



UNIVERSIDADE FEDERAL DE PERNAMBUCO - UFPE
CENTRO DE INFORMÁTICA – CIN
GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO



UMA ANÁLISE SISTÊMICA DOS PROBLEMAS ENFRENTADOS POR
EMPRESAS DE SOFTWARE DURANTE O PROCESSO DE ENGENHARIA DE
REQUISITOS

TRABALHO DE GRADUAÇÃO

Daniel de Andrade Penaforte

RECIFE, JANEIRO DE 2008



UNIVERSIDADE FEDERAL DE PERNAMBUCO - UFPE
CENTRO DE INFORMÁTICA – CIN
GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO



DANIEL DE ANDRADE PENAFORTE

UMA ANÁLISE SISTÊMICA DOS PROBLEMAS ENFRENTADOS POR
EMPRESAS DE SOFTWARE DURANTE O PROCESSO DE ENGENHARIA DE
REQUISITOS

Este trabalho foi apresentado à graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação, sob orientação da Profa. Dra. Carina Frota Alves e co-orientação do Prof. Dr. Antônio Carlos Valença.

RECIFE, JANEIRO DE 2008

“A vida é a arte do encontro, embora haja tanto desencontro pela vida.”

(Vinícius de Moraes)

RESUMO

Durante todo o ciclo de desenvolvimento de software, as organizações se vêem diante de problemas das mais diversas naturezas, sendo muitos deles relacionados à Engenharia de Requisitos. Os problemas na etapa de requisitos são os mais caros de se corrigir nas fases subseqüentes do desenvolvimento de software, tendo também influência direta no atraso do prazo de entrega dos projetos e na satisfação dos clientes com o produto final. Pesquisas apontam fatores como clareza dos requisitos, especificação e requisitos incompletos e mudanças nos requisitos como fatores críticos de sucesso ou falhas nos projetos de software. Tais fatores estão associados, muitas vezes, à falta de um processo de Engenharia de Requisitos definido, onde há certa informalidade na realização das atividades com os requisitos, e a falhas no gerenciamento dos requisitos. Este estudo apresenta uma análise desses problemas críticos enfrentados por empresas de software durante o processo de Engenharia de Requisitos, sob a ótica da Dinâmica de Sistemas. Justamente para ampliar e melhorar o espectro de visão das pessoas com relação a estes problemas, a Dinâmica de Sistemas foi escolhida para embasar este estudo, pois é possível apresentar circuitos de causa-efeito representando o comportamento do sistema como um todo. A partir da identificação dos circuitos é possível realizar leituras da situação tendo noção de como os fatores exercem influência entre si, e de como estas influências afetam o comportamento do sistema de uma forma geral, podendo gerar *insights* e reflexões sobre a situação que levem ao aprendizado e que dêem suporte à tomada de decisão para a resolução dos problemas. Portanto, nesta análise trazemos essa contribuição da Dinâmica de Sistemas, refletindo sobre a necessidade das empresas de software terem um processo de Engenharia de Requisitos bem definido e adequado à sua realidade, como também sobre as dificuldades em se gerenciar os requisitos.

Palavras chave: Engenharia de Requisitos, Problemas no processo de Engenharia de Requisitos, Dinâmica de Sistemas.

AGRADECIMENTOS

A Carina Frota Alves, minha orientadora, pela destreza durante o período de orientação, por mostrar-se sempre prestativa, pela oportunidade, apoio e confiança neste trabalho de conclusão de curso.

A Antônio Carlos Valença, meu co-orientador, a quem devo, além de grandes aprendizados sobre a vida, todo o conhecimento sobre Dinâmica de Sistemas e Pensamento Sistêmico utilizado para a realização desta análise.

Aos meus familiares que, apesar da distância, motivam-me e prestam o apoio necessário para que eu consiga vencer os desafios impostos pela vida e atingir os meus objetivos pessoais.

Aos meus queridos amigos e companheiros de trabalho que prestaram apoio e suporte incondicional durante a realização deste trabalho: César Delmas, Daniel Julião, Guilherme Carvalho e Hector Paulo Oliveira.

Ao querido companheiro André Felipe Santana que em algumas conversas me remeteu a importantes reflexões importantes para o encaminhamento deste trabalho.

Aos meus companheiros de graduação, com os quais tive a oportunidade de compartilhar 4 anos e meio de aprendizado, esforço, dedicação, cumplicidade, alegrias, churrascos, confraternizações e, por muitas vezes, noites em claro para a conclusão de projetos.

A Deus por me permitir este momento.

ASSINATURAS

Este Trabalho de Graduação é resultado dos esforços do aluno Daniel de Andrade Penaforte, sob a orientação da professora Carina Frota Alves, sob o título de “Uma Análise Sistêmica dos Problemas Enfrentados por Empresas de Software Durante o Processo de Engenharia de Requisitos”. Todos abaixo estão de acordo com o conteúdo deste documento e os resultados deste Trabalho de Graduação.

Carina Frota Alves (orientadora)

Antônio Carlos Valença (co-orientador)

Daniel de Andrade Penaforte (aluno)

ÍNDICE

1	INTRODUÇÃO	11
1.1	Motivação e Contexto.....	11
1.2	Objetivos.....	12
1.3	Estrutura do Documento.....	12
2	DINÂMICA DE SISTEMAS	14
2.1	Contextualização Histórica	14
2.2	As Noções de Sistema.....	15
2.3	Os Elementos da Dinâmica de Sistemas.....	17
2.4	A Modelagem pela Dinâmica de Sistemas.....	19
2.4.1	Modelagem Soft ou Mental	21
2.5	Arquétipos Sistêmicos.....	22
2.5.1	Balanceamento com Retardo.....	23
2.5.2	Limite ao Crescimento	24
2.5.3	Sucesso para os Bem-Sucedidos.....	25
2.5.4	Solução Quebra-Galho	25
2.5.5	Transferência de Responsabilidade.....	26
2.5.6	Deriva de Metas.....	27
2.5.7	Escalada.....	27
2.5.8	Crescimento e Subinvestimento	28
2.5.9	Tragédia do Fator Comum.....	29
2.5.10	Adversários Acidentais.....	30
2.5.11	Princípio da Atratividade	31
3	UMA VISÃO GERAL DA ENGENHARIA DE REQUISITOS.....	32
3.1	O Processo de Engenharia de Requisitos.....	33
3.1.1	Elicitação de Requisitos	34
3.1.2	Análise e Negociação de Requisitos.....	36
3.1.3	Documentação de Requisitos	37
3.1.4	Validação de Requisitos	38
3.1.5	Gerenciamento de Requisitos	38
4	UMA PROPOSTA DE ANÁLISE SISTÊMICA DOS PROBLEMAS ENFRENTADOS POR EMPRESAS DE SOFTWARE DURANTE O PROCESSO DE REQUISITOS	40
4.1	Problemas Típicos Enfrentados Durante o Processo de Engenharia de Requisitos	41

4.2	Adaptação da Modelagem <i>Soft</i> da Dinâmica de Sistemas.....	43
4.3	Análise do Problema 1 - Falta de um Processo de Engenharia de Requisitos Definido.....	45
4.4	Análise do Problema 2 - Dificuldade em Gerenciar Requisitos	53
5	CONCLUSÕES E TRABALHOS FUTUROS	58
5.1	Principais Dificuldades e Limitações Encontradas.....	59
5.2	Oportunidades para Trabalhos Futuros	59
6	REFERÊNCIAS BIBLIOGRÁFICAS.....	60

ÍNDICE DE QUADROS

TABELA 1 PRINCIPAIS DIFERENÇAS ENTRE AS ABORDAGENS SOFT E HARD DA DINÂMICA DE SISTEMAS .	20
TABELA 2 PRINCIPAIS PROBLEMAS COM REQUISITOS IDENTIFICADOS NO ESTUDO DE HALL E BEECHAM	41
TABELA 3 PRINCIPAIS PROBLEMAS NO PROCESSO DE ENGENHARIA DE REQUISITOS IDENTIFICADOS NO ESTUDO DE ALVES E PEREIRA.....	42

ÍNDICE DE FIGURAS

FIGURA 1 ESTRUTURA UNIDIRECIONAL DE CAUSA E EFEITO	16
FIGURA 2 DIAGRAMA, CIRCUITO, CICLO DE CAUSALIDADE OU FEEDBACK.....	17
FIGURA 3 RELAÇÃO DE MESMO SENTIDO; (B) RELAÇÃO DE SENTIDO OPOSTO	18
FIGURA 4 RELAÇÃO COM DELAY	18
FIGURA 5 DIAGRAMA DE ENLACES CAUSAIS COM UM CICLO DE BALANCEAMENTO E UM CICLO DE REFORÇO	19
FIGURA 6 DIAGRAMA DE ENLACES CAUSAIS DO ECOSSISTEMA COELHOS E RAPOSAS	22
FIGURA 7 ESTRUTURA GENÉRICA DO ARQUÉTIPO BALANCEAMENTO COM RETARDO	23
FIGURA 8 ESTRUTURA GENÉRICA DO ARQUÉTIPO LIMITE AO CRESCIMENTO	24
FIGURA 9 ESTRUTURA GENÉRICA DO ARQUÉTIPO SUCESSO PARA OS BEM-SUCEDIDOS.....	25
FIGURA 10 ESTRUTURA GENÉRICA DO ARQUÉTIPO SOLUÇÃO QUEBRA-GALHO COM RETARDO	25
FIGURA 11 ESTRUTURA GENÉRICA DO ARQUÉTIPO TRANSFERÊNCIA DE RESPONSABILIDADE	26
FIGURA 12 ESTRUTURA GENÉRICA DO ARQUÉTIPO DERIVA DE METAS	27
FIGURA 13 ESTRUTURA GENÉRICA DO ARQUÉTIPO ESCALADA	27
FIGURA 14 ESTRUTURA GENÉRICA DO ARQUÉTIPO BALANCEAMENTO COM RETARDO	28
FIGURA 15 ESTRUTURA GENÉRICA DO ARQUÉTIPO TRAGÉDIA DO FATOR COMUM.....	29
FIGURA 16 ESTRUTURA GENÉRICA DO ARQUÉTIPO ADVERSÁRIOS ACIDENTAIS	30
FIGURA 17 ESTRUTURA GENÉRICA DO ARQUÉTIPO PRINCÍPIO DA ATRATIVIDADE	31
FIGURA 18 ATIVIDADES DO PROCESSO DE ENGENHARIA DE REQUISITOS NO MODELO ESPIRAL.....	34
FIGURA 19 ETAPAS DO MÉTODO DE ANÁLISE DOS PROBLEMAS COM REQUISITOS	44
FIGURA 20 CICLO DE BALANCEAMENTO DA SOLUÇÃO QUEBRA-GALHO	46
FIGURA 21 CICLO DE REFORÇO DA SOLUÇÃO QUEBRA-GALHO	47
FIGURA 22 ARQUÉTIPO DA SOLUÇÃO QUEBRA-GALHO NA FALTA DE PROCESSO DE ER DEFINIDO....	48
FIGURA 23 CICLO DE REFORÇO DO ARQUÉTIPO DE CRESCIMENTO E SUBINVESTIMENTO	50
FIGURA 24 PRIMEIRO CICLO DE BALANCEAMENTO DO ARQUÉTIPO DE CRESCIMENTO E SUBINVESTIMENTO.....	51
FIGURA 25 SEGUNDO CICLO DE BALANCEAMENTO DO ARQUÉTIPO DE CRESCIMENTO E SUBINVESTIMENTO	52
FIGURA 26 ARQUÉTIPO DE CRESCIMENTO E SUBINVESTIMENTO NA FALTA DE PROCESSO DE ER DEFINIDO.....	53
FIGURA 27 PRIMEIRO CICLO DE REFORÇO DO ARQUÉTIPO DE SUCESSO PARA OS BEM SUCEDIDOS...	55
FIGURA 28 SEGUNDO CICLO DE REFORÇO DO ARQUÉTIPO DE SUCESSO PARA OS BEM SUCEDIDOS..	56
FIGURA 29 ARQUÉTIPO DE SUCESSO PARA OS BEM SUCEDIDOS NA DIFICULDADE EM GERENCIAR REQUISITOS.	57

