design Structure Matrix*) ou DSM.

Segundo Ulrich e Eppinger (2008), os diagramas de Gantt não apresentam de forma explícita as dependências entre tarefas. Essas dependências ditam quais tarefas devem ser completadas antes de outras iniciarem.

Assim como no Diagrama de Gantt, os diagramas de rede do PERT/CPM só conseguem representar os relacionamentos de dependência seqüencial e paralela entre as atividades de um projeto.

Ulrich e Eppinger (2008) salientam que a representação PERT/CPM não permite a indicação de *loops* ou *feedback*s e desta maneira não pode explicitamente mostrar as interdependências entre as atividades.

Assim sendo, com relação ao seqüenciamento das atividades de um projeto baseado na análise do fluxo de informações, o PERT/CPM e o Diagrama de Gantt não permitem o gerenciamento de atividades interdependentes (realimentação e iteração) uma vez que são técnicas que objetivam a análise do fluxo de trabalho (Yassine, 2004).

Danilovic e Sandkull (2004) afirmam que a informação capturada numa análise utilizando a DSM é similar a aquela em um grafo direcionado ou em um PERT/CPM. Entretanto, a representação na forma de matriz torna possível a criação de um modelo de fluxo de informação e de uma análise de interdependência mais abrangente para projetos complexos.

Assim sendo, para lidar com o gerenciamento de atividades através da análise do fluxo de informação, principalmente quando ocorre interdependência entre as atividades, a técnica mais indicada é a Matriz de Estrutura de Projeto ou DSM.

2.5 Matriz de Estrutura de Projetos (DSM)

Segundo Browning (1998), a fonte primária dos conceitos da DSM cresceu com os esforços em se resolver sistemas de equações nos anos 50 e 60 (matemática matricial, diagramas de precedência de rede, diagramas de relacionamento de rede e diagramas N2).

Warfield (1973), um dos percursores da técnica DSM, publicou um trabalho sobre o uso de matrizes binárias na modelagem de sistemas. Segundo ele, as matrizes binárias podem representar a presença ou ausência de um tipo específico de relação entre pares de elementos de sistema, abrindo oportunidades para a sua estruturação. Neste trabalho eram apresentados os primeiros algorítmos para o particionamento de matrizes.

Smith e Eppinger (1997) afirmam que as primeiras aplicações deste tipo de análise voltadas à melhoria dos projetos de engenharia surgiram com Steward (1981).

Steward (1981), em seu trabalho "*The Design Structure System: A Method for Managing the Design of Complex Systems*", idealiza e apresenta as primeiras utilizações da DSM voltadas à identificação e minimização dos ciclos de iteração no planejamento de projeto de sistemas. Os algorítmos de particionamento foram utilizados para demonstrar a técnica através da identificação dos diferentes tipos de relacionamento entre as atividades.

Steward (1981) reforça que a DSM não substitui as técnicas tradicionais de gerenciamento de projetos (por exemplo: PERT/CPM ou Diagramas de Gantt), mas fornece uma análise preliminar que proporciona uma programação das atividades ainda mais efetiva, uma vez que os fluxos de informação são mapeados.

Os primeiros algoritmos utilizados para auxiliar a análise das matrizes na DSM foram desenvolvidos em 1989 por J. L. Rogers da NASA. O uso na indústria comecou em 1990 quando alguns alunos e professores do MIT (*Massachusetts Institute of Technology*) passaram a desenvolver pesquisas em torno da DSM e expandiram suas aplicações. As pesquisas no MIT tiveram a liderança do Prof. Dr. Steven D. Eppinger.

Nos últimos anos, este trabalho se expandiu além do MIT e a necessidade resultante para um nome mais genérico levou ao termo "*Dependency Structure Matrix*" (Browning, 1998). Desde então, inúmeros trabalhos têm sido desenvolvidos e utilizados com sucesso em todo mundo.

Dentre as áreas que passaram a utilizar a DSM como ferramenta de planejamento e gestão de projetos, pode-se citar uma lista extensa de trabalhos publicados nos últimos 18 anos:

- Automobilística
  - Black, Fine e Sachs (1990).
  - Sequeira (1991).
  - Eppinger et al (1993).
  - McCord e Eppinger (1993).
  - Pimmler e Eppinger (1994).
  - Yassine et al (2001).
- Construção Civil
  - Austin et al (2000).
  - Peralta (2002).
  - Rodriguez (2005).
  - Manzione (2006).
- Aeronáutica e Aeroespacial
  - Grose (1994).
  - Browning (2002).
- Semicondutores
  - Osborne (1993).
  - Dong e Whitney (2001).
- Indústria Naval
  - Pieroni e Naveiro (2006).
- Gerenciamento de projetos e processos de engenharia
  - Smith e Eppinger (1995).
  - Browning (1998).
  - Yassine e Braha (2003).
  - Danilovic e Browning (2006).
  - Helo (2006).

Yassine (2004) define a DSM como uma técnica para manipulação de informação que permite a representação de relacionamentos complexos visando determinar a sequência mais sensível para as tarefas. Desta forma e possível melhorar o planejamento, a execução e o gerenciamento de projetos complexos de desenvolvimento do produto.

Austin et al (2000) descrevem a DSM como uma ferramenta usada para identificar ciclos de iteração dentro do processo de projeto, com o objetivo de otimizar a ordenação das atividades.

De uma forma geral, a DSM é uma ferramenta de análise de sistemas e de gerenciamento de projetos. Como ferramenta de análise de sistemas ela fornece uma representação clara e compacta de um sistema complexo e um método de captura de interações, interdependências e interfaces entre elementos de sistema.

Como ferramenta de gerenciamento de projetos, a DSM fornece subsídios de se representar realimentação e dependências de tarefas cíclicas. Isto é muito importante uma vez que a maioria das aplicações de engenharia apresenta esta propriedade.

De maneira a explicar melhor os conceitos da DSM, iremos dividir este tópico nas seguintes partes: Grafos Direcionados e matrizes binárias, formas de representação e leitura de uma DSM, tipos de DSM, particionamento de matrizes e vantagens e desvantagens da DSM.

a) Grafos Direcionados e Matrizes Binárias

Os primeiros estudos na área de modelagem de sistemas utilizavam a teoria dos grafos para representar a relação entre elementos de sistema.

Segundo Feofiloff, Kohayakawa e Wakabayashi (2005), um grafo é um conjunto de pontos (vértices ou nós) conectados por linhas (arestas ou arcos) que é definido por um par de conjuntos (V, A). Sendo assim, V representa um conjunto não-vazio (vértices do grafo) e A representa um conjunto de pares ordenados a=(v,w), onde v e w $\in$ V (arestas do grafo). Quando as arestas que ligam os vértices possuem orientação, o grafo é denominado direcionado (Figura 2.24).

Seja, por exemplo, o grafo G1(V, A) dado por **G1 (V, A)**, onde:

V = {grupo de atividades}, ou seja, V = {1, 2, 3, 4, 5, 6, 7, 8}

A = {(v, w) | relacionamento entre atividades}, ou seja, A = {(1, 4), (1, 2), (1, 3), (2, 3), (2,6), (3,4), (3,5), (4,5), (5,6), (5,7) (6,8), (7,8)}.

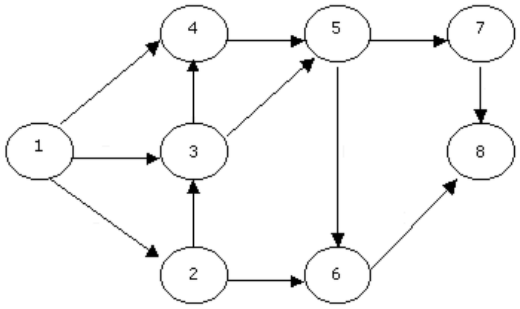


Figura 2.24: Grafo Direcionado (Fonte: Feofiloff, Kohayakawa e Wakabayashi, 2005).

A matriz de representação de um grafo direcionado é binária e quadrada contendo m linhas e colunas e n elementos diferentes de zero, onde m representa o numero de vértices e n o numero de arestas num grafo.

Considere dois elementos de sistema Si e Sj, onde "i" e a linha e "j" a coluna:

S = [S1, S2, S3,...,Sn] com n elementos

Si *R* Sj, ou seja, Si tem relação com Sj.

Si ~~*R*~~ Sj, ou seja, Si não tem relação com Sj.

Segundo Warfield (1973), com esta relação é possível construir uma matriz binária cuja entrada na posição (i,j) será 1 quando Si *R* Sj e será zero quando Si ~~*R*~~ Sj. A matriz resultante será quadrada e terá a seguinte forma:

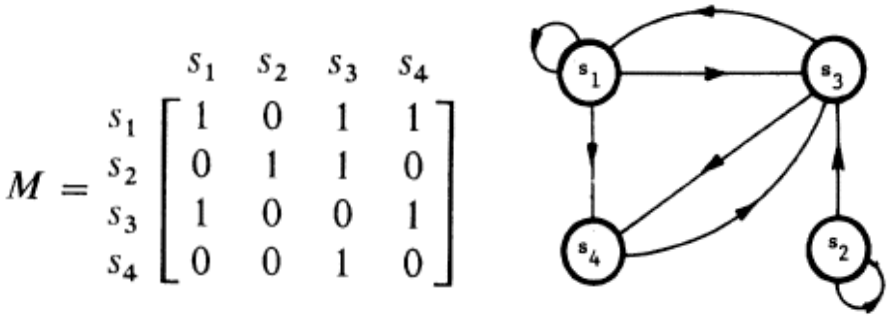


Figura 2.25: Exemplo de uma matriz binária e sua representação na forma de grafo direcionado (Fonte: Warfield 1973).

b) Formas de Representação e Leitura de uma DSM

Gebala e Eppinger (1991) afirmam que o uso de matrizes binárias para modelagem de sistemas é muito útil porque elas podem representar a presença ou a ausência de relacionamento entre pares de sistemas.

A grande vantagem da representação matricial sobre os grafos direcionados esta no seu nível de compactabilidade e na habilidade de prover um mapeamento sistemático entre os elementos de sistema que é claro e fácil de ler independentemente da ordem da matriz. A figura 2.26 mostra a relação entre a representação através de grafos e a DSM.

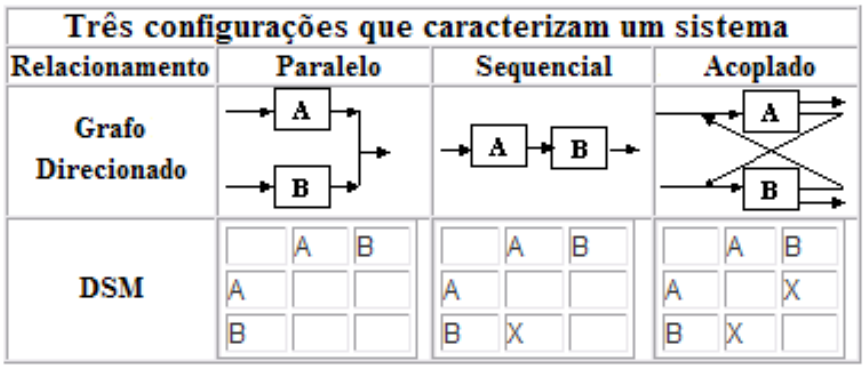


Figura 2.26: Tipos de relacionamento entre elementos de sistema (www.dsmweb.org)

Basicamente, a DSM é uma matriz quadrada com uma linha e uma coluna representando uma mesma atividade. As atividades são listadas em ordem cronológica com atividades precedentes listadas nas linhas superiores. As marcas nas células da matriz indicam se existem relações baseadas em informação, ou seja, se existe a relação entre a atividade "i" e a atividade "j", então esta relação é indicada com uma marca na célula [Aij] correspondente.

Browning (1998) afirma que as marcas nas células não necessariamente precisam ser valores binários ("1" ou "0") ou símbolos ("X" ou "vazio"). Elas também podem ser indicadores que possam representar:

- Grau de dependência entre as atividades.
- Probabilidade de iteração.
- Tipo de fluxo de dados.
- Percentual de retrabalho etc.

As durações das atividades também são consideradas juntamente com seu sequenciamento de forma a auxiliar na definição do cronograma e do caminho crítico em uma etapa posterior.

As marcas referentes a uma linha da matriz representam que aquela atividade recebe uma informação da atividade correspondente na coluna. Similarmente, marcas referentes a uma coluna da matriz representam que aquela atividade envia uma informação para a atividade correspondente na linha.

Segundo Yassine (2004), caso a ordem dos elementos na matriz represente uma sequência no tempo, então as marcas abaixo da diagonal principal representam que uma informação é transferida de uma atividade anterior (*upstream*) para outra posterior (*downstream*). Este tipo de marca é conhecida como marca de alimentação. Por outro lado, marcas acima da diagonal principal representam que uma informação é transferida de uma atividade posterior para uma atividade anterior. Neste caso, este tipo de marca é conhecida como marca de retroalimentação.

A figura 2.27 representa uma DSM e suas marcas:

Figura 2.27: Exemplo de uma DSM 12x12 (Fonte: Adaptado de Manzione 2006)

Na figura 2.27, a leitura através das linhas da matriz indica as atividades que enviam informação para a atividade correspondente. Por exemplo: a atividade correspondente a linha K, por exemplo, recebe informação das atividades B e C. Já a leitura através das colunas da matriz mostra as atividades que enviam informação para a atividade correspondente. Por exemplo: a atividade F envia informações para as atividades D, E, I, J.

As marcas de retroalimentação correspondem aos dados de entrada que não estão disponíveis no momento de se executar uma atividade. No caso da matriz mostrada na figura 2.27, as atividades D e E, por exemplo, dependem de informações de uma atividade posterior, ou seja, a atividade F. Neste caso, as informações da atividade F (que não estão disponíveis) terão que ser estimadas para posteriormente serem verificadas. A relação de interdependência entre a atividade F e as atividades D e E irá gerar ciclos de iteração que irão convergir para uma solução final.

Yassine (2004) denomina a região acima da diagonal principal de região de retroalimentação, enquanto que a região abaixo da diagonal principal é chamada de região de alimentação.

c) Tipos de DSM

Browning (2001) categoriza a DSM em dois tipos:

- Temporal
  - Atividade.
  - Parâmetro.

- Estáticas
  - Equipe ou Organização.
  - Componente.

Segundo Manzione (2006), na DSM temporal, a ordem das linhas e das colunas indica um fluxo através do tempo (fenômeno que ocorre no desenvolvimento de projetos de engenharia). Já as estáticas representam sistemas que convivem simultaneamente como equipes de uma organização ou componentes da arquitetura de um produto.

Em ambos os casos, as matrizes são analisadas através de diferentes algoritmos referenciados na literatura. O tipo de algoritmo a ser utilizado vai depender do tipo de dado que está sendo utilizado na DSM. A tabela 2.5 apresenta os tipos de dados por categoria da DSM e também o tipo de algoritmo relacionado (Browning, 2001):

Tabela 2.5: Dados representados na DSM (Fonte: Adaptado de Browning 2001)

Categoria da DSM	Tipo de Dado de DSM	Representação	Aplicação	Metodo de Analise (Algoritmos)
Temporais	***Atividade***	Tarefas, atividades e relacionamentos de entrada e saida	Programação de projetos, sequenciamento de atividades e redução de tempo de ciclo	Particionamento, "Tearing", "Banding", Simulação e Analise de Auto-Valores
	Parametro	Pontos de decisão de parametros	Sequenciamento de atividades de baixo nivel e construção de processos	Particionamento, "Tearing", "Banding", Simulação e Analise de Auto-Valores
Estaticas	***Equipes***	Caracteristicas de interface multi-times	Projeto Organizacional e integração de equipes	Blocagem ("Clustering")
	Componentes	Relacionamentos multi-componentes	Arquitetura de sistemas	Blocagem ("Clustering")

Como o objetivo deste trabalho é o de otimizar o seqüenciamento de atividades de um processo, será priorizado a aplicação dos algoritmos de particionamento.

d) Particionamento da DSM

Segundo Yassine (2004), o principal objetivo destes algoritmos de particionamento é o de eliminar ou minimizar das marcas de retroalimentação na matriz. A matriz sem marcas de retroalimentação acima da diagonal principal é chamada de matriz triangular inferior (completamente particionada).

Esta minimização ou redução das marcas de retroalimentação irá desencadear a redução do tempo de ciclo do projeto, conforme citado por Browning (1998).

Gebala e Eppinger (1991) afirmam que o objetivo principal do particionamento é o de re-sequenciar as atividades de um projeto visando maximizar a disponibilidade de informação requerida em cada estágio do processo de projeto.

Do ponto de vista da álgebra linear, conforme apresentado por Warfield (1973), o particionamento de uma matriz envolve inicialmente a permutação das linhas e das colunas de uma matriz M, de forma a originar uma matriz triangular inferior $M^{'}$, através da seguinte relação:

$$M^{'} = P^{-1}MP \qquad 2.1$$

Em que:

M' é uma matriz semelhante à matriz M.

P é a matriz de permutação obtida a partir da matriz identidade I.

P^{-1} é a matriz inversa de P

Comparando a matriz M' com a matriz original M, verifica-se que elas contém a mesma informação. Entretanto, a forma é diferente em função das permutações. A operação $P^{-1}MP$ intercambiou as linhas e colunas da matriz M de acordo com a matriz de permutação P desejada.

Desta forma, as matrizes quadradas M e M' são denominadas semelhantes, pois definem num espaço vetorial V um mesmo operador linear (transformação linear T: V → V), ou seja, duas matrizes M e M' são semelhantes se existe uma matriz inversível P, tal que $M' = P^{-1} MP$.

Caso ainda existam elementos acima da diagonal principal da matriz, é sinal que existem ciclos de iteração. Neste caso, o particionamento pode ser feito através da divisão da matriz M' em sub-matrizes quadradas menores que podem ser formadas ao longo da diagonal principal. A matriz M' nesta forma é chamada de triangular blocada. As matrizes C1, C2, C3 e C4 são chamadas de matrizes diagonais ou blocos e o relacionamento entre elas não deve apresentar interdependência.

Na figura 2.28 é possível conferir as etapas básicas do particionamento de uma matriz através do algoritmo *Reachability Matrix Method* desenvolvido por Warfield (1973):

$$M = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \\ \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array} \quad M' = \begin{array}{c} \\ 1 \\ 3 \\ 7 \\ 5 \\ 4 \\ 2 \\ 6 \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & 3 & 7 & 5 & 4 & 2 & 6 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \end{array}.$$

$$M' = \begin{bmatrix} C_1 & 0 & 0 & 0 \\ C_{21} & C_2 & 0 & 0 \\ C_{31} & C_{32} & C_3 & 0 \\ C_{41} & C_{42} & C_{43} & C_4 \end{bmatrix}$$

$$C_1 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \quad C_2 = [1] \quad C_3 = [1] \quad C_4 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Figura 2.28: O processo de particionamento de uma matriz (Fonte: Adaptado de Warfield, 1973)

Segundo Manzione (2006), a otimização da DSM permite o isolamento dos blocos onde ocorrem os ciclos de iteração. Assim, o gerenciamento do projeto se torna mais eficiente com a detecção destes ciclos, abrindo oportunidades para a correta tomada de decisão.