1 INTRODUÇÃO

Este capítulo introdutório tem por objetivo apresentar a motivação de pesquisa para a realização deste trabalho de graduação, definir os objetivos do trabalho e o que se pode esperar da leitura deste texto. Busca também situar o contexto geral que gerou a motivação do estudo e apresenta a estrutura do documento.

1.1 Motivação e Contexto

Durante todo o ciclo de desenvolvimento de software, as organizações se vêem diante de problemas das mais diversas naturezas. Falhas com a comunicação e dificuldades com a identificação das reais necessidades do cliente são alguns destes e que estão relacionados à fase de Engenharia de Requisitos. Segundo [KOTONYA e SOMMERVILLE, 1997], os problemas nesta etapa são os mais caros de corrigir nas fases subseqüentes do desenvolvimento de software, tendo também influência direta no atraso do prazo de entrega dos projetos. Em algumas situações as dificuldades têm origem em questões humanas e sociais inerentes ao processo, em outros momentos em problemas da própria organização. O fato é que tais problemas persistem no cotidiano das organizações e necessitam de uma análise minuciosa a fim de serem gerenciados e minimizados.

A Dinâmica de Sistemas é um conjunto de ferramentas e métodos que tem por objetivo a análise e o estudo do funcionamento de sistemas que sofrem alteração ao longo do tempo. Sterman [STERMAN, 2000] aponta a Dinâmica de Sistemas como uma ferramenta ideal para a modelagem de sistemas complexos: “[...] *Os modelos formais são utilizados para estruturar o pensamento nas tomadas de decisões*”. Os problemas associados ao processo de Engenharia de Requisitos, enfrentados pelas empresas de software durante o desenvolvimento dos seus produtos, caracterizam uma situação complexa e que pode ser modelada com a Dinâmica de Sistemas.

A literatura oferece pesquisas e artigos que levantam problemas freqüentes relacionados ao processo de requisitos nas empresas de software [HALL, 2002] [ALVES, 2007] [PEREIRA, 2007]. Portanto, é possível realizar um estudo qualitativo destes problemas, identificando fatores críticos de influência sobre os mesmos, e, em seguida, aplicar as ferramentas e métodos fornecidos pela Dinâmica de Sistemas a fim de compreender

sob uma nova ótica a realidade dos problemas enfrentados pelas empresas de software durante o processo de Engenharia de Requisitos.

1.2 Objetivos

Este estudo tem como objetivo fazer um levantamento dos principais problemas enfrentados por empresas de software durante o processo de Engenharia de Requisitos e abordá-los sob a ótica da Dinâmica de Sistemas, com vistas a:

- Aprofundar a compreensão destes problemas;
- Identificar modelos que provoquem insight e auxiliem na tomada de decisão para a minimização dos mesmos.

Para tal, o levantamento dos principais problemas se baseará em pesquisas anteriores de caracterização dos principais problemas enfrentados por empresas durante o processo de Engenharia de Requisitos. Através da identificação de fatores críticos relacionados aos problemas e das influências que estes fatores exercem entre si, procuraremos modelar tais problemas com as ferramentas e métodos que a Dinâmica de Sistemas oferece, apresentando o comportamento e a dinâmica que regem tal influência, bem como discutir alertas e possíveis estratégias a fim de provocar reflexões e, conseqüentemente, gerar aprendizado que possa ajudar numa melhor compreensão dos problemas associados ao processo de Engenharia de Requisitos.

1.3 Estrutura do Documento

A organização dos seguintes capítulos deste estudo é a descrita a seguir.

- Capítulo 2 (Dinâmica de Sistemas): traz uma visão geral sobre Dinâmica de Sistemas, apresentando um breve histórico, seus principais elementos e como se realiza a modelagem *soft* da Dinâmica de Sistemas;
- Capítulo 3 (Uma Visão Geral da Engenharia de Requisitos): apresenta a Engenharia de Requisitos no contexto do desenvolvimento de *software*, apresentando o processo de Engenharia de Requisitos segundo o modelo Espiral e detalhando suas etapas;
- Capítulo 4 (Análise Sistêmica dos Problemas Enfrentados por Empresas de Software durante o Processo de Engenharia de Requisitos): descreve o método

de análise utilizado, um levantamento dos principais problemas relacionados ao processo de engenharia de requisitos nas empresas de software e, por fim, apresenta uma análise de alguns deles sob a ótica da Dinâmica de Sistemas;

- Capítulo 5 (Conclusões e Trabalhos Futuros): apresenta as considerações finais ao estudo, limitações e dificuldades da análise, e apresenta possibilidades de trabalhos futuros decorrentes deste estudo.
- Capítulo 6 (Referências Bibliográficas): traz todas as referências utilizadas no embasamento deste estudo.

2 DINÂMICA DE SISTEMAS

2.1 Contextualização Histórica

Durante a Segunda Guerra mundial, o engenheiro elétrico Jay W. Forrester realizava trabalhos para as Forças Armadas americanas através do desenvolvimento de controladores para armas e radares nos laboratórios do MIT (*Massachusetts Institute of Technology*). Na realização destas atividades, Forrester utilizava a então pioneira tecnologia dos sistemas de controle de *feedback*. Em 1956, Forrester decide abandonar os trabalhos com a engenharia e inicia estudos com a utilização dos conceitos dos sistemas de *feedback* nos problemas administrativos.

Em 1961, através da publicação do livro *Industrial Dynamics* [FORRESTER, 1961], tem-se o início da aplicação da Dinâmica de Sistemas nas ciências administrativas. Neste livro, Forrester fez uma análise da utilização dos sistemas de feedback em processos administrativos, com o intuito de investigar os motivos dos fracassos das corporações. Alguns anos mais tarde, em 1968, Forrester inicia a aplicação deste conhecimento em problemas sociais através de contatos com John Collins, professor visitante para questões urbanas no MIT, culminando, no ano seguinte, no lançamento do livro *Urban Dynamics* [FORRESTER, 1969]. Daí em diante, Forrester direcionou-se cada vez mais ao estudo da dinâmica de sistemas sociais e econômicos globais através da realização de projetos no MIT. Em 1973, ele lança mais um livro, o *World Dynamics* [FORRESTER, 1973]. Neste livro, Forrester apresenta modelos de simulação que mostram como aumentos exponenciais na população e no consumo de recursos naturais conduzem a crises de poluição e fome, a menos que se tome alguma medida, preferencialmente nas políticas econômicas. Todo este estudo de Forrester teve um grande impacto no mundo acadêmico, pois, até então, nenhum outro modelo de Dinâmica de Sistemas tivera sido estudado de forma tão minuciosa.

Desde então, a aplicação dos conceitos da Dinâmica de Sistemas tem crescido de forma substancial, dando suporte para análise de problemas em diferentes contextos, como ecologia, educação, economia e gestão empresarial.

2.2 As Noções de Sistema

A palavra sistema tem sua origem em “*systema*”, derivada de “*syn*” e “*hystema*” que querem dizer “conjuntamente” e “estabelecer”, respectivamente. Portanto, partindo inicialmente da etimologia da palavra um sistema quer dizer, literalmente, “estabelecer conjuntos”. Para que possamos entender a definição de sistemas, é necessário possuir uma visão ampla de complexidade, finalidade e interdependência entre os elementos analisados [GORDON, 1969].

Existem definições diversas sobre sistemas na literatura. Segundo Ackoff [ACKOFF, 1974], “*Um sistema é um conjunto de dois ou mais elementos de qualquer tipo; por exemplo, conceitos (como no sistema de números), idéias (como em sistemas filosóficos), objetos (como num sistema telefônico ou num organismo) ou pessoas (como numa sociedade)*”. A fim de distinguir um sistema de um agrupamento qualquer de elementos, o mesmo autor defende ainda que um sistema apresenta as seguintes características:

- a) As propriedades ou o comportamento de cada elemento têm um efeito nas propriedades ou comportamentos do sistema como um todo;
- b) Nenhuma parte tem efeito independente sobre o todo, e cada uma é afetada no mínimo por outra;
- c) Todos os subgrupos de elementos têm efeitos sobre o comportamento do todo e nenhum possui efeito independente sobre ele. Ou seja, as partes de um sistema estão conectadas de tal modo que subgrupos ou subsistemas e partes independentes não podem ser consideradas isoladamente.

Portanto, para se ter um entendimento do comportamento de um sistema, é preciso analisá-lo na sua totalidade. Isto vai de encontro ao processo analítico tradicional, onde (i) separa-se o objeto das partes que o constituem, (ii) tenta-se compreender as partes constituintes separadamente e (iii) reúne-se o conhecimento obtido das partes para a compreensão do todo. Na verdade, o método sistêmico atua na direção contrária do processo analítico tradicional, sendo constituído de três procedimentos gerais [ACKOFF, 1974]:

- a) Identificar um todo que contempla aspecto ou fator de interesse, observando suas conexões com o sistema maior ou outros sistemas que constituem o ambiente;
- b) Identificar as partes e as interações dentro do todo;

- c) Explicar o comportamento e as propriedades do todo, do qual o fator de interesse é parte. Ou seja, compreender o comportamento do sistema no contexto do padrão de interações que ele apresenta.

Os sistemas podem ser classificados de várias formas diferentes a partir das suas características. Uma das classificações mais clássicas da literatura é a de Forrester [FORRESTER, 1976], a qual classifica os sistemas como sendo de ciclo aberto (enlaces abertos) ou de ciclo de *feedback* ou fechado (enlaces fechados).

Os sistemas de ciclo aberto são caracterizados por relações de causa-efeito lineares, onde, apesar da causa implicar num efeito, este por sua vez não realimenta a causa geradora, portanto não existe *feedback* [FERNANDES, 2003]. Trata-se de uma estrutura unidirecional de causa e efeito, onde há saídas que correspondem a entradas, porém as saídas estão isoladas das entradas e não exercem influência sobre elas. O pressuposto nesse tipo de estrutura é que a informação sobre o estado do sistema orienta uma decisão, que leva a uma ação, que leva a um resultado, como mostra a figura 1.

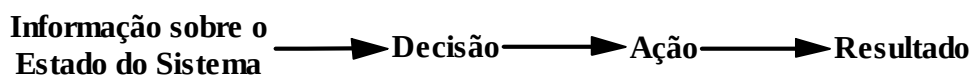


Figura 1 Estrutura unidirecional de causa e efeito. Fonte: [FERNANDES, 2003]

Um sistema de *feedback*, por sua vez, sofre influência do seu próprio comportamento passado, de modo que as saídas influenciam as entradas, estabelecendo um ciclo fechado. Aqui, causa e efeito se confundem, pois qualquer influência de um componente do sistema é, ao mesmo tempo, causa e efeito, ou seja, a causalidade não tem um único sentido [FERNANDES, 2003]. Desta forma, este tipo de estrutura caracteriza um sistema dinâmico, onde o resultado passado influencia o resultado futuro, como mostra a figura 2.



Figura 2 Diagrama, circuito, ciclo de causalidade ou feedback. Fonte: [FERNANDES, 2003]

Checkland [CHECKLAND, 1994] apresenta um conceito compartilhado das diversas visões sobre sistemas, sintetizando um sistema como “[...] *um todo adaptativo; uma entidade com propriedades emergentes, estruturada em níveis e processos de comunicação e controle que permitem a adaptação a uma ambiente de mudança*”. É interessante notar a característica adaptativa dos sistemas nesta definição, que se apresenta tanto na interação do mesmo com o ambiente externo, sujeito a mudanças, como internamente na sua estrutura de níveis.

2.3 Os Elementos da Dinâmica de Sistemas

A Dinâmica de Sistemas possui um conjunto de elementos e conceitos fundamentais necessários à modelagem de sistemas complexos. Segundo Valença [VALENÇA, 1999], o entendimento dos seguintes conceitos são essenciais à construção de diagramas ou *loops* de causalidade:

- a) **Variáveis ou elementos do sistema:** são as entidades ou fatores relevantes do sistema, passíveis de quantificação e de variação ao longo do tempo.
- b) **Relações:** setas que indicam as formas de relacionamento entre variáveis e a direção ou tendência da variável. Graficamente, os sinais “+” ou “m” (mesmo sentido) indicam que a variação no fator causador gera uma variação no mesmo sentido no fator que recebe o efeito. Já os sinais “-” ou “o” (sentido oposto) indicam que a variação no fator causador gera uma variação no sentido contrário no fator que recebe o efeito.

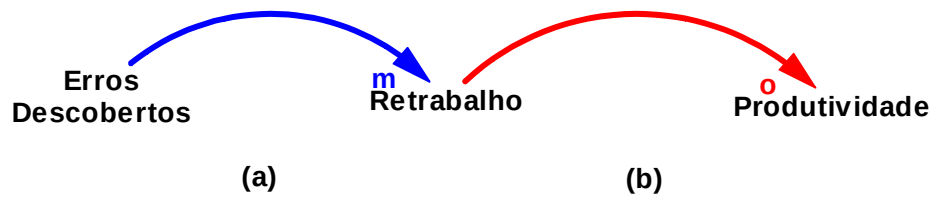


Figura 3 Relação de mesmo sentido; (b) Relação de sentido oposto. Fonte: Adaptado de [MONTENEGRO, 2006]

Na figura 3.a podemos identificar uma relação de mesmo sentido entre as variáveis “Erros Descobertos” e “Retrabalho”, de modo que quanto maior a quantidade de erros descobertos num projeto de software, por exemplo, maior será o retrabalho da equipe. Já em 3.b, a relação entre as variáveis “Retrabalho” e “Produtividade” é de sentido oposto, ou seja, quanto maior o retrabalho da equipe, menor sua produtividade.

- c) **Atrasos, retardos ou *delay***: efeitos que somente são sensíveis após um tempo de espera (ou tempo de retardo), e que são simbolizados por uma ou duas barras paralelas cortando uma seta.

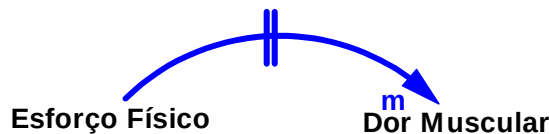


Figura 4 Relação com delay. Fonte: Adaptado de [ANDRADE, 2006]

A figura 4 mostra uma relação que apresenta tempo de retardo. Neste exemplo, quanto maior for o esforço físico de um atleta, maior serão suas dores musculares, porém este efeito não é imediato, ocorre após certo tempo.

- d) **Enlace, circuito, ciclo ou *feedback***: conjunto circular de causas em que uma perturbação num elemento causa uma variação nele próprio como resposta. Podem ser classificados como “balanceamento, equilíbrio ou *feedback* negativo” ou “reforço ou *feedback* positivo”. No primeiro caso, o efeito resultante sobre o próprio loop é no mesmo sentido. Graficamente isto pode ser identificado quando há um número par de relacionamentos de sentido oposto no loop. Já no segundo caso, o efeito resultante sobre o mesmo é no sentido oposto. Graficamente pode ser identificado quando há um número ímpar de relacionamentos de sentido oposto no loop.

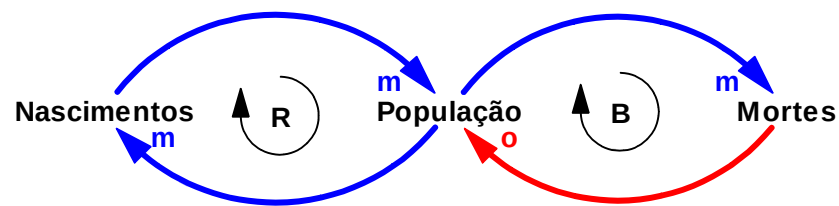


Figura 5 Diagrama de enlances causais com um ciclo de balanceamento e um ciclo de reforço.
Fonte: Adaptado de [MONTENEGRO, 2006]

No exemplo da figura 5, o diagrama de causalidade representa a dinâmica do tamanho da população. No ciclo de reforço, quanto maior o nascimento de pessoas, maior a população, que, por sua vez, causa um efeito de mesmo sentido na natalidade. Trata-se de um “reforço virtuoso”. No ciclo de balanceamento, a mortalidade dos indivíduos está limitando o crescimento da população. A dinâmica apresenta-se da seguinte forma: quanto maior a população, maior a quantidade de mortes e, quanto maior a mortalidade de pessoas, menor a população.

Segundo Peter Senge [SENGE, 2001], num sistema com *feedback* de reforço uma pequena ação pode gerar grandes conseqüências, para melhor ou para pior; trata-se do conhecido “efeito bola-de-neve”. Já os *feedbacks* de balanceamento são responsáveis pela limitação do crescimento infinito de qualquer sistema e, muitas vezes, são caracterizados por serem direcionados para algum objetivo ou meta.

2.4 A Modelagem pela Dinâmica de Sistemas

Um modelo pode ser visto como uma representação externa e explícita de parte da realidade percebida pela pessoa que deseja usar aquele modelo para entender, mudar, gerenciar e controlar parte daquela realidade [RADZICKI, 1997]. Complementando esta definição, Pidd [PIDD, 1997] diz que um sistema pode ser visto e modelado de diferentes formas: “[...] Podem existir vários modelos para um mesmo sistema, dependendo dos olhos de quem o analisa”. Desta forma, percepções, habilidades e valores de julgamento podem influenciar na apresentação de modelos diferentes para o mesmo objetivo de investigação.

Segundo Mohapatra [MOHAPATRA, 1994], os objetivos para a construção de um modelo são: (i) entender como um sistema real trabalha; (ii) conhecer os fatores que influenciam o comportamento deste sistema; (iii) explorar os efeitos de vários cenários e políticas no comportamento do sistema; e (iv) ter a capacidade de compartilhar com outros

participantes os seus resultados. Em outras palavras, trata-se de entender o comportamento do sistema na sua totalidade, identificando fatores de influência sobre o mesmo através da exploração de cenários, e compartilhar o aprendizado com a construção do modelo.