Através do particionamento é possível ver quais tarefas são seqüenciais, paralelas ou interdependentes. Uma vez que a matriz é particionada, as tarefas em série e em paralelo podem ser realizadas. Para as tarefas interdependentes, um planejamento se faz necessário uma vez que deve haver convergência no ciclo de iteração. A figura 2.29 ilustra os relacionamentos:

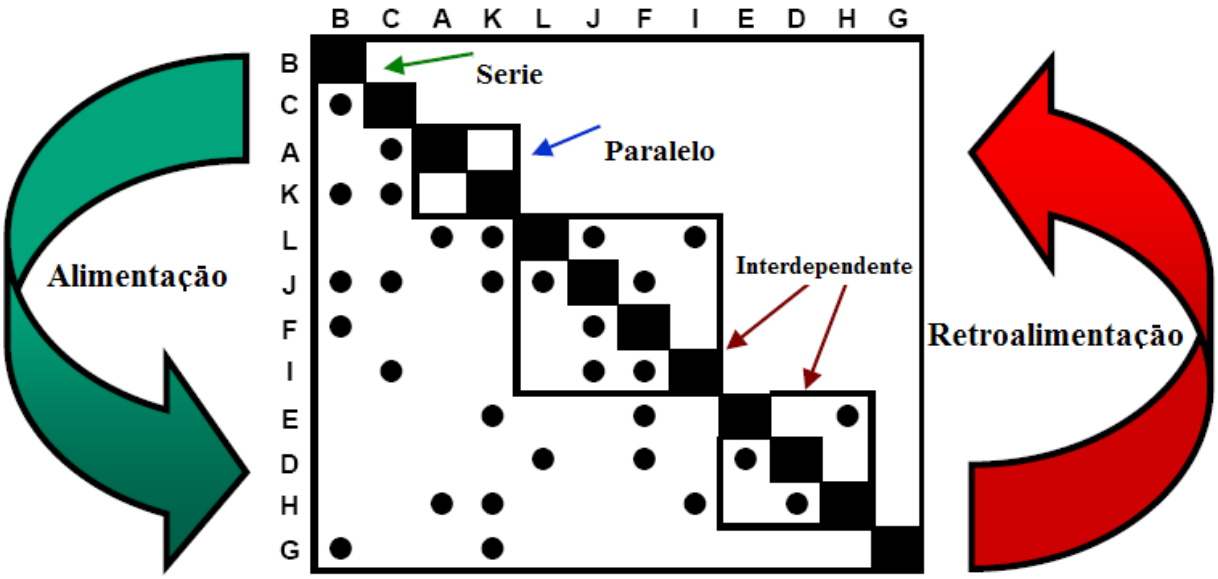


Figura 2.29: Matriz particionada e os relacionamentos entre as atividades (Fonte: Adaptado de Yassine 2004).

e) Principais Vantagens e Desvantagens da DSM

Para Browning (1998), as principais vantagens da DSM são:

- Representação concisa de processos complexos através de uma visão sistêmica.
- Representação clara das potenciais iterações nos processos.

- Forma mais precisa de gerenciamento voltado à antecipação de riscos programados.
- Demonstração apropriada de atividades simultâneas.
- Permite de forma rápida a análise de mudanças de sequência de atividades.
- Fornece visão de quando disponibilizar recursos para reduzir iterações não-intencionais.

Outra vantagem levantada por Manzione (2006) é que os modelos construídos com a DSM reduzem a complexidade do desenvolvimento e projeto de produtos, pois fornecem ferramentas analíticas que revelam fácil e claramente os fluxos de informação entre as atividades.

Por outro lado, as principais desvantagens da DSM relatadas por alguns autores são:

- Para ter bons resultados, a construção da DSM exige grande conhecimento das interações entre as atividades do processo (Dong, 2002).
- O uso da DSM é mais eficaz na melhoria de projetos maduros, uma vez que o mapeamento das interações é facilitado pela experiência (Dong, 2002).
- Segundo Tang, Zhang e Dai (2009), a utilização da DSM em produtos que nunca foram desenvolvidos antes não é aconselhável, uma vez que ainda não há conhecimento suficiente para avaliar as dependências existentes.
- O exame das interações entre diferentes domínios é limitado (Bartolomei, 2007).

É importante ressaltar mais uma vez que a utilização da DSM não invalida a utilização de outras técnicas de programação de projetos, como por exemplo, o PERT/CPM, o Diagrama de Gantt, o Método da Corrente Crítica, entre outros. Neste sentido, a utilização da DSM seria efetuada preliminarmente a aplicação de uma análise de caminho crítica, da preparação de um cronograma de projetos etc.

3 Método Proposto

3.1 Descrição do Método

Este capítulo tem por objetivo apresentar uma proposta de metodologia a ser implementada visando o emprego da matriz de estrutura de projetos (DSM) para otimização das atividades de um processo complexo.

Segundo Yassine (2004), o sucesso na utilização do método de DSM é determinado por uma apropriada decomposição do sistema em estudo e pela precisão dos relacionamentos de dependência coletados.

Cho (2001) sugere que o processo de construção de uma DSM para analise das atividades de um projeto contenha as seguintes etapas:

- Decomposição do projeto em partes gerenciáveis.
- Identificação das entradas, saídas e do tipo de fluxo de informação em cada atividade.
- Mapeamento dos fluxos de informação através da DSM.

De uma maneira similar, Yassine (2004), sugere os seguintes passos na construção de uma DSM por atividade:

- Entrevistar engenheiros e gerentes que atuam no processo em estudo.
- Determinar lista de atividades.

- Questionar a respeito das entradas, saídas, tipos de fluxo, força de interação, etc.
- Alimentar a DSM com as atividades e relacionamentos (fluxo de informação).
- Validar a DSM montada com os engenheiros e gerentes que atuam no processo.
- Otimizar a DSM utilizando os algoritmos necessários.

Neste sentido, fica bastante claro que a aplicação da DSM exige cuidados preliminares de maneira a garantir o sucesso de sua aplicação.

Numa aplicação de DSM voltada a otimização do seqüenciamento de atividades de um processo de projeto, o mapeamento do fluxo de informações tem fundamental importância na identificação de ciclos de iteração que possam existir.

Uma metodologia pesquisada e que contempla essas preocupações foi a proposta por Austin et al (1999) em seu trabalho sobre a aplicação de técnicas de planejamento no gerenciamento do projeto de construções prediais. Segundo os autores, este trabalho apresenta uma metodologia que fornece uma abordagem estruturada para a utilização da DSM (tipo atividade). O objetivo desta metodologia é o de analisar os problemas relacionados à natureza iterativa do projeto baseando-se no fluxo de informação entre as atividades ao invés do fluxo de trabalho.

Nesta metodologia, conhecida como ADePT (*Analytical Design Planning Technique*), o caráter interdisciplinar e a natureza iterativa das atividades de um processo são considerados durante a fase de gerenciamento do tempo do projeto.

Esta metodologia é composta das seguintes etapas (Figura 3.1):

- Modelagem do processo de projeto.
- Criação de uma tabela de dependência de informações entre as atividades.
- Aplicação da Matriz de Estrutura de Projetos (DSM).
- Programação do projeto (construção do cronograma).

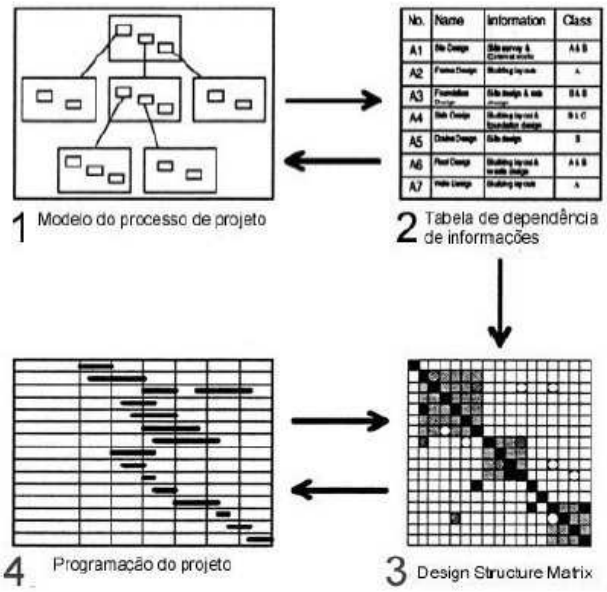


Figura 3.1: Metodologia ADePT (Fonte: Austin et al., 1999)

A primeira etapa, a modelagem do processo de projeto, consiste basicamente em quatro etapas:

- Representação de seus níveis hierárquicos.
- Decomposição das atividades.
- Definição dos requisitos de informação existentes entre elas.
- Representação gráfica do modelo através do diagrama IDEF0.

Segundo Austin et al (1999), a etapa de modelagem do processo deve ser executada e validada por integrantes e gerentes atuantes do projeto de forma a aumentar a precisão do mapeamento do processo e suas atividades.

A segunda etapa da metodologia consiste na construção da uma tabela de dependência de informações que reflita o desdobramento das atividades do processo de projeto levantadas durante a fase anterior.

A terceira etapa do consiste na aplicação da DSM visando a otimização do seqüenciamento de atividades, através do mapeamento do fluxo de informações delineado pela tabela de dependência de informação.

Por fim, a quarta e última etapa da metodologia consiste em utilizar a nova sequência de atividades delineada pela DSM para a elaboração da programação do projeto, utilizando para tal, técnicas especificas para tal fim, como por exemplo, o Diagrama de Gantt.

Um ponto importante a se destacar é que, durante a pesquisa, verificou-se que a metodologia ADePT foi utilizada por diversos autores, porém sempre voltada a aplicações envolvendo o gerenciamento do processo de projetos voltados principalmente para a área de construção civil. Como por exemplo, podemos citar: Austin et al (1999), Peralta (2002), Hoffmeister (2003), Rodriguez (2005), Manzione (2006), entre outros.

Com o intuito de poder direcionar a metodologia desenvolvida por Austin et at (1999) para uma aplicação mais voltada ao gerenciamento do processo de DFX, é proposto um método alternativo que inclui as características deste tipo de processo, levando em conta aspectos e características do ciclo de vida do produto, engenharia simultânea, incorporação de atributos durante as fases de projeto e engenharia de sistemas.

O método proposto visa detalhar a modelagem do processo de projeto visando aumentar o nível de conhecimento do processo e também a precisão do relacionamento entre as atividades de maneira a maximizar os resultados gerados pela DSM.

Assim sendo, este método consite de 2 etapas principais:

- Modelagem do Processo
    - Mapeamento da metodologia de projeto empregada.
    - Definição do modelo de DFX a ser implementado.
    - Descrição dos níveis hierárquicos do processo de DFX.
    - Decomposição hierárquica das atividades do processo.
    - Construção da tabela de dependência de informação.

- Gerenciamento do Processo
    - Aplicação da DSM.
    - Validação dos resultados da análise.
    - Elaboração da programação do processo de projeto.

A figura 3.2 representa as etapas e sub-etapas deste método:

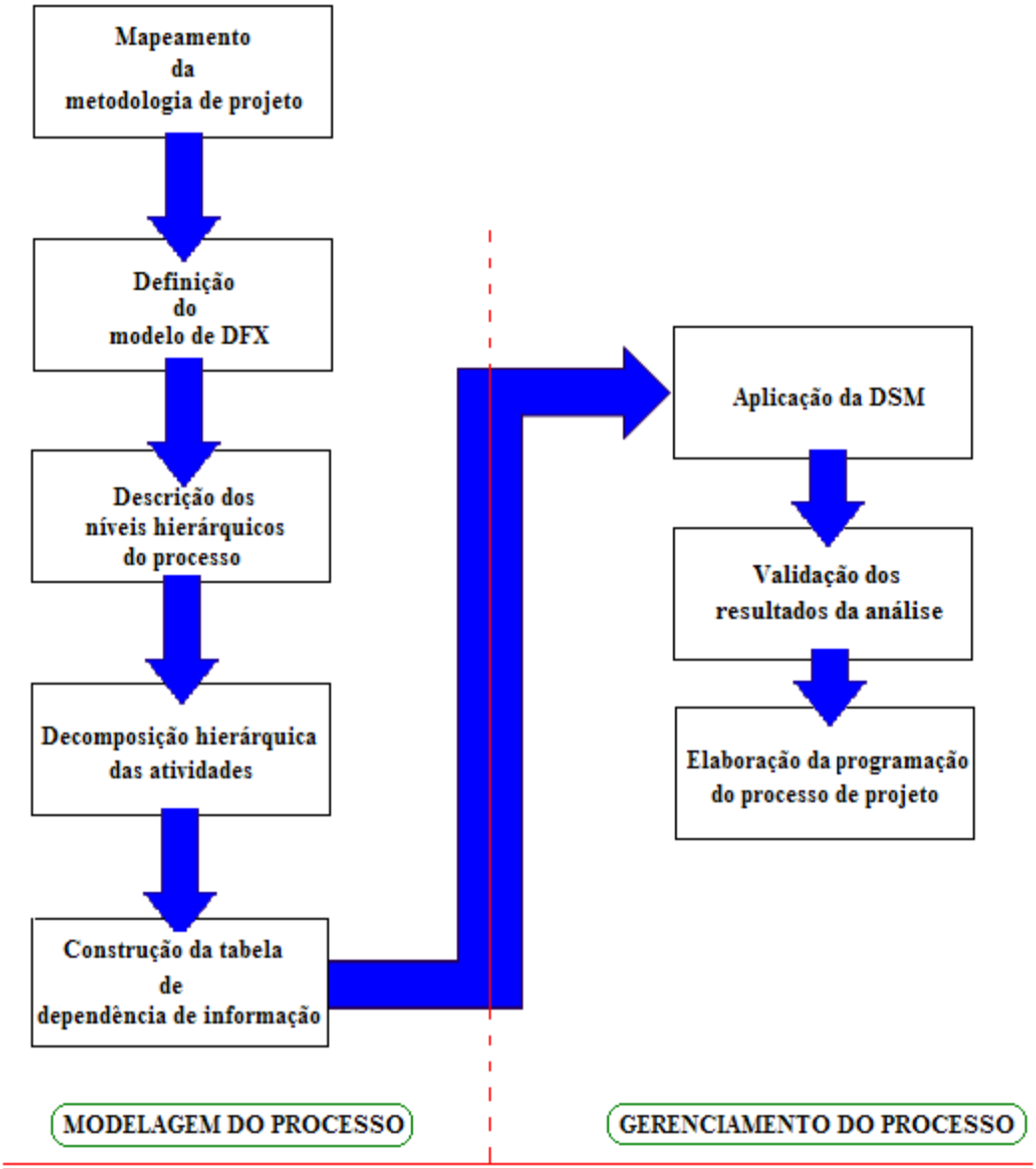


Figura 3.2: Metodologia ADePT aplicada ao DFX.

3.1.1 Modelagem do Processo

Esta etapa, constituída de 5 sub-etapas, é a responsável pelo entendimento do processo de projeto (*design process*) visando o seu gerenciamento. Este entendimento se dá através da representação de um modelo que contemple uma visão geral (alto nível) e uma visão detalhada do processo.

Neste sentido, a modelagem do processo se inicia com o mapeamento da metodologia de projeto empregada e que resultará na sistemática e nas atividades do processo de desenvolvimento do produto, conforme a seguir:

Mapeamento da metodologia de projeto

De uma forma geral, as metodologias de projeto descrevem as macro etapas do processo de projeto, desde a identificação de necessidades até o projeto detalhado do produto.

As metodologias de projeto que foram mencionadas no Capítulo 2 (metodologias de Asimow, Blanchard & Fabrycky e Pahl & Beitz) são alguns exemplos que apresentam as diferentes abordagens e níveis de detalhamento das atividades utilizadas nos processos de projeto.

Definição do modelo de DFX

A segunda etapa da modelagem do processo consiste na definição do modelo de DFX a ser implementado. Conforme descrito no Capitulo 2, num projeto voltado ao ciclo de vida, existem diversos atributos que podem ser maximizados em detrimento do desempenho, aparência e funcionabilidade. Dentre estes atributos podemos citar: confiabilidade, mantenabilidade, segurança, facilidade de manufatura, meio ambiente, entre outros.

Definição dos níveis hierárquicos do processo

Em função das macro atividades da metodologia de projeto mapeada e do modelo de DFX a ser implementado no projeto do produto, são descritos os níveis hierárquicos do processo.

Os níveis hierárquicos de um processo de DFX representam macro atividades e são definidos em função de aspectos técnicos e gerenciais (característicos a cada atributo) que englobam a maneira como os atributos desejáveis de projeto são incorporados e como o processo deve ser gerenciado.

Vale ressaltar que estes aspectos são trabalhados de forma cooperativa em times multi-funcionais desde as fases mais iniciais do projeto. Sendo assim, cada nível hierárquico do processo de DFX será distribuído em cada uma das macro etapas do projeto do produto.

Decomposição hierárquicas das atividades

Após a definição e a alocação de cada nível hierárquico do processo de DFX nas macro etapas de projeto, torna-se necessário a decomposição hierárquica das atividades visando a representação do fluxo de informação entre elas.

Conforme mencionado no Capítulo 2, uma decomposição hierárquica de atividades pode ser realizada através da utilização de uma estrutura analítica de projeto (EAP), onde as macro-atividades são desdobradas em partes controláveis visando a obtenção das atividades que terão seu fluxo de informações mapeado.

Peralta (2002) afirma que o detalhamento deve atingir um nível onde as tarefas ou pacotes de trabalho são identificados e que cada tarefa possa ser planejada, orçada, monitorada e controlada.

Manzione (2006), por sua vez, afirma que os critérios de decomposição podem ser colhidos a partir de reuniões, onde sugestões de representantes dos times multi-funcionais que atuam no processo possam ser colhidas, reforçando assim a importância da participação dos profissionais envolvidos no planejamento das atividades do processo.

Construção da tabela de dependência de informação

Com as atividades do processo definidas, a próxima etapa visa representar o fluxo de informação existente entre cada uma das atividades.

Conforme mencionado no Capítulo 2, existem várias metodologias que são aplicáveis a representação gráfica do fluxo de informação em um projeto, como por exemplo os diagramas IDEF0 e as redes de Petri. Em todos os casos, o objetivo principal é o de poder mapear as informações de entrada e de saída de cada atividade visando a melhoria de um processo.

Uma maneira simples de representar este fluxo é através da construção de uma tabela de dependência de informação que leve em consideração o mapeamento do fluxo de informação entre as atividades do processo.

Conforme apresentado em alguns trabalhos por alguns autores, como Peralta (2002) e Manzione (2006), esta tabela deve contemplar informações que auxiliem a definição dos relacionamentos sequenciais, paralelos e interdependentes na matriz. Dentre estas informações podemos citar a descrição das atividades, as informações requeridas, as atividades de origem e o tempo de execução.

De forma a promover uma maior precisão dos resultados da DSM, é sugerido que durante a construção da tabela de dependência de informação alguns integrantes das áreas envolvidas no processo sejam convocadas. Além disso, uma validação final com os gerentes das áreas também é sugerida para aprimorar ainda mais os resultados (conforme proposto por Yassine, 2004).

Basicamente, uma tabela de dependência de informações tem a seguinte configuração, conforme tabela 3.1:

Tabela 3.1: Tabela de dependência de informações (Fonte: Adaptado de Manzione 2006)

TABELA DE DEPENDENCIA DE INFORMAÇAO					
Atividade		**Informação Requerida**		**Informação Disponibilizada**	
Número	Descrição	Descrição	Atividade de Origem	Descrição	Tempo de Execução

É importante ressaltar que durante a etapa de construção da tabela, possíveis modificações na etapa de decomposição hierárquica das atividades podem vir a ser necessárias, caso alguma atividade que forneça alguma informação relevante a outra não esteja delineada.

Com a definição das atividades (devidamente alinhadas com sua hierarquia) e com o mapeamento do fluxo de informação entre as atividades, através da construção de uma tabela de dependência de informação, é possível concluir a etapa de modelagem do processo e iniciar a etapa de gerenciamento do processo.

3.1.2 Gerenciamento do Processo

A etapa de gerenciamento do processo é constituída de 3 sub-etapas e tem como objetivo a aplicação da DSM no processo de projeto visando a construção de um cronograma que incorpora um sequenciamento das atividades otimizado em função do fluxo de informação. As etapas são:

Aplicação da DSM

A primeira sub-etapa consiste na aplicação da DSM. Conforme mencionado no Capítulo 2, esta etapa pode ser dividida nos seguintes passos:

- Alocação das atividades do processo nas linhas e colunas da matriz.
    - As atividades serão inicialmente dispostas em função das etapas de projeto e dos níveis hierárquicos definidos na etapa de modelagem do processo.
- Identificação dos tipos de relacionamento entre as atividades a partir das marcas de alimentação e retroalimentação na matriz.
- Identificação do tipo de fluxo entre as atividades (conforme Cho, 2001).
- Otimização do seqüenciamento das atividades da matriz a partir da utilização de algoritmos de particionamento.
- Identificação dos blocos acoplados, ou seja contendo ciclos de iteração.
- Avaliação dos resultados obtidos.