Na modelagem com Dinâmica de Sistemas, há duas abordagens que podem ser empregadas para criação do modelo que caracterize um sistema: a *soft* ou mental e a *hard* ou formal. Pidd [PIDD, 1997] apresenta um estudo comparativo entre estas duas abordagens, diferenciando-as de acordo com algumas características. As principais diferenças entre as mesmas encontram-se sumarizadas na tabela comparativa abaixo:

Tabela 1 Principais diferenças entre as abordagens soft e hard da Dinâmica de Sistemas.
Fonte: Adaptado de [PIDD, 1997]

	<i>Soft</i> (mental)	<i>Hard</i> (formal)
Definição do problema	Vista como problemática, pluralista	Vista como direta, unitária
O modelo	Uma forma de gerar debates e <i>insight</i> a respeito do mundo real	Uma representação do mundo real
Objetivo	<i>Insight</i> e aprendizagem	Soluções e otimizações
Resultado	Progresso pela aprendizagem	Um produto ou recomendação

Na modelagem *hard*, acredita-se que todo problema pode ser definido e, portanto, a sua respectiva solução poderá ser encontrada. Já na modelagem *soft*, tem-se a percepção de uma situação problemática, situação abstrata e extremamente complexa onde não é possível definir o problema com precisão [ANDRADE, 2006]. As duas abordagens não são excludentes, de modo que a *soft* pode ser utilizada na fase inicial de conceituação e definição do problema, enquanto a *hard* pode ser utilizada em estágios posteriores [FERNANDES, 2003].

O escopo desta pesquisa enfoca na construção de modelos sistêmicos de situações problemáticas típicas enfrentadas por empresas de software durante o processo de requisitos.

O nosso objetivo principal é gerar aprendizado e *insight* sobre a realidade através da discussão dos modelos aqui apresentados. Diante disto, a modelagem *soft* mostra-se mais adequada à nossa proposta e, portanto, abordaremos a mesma de forma mais detalhada na próxima seção.

2.4.1 Modelagem Soft ou Mental

Segundo Peter Senge [SENGE, 2001], o desenvolvimento metodológico para a abordagem Soft da dinâmica de sistemas contempla os seguintes passos:

- 1) **Definição de uma Situação Complexa de Interesse:** onde é buscada uma situação complexa importante para a organização, ou para o conjunto de indivíduos interessados;
- 2) **Apresentação da História dos Eventos:** onde o objetivo é assinalar eventos relevantes relacionados com a situação ao longo do período considerado;
- 3) **Identificação de Fatores Chaves:** onde é necessário identificar que fatores ou variáveis podem ser assinalados como chave para a compreensão da situação;
- 4) **Observação do Comportamento:** que consiste em observar padrões de comportamento, através de gráficos ou curvas, de alguns elementos chaves ao longo do tempo;
- 5) **Identificação das Influências:** onde o objetivo é identificar as relações causais entre os fatores, a partir da comparação das curvas, hipóteses preliminares e intuições a respeito das influências recíprocas;
- 6) **Identificação de Modelos Mentais:** onde o objetivo é identificar os modelos mentais presentes, ou seja, levantar crenças e pressupostos que os atores envolvidos mantêm em suas formas de enxergar a realidade, e que estão ajudando a criar a situação em questão.

A partir da aplicação desta abordagem da Dinâmica de Sistemas, obtêm-se como resultado diagramas de enlaces causais. Tais diagramas são mapas cognitivos sobre a situação-problema em questão. Eles são úteis para o entendimento compartilhado do funcionamento do sistema, e sua estrutura determina o comportamento do sistema [GOODMAN, 1989]. A figura 6 ilustra um diagrama de enlaces causais resultante da utilização da modelagem *soft* da Dinâmica de Sistemas do Ecossistema de Coelhos e Raposas [VALENÇA, 1995].

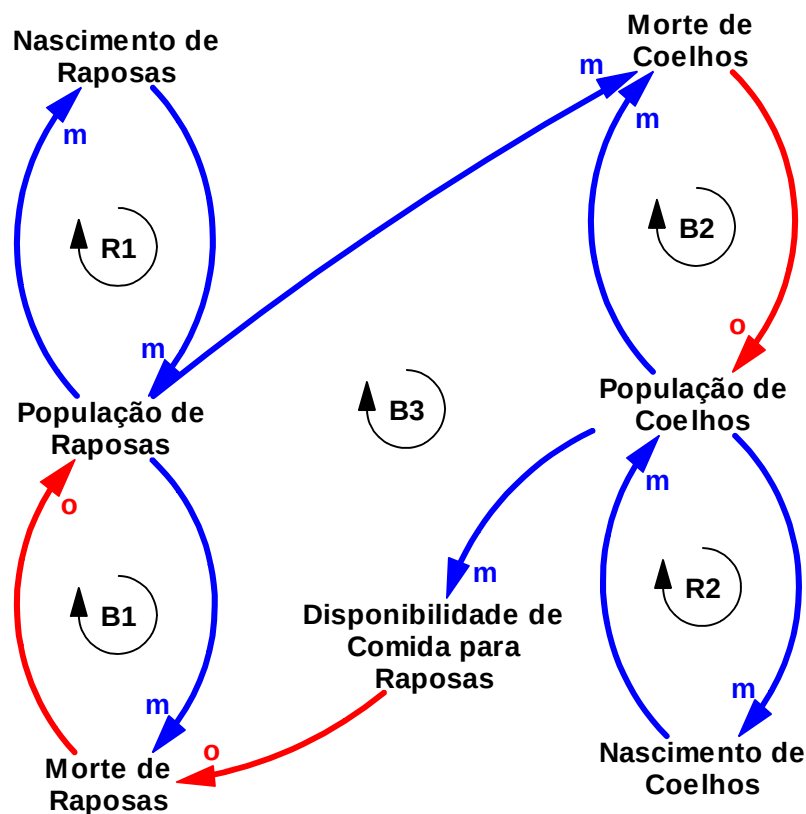


Figura 6 Diagrama de enlances causais do Ecossistema Coelhos e Raposas. Fonte: [VALENÇA, 1995]

Nesta figura, identificamos cinco ciclos ou circuitos causais que representam a dinâmica do ecossistema de coelhos e raposas. Nos circuitos R1 e B1, enquanto a *População de Raposas* é reforçada pelo *Nascimento de Raposas*, a *Morte de Raposas* limita o crescimento da *População*. Os circuitos R2 e B2 apresentam a mesma dinâmica, sendo para a *População de Coelhos*. No circuito B3, um aumento da *População de Raposas* causa um maior número de *Morte de Coelhos*, que diminui a *População de Raposas*, diminuindo também a *Disponibilidade de Comida para as Raposas* e, conseqüentemente, aumentando a *Morte de Raposas*, que leva a diminuição da *População de Raposas*. A partir daí inicia-se um novo ciclo que apresentará o comportamento inverso, em função disto ele é classificado como um circuito de balanceamento.

2.5 Arquétipos Sistêmicos

A palavra arquétipos vem do grego *archetypos*, que significa “primeiro da sua espécie”. Arquétipos sistêmicos são estruturas sistêmicas genéricas compostas por relações de causa-efeito cíclicas que se repetem em diferentes contextos, geralmente, sem que as pessoas

tenham consciência de seus efeitos na situação em questão. Por terem um comportamento previsível, a revelação destas estruturas pode inspirar estratégias de ação eficazes para as situações problemáticas que elas representam [SENGE, 2001].

Os arquétipos são padrões recorrentes de estruturas que estão associados a um comportamento determinado e que exibe as características de um sistema complexo. Representam um *insight* sistêmico ou um *insight* sobre um padrão de comportamento. Ao longo de 50 anos de pesquisa, foram identificados 11 arquétipos, descritos abaixo através da sua estrutura básica e do seu comportamento [VALENÇA, 1999].

2.5.1 Balanceamento com Retardo

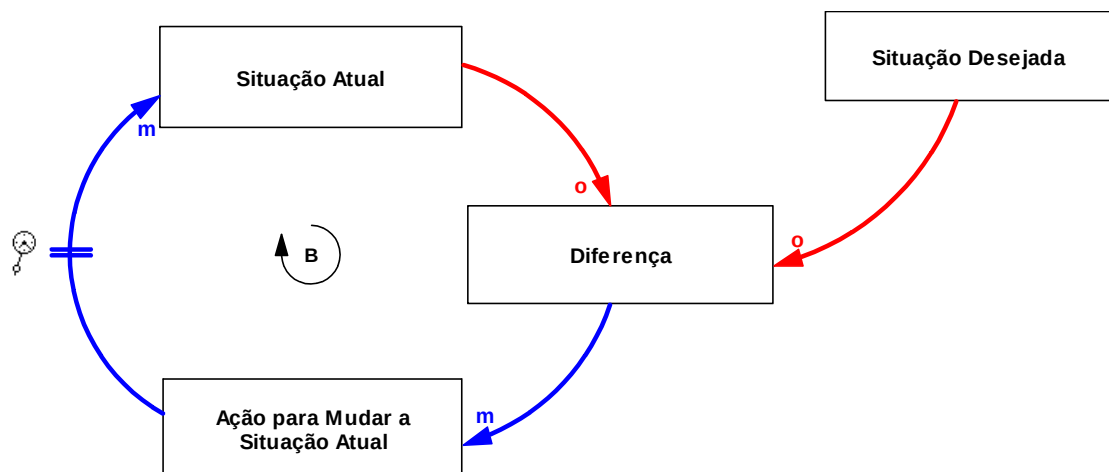


Figura 7 Estrutura genérica do arquétipo Balanceamento com Retardo. Fonte: [VALENÇA, 1999]

Neste arquétipo, há uma *Diferença* entre a *Situação Desejada* e a *Situação Atual* (onde a *Situação Atual* é menor que a *Situação Desejada*), que provoca uma *Ação para Mudar a Situação Atual* que, por sua vez, atua no sentido de reduzir a *Diferença*. O tempo entre a *Ação* e a *mudança na Situação Atual* provoca um comportamento oscilante. Isto é, como a *Situação Atual* não muda imediatamente, continuam as ações no sentido de mudá-la e, quando finalmente os efeitos da ação começam a atuar, a *Situação Atual* se torna maior que a *Situação Desejada* e o ciclo passa a funcionar no sentido de igualar as situações. As oscilações vão diminuindo até que ambos os valores se igualem. Quanto maior o tempo de resposta do sistema, maior o tempo de oscilação.