Validação dos resultados da análise

Nesta etapa foram validados os resultados fornecidos pela DSM. Esta validação deve ser realizada por membros do time de trabalho que atuam no processo em estudo, de forma a corrigir possíveis incoerências que possam ser mostradas após o termino da análise. Em caso de possíveis correções, a etapa anterior deve ser corrigida e reiniciada.

Elaboração da programação do processo de projeto

Por fim, a segunda e ultima etapa do gerenciamento do processo consiste na elaboração do cronograma do processo a partir da utilização de um diagrama de Gantt. Técnicas como o PERT/CPM, por exemplo, podem ser utilizados previamente ao diagrama de Gantt para refinar ainda mais o planejamento do processo.

Vale ressaltar que durante esta etapa, possíveis revisões no cronograma podem causar a revisão da análise de DSM previamente realizada, sendo assim um processo interativo.

4 Aplicação do Método Proposto

4.1 Introdução

A aplicação do método proposto apresentado neste trabalho foi desenvolvido em uma indústria do ramo aeronáutico situada em São José dos Campos, estado de São Paulo.

O principal objetivo é o de aplicar a matriz de estrutura de projetos (DSM) no gerenciamento do processo responsável pela aplicação dos conceitos e requisitos de segurança no desenvolvimento de uma aeronave comercial (processo conhecido como *Safety Assessment*), visando estabelecer a viabilidade de sua aplicação na prática.

Para tal objetivo, a aplicação do método se baseará na metodologia proposta no Capitulo 3, que tem por base a metodologia ADePT, desenvolvida por Austin et al (1999). A metodologia utilizada levará adicionalmente em conta aspectos do DFX, contando com duas etapas principais: modelagem do processo e o gerenciamento do processo.

Vale ressaltar que o enfoque principal deste método é o de fornecer uma abordagem estruturada para a utilização da DSM visando o gerenciamento do processo a partir do mapeamento do fluxo de informações entre suas atividades. Os resultados obtidos a partir da aplicação do método proposto serão também apresentados neste capítulo.

O processo de *Safety Assessment* é considerado um processo longo e complexo, uma vez que se inicia durante as fases mais preliminares do projeto e termina ao final da certificação de tipo da aeronave, tendo uma duração média aproximadade 60 meses (5 anos).

Além disso, apresenta interdependência entre atividades (ciclos de iteração), que é uma característica comum em processos complexos de desenvolvimento de produto.

Assim sendo, o estudo do processo de *Safety Assessment* se encaixa perfeitamente na proposta desta técnica, uma vez que é composto por diversos relacionamentos de natureza iterativa.

4.2 Modelagem do Processo de Projeto

Entre os meses de Julho e Outubro de 2009, o processo de *Safety Assessment* foi analisado através da análise documental de normas, práticas recomendadas (ARP-4761 e ARP-4754), procedimentos internos e também de entrevistas (com foco na definição e no detalhamento das atividades dentro de cada etapa do processo) efetuadas com engenheiros da área de integração e segurança de sistemas e de desenvolvimento do produto. O objetivo deste estudo foi o de obter subsídios para a modelagem do processo.

A experiência recente de desenvolvimento da engenharia de sistemas aponta para a necessidade do estudo do ciclo completo de vida do sistema, ainda na fase de concepção.

Esta necessidade condiz com o atual modelo de desenvolvimento de aeronaves comerciais, onde diversos fatores de seu ciclo de vida são considerados preliminarmente durante as fases de projeto através de uma abordagem de desenvolvimento integrado do produto ou engenharia simultânea.

A figura 4.1, representa este conceito aplicado ao desenvolvimento de uma aeronave comercial.

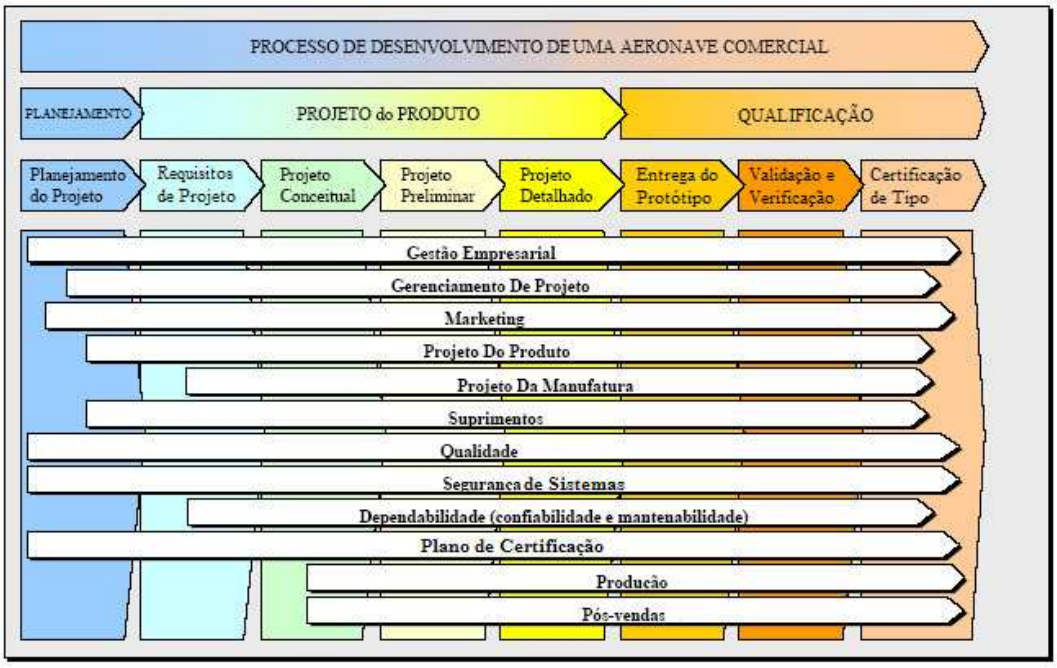


Figura 4.1: Desenvolvimento integrado de uma aeronave comercial (Adaptado de Back, 2007).

Dentre os principais fatores do ciclo de vida considerados na abordagem de desenvolvimento integrado de uma aeronave encontra-se a segurança de sistemas, que é abordada no processo de *Safety Assessment*, e que é fundamental no processo de certificação do produto.

Conforme observação na empresa pesquisada e através do estudo da literatura, a metodologia de projeto a ser tomada com base neste estudo se baseia na sistemática de Blanchard e Fabrycky (1981), que vêem o projeto como uma função no ciclo de vida de um sistema.

Segundo Back (2007), esta metodologia é a que possui a melhor visão global do processo de desenvolvimento de produtos, contendo conceitos de projeto do produto para o consumidor e para o ciclo de vida do produto, próxima da atual visão da engenharia simultânea.

Assim sendo, o projeto não deve se resumir apenas em transformar uma necessidade em uma configuração definitiva de um sistema, mas considerar também as conseqüências futuras em relação a aspectos como produtividade, confiabilidade, manutenibilidade, sistemas de apoio logístico, custos e complexidade de desativação do sistema e o custo total de vida útil do sistema.

Esta visão de engenharia de sistemas (enfatizando o processo de realização do produto sob uma visão centrada no sistema e atrelada a requisitos – conforme modelo V&V) representa a inter-relação entre o processo de desenvolvimento dos sistemas e o processo de *Safety Assessment*.

Como visto na revisão de literatura, a metodologia proposta por Blanchard e Fabrycky (1981) é composta das seguintes macro etapas:

- Projeto Conceitual.
- Projeto Preliminar.
- Projeto Detalhado dos Sistemas / Qualificação do Produto (Validação e Verificação).
- Produção e Entrega.
- Suporte ao Ciclo de Vida dos Sistemas.
- Retirada do Sistema.

Apesar de não fazer parte das etapas propostas pela metodologia de projeto segundo Blanchard e Fabrycky (1981), a qualificação do produto foi considerada de forma a incluir o *loop* de requisitos e de projeto descrito pelo processo que rege a engenharia de sistemas (Figura 2.11).

Como simplificação, são consideradas apenas as etapas de projeto conceitual, projeto preliminar, projeto detalhado e qualificação.

A primeira etapa trata do projeto conceitual, que se concretiza através da aplicação do processo de engenharia de sistemas, mediante a análise funcional e definição de requisitos de projeto no nível de desenvolvimento de aeronave.

A etapa seguinte refere-se ao projeto preliminar, que começa com a análise funcional em nivel de sistema. O projeto preliminar inclui o processo de análise funcional e a locação de requisitos, síntese de sistemas e definição da configuração na forma de especificações detalhadas.

A fase do projeto detalhado inicia com uma configuração derivada das atividades do projeto preliminar visando a descrição detalhada da arquitetura dos sistemas (nível de item).

Em cada nível de desenvolvimento do projeto (avião, sistema ou item) existem validações que são realizadas para assegurar que o projeto e/ou os requisitos estejam corretos e completos para que o processo possa prosseguir ao próximo estágio, sucessivamente até a sua finalização da certificação de tipo.

O diagrama V&V (Figura 4.2) representa as principais etapas de desenvolvimento de uma aeronave, sob a ótica da engenharia de sistemas e, como já apresentado, reflete uma abordagem *top-down* e *bottom-up* do processo. O lado esquerdo do modelo representa a alocação e validação dos requisitos nos níveis de desenvolvimento da aeronave e o lado direito representa as integrações, testes e verificações que irão qualificar o produto.

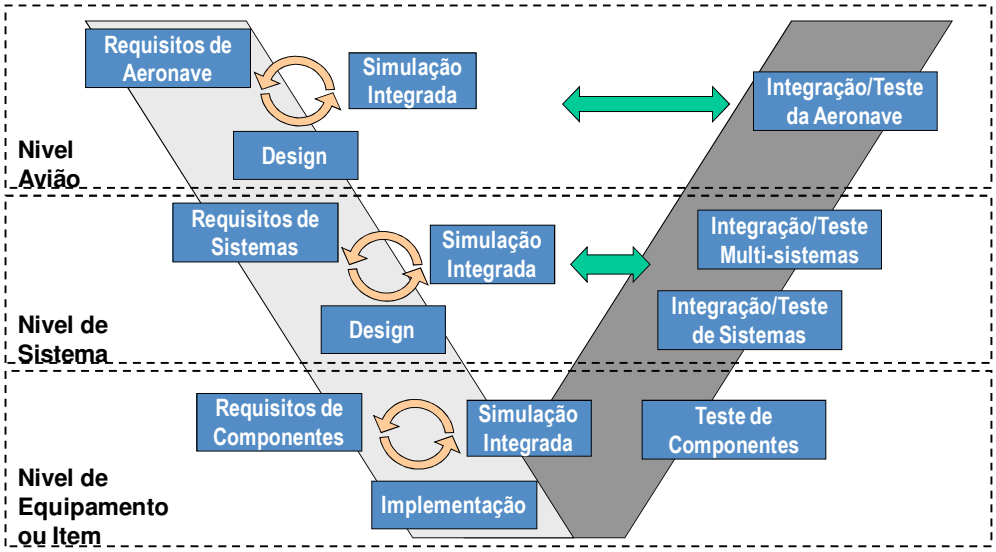


Figura 4.2: O diagrama "V&V" e os níveis de desenvolvimento de uma aeronave.

A simulação integrada, indicada neste diagrama, representa o meio de validação dos efeitos das condições de falha (seja em nivel avião, sistema ou item), através de critérios de performance e controlabilidade definidos pela engenharia de desevolvimento do produto.

Por fim, a última etapa do desenvolvimento integrado de uma aeronave comercial consiste na sua qualificação, ou seja, na validação e verificação. O principal objetivo desta etapa é comparar aquilo que foi produzido (sistema, subsistema, item, etc) com o que foi especificado pelos principais requisitos de certificação da aeronave.

Conforme apresentado no Capitulo 2, o processo de *Safety Assessment* é composto de 3 perspectivas principais: Funcional, Instalativa e Operacional. Para efeito de simplicação , a aplicação do método proposto só considerará as atividades voltadas a perspectiva funcional. Neste caso, os principais níveis hierárquicos, apresentados no Capitulo 2, são: a Análise Funcional, o PSSA e o SSA.

Em cada um deste níveis hierárquicos e em cada um dos níveis de desenvolvimento da aeronave (avião, sistema e item) são utilizadas técnicas de análise de risco voltadas a fundamentar todo o processo de *Design for Safety*. Estas análises são o FHA, FTA (qualitativa e quantitativa) e a FMEA.

A tabela 4.1 apresenta as principais etapas de projeto definidas por Blanchard e Fabrycky (1981). Em cada macro etapa do processo e em cada nível de desenvolvimento são detalhados os produtos resultantes do processo.

Tabela 4.1: Etapas do processo de *Safety Assessment* por etapa de projeto.

Etapas de Projeto segundo Blanchard e Fabrycky e os Produtos do Processo de "Safety Assessment"			
Etapa do Projeto	**Nivel de Desenvolvimento**	**Produto da Etapa**	**Detalhamento**
Projeto Conceitual	Avião	Arquitetura conceitual e requisitos de alto nivel	- Requisitos funcionais. - Requisitos de independência. - Metas de probabilidade dos sistemas. - Definição conceitual da arquitetura dos sistemas. - Considerações iniciais para análise de riscos particulares.
Projeto Preliminar	Sistema	Arquitetura preliminar dos sistemas e requisitos de nivel de sistema	- Requisitos funcionais. - Requisitos de independência. - Metas de probabilidade dos subsistemas. - Niveis de desenvolvimento (DAL's) para SW e HW. - Definição preliminar da arquitetura dos sistemas. - Análises de riscos particulares. - Análises de modo comum.
Projeto Detalhado	Sistema / Item	Arquitetura detalhada dos sistemas e requisitos de nivel de item	- Modos de falha de componentes do sistema. - Taxas de falha dos componentes. - Descrição detalhada da arquitetura dos sistemas. - Especificação detalhada para SW e HW. - Análises de modo comum. - Análise zonal de segurança.
Qualificação	Todos	Integração, testes, qualificação e certificação da aeronave	- Validação e verificação no nivel de item. - Validação e verificação no nivel de sistema. - Validação e verificação no nivel avião. - Emissão do certificado de tipo.

Conforme já mostrado, num projeto complexo, o efetivo gerenciamento do tempo de ciclo exige o desdobramento das macro-atividades em atividades mais facilmente gerenciáveis. Este desdobramento pode ser efetuado através das estruturas analíticas do projeto.

Na aplicação do método proposto foram consideradas as estruturas analíticas de projeto orientadas a partir de cada das macro etapas de projeto conceitual, projeto preliminar, projeto detalhado e qualificação. De forma a incluir todas as etapas de projeto do produto mostradas na figura 4.1 (desenvolvimento integrado da aeronave), é considerada também a etapa referente à definição dos requisitos do projeto (neste caso, os requisitos de certificação da aeronave).

Para a decomposição hierárquica das atividades, foram entrevistados profissionais da área de integração de sistemas, que são os responsáveis por gerir o processo de *Safety Assessment* dentro da empresa. Esta decomposição resultou em uma lista de sessenta (60) atividades, que estão divididas por cada macro etapa de projeto e pelo nível de desenvolvimento, como a seguir:

Definição dos Requisitos de Projeto

Esta etapa se caracteriza pela definição da missão e consequentemente da base de certificação a ser utilizada pela empresa fabricante. No primeiro caso, o objetivo é o de definir o tipo de aeronave a ser desenvolvida em função dos requisitos de mercado (exemplo: aeronave de transporte de passageiros, aeronave de transporte de carga, aeronave militar, etc). Já no segundo caso, o objetivo é o de definir a base de certificação, ou seja, quais os requisitos a serem aplicados em função do tipo de aeronave a ser desenvolvida (Exemplo: FAR 25.1309 para projetos da aviação civil ou a MIL-STD-882D para projeto de aeronaves militares).

Tabela 4.2: Definição de requisitos (principais atividades).

DECOMPOSIÇÃO HIERÁRQUICA DAS ATIVIDADES DO PROCESSO		
Descrição	**Etapa de Projeto**	**Nivel de Desenvolvimento**
Definição da missão da aeronave	Definição de Requisitos	Avião
Definição da base de certificação	Definição de Requisitos	Avião

Projeto Conceitual

Nesta etapa ocorre a primeira visão conceitual da arquitetura da aeronave, onde são definidos, por exemplo, a posição dos motores, o tipo de asa, o tipo de fuselagem, os tipos de tecnologia sistemas a serem empregados (comandos de vôo *fly-by-wire*, barramentos digitais de comunicação entre equipamentos etc), entre outros.

Com estas definições conceituais, são definidas as funções de alto nível da aeronave e são realizadas as primeiras análises de risco, através das técnicas de FHA e FTA (qualitativa). É nesta etapa que também são definidos os primeiros requisitos de segurança em alto nível (funcionais e de independência) a serem desdobrados para os sistemas da aeronave.

Com a definição das funções de alto nível da aeronave, dá-se início efetivamente ao processo de *Safety Assessment*. A figura 4.3 apresenta um exemplo de desdobramento funcional do processo utilizado durante a etapa de projeto conceitual.

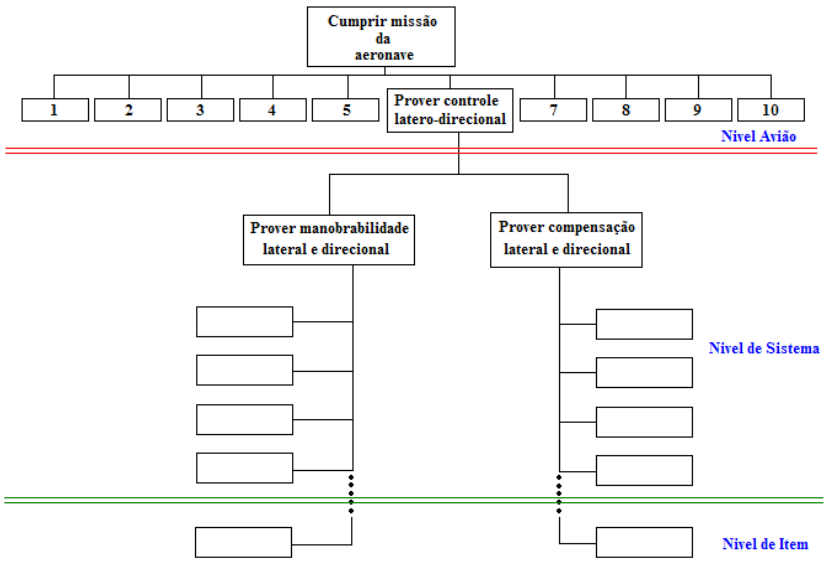


Figura 4.3: Exemplo do desdobramento funcional no processo.

A tabela 4.3 apresenta com um nível maior de detalhamento as atividades que compõem a etapa de projeto conceitual.

Tabela 4.3: Projeto Conceitual (principais atividades).

DECOMPOSIÇÃO HIERÁRQUICA DAS ATIVIDADES DO PROCESSO		
Descrição	**Etapa de Projeto**	**Nivel de Desenvolvimento**
Definição conceitual da arquitetura da aeronave	Projeto Conceitual	Avião
Definição das funções de alto nivel da aeronave	Projeto Conceitual	Avião
Identificação do perfil médio vôo da aeronave	Projeto Conceitual	Avião
Definição das condições de falha de cada função de alto nivel	Projeto Conceitual	Avião
Determinação dos criterios de performance e controlabilidade para a determinação dos efeitos das condições de falha	Projeto Conceitual	Avião
Determinação dos efeitos das condições de falha para a aeronave, tripulação e passageiros	Projeto Conceitual	Avião
Validação dos efeitos das condições de falha atraves de simulação integrada	Projeto Conceitual	Avião
Atribuição do nivel de severidade das condições de falha.	Projeto Conceitual	Avião
Identificação dos requisitos de segurança no nivel avião.	Projeto Conceitual	Avião
Validação dos requisitos de segurança	Projeto Conceitual	Avião
Documentação dos resultados do AFHA	Projeto Conceitual	Avião
Descrição preliminar do pacote de sistemas	Projeto Conceitual	Avião
Correlação das funções de alto nivel da aeronave com os sistemas contribuintes ("matriz funções de alto nivel x funções sistemas")	Projeto Conceitual	Avião
Correlação da perda ou a degradação de uma função de alto nivel com os sistemas contribuintes ("matriz de combinação critica")	Projeto Conceitual	Avião
Definição das falhas combinadas que podem levar a perda ou degradação das funções de alto nivel	Projeto Conceitual	Avião
Construção das árvores de falha qualitativas para cada condição de falha (top event)	Projeto Conceitual	Avião
Definição dos requisitos de independência (inter-sistemas)	Projeto Conceitual	Avião
Validação dos requisitos de independência (inter-sistemas)	Projeto Conceitual	Avião

Projeto Preliminar

Nesta etapa, os esforços do *Safety Assessment* se concentram no nível dos sistemas através da aplicação de técnicas de análise de risco como o FHA e a FTA (qualitativo), visando a identificação os requisitos de segurança para cada sistema em estudo. Após definidos, tanto a arquitetura preliminar, quanto os requisitos de segurança dos sistemas são confrontados de forma a validar se a arquitetura proposta atende a estes requisitos de segurança. Caso não haja compatibilidade, a arquitetura deve ser revisitada. Outro ponto importante desta etapa consiste na definição dos níveis de desenvolvimento (DALs) para *software* e *hardware* embarcado a serem utilizados pelos sistemas. A tabela 4.4 apresenta as atividades do projeto preliminar.

Tabela 4.4: Projeto Preliminar (principais atividades).

DECOMPOSIÇÃO HIERÁRQUICA DAS ATIVIDADES DO PROCESSO		
Descrição	**Etapa de Projeto**	**Nivel de Desenvolvimento**
Descrição preliminar da arquitetura dos sistemas (sub-sistemas)	Projeto Preliminar	Sistema
Definição das funções de sistema	Projeto Preliminar	Sistema
Definição das condições de falha para cada função de nivel de sistema	Projeto Preliminar	Sistema
Identificação das condições ambientais agravantes.	Projeto Preliminar	Sistema
Determinação dos criterios de performance, controlabilidade, qualidade de voo e fatores humanos para a determinação dos efeitos das condições de falha	Projeto Preliminar	Sistema
Determinação dos efeitos das condições de falha para a aeronave, tripulação e passageiros (nivel de sistema)	Projeto Preliminar	Sistema
Validação dos efeitos das condições de falha atraves de simulação integrada (nivel de sistema)	Projeto Preliminar	Sistema
Atribuição do nivel de severidade das condições de falha (nivel de sistema)	Projeto Preliminar	Sistema
Identificação dos requisitos de segurança no nivel de sistema	Projeto Preliminar	Sistema
Validação dos requisitos de segurança no nivel de sistema	Projeto Preliminar	Sistema
Apresentação do SFHA para as autoridades certificadoras	Projeto Preliminar	Sistema
Concordância da autoridade com relação aos resultados do SFHA	Projeto Preliminar	Sistema
Documentação dos resultados do SFHA	Projeto Preliminar	Sistema
Definição preliminar da arquitetura dos sistemas (itens)	Projeto Preliminar	Sistema
Seleção das condições de falha com nivel de severidade "catastrophic", "hazardous" e "major" definidos no SFHA	Projeto Preliminar	Sistema
Construção das árvores de falha qualitativas para cada condição de falha (top event)	Projeto Preliminar	Sistema
Definição das metas de probabilidade dos sub-sistemas	Projeto Preliminar	Sistema
Validação se a arquitetura proposta atende aos requisitos de seguranca	Projeto Preliminar	Sistema
Definição das funções de suporte ou interface	Projeto Preliminar	Sistema
Definição dos requisitos de independência (intra-sistemas)	Projeto Preliminar	Sistema
Atribuição dos "Development Assurance Levels" (DAL) - Definição das condições de falha associadas a "software" ou "hardware eletronico programavel". - Definição da lista de equipamentos que contem as funções lógicas associadas. - Definição da lista de funções desempenhadas pelos equipamentos associados.	Projeto Preliminar	Sistema

Projeto Detalhado

Nesta etapa, os esforços do processo de *Safety Assessment* se concentram no nível dos itens de sistema (equipamentos) através da aplicação de técnicas como o FMEA e FTA (quantitativa). Em função dos requisitos de segurança e dos níveis de desenvolvimento definidos nos níveis superiores, a arquitetura dos sistemas é finalmente detalhada e congelada (tabela 4.5).

Tabela 4.5: Projeto Detalhado (principais atividades).

DECOMPOSIÇÃO HIERÁRQUICA DAS ATIVIDADES DO PROCESSO		
Descrição	**Etapa de Projeto**	**Nivel de Desenvolvimento**
Definição detalhada da arquitetura dos sistemas (congelamento)	Projeto Detalhado	Item
Elaboração das FMEAs de nivel de sistema (Definição dos modos de falha e dos efeitos associados aos itens do sistema, Teste e analise em nivel de item e definição das taxas de falha para cada modo de falha associado).	Projeto Detalhado	Item
Construção das árvores de falha quantitativas para cada condição de falha catastrophic e hazardous (top event)	Projeto Detalhado	Item
Definição das tarefas de manutenção *safety related* (CCMRs) e seus intervalos correspondentes	Projeto Detalhado	Item
Considerações para a preparação da lista de equipamentos mínimos para despacho (relatório de flexibilidade operacional)	Projeto Detalhado	Item
Considerações para a preparação do manual de vôo da aeronave (AFM)	Projeto Detalhado	Item

Qualificação do Produto

A qualificação do produto consiste na validação e verificação dos requisitos de segurança definidos nos níveis de desenvolvimento. Enquanto as validações ocorrem ao final de cada nível (através de reuniões e de tarefas de *check-list*), a verificação ocorre dos níveis mais inferiores aos níveis mais superiores do desenvolvimento do produto (item, sistema, multi-sistemas e aeronave) através de integração e testes (simulador de vôo, bancadas de ensaio, protótipo, etc).

Ao final de todos os testes é verificado se o projeto da aeronave atende aos requisitos definidos pela base de certificação e, caso aprovado pelas autoridades, o projeto da aeronave está qualificado e é emitida a certificação de tipo. A tabela 4.6 apresenta as atividades.

Tabela 4.6: Qualificação do Produto (principais atividades).

DECOMPOSIÇÃO HIERÁRQUICA DAS ATIVIDADES DO PROCESSO		
Descrição	**Etapa de Projeto**	**Nivel de Desenvolvimento**
Verificação das metas de probabilidade definidas no PSSA	Qualificação	Item
Integração e teste no nivel de sistemas	Qualificação	Sistema
Confirmação da arquitetura dos sistemas	Qualificação	Sistema
Validação da severidade das condições de falha definidas no SFHA	Qualificação	Sistema
Integração e teste multi-sistemas	Qualificação	Sistema
Confirmação da arquitetura final multi-sistemas	Qualificação	Sistema
Integração e teste no nivel de aeronave	Qualificação	Avião
Confirmação da arquitetura final dos sistemas na aeronave	Qualificação	Avião
Documentação dos resultados do SSA	Qualificação	Avião
Verificação dos objetivos de segurança definidos no AFHA	Qualificação	Avião
Confirmação que o projeto atende ao requisito RBAC 25.1309	Qualificação	Avião
Apresentação dos relatórios de "Safety Assessment" para as autoridades certificadoras (AFHA, SFHA, PSSA e SSA).	Qualificação	Avião
Emissão do certificado de tipo pela autoridade certificadora	Qualificação	Aviao

A última etapa da modelagem do processo de *Safety Assessment* consistiu na representação do fluxo de informação entre as atividades decompostas.

De forma a simplificar o trabalho, o fluxo de informação não foi representado graficamente. Neste caso, as entradas e as saídas para cada atividade bem como as atividades responsáveis por enviar e receber as informações foram identificadas através de uma tabela de dependência de informação.

Conforme sugerido por Manzione (2006), a tabela de dependência de informação deve conter as atividades do processo, as informações disponibilizadas e requeridas por cada atividade e a origem destas informações. Foram incluídas também as macro etapas de projeto, os níveis de desenvolvimento e o tempo total médio estimado de execução de cada atividade em dias. As tabelas 4.7, 4.8, 4.9, 4.10, 4.11 e 4.12 apresentam as dependências de informação.

Para o levantamento dos relacionamentos entre as atividades foram realizadas reuniões com membros das áreas de integração de sistemas e de desenvolvimento do produto da empresa visando o correto mapeamento do fluxo de informações. Vale ressaltar que documentações e apresentações internas também foram consultadas para a execução desta atividade.

Tabela 4.7: Tabela de dependência de informação (atividades de 1 a 10).

TABELA DE DEPENDENCIA DE INFORMAÇAO						
Atividade				**Informação Requerida**	**Informação Disponibilizada**	
Número	**Descrição**	**Etapa de Projeto**	**Nivel de Desenvolvimento**	**Atividade de Origem**	**Breve Descrição**	**Tempo Médio de Execução (dias)**
1	Definição da missão da aeronave	Definição de Requisitos	Avião	N/A	Tipo de aeronave a ser desenvolvida em função dos requisitos de mercado	7
2	Definição da base de certificação	Definição de Requisitos	Avião	1	Requisitos de certificação (em função do tipo de aeronave a ser desenvolvida)	1
3	Definição conceitual da arquitetura da aeronave	Projeto Conceitual	Avião	1	Definições básicas da arquitetura da aeronave (tipo e posição dos motores, tipo de asa, fuselagem, sistemas, etc)	60
4	Definição das funções de alto nivel da aeronave	Projeto Conceitual	Avião	1, 3	Funções de alto nivel	7
5	Identificação do perfil médio vôo da aeronave	Projeto Conceitual	Avião	1	Perfil médio de vôo (fases de vôo e os tempos em cada fase).	5
6	Definição das condições de falha de cada função de alto nivel	Projeto Conceitual	Avião	4	Condições de falha das funções de alto nivel	7
7	Determinação dos criterios de performance e controlabilidade para a determinação dos efeitos das condições de falha	Projeto Conceitual	Avião	3, 5	Velocidade de ultrapassagem de final de pista, desempenho na decolagem monomotor, caracteristica de vento lateral durante o pouso, nivel de assimetria nos comandos de vôo, etc.	30
8	Determinação dos efeitos das condições de falha para a aeronave, tripulação e passageiros	Projeto Conceitual	Avião	5, 6, 7, 9	Efeitos das condições de falha na aeronave, tripulação e passageiros.	20
9	Validação dos efeitos das condições de falha atraves de simulação integrada	Projeto Conceitual	Avião	7, 8	Resultados da simulação do comportamento da aeronave em função das condições de falha	10
10	Atribuição do nivel de severidade das condições de falha.	Projeto Conceitual	Avião	2, 8, 9, 12	Nivel de severidade ("catastrophic", "hazardous", "major", "minor", "no safety effect")	7

Tabela 4.8: Tabela de dependência de informação – continuação (atividades de 11 a 20).

TABELA DE DEPENDENCIA DE INFORMAÇAO						
Atividade				Informação Requerida	Informação Disponibilizada	
Número	Descrição	Etapa de Projeto	Nivel de Desenvolvimento	Atividade de Origem	Breve Descrição	Tempo Médio de Execução (dias)
11	Identificação dos requisitos de segurança no nivel avião.	Projeto Conceitual	Avião	10	Sumarização dos requisitos para cada funcao de alto nivel em função das severidades definidas.	7
12	Validação dos requisitos de segurança	Projeto Conceitual	Avião	11	Requisitos estão suficientemente corretos e completos (reuniões e check-list)	7
13	Documentação dos resultados do AFHA	Projeto Conceitual	Avião	4, 5, 6, 8, 10, 12	Relatorio final do AFHA	20
14	Descrição preliminar do pacote de sistemas	Projeto Conceitual	Avião	3	Definição dos pacotes dos grandes sistemas (elétrico, hidráulico, comandos de vôo, etc).	60
15	Correlação das funções de alto nivel da aeronave com os sistemas contribuintes ("matriz funções de alto nivel x funções sistemas")	Projeto Conceitual	Avião	4, 14	Matriz funções de alto nivel x funções de sistemas	15
16	Correlação da perda ou a degradação de uma função de alto nivel com os sistemas contribuintes ("matriz de combinação critica")	Projeto Conceitual	Avião	13, 15	Matriz de combinação critica	15
17	Definição das falhas combinadas que podem levar a perda ou degradação das funções de alto nivel	Projeto Conceitual	Avião	16	Falhas combinadas que levam a perda ou degradação das funções de alto nivel	15
18	Construção das árvores de falha qualitativas para cada condição de falha (top event)	Projeto Conceitual	Avião	4, 17	Arvores de falha qualitativas	30
19	Definição dos requisitos de independência (inter-sistemas)	Projeto Conceitual	Avião	18, 20	Requisitos de independência (inter-sistemas)	60
20	Validação dos requisitos de independência (inter-sistemas)	Projeto Conceitual	Avião	14, 19	Apresentação dos requisitos de independência para a engenharia de desenvolvimento do produto	15

Tabela 4.9: Tabela de dependência de informação – continuação (atividades de 21 a 30).

TABELA DE DEPENDENCIA DE INFORMAÇAO						
Atividade				Informação Requerida	Informação Disponibilizada	
Número	Descrição	Etapa de Projeto	Nivel de Desenvolvimento	Atividade de Origem	Breve Descrição	Tempo Médio de Execução (dias)
21	Descrição preliminar da arquitetura dos sistemas (sub-sistemas)	Projeto Preliminar	Sistema	14, 20	Definição preliminar dos sub-sistemas.	30
22	Definição das funções de sistema	Projeto Preliminar	Sistema	12, 20, 21, 32,39	Definição das funções de cada sistema	30
23	Definição das condições de falha para cada função de nivel de sistema	Projeto Preliminar	Sistema	22	Definição das condições de falha para cada função de sistema	30
24	Identificação das condições ambientais agravantes.	Projeto Preliminar	Sistema	2	Condições agravantes (pista contaminada, ventos de travez, vôo noturno, granizo, gelo, chuva, etc)	5
25	Determinação dos criterios de performance, controlabilidade, qualidade de voo e fatores humanos para a determinação dos efeitos das condições de falha	Projeto Preliminar	Sistema	5, 21	Considerações de fatores humanos, velocidade de ultrapassagem de final de pista, desempenho na decolagem monomotor, caracteristica de vento lateral durante o pouso, nivel de assimetria nos comandos de vôo, etc.	21
26	Determinação dos efeitos das condições de falha para a aeronave, tripulação e passageiros (nivel de sistema)	Projeto Preliminar	Sistema	5, 23, 24, 25, 27, 32	Efeitos das condições de falha na aeronave, tripulação e passageiros.	45
27	Validação dos efeitos das condições de falha atraves de simulação integrada (nivel de sistema)	Projeto Preliminar	Sistema	21, 26	Resultados da simulação do comportamento dos sistemas em função das condições de falha.	30
28	Atribuição do nivel de severidade das condições de falha (nivel de sistema)	Projeto Preliminar	Sistema	2, 26, 27, 30, 32, 51	Nivel de severidade ("catastrophic", "hazardous", "major", "minor", "no safety effect")	30
29	Identificação dos requisitos de segurança no nivel de sistema	Projeto Preliminar	Sistema	28	Sumarização dos requisitos para cada funcao em nivel de sistema em função das severidades definidas.	7
30	Validação dos requisitos de segurança no nivel de sistema	Projeto Preliminar	Sistema	29	Requisitos estão suficientemente corretos e completos (reuniões e check-list)	10

Tabela 4.10: Tabela de dependência de informação – continuação (atividades de 31 a 40).

TABELA DE DEPENDENCIA DE INFORMAÇAO						
Atividade				Informação Requerida	Informação Disponibilizada	
Número	Descrição	Etapa de Projeto	Nivel de Desenvolvimento	Atividade de Origem	Breve Descrição	Tempo Médio de Execução (dias)
31	Apresentação do SFHA para as autoridades certificadoras	Projeto Preliminar	Sistema	22, 23, 26, 28, 29	Resultados do SFHA	40
32	Concordância da autoridade com relação aos resultados do SFHA	Projeto Preliminar	Sistema	31	Aceitacao ou nao dos resultados do SFHA	20
33	Documentação dos resultados do SFHA	Projeto Preliminar	Sistema	22, 23, 24, 26, 28, 29, 32	Documentação final do SFHA	35
34	Definição preliminar da arquitetura dos sistemas (itens)	Projeto Preliminar	Sistema	21, 38, 48, 50, 53, 55	Definição preliminar da arquitetura dos sistemas no nivel de item baseado em decisoes de projeto (rationale).	40
35	Seleção das condições de falha com nivel de severidade "catastrophic", "hazardous" e "major" definidos no SFHA	Projeto Preliminar	Sistema	33	Condições de falha "catastrophic", "hazardous" e "major".	5
36	Construção das árvores de falha qualitativas para cada condição de falha (top event)	Projeto Preliminar	Sistema	34, 35	Árvores de falha qualitativas.	45
37	Definição das metas de probabilidade dos sub-sistemas	Projeto Preliminar	Sistema	33, 36	Valores quantitativos de probabilidade de falha dos sistemas	15
38	Validação se a arquitetura proposta atende aos requisitos de seguranca	Projeto Preliminar	Sistema	34, 37	Concordância da arquitetura proposta (atende ou não atende aos requisitos de seguranca)	10
39	Definição das funções de suporte ou interface	Projeto Preliminar	Sistema	34	Funções relacionadas a fontes de potência, refrigeração de equipamentos eletrônicos, barramento de dados, etc.	25
40	Definição dos requisitos de independência (intra-sistemas)	Projeto Preliminar	Sistema	36, 38	Requisitos de independência (intra-sistemas)	60

Tabela 4.11: Tabela de dependência de informação – continuação (atividades de 41 a 50).