2.5.2 Limite ao Crescimento

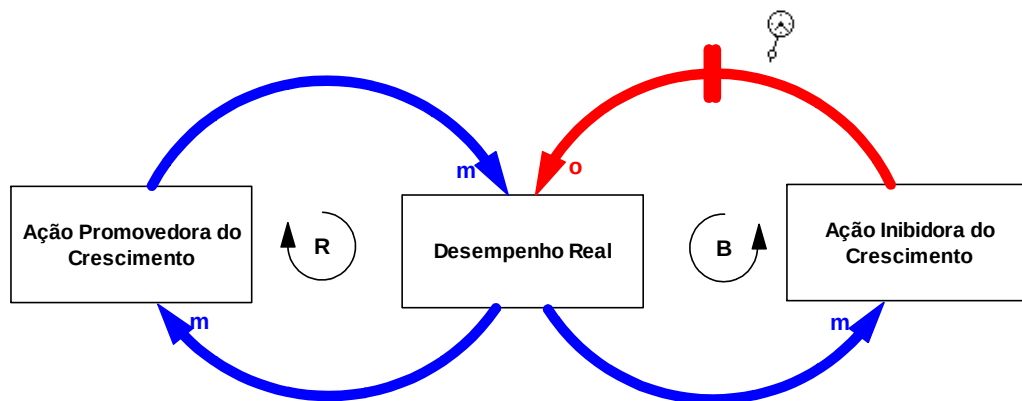


Figura 8 Estrutura genérica do arquétipo Limite ao Crescimento. Fonte: [VALENÇA, 1999]

Este arquétipo apresenta uma *Ação Promovedora do Crescimento* que atua no sentido de reforçar o *Desempenho Real*. A melhoria no *Desempenho Real* também favorece a *Ação Promovedora do Crescimento*, gerando assim um ciclo virtuoso de crescimento. Enquanto isso está ocorrendo, o *Desempenho Real* se depara com fatores limitantes (*Condição Limite*) que desaceleram o crescimento, atuando sobre a *Ação Promovedora do Crescimento*.

Em situações onde essa estrutura ocorre geralmente o ciclo de reforço se destaca no início e o de balanceamento quase não é percebido. Quando o *Desempenho Real* atinge um valor significativo, o ciclo de balanceamento começa a atuar de forma mais expressiva. O foco, normalmente, está voltado para o ciclo de reforço, pois este é provavelmente o que produz os resultados desejados. Então, quando a redução do *Desempenho* começa a ocorrer, causa certa confusão e a ação tomada para corrigir o problema geralmente é dar ênfase à ação que promove o crescimento, porém os esforços não mais produzem o resultado esperado e isso gera mais confusão. A estrutura atingiu o ponto em que o próprio *Desempenho* está inibindo um aumento de *Desempenho*.

2.5.3 Sucesso para os Bem-Sucedidos

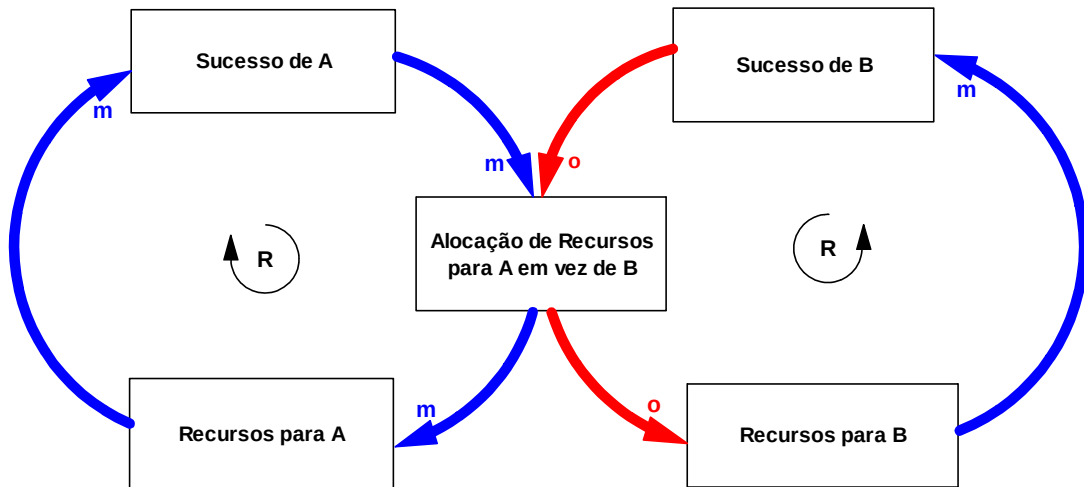


Figura 9 Estrutura genérica do arquétipo Sucesso para os Bem-Sucedidos. Fonte: [VALENÇA, 1999]

Nessa estrutura, consideremos os recursos limitados e que, dessa forma, quando alocados para A em vez de para B, mais *Recursos para A*, mais *Sucesso de A* e mais *Alocação de Recursos para A em vez de para B*. Mesmo obtendo algum recurso, B recebe menos que A, logo tem menos *Sucesso* e terá cada vez menos recursos alocados.

2.5.4 Solução Quebra-Galho

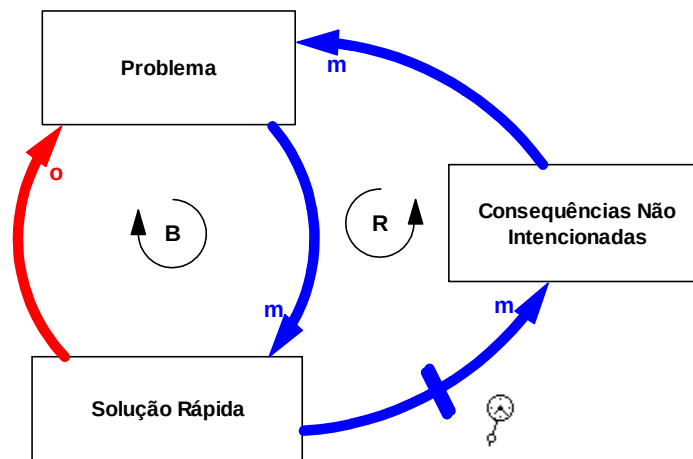


Figura 10 Estrutura genérica do arquétipo Solução Quebra-Galho com Retardo. Fonte: [VALENÇA, 1999]

Essa estrutura apresenta um ciclo de balanceamento que tende a resolver um Problema mediante uma *Solução Rápida* e um ciclo de reforço que, com o passar do tempo, gera *Consequências Não Intencionadas* e faz ressurgir o *Problema*, geralmente de forma mais grave. O que torna essa estrutura preocupante é que o tempo de espera entre a *Solução Rápida*

e as *Conseqüências Não Intencionadas* pode ser grande o suficiente para desassociar o efeito da causa.

2.5.5 Transferência de Responsabilidade

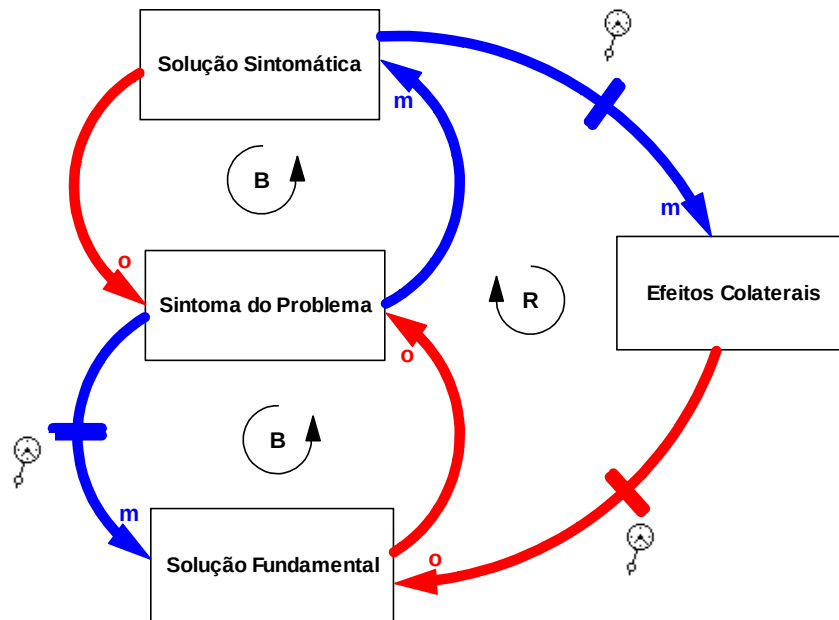


Figura 11 Estrutura genérica do arquétipo Transferência de Responsabilidade. Fonte: [VALENÇA, 1999]

Essa estrutura apresenta dois ciclos de balanceamento e um de reforço. Os dois de balanceamento interagem como se fossem um único ciclo de reforço atuando no mesmo sentido do ciclo de reforço e levando a estrutura como um todo a comportar-se no sentido contrário ao desejado. Quando o *Sintoma do Problema* aparece, a busca pela solução Sintomática é o caminho mais rápido e fácil. Porém, além de aliviar o *Sintoma do Problema* ela também gera *Efeitos Colaterais* que, ao longo do tempo, reduzem a busca da *Solução Fundamental* fazendo com que o *Sintoma do Problema* reapareça.

2.5.6 Deriva de Metas

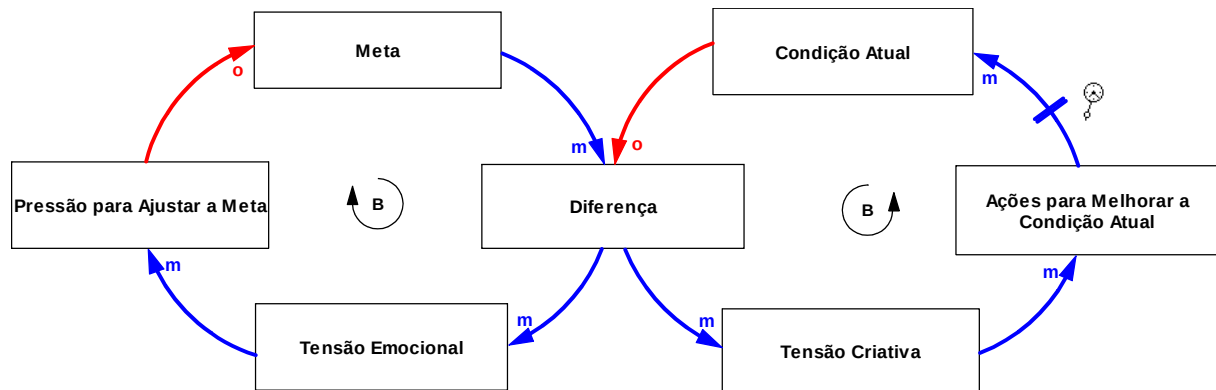


Figura 12 Estrutura genérica do arquétipo Deriva de Metas. Fonte: [VALENÇA, 1999]

Essa estrutura é formada por dois ciclos de balanceamento que interagem de forma que a atividade de um corrói a meta que o outro ciclo tendia a alcançar. Quanto maior a *Diferença* entre a *Meta* desejada e minha *Condição Atual*, maior a *Tensão Emocional* e *Criativa*. A *Tensão Emocional* aumenta a *Pressão para Ajustar as Metas* que diminui a *Meta* e diminui a *Diferença*, diminuindo assim a *Tensão Emocional*. No outro ciclo, o aumento da *Tensão Criativa* leva a um aumento das *Ações para Melhorar a Condição Atual* que, passado algum tempo, aumenta a *Condição Atual* e diminui a *Diferença*, diminuindo a *Tensão Criativa*. Como a melhoria da *Condição Atual* leva algum tempo para ocorrer, o que se verifica é que o ciclo que atua na diminuição da *Meta* é mais incisivo, reduzindo a *Meta* para um patamar inferior. Quanto mais lenta a resposta para melhorar a *Condição Atual*, maior a deriva da *Meta*.

2.5.7 Escalada

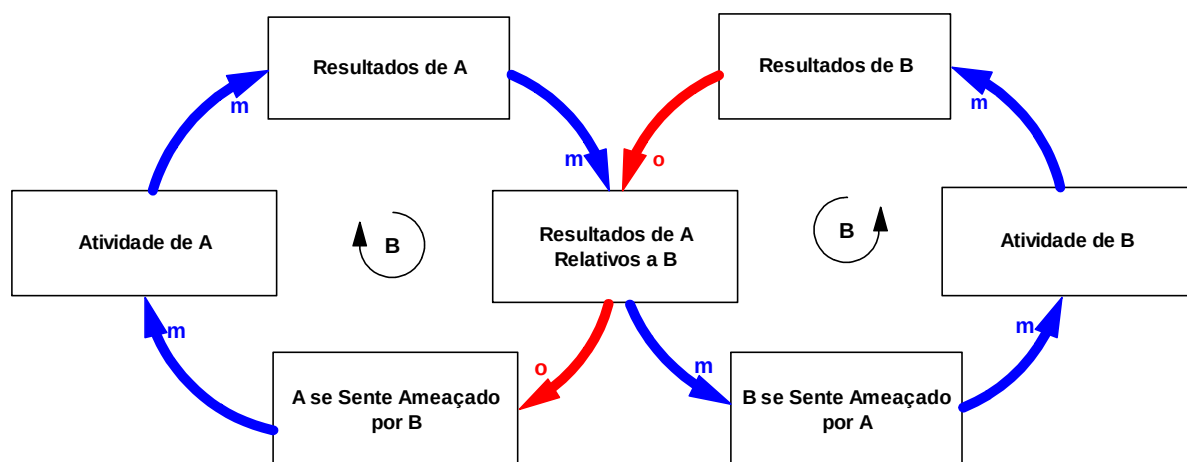


Figura 13 Estrutura genérica do arquétipo Escalada. Fonte: [VALENÇA, 1999]

Esta estrutura apresenta dois ciclos de balanceamento que atuam em conjunto e apresentam um resultado equivalente a um único ciclo de reforço. Se os *Resultados de A e B* são iguais, o *Resultado de A relativo a B* é zero e nada acontece. Porém se os *Resultados de A* são superiores aos de B, isso provocará a seguinte dinâmica: mais *Resultados de A Relativo a B*, menos *A se sente ameaçado de por B* e mais *B se sente ameaçado por A*, menos *Atividade de A* e mais *Atividade de B*, menos *Resultados de A* e mais *Resultado de B*, menos *Resultados de A Relativo a B*, o que provoca uma reação de A na rodada seguinte, seguida de uma reação de B e assim por diante.

2.5.8 Crescimento e Subinvestimento

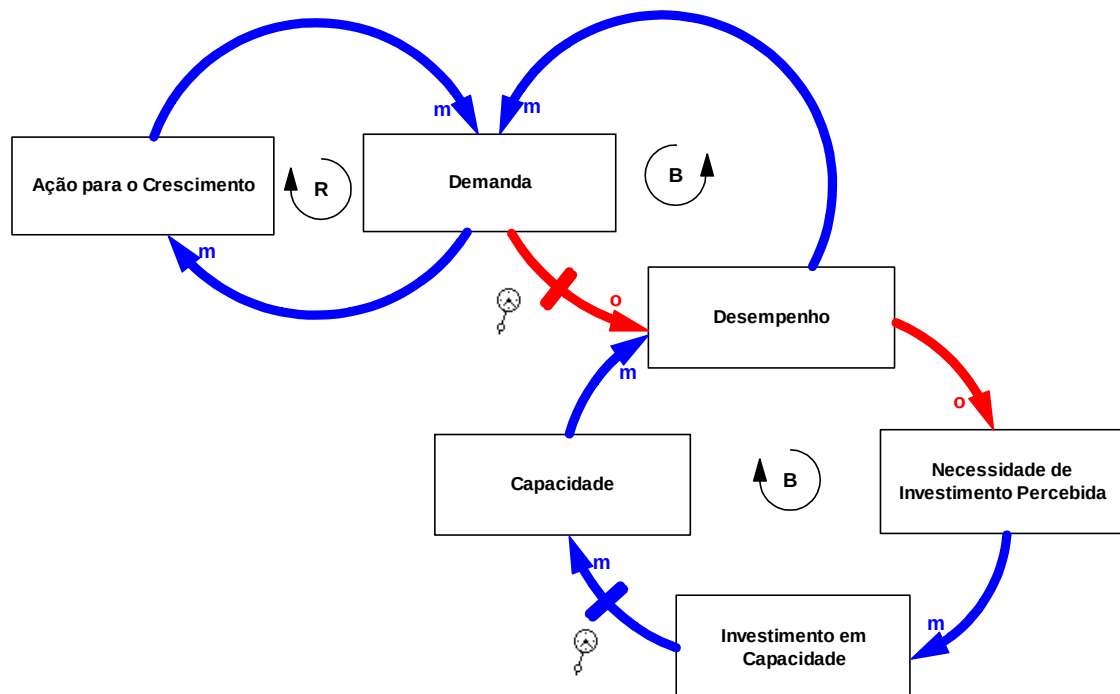


Figura 14 Estrutura genérica do arquétipo Balanceamento com Retardo. Fonte: [VALENÇA, 1999]

O comportamento desse arquétipo é semelhante ao do *Limite ao Crescimento*. Esse arquétipo caracteriza-se por um ciclo de reforço e dois de balanceamento. No ciclo de reforço, a *Ação para o Crescimento* aumenta a *Demanda* que, por sua vez, reforça a *Ação para o Crescimento*. No primeiro ciclo de balanceamento, o aumento na *Demanda* leva a uma redução no *Desempenho*, que reduz a *Demanda*. No segundo ciclo de balanceamento, a redução do *Desempenho* mostra a *Necessidade de Investimento Percebida*, que leva a mais *Investimento em Capacidade* e a um melhor *Desempenho*.

2.5.9 Tragédia do Fator Comum

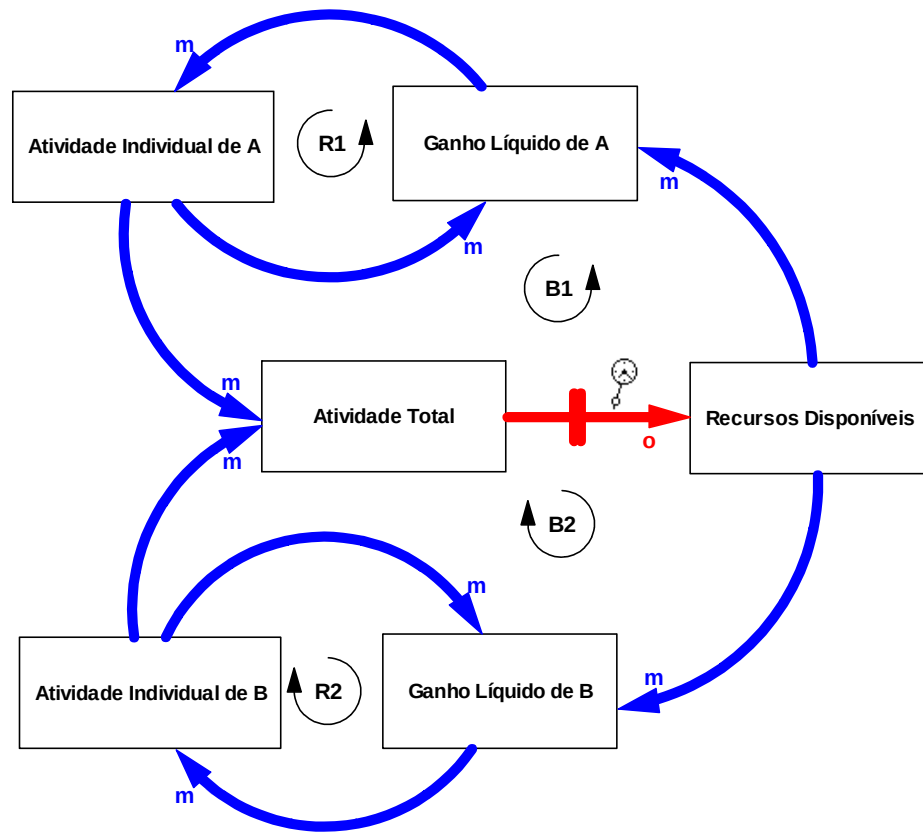


Figura 15 Estrutura genérica do arquétipo Tragédia do Fator Comum. Fonte: [VALENÇA, 1999]

Esse arquétipo apresenta duas estruturas de Limitação ao Crescimento unidas pelo mesmo fator limitante: o recurso, ou seja, o fator comum a ambas. São notados dois ciclos de reforço formados por: *Atividade Individual de A* e *Ganho Líquido de A*, e *Atividade Individual de B* e *Ganho Líquido de B*. As atividades individuais, por sua vez, atuam sobre a *Atividade Total* que, como o passar do tempo, diminui os *Recursos Disponíveis*, reduzindo assim o *Ganho líquido de A e B* que dependia dos *Recursos Disponíveis*, cessando dessa forma o crescimento da *Atividade Individual de A e B*.