TABELA DE DEPENDENCIA DE INFORMAÇAO						
Atividade				Informação Requerida	Informação Disponibilizada	
Número	Descrição	Etapa de Projeto	Nivel de Desenvolvimento	Atividade de Origem	Breve Descrição	Tempo Médio de Execução (dias)
41	Atribuição dos "Development Assurance Levels" (DAL) - Definição das condições de falha associadas a "software" ou "hardware eletronico programavel". - Definição da lista de equipamentos que contem as funções lógicas associadas. - Definição da lista de funções desempenhadas pelos equipamentos associados.	Projeto Preliminar	Sistema	33, 38, 39	Niveis de desenvolvimento para "software" e "hardware eletrônico programável" (A, B, C, D, E). - Condições de falha associadas a "software" e "hardware eletrônico programável". - Modulos de controle, circuitos de sinalização, sistemas monitores, unidades de aquisição de dados, etc. - Lista de funções desempenhadas pelos equipamentos contendo funções lógicas.	45
42	Definição detalhada da arquitetura dos sistemas (congelamento)	Projeto Detalhado	Item	38, 40, 41	Arquitetura detalhada dos sistemas (apresentação com os resultados das árvores de falha e dos FHAs)	15
43	Elaboração das FMEAs de nivel de sistema (Definição dos modos de falha e dos efeitos associados aos itens do sistema, Teste e analise em nivel de item e definição das taxas de falha para cada modo de falha associado).	Projeto Detalhado	Item	42, 48	Modos de falha e efeitos de cada item do sistema relacionados às condições de falha "catastrophic", "hazardous" e "major" // Confirmação dos modos de falha e definição das taxas de falha através de ensaios e análises especificos (ensaios acelerados, MIL-HDBK-217, experiência em campo, etc) // Taxas de falha referentes a cada modo de falha dos itens.	90
44	Construção das árvores de falha quantitativas para cada condição de falha catastrophic e hazardous (top event)	Projeto Detalhado	Item	35, 42, 43	Árvores de falha quantitativas.	120
45	Definição das tarefas de manutenção "safety related" (CCMRs) e seus intervalos correspondentes	Projeto Detalhado	Item	44	Tarefas de manutenção relacionadas às dormências e seus respectivos intervalos de inspeção.	30
46	Considerações para a preparação da lista de equipamentos mínimos para despacho (relatório de flexibilidade operacional)	Projeto Detalhado	Item	44	árvores de falha quantitativas potenciais para definição da lista de quipamentos mínimos para despacho da aeronave.	20
47	Considerações para a preparação do manual de vôo da aeronave (AFM)	Projeto Detalhado	Item	44	Condições especiais para a elaboração de procedimentos da tripulação no caso de falhas de sistemas / equipamentos em vôo.	20
48	Verificação das metas de probabilidade definidas no PSSA	Qualificação	Item	37, 44	Confirmação se os valores calculados pelas árvores de falhas atendem às metas quantitativas de probabilidade definidas no PSSA.	10
49	Integração e teste no nivel de sistemas	Qualificação	Sistema	42	Resultados de testes dos sistemas em RIG ou em protótipo.	60
50	Confirmação da arquitetura dos sistemas	Qualificação	Sistema	49	Confirmação da arquitetura dos sistemas.	15

Tabela 4.12: Tabela de dependência de informação – continuação (atividades de 51 a 60).

TABELA DE DEPENDENCIA DE INFORMAÇAO						
Atividade				Informação Requerida	Informação Disponibilizada	
Número	Descrição	Etapa de Projeto	Nivel de Desenvolvimento	Atividade de Origem	Breve Descrição	Tempo Médio de Execução (dias)
51	Validação da severidade das condições de falha definidas no SFHA	Qualificação	Sistema	32, 50	Confirmação ou não das criticalidades definidas no SFHA através dos veículos de validação.	25
52	Integração e teste multi-sistemas	Qualificação	Sistema	42, 50, 51	Resultados de testes dos sistemas em RIG ou em protótipo.	60
53	Confirmação da arquitetura final multi-sistemas	Qualificação	Sistema	52	Confirmação da arquitetura final multi-sistemas.	15
54	Integração e teste no nivel de aeronave	Qualificação	Avião	42, 53	Resultados de testes da aeronave em vôos de desempenho operacional, ensaios de propagação de falhas, "cold soak", "hot soak", etc.	200
55	Confirmação da arquitetura final dos sistemas na aeronave	Qualificação	Avião	54	Confirmação da arquitetura final dos sistemas da aeronave	30
56	Documentação dos resultados do SSA	Qualificação	Avião	20, 33, 40, 41, 44, 45, 46, 47	Relatório do SSA	30
57	Verificação dos objetivos de segurança definidos no AFHA	Qualificação	Avião	56	Confirmação que os objetivos de segurança são atingidos no nível de aeronave.	60
58	Confirmação que o projeto atende ao requisito RBAC 25.1309	Qualificação	Avião	2, 56, 57	Confirmação que o projeto atende ao requisito RAC 25.1309.	10
59	Apresentação dos relatórios de "Safety Assessment" para as autoridades certificadoras (AFHA, SFHA, PSSA e SSA).	Qualificação	Avião	13, 33, 40, 41, 50, 53, 55, 56, 57, 58	Resultados e considerações finais do "Safety Assessment".	60
60	Emissão do certificado de tipo pela autoridade certificadora	Qualificação	Aviao	58, 59	Certificado de tipo da aeronave	20

4.3 Gerenciamento do Processo de Projeto

4.3.1 Aplicação da DSM

A próxima etapa da aplicação do método proposto consiste na aplicação da DSM visando a otimização do fluxo de informação através do resequenciamento das atividades do processo.

A DSM utilizada foi categorizada como sendo temporal e baseada em atividade, segundo (Browning, 2001), uma vez que o seu objeto de estudo é focado no resequenciamento de atividades visando a redução do tempo de ciclo.

Para este estudo foi utilizado a ferramenta computacional "DSM@MIT", proposta por Soo-Haeng Cho e Steven D. Eppinger, através do trabalho entitulado: "*An Integrated Method for Managing Complex Engineering Projects Using the Design Structure Matrix and Advanced Simulation*", desenvolvido em 2001.

Esta ferramenta consiste de um suplemento do *software* MS-Excel (macro) que também possui interface com o *software* MS-Project. Ela está disponibilizada gratuitamente através do endereço eletrônico "www.dsmweb.org" e foi acessada no mês de Janeiro de 2009. Cho (2001), Peralta (2002), Manzione (2006) e Pierone e Naveiro (2006) foram alguns dos autores pesquisados que utilizaram a ferramenta.

O primeiro passo na aplicação da DSM consiste em listar as atividades nas linhas da matriz com sua ordem sendo também refletida nas colunas.

Foram listadas as atividades do processo relacionadas às etapas de definição de requisitos (de 1 a 2), projeto conceitual (3 a 20), projeto preliminar (21 a 41), projeto detalhado (42 a 47) e qualificação (48 a 60).

A ordem nas quais as atividades foram listadas se baseou basicamente no seqüenciamento proposto pela ARP-4761 (Figura 2.15) e também por aquilo que é aplicado na prática na empresa estudada.

O segundo passo consiste em marcar as dependências entre as atividades na DSM de maneira a mapear o fluxo de informações. Para tal, foram utilizadas as informações contidas nas tabelas de dependência de informação. Além disso, foram considerados os tipos de fluxo que caracterizam o relacionamento entre cada uma das atividades (conforme Cho, 2001).

Neste sentido, o tipo "1" nas células da matriz indica a necessidade da conclusão da atividade anterior para o inicio da atividade posterior.

Já o tipo "2", indica que uma atividade posterior pode ser iniciada sem que a atividade anterior esteja complemante finalizada.

A figura 4.4, representa a entrada de dados da DSM, contendo o seqüenciamento de atividades antes de seu particionamento através do algoritmo de Cho (2001).

4.3.2 Validação dos Resultados da Análise

Nesta etapa foram validados os resultados apresentados pela DSM através de duas reuniões: a primeira realizada com membros que atuam diretamente no processo de *Safety Assessment* e a outra com o gerente da area de integração de sistemas.

Os resultados apresentados nas tabelas 4.5, 4.6, 4.7, 4.8 e 4.9 se mostraram coerentes com aquilo que era esperado para a melhoria do proceso e não foi necessário efetuar nenhuma revisão na etapa de aplicação da DSM e nem mesmo nas etapas preliminares de gerenciamento do processo.

Assim sendo, os resultados foram validados e aprovados para a construção do cronograma proposto para o processo.

Figura 4.4: DSM com a entrada de dados antes de seu particionamento.

Project Name			<NOTE>			1. Enter the name of a project in cell '2A'.																																																							
Processo de "Safety Assessment"						2. Enter all task names in column 'A'.																																																							
Task Name		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
Definição da missão da aeronave	1																																																												
Definição da base de certificação	2	1																																																											
Definição conceitual da arquitetura da aeronave	3	1																																																											
Definição das funções de alto nivel da aeronave	4	1		1																																																									
Identificação do perfil médio vôo da aeronave	5	1																																																											
Definição das condições de falha de cada função de alto nivel	6				2																																																								
Determinação dos criterios de performance e controlabilidade para a determinação dos	7			1		1																																																							
Determinação dos efeitos das condições de falha para a aeronave, tripulação e passag	8					1	2	1		2																																																			
Validação dos efeitos das condições de falha atraves de simulação integrada	9							1	2																																																				
Atribuição do nivel de severidade das condições de falha.	10		1						1	2			2																																																
Identificação dos requisitos de segurança no nivel avião.	11										2																																																		
Validação dos requisitos de segurança	12											2																																																	
Documentação dos resultados do AFHA	13				1	1	1		1		1		1																																																
Descrição preliminar do pacote de sistemas	14			1																																																									
Correlação das funções de alto nivel da aeronave com os sistemas contribuintes ("matr	15				1										1																																														
Correlação da perda ou a degradação de uma função de alto nivel com os sistemas cor	16													1		1																																											
Definição das falhas combinadas que podem levar a perda ou degradação das funções	17																1																																												
Construção das árvores de falha qualitativas para cada condição de falha (top event)	18				1													1																																											
Definição dos requisitos de independência (inter-sistemas)	19																		2		2																																								
Validação dos requisitos de independência (inter-sistemas)	20														1					2																																									
Descrição preliminar da arquitetura dos sistemas	21														1						1																																								
Definição das funções de sistema	22												1								1	2											1							1																					
Definição das condições de falha para cada função de nivel de sistema	23																						2																																						
Identificação das condições ambientais agravantes.	24		1																																																										
Determinação dos criterios de performance, controlabilidade, qualidade de voo e fatores	25					1																1																																							
Determinação dos efeitos das condições de falha para a aeronave, tripulação e passag	26					1																		2	1	1		2																																	
Validação dos efeitos das condições de falha atraves de simulação integrada (nivel de	27																					1					2																																		
Atribuição do nivel de severidade das condições de falha (nivel de sistema)	28		1																								2	2			2		1																			1									
Identificação dos requisitos de segurança no nivel de sistema	29																												2																																
Validação dos requisitos de segurança no nivel de sistema	30																													2																															
Apresentação do SFHA para as autoridades certificadoras	31																						1	1			1		1	1																															
Concordância da autoridade com relação aos resultados do SFHA	32																															1																													
Documentação dos resultados do SFHA	33																						1	1	1		1		1	1			1																												
Definição preliminar da arquitetura dos sistemas	34																					2																	1										1		1			1		1					
Seleção das condições de falha com nivel de severidade "catastrophic", "hazardous" e	35																																	1																											
Construção das árvores de falha qualitativas para cada condição de falha (top event)	36																																		2	1																									
Definição das metas de probabilidade dos sub-sistemas	37																																	1			2																								
Validação se a arquitetura proposta atende aos requisitos de seguranca	38																																		2			2																							
Definição das funções de suporte ou interface	39																																		1																										
Definição dos requisitos de independência (intra-sistemas)	40																																				1		1																						
- Definição das condições de falha associadas a "softw are" ou "hardw are eletronico	41																																	1					1	1																					
Definição detalhada da arquitetura dos sistemas (congelamento)	42																																						1		1	1																			
Elaboração das FMEAs de nivel de sistema (Definição dos modos de falha e dos efeitos	43																																										2						1												
Construção das árvores de falha quantitativas para cada condição de falha catastroph	44																																			1							2	2																	
Definição das tarefas de manutenção "safety related" (CCMRs) e seus intervalos corre	45																																												2																
Considerações para a preparação da lista de equipamentos mínimos para despacho (re	46																																												2																
Considerações para a preparação do manual de vôo da aeronave (AFM)	47																																												2																
Verificação das metas de probabilidade definidas no PSSA	48																																					1							2																
Integração e teste no nivel de sistemas	49																																										2																		
Confirmação da arquitetura dos sistemas	50																																																	1											
Validação da severidade das condições de falha definidas no SFHA	51																																1																		1										
Integração e teste multi-sistemas	52																																										1								1	1									
Confirmação da arquitetura final multi-sistemas	53																																																				1								
Integração e teste no nivel de aeronave	54																																										1											1							
Confirmação da arquitetura final dos sistemas na aeronave	55																																																						1						
Documentação dos resultados do SSA	56																				1													1							1	1			1	1	1	1													
Verificação dos objetivos de segurança definidos no AFHA	57																																																								1				
Confirmação que o projeto atende ao requisito RBAC 25.1309	58		1																																																						1	1			
Apresentação dos relatórios de "Safety Assessment" para as autoridades certificadora	59													1																				1							1	1									1			1		1	1	1	1		
Emissão do certificado de tipo pela autoridade certificadora	60																																																										1	1	

Marcas abaixo da diagonal principal (marcas de alimentação) representam uma informação transferida de uma atividade anterior para outra posterior. Por outro lado, marcas acima da diagonal principal representam uma informação transferida de uma atividade posterior para uma atividade anterior (marcas de retroalimentação, *feedback* ou ciclo).

Conforme apresentado por Browning (2001), a DSM acima pode ser caracterizada como sendo temporal (uma vez que a ordem das linhas e das colunas indica um fluxo através do tempo) e por atividade, já que o estudo se baseia na otimização do fluxo de informação de um processo de engenharia.

Depois de construída a matriz, a próxima etapa consiste no redesenho do seqüenciamento das atividades do processo através da utilização de um algoritmo de particionamento presente na ferramenta computacional DSM@MIT.

Segundo Cho (2001), este algoritmo adota o método de particionamento desenvolvido por Warfield (1973) conhecido como "*Reachability Matrix Method*". Neste método, uma matriz binária é processada de forma a decompor um projeto em níveis hierárquicos onde as atividades estarão inseridas. Segundo Manzione (2006), estes níveis hierárquicos podem ser constituidos de núcleos de atividades que podem ser executadas independentemente (sejam sequenciais ou simultâneas), ou de blocos acoplados, que representam atividades interdependentes (dentro das quais há ao menos uma marca de retroalimentação acima da diagonal principal).

Após o particionamento, as atividades são rearranjadas e os blocos acoplados são identificados. Um algoritmo fornece duas visões para o particionamento: a primeira baseada na regra do AEAP (*As Early As Possible*), ou seja, que assume que uma atividade se inicia imediatamente após todas as informações necessárias estarem disponíveis; e a segunda baseada na regrado ALAP (*As Late As Possible*), ou seja, que assume que uma atividade se inicia somente quando for necessário, sem que haja atraso no projeto.

As figuras 4.5 e 4.6 representam as matrizes particionadas sob regras AEAP e ALAP:

Figura 4.5: DSM particionada sob a regra AEAP.

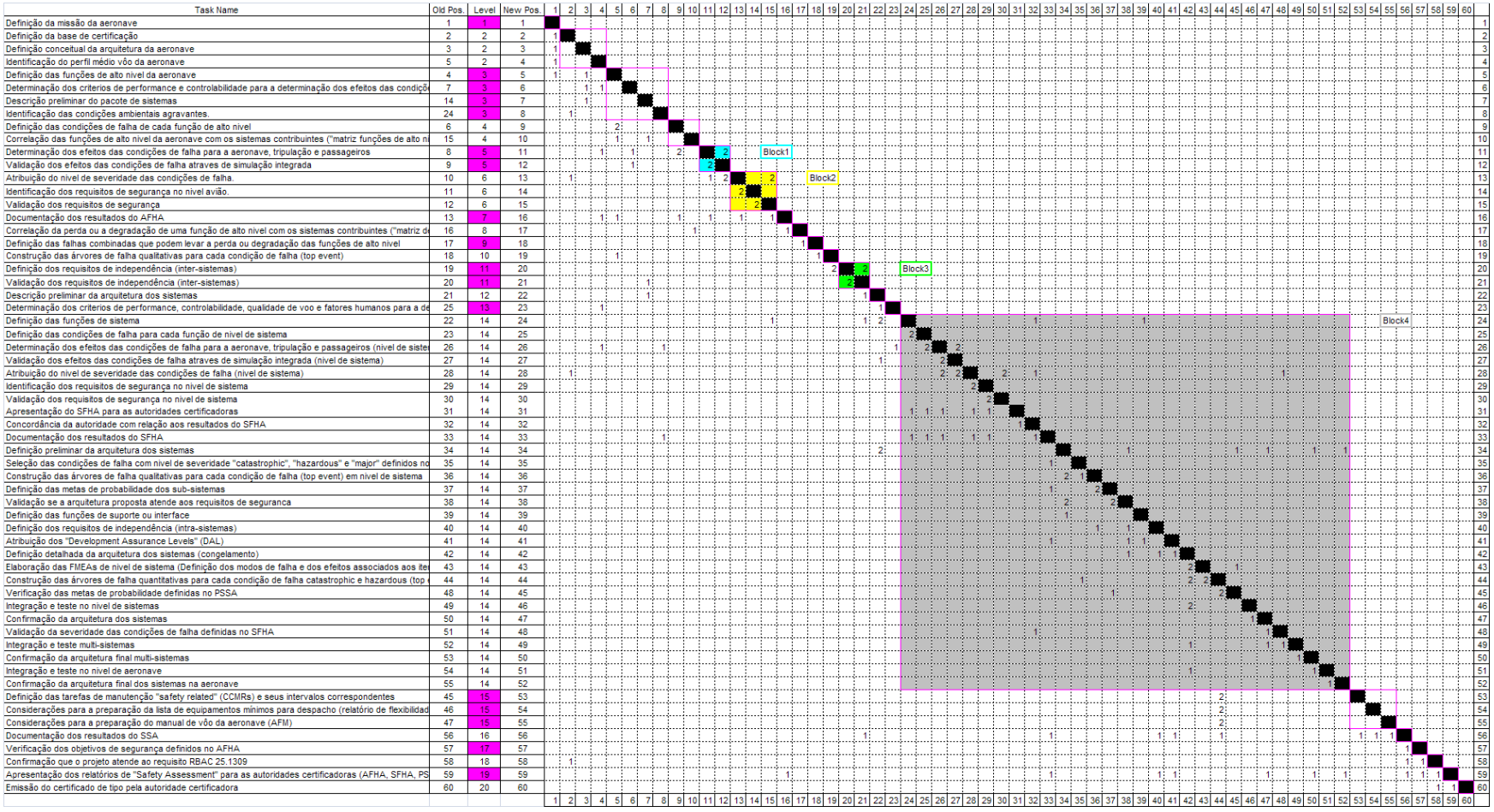
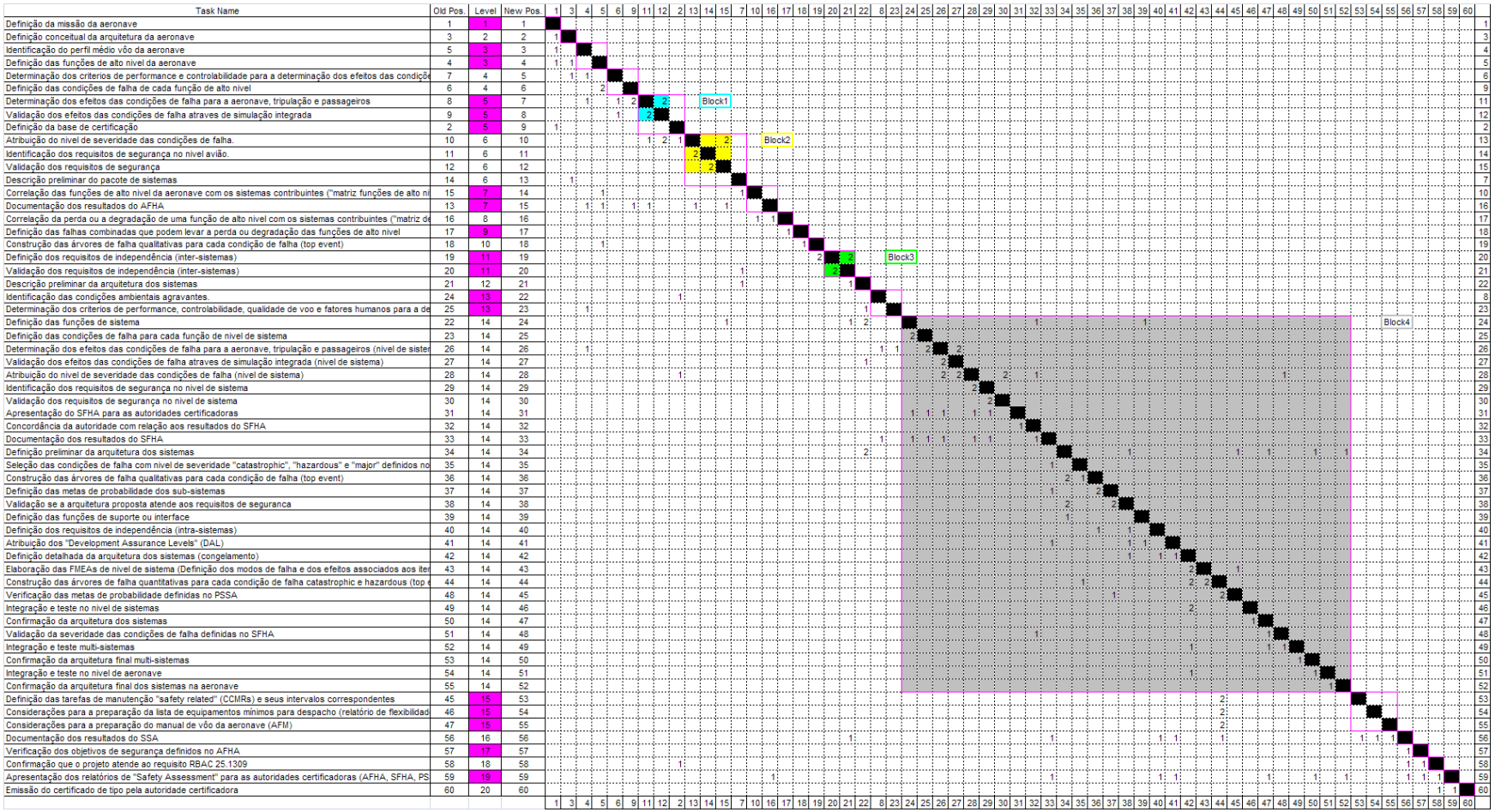


Figura 4.6: DSM particionada sob a regra ALAP.



Task Name	Old Pos.	Level	New Pos.
Definição da missão da aeronave	1	1	1
Definição conceitual da arquitetura da aeronave	3	2	2
Identificação do perfil médio vôo da aeronave	5	3	3
Definição das funções de alto nivel da aeronave	4	3	4
Determinação dos criterios de performance e controlabilidade para a determinação dos efeitos das condiçõe	7	4	5
Definição das condições de falha de cada função de alto nivel	6	4	6
Determinação dos efeitos das condições de falha para a aeronave, tripulação e passageiros	8	5	7
Validação dos efeitos das condições de falha atraves de simulação integrada	9	5	8
Definição da base de certificação	2	5	9
Atribuição do nivel de severidade das condições de falha.	10	6	10
Identificação dos requisitos de segurança no nivel avião.	11	6	11
Validação dos requisitos de segurança	12	6	12
Descrição preliminar do pacote de sistemas	14	6	13
Correlação das funções de alto nivel da aeronave com os sistemas contribuintes ("matriz funções de alto ni	15	7	14
Documentação dos resultados do AFHA	13	7	15
Correlação da perda ou a degradação de uma função de alto nivel com os sistemas contribuintes ("matriz de	16	8	16
Definição das falhas combinadas que podem levar a perda ou degradação das funções de alto nivel	17	9	17
Construção das árvores de falha qualitativas para cada condição de falha (top event)	18	10	18
Definição dos requisitos de independência (inter-sistemas)	19	11	19
Validação dos requisitos de independência (inter-sistemas)	20	11	20
Descrição preliminar da arquitetura dos sistemas	21	12	21
Identificação das condições ambientais agravantes.	24	13	22
Determinação dos criterios de performance, controlabilidade, qualidade de voo e fatores humanos para a de	25	13	23
Definição das funções de sistema	22	14	24
Definição das condições de falha para cada função de nivel de sistema	23	14	25
Determinação dos efeitos das condições de falha para a aeronave, tripulação e passageiros (nivel de siste	26	14	26
Validação dos efeitos das condições de falha atraves de simulação integrada (nivel de sistema)	27	14	27
Atribuição do nivel de severidade das condições de falha (nivel de sistema)	28	14	28
Identificação dos requisitos de segurança no nivel de sistema	29	14	29
Validação dos requisitos de segurança no nivel de sistema	30	14	30
Apresentação do SFHA para as autoridades certificadoras	31	14	31
Concordância da autoridade com relação aos resultados do SFHA	32	14	32
Documentação dos resultados do SFHA	33	14	33
Definição preliminar da arquitetura dos sistemas	34	14	34
Seleção das condições de falha com nivel de severidade "catastrophic", "hazardous" e "major" definidos no	35	14	35
Construção das árvores de falha qualitativas para cada condição de falha (top event)	36	14	36
Definição das metas de probabilidade dos sub-sistemas	37	14	37
Validação se a arquitetura proposta atende aos requisitos de seguranca	38	14	38
Definição das funções de suporte ou interface	39	14	39
Definição dos requisitos de independência (intra-sistemas)	40	14	40
Atribuição dos "Development Assurance Levels" (DAL)	41	14	41
Definição detalhada da arquitetura dos sistemas (congelamento)	42	14	42
Elaboração das FMEAs de nivel de sistema (Definição dos modos de falha e dos efeitos associados aos iter	43	14	43
Construção das árvores de falha quantitativas para cada condição de falha catastrophic e hazardous (top e	44	14	44
Verificação das metas de probabilidade definidas no PSSA	48	14	45
Integração e teste no nivel de sistemas	49	14	46
Confirmação da arquitetura dos sistemas	50	14	47
Validação da severidade das condições de falha definidas no SFHA	51	14	48
Integração e teste multi-sistemas	52	14	49
Confirmação da arquitetura final multi-sistemas	53	14	50
Integração e teste no nivel de aeronave	54	14	51
Confirmação da arquitetura final dos sistemas na aeronave	55	14	52
Definição das tarefas de manutenção "safety related" (CCMRs) e seus intervalos correspondentes	45	15	53
Considerações para a preparação da lista de equipamentos mínimos para despacho (relatório de flexibilidad	46	15	54
Considerações para a preparação do manual de vôo da aeronave (AFM)	47	15	55
Documentação dos resultados do SSA	56	16	56
Verificação dos objetivos de segurança definidos no AFHA	57	17	57
Confirmação que o projeto atende ao requisito RBAC 25.1309	58	18	58
Apresentação dos relatórios de "Safety Assessment" para as autoridades certificadoras (AFHA, SFHA, PS	59	19	59
Emissão do certificado de tipo pela autoridade certificadora	60	20	60

Após o particionamento ter sido realizado sob as regras AEAP e ALAP, é possível verificar que algumas atividades tiveram seu seqüenciamento modificado em função da otimização do fluxo de informação. As posições originais e as novas posições podem ser vistas através das colunas "*Old Pos*" e "*New Pos*" em cada uma das matrizes.

As 60 atividades do processo foram associadas a 20 níveis hierárquicos através do algoritmo de particionamento que estão identificados na coluna "Level". A mudança de cor na coluna "*Level*" indica a alteração de um nível para outro.

Tanto na matriz AEAP quanto na matriz ALAP foram gerados 4 blocos acoplados contendo as atividades cíclicas ou interdependentes. Os blocos foram identificados através de uma numeração de 1 a 4 e através de um código de cores: "*Block*" 1, "*Block*" 2, "*Block*" 3 e "*Block*" 4. Os relacionamentos seqüenciais e simultâneos entre as atividades também foram identificados dentro de cada nível hierárquico associado.

Outra forma de se representar o mapeamento das atividades através da DSM particionada é através do colapso de blocos acoplados em blocos "atividade", visando uma leitura mais concisa do processo (pares 11;11, 12;12, 17;17 e 20;20). Ver figura 4.7.

Nesta visão, segundo Cho (2001) apud Eppinger et al (1993), o fluxo de informação de uma atividade simples para um bloco "atividade" é marcado como sendo do tipo "2", quando existe ao menos um fluxo deste tipo entre uma atividade simples e uma atividade contida no bloco. Em contrapartida, qualquer fluxo de informação de um bloco "atividade" para qualquer atividade simples ou mesmo outro bloco "atividade" será também do tipo "2", uma vez que é muito provável que informações preliminares de atividades dentro dos blocos acoplados sejam transferidas para atividades posteriores antes de convergirem para sua forma final ao final das iterações.

As figuras 4.7 e 4.8 representam as visões colapsadas (geradas através da ferramenta DSM@MIT) sob as regras AEAP e ALAP respectivamente:

Figura 4.7: DSM colapsada particionada sob a regra AEAP.

Task Name	Level	New Pos.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	
Definição da missão da aeronave	1	1																													1
Definição da base de certificação	2	2	1																												2
Definição conceitual da arquitetura da aeronave	2	3	1																												3
Identificação do perfil médio vôo da aeronave	2	4	1																												4
Definição das funções de alto nivel da aeronave	3	5	1		1																										5
Determinação dos criterios de performance e controlabilidade para a determinação dos efeito	3	6			1	1																									6
Descrição preliminar do pacote de sistemas	3	7			1																										7
Identificação das condições ambientais agravantes.	3	8		1																											8
Definição das condições de falha de cada função de alto nivel	4	9					2																								9
Correlação das funções de alto nivel da aeronave com os sistemas contribuintes ("matriz fun	4	10					1		1																						10
Block1:	5	11				1		1			2																				11
Block2:	6	12		1									2																		12
Documentação dos resultados do AFHA	7	13				1	1				1		2	2																	13
Correlação da perda ou a degradação de uma função de alto nivel com os sistemas contribuir	8	14										1			1															14
Definição das falhas combinadas que podem levar a perda ou degradação das funções de al	9	15														1															15
Construção das árvores de falha qualitativas para cada condição de falha (top event)	10	16					1										1														16
Block3:	11	17							1									2													17
Descrição preliminar da arquitetura dos sistemas	12	18							1										2												18
Determinação dos criterios de performance, controlabilidade, qualidade de voo e fatores hum	13	19				1														1											19
Block4:	14	20		1		1				1				2					2	2	1											20
Definição das tarefas de manutenção "safety related" (CCMRs) e seus intervalos correspond	15	21																				2									21
Considerações para a preparação da lista de equipamentos mínimos para despacho (relatóric	15	22																				2									22
Considerações para a preparação do manual de vôo da aeronave (AFM)	15	23																				2									23
Documentação dos resultados do SSA	16	24																	2			2	1	1	1						24
Verificação dos objetivos de segurança definidos no AFHA	17	25																								1					25
Confirmação que o projeto atende ao requisito RBAC 25.1309	18	26		1																						1	1				26
Apresentação dos relatórios de "Safety Assessment" para as autoridades certificadoras (AF	19	27													1							2				1	1	1			27
Emissão do certificado de tipo pela autoridade certificadora	20	28																										1	1		28
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	

Figura 4.8: DSM colapsada particionada sob a regra ALAP.

Task Name	Level	New Pos.	1	3	4	5	6	9	11	2	12	7	10	13	14	15	16	17	18	8	19	20	21	22	23	24	25	26	27	28	
Definição da missão da aeronave	1	1																													1
Definição conceitual da arquitetura da aeronave	2	2	1																												3
Identificação do perfil médio vôo da aeronave	3	3	1																												4
Definição das funções de alto nivel da aeronave	3	4	1	1																											5
Determinação dos criterios de performance e controlabilidade para a determinação do	4	5		1	1																										6
Definição das condições de falha de cada função de alto nivel	4	6				2																									9
Block1:	5	7			1		1	2																							11
Definição da base de certificação	5	8	1																												2
Block2:	6	9							2	1																					12
Descrição preliminar do pacote de sistemas	6	10		1																											7
Correlação das funções de alto nivel da aeronave com os sistemas contribuintes ("ma	7	11				1						1																			10
Documentação dos resultados do AFHA	7	12			1	1		1	2		2																				13
Correlação da perda ou a degradação de uma função de alto nivel com os sistemas c	8	13											1	1																	14
Definição das falhas combinadas que podem levar a perda ou degradação das funçõ	9	14													1																15
Construção das árvores de falha qualitativas para cada condição de falha (top event)	10	15				1										1															16
Block3:	11	16										1					2														17
Descrição preliminar da arquitetura dos sistemas	12	17										1						2													18
Identificação das condições ambientais agravantes.	13	18								1																					8
Determinação dos criterios de performance, controlabilidade, qualidade de voo e fator	13	19			1														1												19
Block4:	14	20			1					1	2							2	2	1	1										20
Definição das tarefas de manutenção "safety related" (CCMRs) e seus intervalos corr	15	21																				2									21
Considerações para a preparação da lista de equipamentos mínimos para despacho (	15	22																				2									22
Considerações para a preparação do manual de vôo da aeronave (AFM)	15	23																				2									23
Documentação dos resultados do SSA	16	24																2				2	1	1	1						24
Verificação dos objetivos de segurança definidos no AFHA	17	25																								1					25
Confirmação que o projeto atende ao requisito RBAC 25.1309	18	26								1																1	1				26
Apresentação dos relatórios de "Safety Assessment" para as autoridades certificado	19	27												1								2				1	1	1			27
Emissão do certificado de tipo pela autoridade certificadora	20	28																										1	1		28
			1	3	4	5	6	9	11	2	12	7	10	13	14	15	16	17	18	8	19	20	21	22	23	24	25	26	27	28	

É importante ressaltar que as iterações contidas nos blocos acoplados são intencionais (planejadas) e que as iterações não-intencionais (não-planejadas) não estão representadas na aplicação do método proposto.

Outra visão que pode ser gerada através de outro algoritmo contido na ferramenta é aquela onde os níveis de folga (diferença entre as visões AEAP e ALAP) são delineados em função dos resultados mostrados nas matrizes colapsadas.

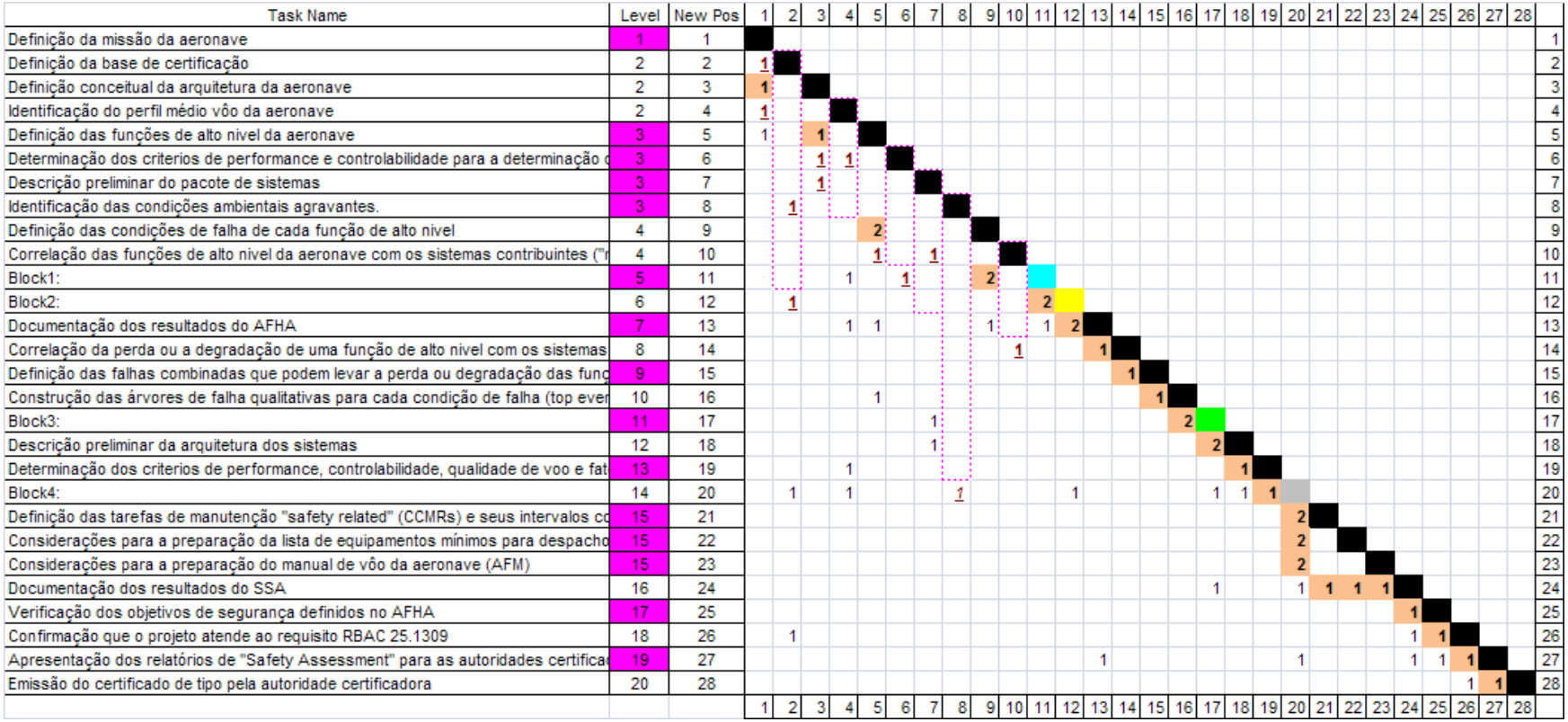
As linhas tracejadas são delineadas em torno das colunas que representam as atividades onde a folgas podem existir entre os níveis hierárquicos (Exemplo: coluna 2 entre as linhas 2 e 11, coluna 8, entre as linhas 8 e 19, etc). Neste caso, uma determinada atividade pode estar localizada entre níveis diferentes.

Outra visibilidade representada por este algoritmo é aquela que mapeia o tipo de dependência de informação entre as atividades. As dependências vinculativas são representadas através de marcas em algumas células. Segundo Cho (2001) as células pintadas em laranja (Exemplo: relacionamentos 5;3, 9;5, 13;12, etc) representam um vinculo crítico uma vez que é aquele que ocorre quando uma dependência entre duas atividades tem “folga zero” entre níveis hierárquicos após o particionamento da matriz. As marcas sublinhadas (Exemplo: 6;3, 11;6, 14;10, etc), por sua vez, representam os vínculos não-críticos, ou seja, onde existe certa liberdade de movimento dentro de um planejamento.

Dependências do tipo não-vinculativas são representadas através das demais marcas na matriz, como por exemplo, os relacionamentos 5;1, 16;5, 20;12, entre outros.

A figura 4.9 representa a forma colapsada AEAP contendo as visões relativas aos níveis de folga e os tipos de dependência entre as atividades:

Figura 4.9: DSM colapsada particionada sob a regra AEAP com os níveis de folga e o tipo de dependência entre as atividades.



Task Name	Level	New Pos	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	
Definição da missão da aeronave	1	1																													1
Definição da base de certificação	2	2	1																												2
Definição conceitual da arquitetura da aeronave	2	3	1																												3
Identificação do perfil médio vôo da aeronave	2	4	1																												4
Definição das funções de alto nivel da aeronave	3	5	1		1																										5
Determinação dos criterios de performance e controlabilidade para a determinação d	3	6			1	1																									6
Descrição preliminar do pacote de sistemas	3	7			1																										7
Identificação das condições ambientais agravantes.	3	8		1																											8
Definição das condições de falha de cada função de alto nivel	4	9					2																								9
Correlação das funções de alto nivel da aeronave com os sistemas contribuintes ("r	4	10					1		1																						10
Block1:	5	11				1		1			2																				11
Block2:	6	12		1									2																		12
Documentação dos resultados do AFHA	7	13				1	1				1		1	2																	13
Correlação da perda ou a degradação de uma função de alto nivel com os sistemas	8	14										1			1															14
Definição das falhas combinadas que podem levar a perda ou degradação das funç	9	15														1															15
Construção das árvores de falha qualitativas para cada condição de falha (top ever	10	16					1										1														16
Block3:	11	17							1									2													17
Descrição preliminar da arquitetura dos sistemas	12	18							1										2												18
Determinação dos criterios de performance, controlabilidade, qualidade de voo e fat	13	19				1														1											19
Block4:	14	20		1		1				1				1					1	1	1										20
Definição das tarefas de manutenção "safety related" (CCMRs) e seus intervalos co	15	21																				2									21
Considerações para a preparação da lista de equipamentos mínimos para despacho	15	22																				2									22
Considerações para a preparação do manual de vôo da aeronave (AFM)	15	23																				2									23
Documentação dos resultados do SSA	16	24																	1			1	1	1	1						24
Verificação dos objetivos de segurança definidos no AFHA	17	25																								1					25
Confirmação que o projeto atende ao requisito RBAC 25.1309	18	26		1																						1	1				26
Apresentação dos relatórios de "Safety Assessment" para as autoridades certifica	19	27													1							1				1	1	1			27
Emissão do certificado de tipo pela autoridade certificadora	20	28																										1	1		28
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	

4.3.3 Elaboração da Programação das Atividades do Processo

Com a definição das precedências entre as atividades em função do novo seqüenciamento e com as durações médias estimadas inicialmente para cada atividade, o processo de *Safety Assessment* pode ser programado através da construção de um cronograma.