2.5.10 Adversários Acidentais

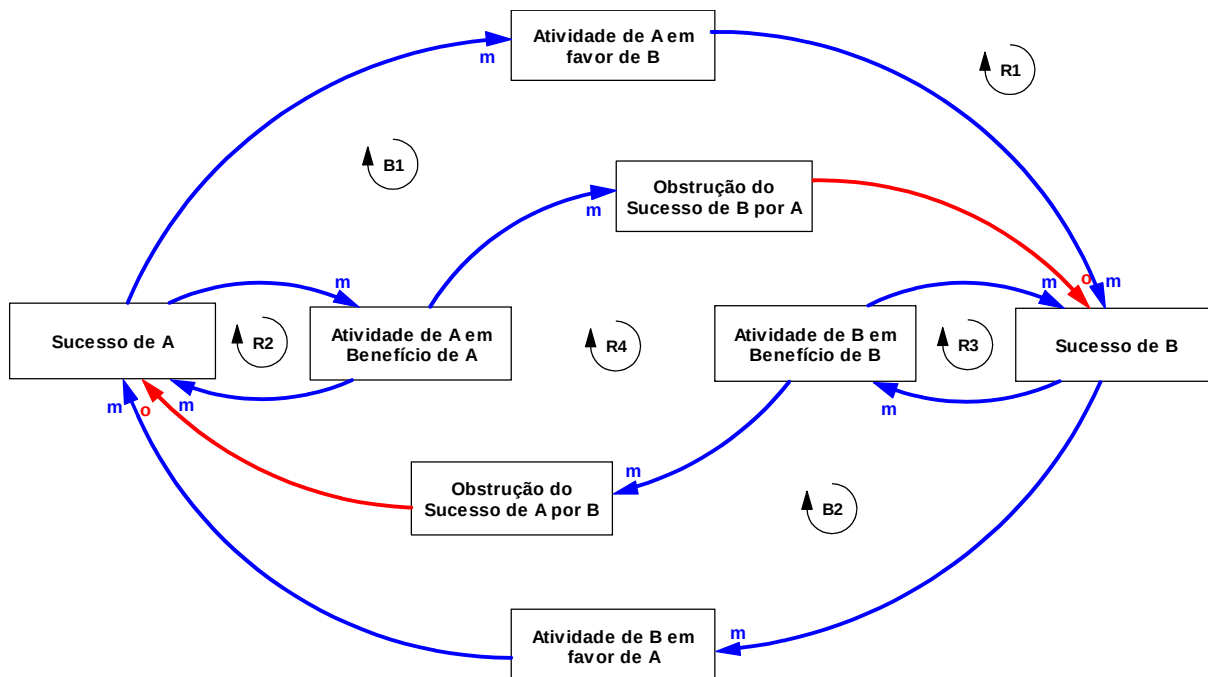


Figura 16 Estrutura genérica do arquétipo Adversários Acidentais. Fonte: [VALENÇA, 1999]

A estrutura do arquétipo Adversários Acidentais é composta de quatro ciclos de reforço e dois ciclos de balanceamento. O ciclo mais externo (R1), contém as variáveis: o *Sucesso de A*, *Atividade de A para o Sucesso de B*, *Sucesso de B* e *Atividade de B para o sucesso de A*, o que representa um ciclo de reforço cooperativo. Estas variáveis também estão promovendo seus próprios crescimentos, estando representadas pelos dois ciclos de reforço internos R2 e R3, contendo: o *Sucesso de A* e *Atividade de A em Benefício de A*, e *Sucesso de B* e *Atividade de B em Benefício de B*. No entanto, estas atividades, atuando em benefício próprio, sem o desejar, podem diminuir o sucesso do outro (parceiro). Vê-se este paradoxo nos ciclos (B1) e (B2). A *Atividade de A em Benefício de A* causa a *obstrução do Sucesso de B por A* que por seu turno diminui o *Sucesso de B*, reduzindo a *Atividade de B em favor de A* e, conseqüentemente, o *Sucesso de A* fechando-se o ciclo (B2). No ciclo (B1) ocorre o mesmo processo com o *Sucesso de B*. O último ciclo desse arquétipo revela um ciclo de reforço (R4) que mostra como A e B acabam se tornando adversários em vez de parceiros, isto é, o sucesso de um diminui o sucesso do outro. Exemplo: quanto mais *Sucesso de A* mais *Atividade de A em Benefício de A* e mais *Obstrução do Sucesso de B por A* que também diminui o *Sucesso de B* e reduz a *Atividade de B em Benefício de B*, diminuindo a *Obstrução do Sucesso de A por B* que, finalmente, promove o aumento o *Sucesso de A*.

2.5.11 Princípio da Atratividade

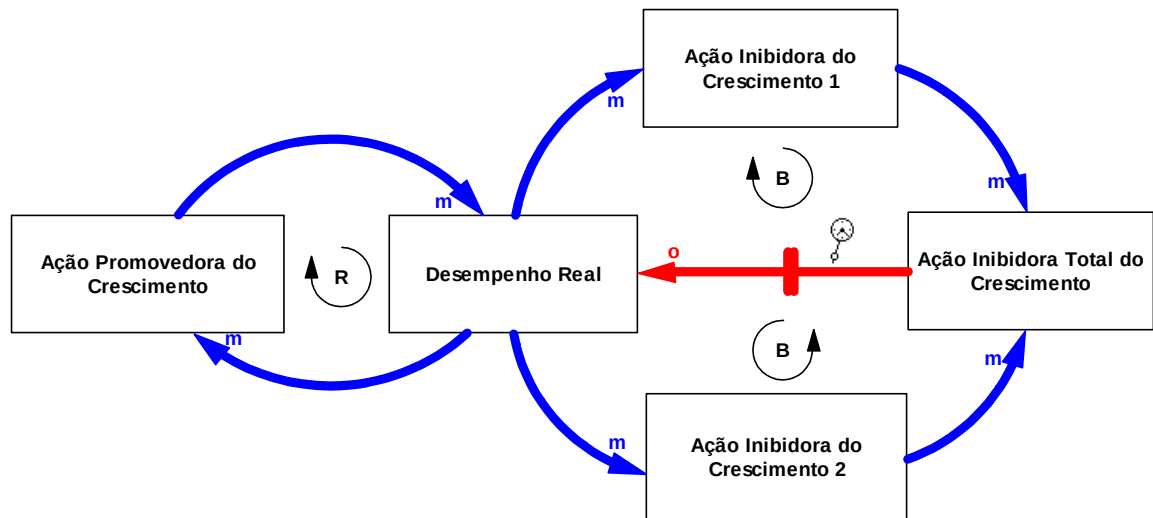


Figura 17 Estrutura genérica do arquétipo Princípio da Atratividade. Fonte: [VALENÇA, 1999]

Essa estrutura apresenta uma *Ação Promovedora do Crescimento* que atua no sentido de reforçar o *Desempenho Real*. A melhoria no *Desempenho* também favorece a *Ação Promovedora do Crescimento*, gerando assim um ciclo virtuoso de crescimento. Enquanto isso está ocorrendo, o *Desempenho* se depara com condições limites que desaceleram o crescimento levando às *Ações Inibidoras do Crescimento*.

3 UMA VISÃO GERAL DA ENGENHARIA DE REQUISITOS

A Engenharia de Requisitos é o processo de aquisição, refinamento e verificação das necessidades do cliente para um sistema de software, objetivando-se ter uma especificação completa e correta dos requisitos de software [IEEE STD830, 1984]. Uma das primeiras medidas de sucesso de um sistema de software é verificar se ele atende às necessidades dos *stakeholders* [NUSEIBEH, 2000]. Portanto, pode-se considerar a Engenharia de Requisitos como uma das etapas mais importantes do processo de Engenharia de Software.

Segundo Kotonya [KOTONYA, 1997], um requisito pode descrever:

- Uma facilidade no nível do usuário; por exemplo, um corretor de gramática e ortografia.
- Uma propriedade muito geral do sistema; por exemplo, o sigilo de informações sem a devida autorização.
- Uma restrição específica no sistema; por exemplo, o tempo de varredura de um sensor.
- Uma restrição no desenvolvimento do sistema; por exemplo, a linguagem que deverá ser utilizada para o desenvolvimento do sistema.

Os requisitos podem ser classificados como **funcionais**, **não-funcionais** e **organizacionais** [KOTONYA, 1997] [LOUCOPOULOS, 1995]. Os requisitos funcionais correspondem às funcionalidades que o sistema de software deve satisfazer, ou seja, descrevem o que o sistema deve fazer. Os requisitos não-funcionais estão associados a aspectos de qualidade e restrições, aspectos de desempenho, aspectos sociais e políticos, segurança, usabilidade, etc., ou seja, fixam restrições sobre como os requisitos funcionais deverão ser implementados. Os requisitos organizacionais dizem respeito à estrutura da organização, ou seja, metas, filosofia e políticas adotadas, seus funcionários e suas expectativas, entre outros [ALENCAR, 1999].

Na realidade das organizações, há um grande hiato entre as reais necessidades do usuário e os requisitos do usuário [KUJALA, 2001]. Isto se deve tanto à falta de experiência dos engenheiros de requisitos quanto à utilização de técnicas ineficientes e/ou insuficientes na fase de elicitação de requisitos [COULIN, 2005]. Isto, segundo Bell [BELL, 1996], faz com

que os requisitos coletados sejam incompletos, inconsistentes e ambíguos, podendo impactar de forma negativa no desenvolvimento e na qualidade do produto final.

A Engenharia de Requisitos é uma área ampla e multidisciplinar, onde aspectos sociais e humanos desempenham um importante papel no processo de engenharia de requisitos [ZAVE, 1997] [NUSEIBEH, 2000]. Abaixo, este processo é abordado de forma mais detalhada.

3.1 O Processo de Engenharia de Requisitos

Na literatura, há vários modelos de Processo de Engenharia de Requisitos que apóiam o desenvolvimento de software [CONSTANTINE, 1999] [KOTONYA, 1997] [SOMMERVILLE, 1997]. Logo, não existe um modelo único para tratar dos requisitos e cada empresa deve adequar os modelos existentes à sua realidade.