Este cronograma, com visualização através do Diagrama de Gantt, foi obtido pela transferência dos dados de saída das DSM's (visão AEAP) para o software MS-Project. A visão AEAP foi escolhida pelo fato de apresentar a programação de atividades num menor tempo possível.

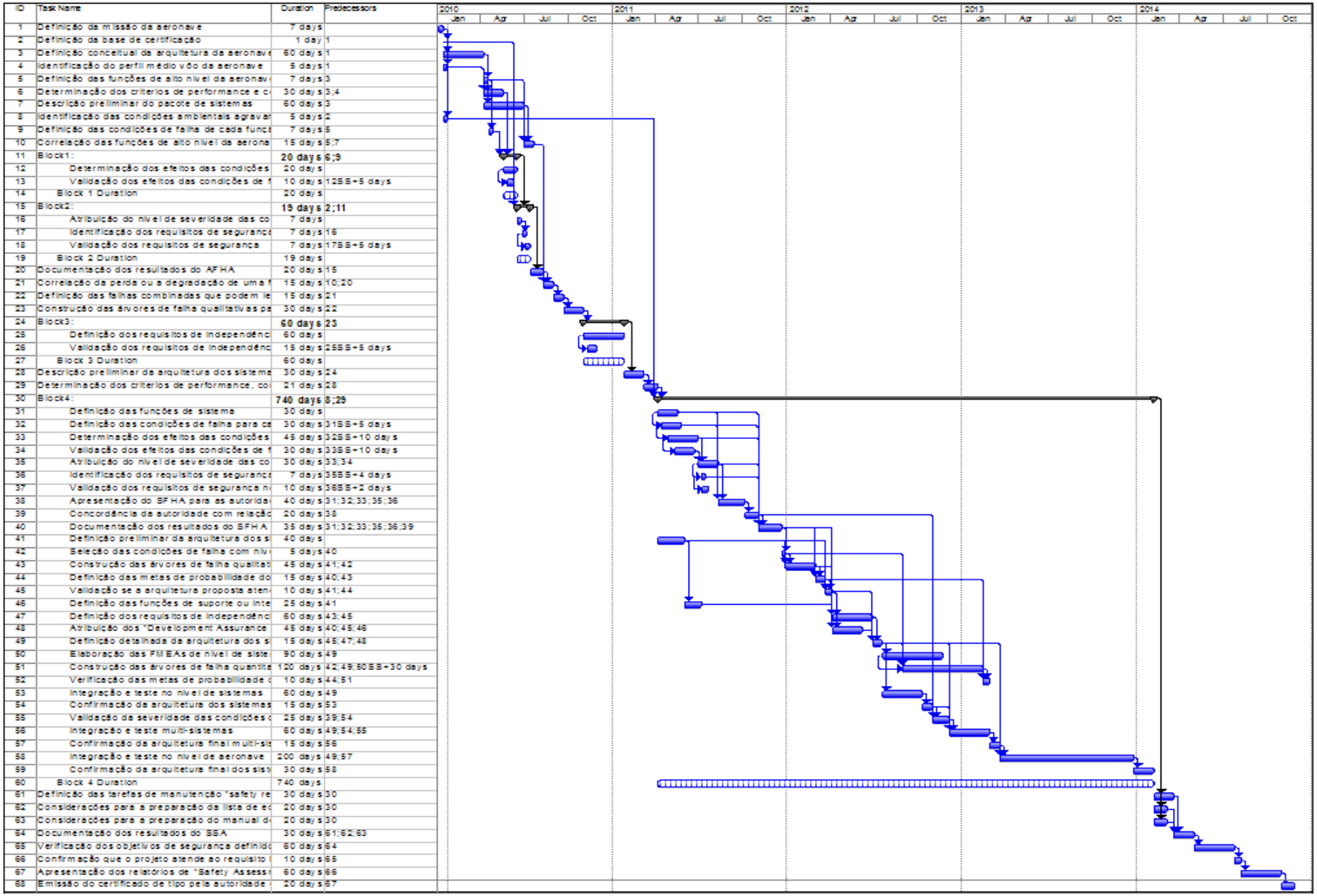
As precedências foram determinadas automaticamente pela ferramenta DSM@MIT, que considerou para tal o seqüenciamento otimizado determinado pela DSM. Os níveis hierárquicos, contendo atividades seqüenciais, paralelas ou blocos acoplados também foram levados em consideração na determinação das atividades predecessoras. Nos casos onde o fluxo de informação é do tipo 1, a precedência considerada foi do tipo "*Finish-to-Start*", ou seja, quando é necessário o término de uma atividade para o início da outra. Quando o fluxo é do tipo 2, a precedência considerada foi do tipo "Start-to-Start", que determina que uma atividade posterior só se inicie quando a atividade anterior se iniciar. Nestes casos, foram considerados intervalos de tempo (atrasos ou "*lags*") entre o início de uma e o início da outra.

É importante ressaltar que num Diagrama de Gantt, o tempo de convergência entre duas atividades interdependentes pode não ser o mesmo representado pelas respectivas durações. Neste caso, segundo Cho (2001), a utilização de atividades fantasmas (*dummy tasks*) nos blocos acoplados pode contornar este problema, uma vez que o tempo estimado desta atividade pode ser atualizado no decorrer do progresso do ciclo de iteração.

Os recursos envolvidos em cada atividade não foram considerados nesta programação.

A figura 4.10 representa o Diagrama de Gantt após o processamento da DSM. As barras em azul representam as durações das atividades do processo enquanto que as barras claras com linhas verticais representam as atividades fantasmas.

Figura 4.10: Diagrama de Gantt após processamento da DSM (visão AEAP).



ID	Task Name	Duration	Predecessors
1	Definição da missão da aeronave	7 days	
2	Definição da base de certificação	1 day	1
3	Definição conceitual da arquitetura da aeronave	60 days	1
4	Identificação do perfil médio vôo da aeronave	5 days	1
5	Definição das funções de alto nível da aeronav	7 days	3
6	Determinação dos critérios de performance e c	30 days	3;4
7	Descrição preliminar do pacote de sistemas	60 days	3
8	Identificação das condições ambientais agravar	5 days	2
9	Definição das condições de falha de cada funçã	7 days	5
10	Correlação das funções de alto nível da aerona	15 days	5;7
11	Block1:	20 days	6;9
12	Determinação dos efeitos das condições	20 days	
13	Validação dos efeitos das condições de f	10 days	12SS+5 days
14	Block 1 Duration	20 days	
15	Block2:	19 days	2;11
16	Atribuição do nível de severidade das co	7 days	
17	Identificação dos requisitos de segurança	7 days	16
18	Validação dos requisitos de segurança	7 days	17SS+5 days
19	Block 2 Duration	19 days	
20	Documentação dos resultados do AFHA	20 days	15
21	Correlação da perda ou a degradação de uma f	15 days	10;20
22	Definição das falhas combinadas que podem le	15 days	21
23	Construção das árvores de falha qualitativas pa	30 days	22
24	Block3:	60 days	23
25	Definição dos requisitos de independênci	60 days	
26	Validação dos requisitos de independênc	15 days	25SS+5 days
27	Block 3 Duration	60 days	
28	Descrição preliminar da arquitetura dos sistema	30 days	24
29	Determinação dos critérios de performance, co	21 days	28
30	Block4:	740 days	8;29
31	Definição das funções de sistema	30 days	
32	Definição das condições de falha para ca	30 days	31SS+5 days
33	Determinação dos efeitos das condições	45 days	32SS+10 days
34	Validação dos efeitos das condições de f	30 days	33SS+10 days
35	Atribuição do nível de severidade das co	30 days	33;34
36	Identificação dos requisitos de segurança	7 days	35SS+4 days
37	Validação dos requisitos de segurança ni	10 days	36SS+2 days
38	Apresentação do SFHA para as autoridad	40 days	31;32;33;35;36
39	Concordância da autoridade com relação	20 days	38
40	Documentação dos resultados do SFHA	35 days	31;32;33;35;36;39
41	Definição preliminar da arquitetura dos s	40 days	
42	Seleção das condições de falha com nív	5 days	40
43	Construção das árvores de falha qualitat	45 days	41;42
44	Definição das metas de probabilidade do	15 days	40;43
45	Validação se a arquitetura proposta aten	10 days	41;44
46	Definição das funções de suporte ou inte	25 days	41
47	Definição dos requisitos de independênci	60 days	43;45
48	Atribuição dos "Development Assurance	45 days	40;45;46
49	Definição detalhada da arquitetura dos s	15 days	45;47;48
50	Elaboração das FMEAs de nível de siste	90 days	49
51	Construção das árvores de falha quantita	120 days	42;49;50SS+30 days
52	Verificação das metas de probabilidade d	10 days	44;51
53	Integração e teste no nível de sistemas	60 days	49
54	Confirmação da arquitetura dos sistemas	15 days	53
55	Validação da severidade das condições d	25 days	39;54
56	Integração e teste multi-sistemas	60 days	49;54;55
57	Confirmação da arquitetura final multi-sis	15 days	56
58	Integração e teste no nível de aeronave	200 days	49;57
59	Confirmação da arquitetura final dos sist	30 days	58
60	Block 4 Duration	740 days	
61	Definição das tarefas de manutenção "safety re	30 days	30
62	Considerações para a preparação da lista de eq	20 days	30
63	Considerações para a preparação do manual d	20 days	30
64	Documentação dos resultados do SSA	30 days	61;62;63
65	Verificação dos objetivos de segurança definido	60 days	64
66	Confirmação que o projeto atende ao requisito	10 days	65
67	Apresentação dos relatórios de "Safety Assess	60 days	66
68	Emissão do certificado de tipo pela autoridade	20 days	67

5 Resultados e Discussões

Neste capitulo serão discutidos os resultados apresentados na aplicação do método proposto, onde a DSM foi aplicada com o objetivo de propor uma sistemática de otimização do processo através da identificação de pontos de melhoria visando a redução do tempo de ciclo do projeto.

Um dos primeiros resultados obtidos foi a identificação dos tipos de marca de dependência entre as atividades na matriz (conforme figura 4.4). Foram identificadas 15 marcas de retroalimentação ou ciclos de iteração (onde a informação é transferida de uma atividade posterior para uma atividade anterior), o que denota ao processo uma relevante complexidade. As outras 129 marcas representam informação transferida de uma atividade anterior para outra posterior (marcas de alimentação).

Após o particionamento da matriz através da ferramenta computacional DSM@MIT foram obtidas duas visões, onde a primeira assume que uma atividade se inicia imediatamente após todas as informações necessárias estarem disponíveis (AEAP) e a segunda que assume que uma atividade se inicia somente quando for necessário, sem que haja atraso no projeto (ALAP). Uma vez o objetivo do trabalho é o de propor a redução do tempo de ciclo do projeto, será dada ênfase na análise da visão AEAP.

Na visão AEAP (figura 4.5) da DSM foram identificados 20 níveis hierárquicos, representados por 4 blocos acoplados (contendo atividades interdependentes), 4 agrupamentos de atividades independentes e 12 atividades solitárias.

Com relação aos blocos acoplados, é possível destacar os seguintes pontos:

- *Block* 1: Este bloco é composto pelas atividades 8 e 9 referentes ao FHA em nível aeronave dentro da etapa de projeto conceitual. A interdependência neste caso ocorre uma vez que os efeitos das condições de falha para a aeronave, tripulação e passageiros devem ser validados posteriormente através de simulação integrada e neste momento ocorrem diversos ajustes antes da convergência. Com isso, é necessário garantir que os recursos (pessoal capacitado, ferramentas e procedimentos) estejam disponíveis para as simulações integradas de forma a reduzir o tempo de iteração.

- *Block* 2: Este bloco é composto pelas atividades 10, 11 e 12 referentes também ao FHA em nível aeronave, dentro da etapa de projeto conceitual. A interdependência ocorre no ajuste proposto pela validação dos requisitos de segurança, definidos em função das severidades atribuídas a cada condição de falha. Esta validação se dá através de reuniões internas e *checklist*, onde ajustes ocorrem antes da convergência. Da mesma maneira que no bloco anterior, é necessário garantir que os recursos: *checklist* preparado e pessoas capacitadas e disponíveis para reuniões de validação. Um bom conhecimento na interpretação dos requisitos também é essencial.

- *Block* 3: O bloco é composto pelas atividades 19 e 20, referentes a etapa de definição dos requisitos de independência inter-sistemas dentro do projeto conceitual e que é realizada através da construção de árvores de falha qualitativas. A atividade de validação dos requisitos de independência se relaciona iterativamente com a definição destes requisitos. A convergência se dá através de ajustes propostos durante discussões com a engenharia de desenvolvimento do produto. A garantia dos recursos necessários (diagrama de blocos funcionais dos sistemas e especialistas capacitados) é imprescindível para a rápida convergência neste momento do processo.

- *Block* 4: Este bloco se constitui no coração de todo o processo de *Safety Assessment*, uma vez que engloba etapas importantíssimas de definição de arquitetura dos sistemas em função daquilo que é exigido pelos requisitos de segurança. Composto por 29 atividades, este bloco se inicia na execução do FHA em nível de sistema e termina na conclusão das verificações (integrações e testes) no nível de aeronave. O seu tamanho é conseqüência da grande concentração de ciclos de iteração (12 ao todo), onde a falha em atender um requisito de segurança nos testes finais nos protótipos (verificação) pode acarretar a alteração nas funções de sistema e, conseqüentemente, afetar as atividades posteriores. Uma vez que é um bloco que abrange um grande número de atividades (desde o projeto preliminar até a qualificação), um esforço detalhado e antecipado de planejamento é essencial para ter como resultado a garantia dos recursos necessários.

Além dos blocos acoplados e das 12 atividades solitárias é possível também verificar alguns agrupamentos de atividades que foram gerados após o particionamento da matriz. Estes agrupamentos são compostos por atividades independentes, uma vez que não há a troca de informação entre elas.

Os agrupamentos identificados foram:

- Agrupamento 1: Atividades 2, 3 e 5.
- Agrupamento 2: Atividades 4, 7, 14 e 24.
- Agrupamento 3: Atividades 6 e 15.
- Agrupamento 4: Atividades 45, 46 e 47.

Com este cenário gerado pelo particionamento da matriz sob a ótica AEAP, é possível avaliar o novo seqüenciamento proposto pela DSM.

Atualmente a empresa estudada adota um seqüenciamento de processo quase linear (com reduzido número de *overlap* de atividades), onde o planejamento para execução das atividades não leva em conta a análise do fluxo de informações. Sendo assim, a distribuição das atividades e dos recursos não é bem equilibrada durante o desenvolvimento do produto.

O impacto deste desequilíbrio gera, em muitos casos, as seguintes conseqüências:

- Necessidade adicional de recursos nas etapas finais do processo.
- Aumento da quantidade de horas extras.
- Aumento dos custos de desenvolvimento do produto.
- Risco de não cumprimento dos prazos de certificação do produto.
- Aumento do tempo de ciclo.
- Não atendimento das expectativas dos clientes podendo afetar a imagem da empresa.

Com a aplicação do método, foi possível identificar algumas oportunidades de melhoria no que se refere tanto à otimização do seqüenciamento das atividades (visando a redução do tempo de ciclo), quanto na possibilidade da melhoria do planejamento de recursos em função da maior visibilidade e do *timing* de todo o processo.

Com relação à otimização do seqüenciamento das atividades, vale destacar algumas oportunidades com potencial de redução do tempo de ciclo em função desta proposta de seqüenciamento:

- As atividades contidas nos agrupamentos 1, 2, 3 e 4 podem ser executadas paralelamente em função da independência de informação entre elas. Com exceção dos agrupamentos 1 e 4, as atividades dos agrupamentos 2 e 3 não são executadas simultaneamente hoje na empresa.

- As atividades interdependentes contidas nos blocos acoplados 1, 2 e 3 podem ser executadas em paralelo, considerando para tal uma restrição (em dias) para o inicio das atividades posteriores. No caso do bloco acoplado 3, a execução em paralelo só é valida se considerada apenas a análise funcional. Na prática isto não ocorre uma vez que a análise instalativa exige que todos os requisitos de independência inter-sistemas estejam definidos antes da validação.

- O grupo de atividades (22, 23, 26 e 27), (28, 29 e 30) e (43, 44), contidas no bloco acoplado 4, também podem ser executadas em paralelo, considerando para tal uma restrição (em dias) para o início das atividades posteriores. É importante ressaltar que hoje a empresa adota um planejamento no qual estas atividades já são executadas de forma simultânea.

- A atividade 24, antes iniciada dentro da etapa de FHA em nível de sistema, poderia ser adiantada para o período de execução do FHA em nível aeronave, uma vez que é uma atividade que não depende de informações de atividades do processo para ser executada. A movimentação desta atividade para este novo estágio do processo é valida uma vez que a disponibilidade de mão de obra neste momento é maior.

- A atividade 15 (referente à etapa de definição dos requisitos de independência inter-sistemas) poderia ser adiantada para a etapa de FHA em nível aeronave, uma vez que as informações para sua execução e a mão de obra necessária para sua execução já se encontram disponíveis.

Vale ressaltar que a decisão de executar ou não certas atividades de forma paralela dependerá de decisões gerenciais em função da existência ou não de recursos disponíveis naquele momento.

A análise da figura 4.9, que representa a DSM colapsada particionada sob a regra AEAP contendo os níveis de folga e o tipo de dependência entre as atividades, também mostra alguns pontos importantes:

- As atividades 2, 4, 6, 7, 8 e 10 possuem folga diferente de zero e podem estar localizadas em níveis diferentes, dependendo do tipo de visão (AEAP ou ALAP) que está sendo considerada. Este tipo de visão pode ser importante no momento em que estão sendo definidas as divisões de recursos para o programa.

- Os relacionamentos linha/coluna (i,j): (3,1), (5,3), (9,5), (11,9), (12,11), (13,12), (14,13), (15,14), (16, 15), (17, 16), (18,17), (19,18), (20,19), (21,20), (22,20), (23,20), (24,21), (24,22), (24,23), (25,24), (26,25), (27,26) e (28,27), representam dependências vinculativas críticas, uma vez que ocorrem quando dependências entre duas atividades tem "folga zero" entre níveis hierárquicos. Segundo Cho (2001), esta visão fornece uma boa orientação para a melhoria do processo em estágios iniciais de planejamento quando dados detalhados de sua duração não estão disponíveis.

Após o particionamento da DSM (visão AEAP) foi realizada uma proposta para a programação das atividades através da utilização do Diagrama de Gantt (Figura 4.10).

Esta nova programação indicou um processo com uma duração de 59 meses, considerando seu início hipotético no mês de Janeiro de 2010. Vale ressaltar que como as atividades 1, 2, 3 e 5 (com duração total de 73 dias) não fazem parte do processo de *Safety Assessement*, a duração estimada prevista para o processo é de 57 meses (4 anos e 9 meses).

Uma vez que este processo tem tido em média uma duração de 60 meses (5 anos) nos últimos programas da empresa, é obtido somente com o reseqüenciamento das atividades uma redução no tempo de ciclo do projeto de 3 meses, o que pode representar para a empresa uma economia de recursos e também a possibilidade de ter disponível um tempo maior para atividades de testes de operação nos protótipos visando a detecção de problemas que possam vir a acontecer prematuramente em campo.

6 Conclusões e Sugestões para Próximos Trabalhos

A proposta deste trabalho foi a de propor um método visando a aplicação da matriz de estrutura de projetos (DSM) no planejamento das atividades do processo de *Safety Assessment* visando identificar potenciais melhorias que levem à redução do tempo de ciclo de desenvolvimento de aeronaves comerciais e também avaliar a viabilidade de sua utilização na prática da empresa.

A pesquisa se baseou no estudo da base teórica e metodológica por trás dos requisitos e práticas recomendadas utilizados nos dias de hoje. Com base neste estudo, foi possível compreender os conceitos de ciclo de vida do produto, engenharia simultânea, projeto voltado à segurança (*Design or Safety*), engenharia de sistemas, sistemas complexos, técnicas de análise de risco, certificação aeronáutica e por fim o processo de *Safety Assessment*.

O estudo de todos estes conceitos possibilitou dimensionar a complexidade intrínseca existente no processo, uma vez que ele é um dos grandes responsáveis em garantir o cumprimento dos requisitos de segurança no projeto de aeronaves.

Como um fator chave para a sua correta e completa execução, verificou-se que o planejamento das atividades deste processo deve ser realizado de maneira a atender aos custos de desenvolvimento e principalmente aos prazos de certificação e entrega do produto ao cliente.

Com o intuito de melhor entender as características existentes no planejamento de processos complexos, foram pesquisados os conceitos de iteração e tempo de ciclo em projetos, paralelismo, interdependência e decomposição hierárquica de atividades e as principais técnicas de programação de projetos.

O estudo destas características mostrou que a boa prática da engenharia simultânea requer a habilidade de uma organização em compartilhar informações entre membros de times multi-funcionais de desenvolvimento do produto. Sendo assim, foi possível entender que o correto gerenciamento do fluxo de informações é um fator importante na redução de tempo de ciclo de um projeto.

Dentre as técnicas de programação de projeto pesquisadas, a DSM foi considerada a mais indicada para lidar com gerenciamento do fluxo de informações, uma vez que, diferentemente das outras técnicas pesquisadas, possibilita o correto mapeamento de atividades interdependentes. Verificou-se também que a sua utilização não inviabiliza a utilização das outras técnicas, pelo contrário, serve como complemento para um planejamento mais robusto e correto.

Outro ponto a destacar sobre a DSM é a sua capacidade de otimizar o seqüenciamento das atividades de um processo. Esta característica faz da DSM uma técnica que permite minimizar as iterações que possam existir no processo de modo a reduzir o seu tempo de ciclo.

Os pontos chave para a correta aplicação desta técnica são, sem dúvida, a modelagem e o gerenciamento do processo. Estes dois pontos foram delineados em um método (baseado numa metodologia já existente) visando a decomposição hierárquica das atividades e a construção dos relacionamentos de dependência entre as atividades. Caso estas duas etapas não estejam bem estruturadas e precisas, os resultados gerados podem não ser satisfatórios.

Sendo assim, a complexidade exigida para a construção da DSM, representada pela necessidade de um conhecimento detalhado do processo nos niveis mais elementares das atividades, se constitui de um fator de atenção na hora de avaliar o uso intensivo desta técnica.

Tendo em vista que a empresa estudada não utiliza um planejamento de processo com abordagem no fluxo de informações, foi detectada a oportunidade de se aplicar a DSM na avaliação deste processo.

Neste sentido, os resultados apresentados demostraram que o atual planejamento do processo *Safety Assessment* praticado pela empresa possui algumas oportunidades de melhoria que poderiam ser implantadas para os próximos programas da empresa.

Estas melhorias identificadas pela DSM podem trazer para a empresa a possibilidade de redução no tempo de ciclo do projeto e também um maior gerenciamento dos recursos, podendo ser utilizada em conjunto com uma tabela de prontidão de mão de obra e ferramentas.

Além disso, a DSM apresenta uma característica dinâmica onde a capacidade de resposta apresentada nos seus resultados aumenta conforme o nível de conhecimento do processo aumenta.

Outro ponto a se destacar, é a grande capacidade de visualização do sequenciamento e dos pontos de atenção no processo, principalmente naqueles onde os blocos acoplados estão presentes. Com essa visualização nos estágios iniciais de um projeto, é possível desenvolver um planejamento muito mais adequado para o programa.

Finalmente, a DSM se mostrou uma técnica viável para a aplicação no gerenciamento de processo de projeto na empresa, sendo que a sua utilização pode ser complementada com a aplicação de outras técnicas já utilizadas, como o PERT/CPM e o Método da Corrente Crítica.

Com sugestão para os próximos trabalhos, é possível destacar algumas idéias:

- Utilização da DSM na avaliação do planejamento das atividades do processo com base nas perspectivas instalativa e operacional do *Safety Assessment*, uma vez que também apresentam um grande nível de interação entre as atividades.
- Utilização da DSM para a programação de projetos utilizando uma abordagem probabilística para modelamento das durações das atividades do processo durante a construção do cronograma.

- Estudos referentes à interação de parâmetros de projeto de sistemas utilizados durante o desenvolvimento do produto visando a identificação ciclos de iteração.
- Estudos para avaliar as interações existentes entre times de desenvolvimento do produto visando otimizar a etapa de integração de sistemas.
- Estudos visando o modelamento e análise de arquiteturas de produto e sistemas com foco na definição das interações espaciais, energéticas, de informação ou de material, existentes entre subsistemas e componentes (neste caso utilizando algoritmos de *clustering*.
- Utilização das DMMs ou Matrizes de Mapeamento de Domínios (*Domain Mapping Matrices*) para estudos de iteração entre domínios. As abordagens de DSM e DMM são complementares umas em relação à outra. Enquanto a DSM foca em um único domínio (Exemplo: atividades de um projeto) a DMM foca na interação entre os domínios (Exemplo: Confrontar a arquitetura do produto com a organização, os requisitos do produto com seus requisitos funcionais ou mesmo a arquitetura do produto com os requisitos do cliente).

Referências

ATA - AIR TRANSPORT ASSOCIATION: ANNUAL TRAFFIC AND OPS. Disponível em < http://www.airlines.org/economics/traffic/World+Airline+Traffic.htm>. Acesso em 10 de janeiro de 2010.

ANAC (AGÊNCIA NACIONAL DE AVIAÇÃO CIVIL). Disponível em < www.anac.gov.br >. Acesso em 5 de Fevereiro de 2009.

AUSTIN, S. et. al. "Analytical Design Planning Technique: A Model of the Detailed Building Design Process" Design Studies Vol. 20 N.3. p. 279-296. May 1999.

AUSTIN, S. et. al. "Analytical Design Planning Technique (ADePT) – A Dependency Structure Matrix Tool to Schedule the Building Desing Process" Construction Management and Economics, p.173-182. 2000.

BACK, N. "Capacitação em Desenvolvimento Integrado de Produtos" Núcleo de Desenvolvimento Integrado de Produtos (NEDIP), Material de Trabalho Referente à Palestra Realizada na Universidade Federal de Santa Maria, RS, 2007.

BARTOLOMEI, J.E. "Qualitative Knowledge Construction for Engineering Systems: Extending the Design Structure Matrix Methodology in Scope and Procedure" 2007. 191p. Doctor of Philosophy in Engineering Systems, Massachusetts Institute of Technology.

BLACK, T.A.; FINE, C.H.; SACHS, E. M. "A Method for Systems Design Using Precedence Relationships: An Application to Automotive Brake Systems" MIT 1990.

BLANCHARD B.S., FABRYCKY W.J. “Systems Engineering and Analysis” Prentice Hall, NJ, 1981.

BLANCHARD, B.S “System Engineering Management” John Wiley & Sons, 2003. 514p.

BOEING – THE ROLE OF HUMAN FACTORS IN IMPROVING AVIATION SAFETY. Disponível em < http://www.boeing.com/commercial/aeromagazine/aero_08/human_textonly .html >. Acesso em 11 de Dezembro de 2009.

BRALLA, J.G. “Design for Excellence” McGraw-Hill, 1996. 326p.

BROWNING, T.R. “Use of Dependency Structure Matrices for Product Development Cycle Time Reduction” Proceedings of the 5th International Conference on Concurrent Engineering: Research and Applications, Tokyo, Japan, 1998.

BROWNING, T.R.; EPPINGER, S.D. “Modeling the Impact of Process Architecture on Cost and Schedule Risk in Product Development” MIT, Sloan School of Management, Working Paper, 2000.

BROWNING, T.R. “Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and New Directions” IEEE Transactions on Engineering Management, Vol. 48, n.3, 2001.

BROWNING, T.R. “Process Integration Using the Design Structure Matrix” Systems Engineering Wiley Periodicals, Vol.5, N.3, p.180-193, 2002.

BUEDE, D.M. “The Engineering Design of Systems Models and Methods” Wiley-Interscience, 1999. 488p.

CACCIABUE, P.C. “A Methodology of Human Factors Analysis for Systems Engineering: Theory and Applications” IEEE Trasactions on Systems, Man and Cybernetics, v.27, n.3, 1997.

CHASE, R.B.; JACOBS, F.R.; AQUILANO, N.J. “Administração da Produção e Operações” McGraw-Hill, 2006. 602p.

CHO, S.H. “An Integrated Method for Managing Complex Engineering Projects Using the Design Structure Matrix and Advanced Simulation” 2001. 125p. Master of Science, Department of Mechanical Engineering, MIT.

DANILOVIC, M.; SANDKULL, B. “The Use of Dependence Structure Matrix and Domain Mapping Matrix in Managing Uncertainty in Multiple Project Situations” International Journal of Project Management, 2004. p. 193-203.

DANILOVIC, M.; BROWNING, T.R. “Managing Complex Product Development Projects with Design Structure Matrices and Domain Mapping Matrices” International Journal of Project Management, p.300-314, 2006.

DEDINI, F.G. “Metodologia e Sistemática de Projeto” apostila IM-136 Unicamp 2007.

DONG, Q.; Whitney, D.E. “Designing a Requirement Driven Product Development Process” MIT, Engineering Systems Division, Working Paper Series, 2001.

DONG, Q. “Predicting and Managing System Interactions at Early Phase of the Product Development Process” 2002. 296p. Doctor of Philosophy in Mechanical Engineering, Massachusetts Institute of Technology.

EPPINGER, S.D. “Managing Complex Systems Development Projects” Massachusetts Institute of Technology, Sloan School of Management, Engineering Systems Division, 2002.

EPPINGER, S.D. “Three Concurrent Engineering Problems in Product Development” MIT Cambridge, MA. 1995.

EPPINGER, S.D. et. al. “A Model-Based Method for Organizing Tasks in Product Development” Research in Engineering Design, 1993. 21p.

EPPINGER, S.D.; SALMINEN, V. “Patterns of Product Development Interactions” International Conference on Engineering Design (ICED) Glasgow, Scotland, 2001.

ERICSON II, C.A. “Hazard Analysis Techniques for System Safety” Wiley-Interscience, 2005. 499p.

FAA REGULATORY AND GUIDANCE LIBRARY. Disponível em < http://rgl.faa.gov >. Acesso em 10 de Janeiro de 2009.

FAR 25.1309 “Airworthiness Standards: Transport Category Airplanes, Sub-Part F: Equipment” Code of Federal Regulations - Federal Aviation Administration (FAA), 2007.

FEOFILOFF, P.; KOHAYAKAWA, Y.; WAKABAYASHI, Y. “Uma Introdução Sucinta a Teoria dos Grafos” Apostila do Instituto de Matematica e Estatistica da USP, 2005.

GEBALA, D.A.; EPPINGER, S.D. “Methods for Analysing Design Procedures” Massachusetts Institute of Technology, 1991, 34p.

GIUDICE, F.; LA ROSA, G.; RISITANO, A. “Product Design for the Environment: a Life Cycle Approach” CRC Press, 2006. 520p.

GOLDT, S.C. “Implementing Cycle Time Reduction in Product Development” 1995. 72p. Master of Science in Management & Mechanical Engineering, Massachusetts Institute of Technology.

GROSE David L., "Reengineering the Aircraft Design Process", Proceedings of the Fifth AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Panama City Beach, FL, 1994.

GUIA PMBOK “Conjunto de Conhecimentos em Gerenciamento de Projetos” Project Management Institute, 3ª edição, 2004. 405p.

HASSON, J.; CROTTY, D. “Boeing’s Safety Assessment Processes for Commercial Airplane Designs” 16th Digital Avionics Systems Conference AIAA/IEEE 1997.

HAUGAN, G.T. “Effective Work Breakdown Structures” Management Concepts, 2001. 120p.

HITCHINS, D. “Systems Methodology” Salisbury, UK. 2003.

HELO, P.T. “Product Configuration Analysis with Design Structure Matrix” Industrial Management & Data Systems, Vol. 106 No. 7, 2006, p. 997-1011.

HOFFMEISTER, A.D. “Sistematização do Processo de Planejamento de Projetos: Definição e Sequenciamento das Atividades para o Desenvolvimento de Produtos Industriais” 2003. 135p. Dissertação de Mestrado, Departamento de Engenharia Mecânica, Universidade Federal de Santa Catarina (UFSC).

HONOUR, E. “Systems Engineering and Complexity” Proceedings of the “Conference on Systems Engineering Research”, Los Angeles, CA 2006.

HUANG, G.Q. “Design for X - Concurrent Engineering Imperatives” Springer Press, 1996. 508p.

INCOSE “Systems Engineering Handbook” Release 1.0 1998.

IPCC: SPECIAL REPORTS ON CLIMATE CHANGE. Disponível em < http://www.grida.no//publications/other/ipcc%5Fsr/?src=/climate/ipcc/aviation/133.htm>. Acesso em 11 de janeiro de 2010.

KNOPF, G.K.; GUPTA, S.M.; LAMBERT, A.J.D. “Environment Conscious Manufacturing” CRC Press, 2007. 560p.

LEVESON, N. “A New Approach to System Safety Engineering” Aeronautics and Astronautics, Massachusetts Institute of Technology. 2002. 382p.

LLOYD, E.; TYE, W. “Systematic Safety” Civil Aviation Authority UK, 1998. 159p.

MAHESWARY, J.U.; VARGHESE, K. “Product Scheduling Using Dependency Structure Matrix” International Journal of Project Management, p.223-230, 2004.

MANZIONE, L. “Estudo de Métodos de Planejamento do Processo de Projeto de Edificios” 2006. 267p. Dissertação de Mestrado, Departamento de Engenharia de Construção Civil, Universidade de São Paulo (POLI-USP).

McCORD, K.R.; EPPINGER, S.D. “Managing the Integration Problem in Concurrent Engineering” MIT, Sloan School of Management, Working Paper, 1993.

McINTYRE, G.R. “The Application of System Safety Engineering and Management Techniques at the US FAA” Safety Science, p. 325–335, 2002.

MIL-STD-882D “Standard Practice for System Safety” US Department of Defense (DoD), 2000. 32p.

MUSGRAVE, G.; LARSEN, A.; SGOBBA, T. “Safety Design for Space Systems” Butterworth-Heinemann, 2009. 992p.

NASA “Systems Engineering Handbook” National Aeronautics and Space Administration, 1995. 164p.

OSBORNE, S.M. “Product Development Cycle Time Characterization Through Modeling of Process Iteration” 1993. 84p. Master of Science in Management and Engineering, Massachusetts Institute of Technology (MIT).

PARK, J.Y.; PARK, Y.W. “Model-Based Concurrent Systems Design for Safety” Concurrent Engineering: Research and Applications, p.287-294, 2004.

PERALTA, A.C. “Um Modelo do Processo de Projeto Baseado na Engenharia simultânea em Empresas Construtoras Incorporadoras de Pequeno Porte” 2002. 143p. Dissertação de Mestrado, Programa de Pós-Gradução em Engenharia de Produção, Universidade Federal de Santa Catarina (UFSC).

PIERONE, E.; NAVEIRO, R.M. “Integrating Project Management, Concurrent Engineering and Engineering Design to Improve Ship Design” Third International Conference on Production Research Americas (ICPR), 2006. 13p.

PIMMLER, T.U.; EPPINGER, S.D. “Integration Analysis of Product Decompositions” ASME Design Theory and Methodology Conference, Minneapolis, MN, 1994. 10p.

PRASAD, B. “Concurrent Engineering Fundamentals” Prentice Hall, 1996. 478p.

RODRIGUEZ, M.A. “Coordenação Técnica de Projetos: Caracterização e Subsídios para sua Aplicação na Gestão do Processo de Projeto de Edificações” 2005. 186p. Tese de Doutorado - Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina (UFSC).

ROLAND, H.E.; MORIARTY, B. “System Safety Engineering and Management” Wiley-Interscience, 1990. 382p.

SAE ARP-4761 “Guidelines and Methods for Conducting the Safety Assessment Process” SAE International, 1996.

SAE ARP-4754 “Certification Considerations for Higly-Integrated or Complex Aircraft Systems” SAE International, 1996.

SAFOUTIN, M.J.; SMITH, R.P. “The Iterative Component of Design” Proceedings of the IEMC, 1996.

SALZMANN, D.J.C; VAN DER TEMPEL, J. “A Safety-Based Design Philosophy for the Ampelman” Proceedings of the EWEC (“European Wind Energy Association”), 2006.

SEQUEIRA, M.W. “Use of the Design Structure Matrix in the Improvement of an Automobile Development Process” 1991. 80p. Master of Science in Materials Science and Engineering, Massachusetts Institute of Technology (MIT).

SERRA, P.R. “Análise de Confiabilidade e Segurança de Sistemas Aeronáuticos” Apostila do Curso de Análise de Confiabilidade e Segurança de Sistemas (ITA) 2006.

SINGH, N. “Systems Approach to Computer-Integrated Design and Manufacturing” John Wiley & Sons, 1996. 643p.

SMITH, R.P.; EPPINGER, S.D. “Characteristics and Models of Iteration in Engineering Design” Proceedings of the “International Conference on Engineering Design”, The Hague 1993.

SMITH, R.P.; EPPINGER, S.D. “A Predictive Model of Sequential Iteration in Engineering Design” Management Science / Vol. 43, N. 8, p.1104-1120, 2007.

STEWARD, D.V. “The Design Structure System – A Method for Managing the Design of Complex Systems” IEEE Transactions on Engineering Management, p.71-74. 1981.

STOREY, N. “Design for Safety” Proceeding of the 7th Safety-Critical Systems Symposium, Huntingon, UK, p.1-25, 1999.

TANG, D.; ZHANG, G.; DAI, S. “Design as Integration of Axiomatic Design and Design Structure Matrix” Journal of Robotics and Computer-Integrated Manufacturing, Vol.25, 2009, p.610-619.

ULRICH, K.T.; EPPINGER, S.D. “Product Design and Development” McGraw-Hill, 4th edition, 2008. 368p.

VALERIANO, D. “Gerência em Projetos” Makron Books, 1998. 438p.

WARFIELD, J.N. “Binary Matrices in System Modeling” IEEE Transactions on Systems, Man and Cybernetics, p.441-449. 1973.

WELLS, A.T; RODRIGUES, C.C. "Commercial Aviation Safety” McGraw-Hill Professional 2006. 475p.

YASSINE, A.; FALKENBURG, D.; CHELST, K. “Engineering Design Management: An Information Structure Approach” International Journal of Production Research, 1999. 20p.

YASSINE, A.; BRAHA, D. “Complex Concurrent Engineering and the Design Structure Matrix Method” Concurrent Engineering: Research and Applications, p.165-176. 2003.

YASSINE, A. “An Introduction to Modeling and Analyzing Complex Product Development Process Using the Design Structure Matrix Method” Product Development Research Laboratory, University of Illinois at Urbana-Champaign, 2004.