Neste trabalho, o modelo de Processo de Engenharia de Requisitos utilizado como referência será o espiral, proposto por Kotonya [KOTONYA, 1997]. Logo, o processo é composto pelas seguintes etapas: elicitação, análise e negociação, documentação e validação de requisitos. Trata-se de um processo iterativo e incremental que contempla todo o ciclo de desenvolvimento de sistemas de software. A figura 18 ilustra o modelo espiral, com suas atividades e sua característica iterativa e incremental bem evidente.

Cada uma dessas atividades do modelo espiral é repetida até que seja tomada a decisão de aceitação do documento de requisitos. Abaixo, cada atividade do modelo é devidamente detalhada.

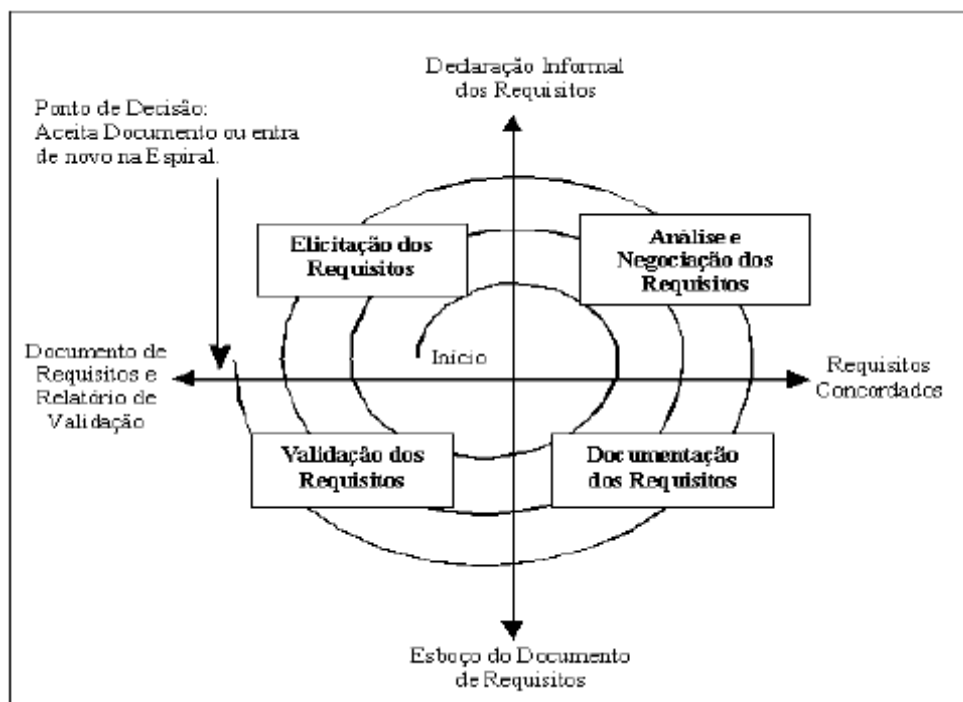


Figura 18 Atividades do Processo de Engenharia de Requisitos no modelo espiral. Fonte: [KOTONYA, 1997]

3.1.1 Elicitação de Requisitos

Etapa inicial do Processo de Engenharia de Requisitos, objetiva levantar conhecimento relevante sobre o problema a ser resolvido. Geralmente requer interação direta com clientes e usuários finais do sistema para a realização da coleta dos requisitos. Além de identificar as necessidades dos usuários, esta atividade também requer uma cuidadosa análise da organização, do domínio da aplicação e dos processos organizacionais [KOTONYA, 1997].

Uma grande dificuldade nesta fase é que os usuários nem sempre têm uma idéia precisa do sistema a ser desenvolvido, ou então apresentam dificuldades em descrever seu conhecimento sobre o domínio do problema. Logo, um dos principais problemas encontrados durante esta etapa é a dificuldade de entender as reais necessidades dos usuários [CASTRO, 1995] [NUSEIBEH, 2000]. No entanto, existem técnicas que buscam reduzir estas dificuldades durante a elicitação dos requisitos do sistema de software. Uma delas, inclusive bastante utilizada, é a técnica de cenários, utilizada para representar as tarefas que os usuários executam normalmente e aquelas que eles desejam executar [LEITE, 1997] [SCHNEIDER, 1998].

Segundo Nuseibeh [NUSEIBEH, 2000], as técnicas para elicitação de requisitos podem ser classificadas como:

- **Técnicas tradicionais:** questionários, entrevistas, análise de documentação existente [WEIRINGA, 1996] [MAMANI, 1999].
- **Técnicas de elicitação de grupo:** são técnicas de dinâmica de grupo com o objetivo de entender de forma mais detalhada as necessidades dos usuários; por exemplo, *brainstorming*.
- **Prototipação:** utilizada para elicitar requisitos quando há um alto grau de incerteza ou quando é necessário um rápido *feedback* dos usuários [DAVIS, 1992].
- **Técnicas de Modelagem:** fornecem um modelo específico das informações que serão adquiridas, e usa esse modelo para orientar o processo de elicitação; por exemplo, KAOS [LAMSWEEERDE, 1998] e I* [YU, 1995] (baseadas em metas) e CREWS [MAIDEN, 1998] (baseada em cenários).
- **Técnicas Cognitivas:** inclui uma série de técnicas originalmente desenvolvidas para aquisição de conhecimento em sistemas baseados em conhecimento; por exemplo, análise de protocolo, *laddering*, *card sorting*, *repertory grids* [SHAW, 1996].
- **Técnicas Contextuais:** surgiu como uma alternativa para as técnicas tradicionais e cognitivas. Inclui técnicas de etnografia e análise social [VILLER, 1999].

De acordo com Patrício [PATRICIO, 2004], importantes contribuições na elicitação de requisitos têm sido dadas pela área de marketing, uma vez que ela lida com aspectos de avaliação e aceitação de produtos ou serviços sob a perspectiva dos consumidores e usuários. A Engenharia de Requisitos tem usado também, mais recentemente, métodos das ciências sociais, como investigação naturalista [POTTS, 1997] e etnografia, buscando melhorar o entendimento das necessidades e objetivos dos usuários, como também obter conhecimentos dos aspectos importantes de interação entre o usuário e o sistema final. A etnografia é uma técnica que envolve o estudo dos usuários em seu ambiente natural, onde engenheiros de requisitos participam ativa ou passivamente das atividades normais dos mesmos, por um longo período de tempo, coletando informações das atividades realizadas [SOMMERVILLE, 1997]. Segundo Zowghi e Coulin [ZOWGHI, 2005], esta técnica é bastante útil na investigação de aspectos relacionados à usabilidade e trabalho cooperativo, onde há necessidade de entender a interação dos usuários com as interfaces do sistema de software.

Uma boa elicitação de requisitos deve conter quatro atividades críticas: (i) **contextualizar objetivos**, que inclui os objetivos gerais do negócio, uma breve descrição do problema a ser resolvido e o porquê do desenvolvimento do sistema; (ii) **aquisição de conhecimentos anteriores**, que inclui informações sobre a organização, sobre o domínio da aplicação e sobre a existência de sistemas semelhantes em uso; (iii) **organização do conhecimento**, que inclui identificação dos *stakeholders* e seus papéis na organização, prioridade de objetivos e possíveis rejeições de requisitos que não contribuem diretamente para o desenvolvimento do sistema; e (iv) **coleta dos requisitos dos stakeholders**, que inclui as consultas realizadas junto aos *stakeholders* para descobrir seus requisitos e derivá-los a partir do domínio da aplicação e ambiente organizacional no qual o sistema será inserido [KOTONYA, 1997].

A escolha da técnica apropriada para elicitar requisitos depende do tempo e dos recursos disponíveis, como também do tipo de informação necessária. Portanto, a elicitação de requisitos trata-se de uma atividade bastante complexa, embora seja fundamental para garantir que o sistema final atenderá as necessidades e desejos dos usuários.

3.1.2 Análise e Negociação de Requisitos

Dentro do ciclo de iteração do modelo espiral, a etapa de análise e negociação de requisitos sucede a elicitação. Mesmo assim, nessa abordagem, nada impede que a análise seja feita em paralelo com a elicitação, pois, durante a coleta, problemas podem ser identificados e a análise seja necessária. Uma vez coletados todos os requisitos, faz-se necessário analisá-los e acordá-los com os *stakeholders*. Nesta etapa deve-se realizar a checagem dos requisitos elicitados para garantir que eles não estejam contraditórios e que nenhuma limitação ou serviço necessários foram esquecidos, averiguar a sua real necessidade e se estes estão alinhados às metas e objetivos de negócio da organização, bem como com o problema a ser abordado [KOTONYA, 1997].

Durante a análise dos requisitos são encontrados problemas como esquecimento, conflito, ambigüidade e duplicação. Com o objetivo de resolver os conflitos dos requisitos, a atividade de negociação dos requisitos busca um acordo das partes interessadas. Em princípio, a negociação dos requisitos deveria ser objetiva, baseada nas necessidades técnicas e organizacionais [KOTONYA, 1997], porém, na realidade, é fortemente influenciada pelas necessidades pessoais dos indivíduos.

Algumas das tarefas da negociação são apresentadas a seguir [KOTONYA, 1997]:

- **Discussão dos requisitos:** consiste do envolvimento dos *stakeholders* para acordar os requisitos conflitantes.
- **Priorização dos requisitos:** consiste em ordenar os requisitos segundo uma escala de prioridades.

Ao final desta etapa espera-se que os requisitos coletados estejam dentro de um grau de conformidade aceitável, ou seja, estejam completos, consistentes e estruturados segundo uma escala de importância e prioridade.

3.1.3 Documentação de Requisitos

Após os conflitos entre os requisitos serem acordados entre os *stakeholders*, faz-se necessária a documentação dos mesmos. Esta etapa objetiva documentar os requisitos de forma clara e objetiva, de forma que seja entendível por qualquer *stakeholder*. De acordo com Pressman [PRESSMAN, 2006], o documento de requisitos é um contrato entre os *stakeholders* e a equipe de desenvolvimento de software. Ryan [RYAN, 1993] diz que o documento de requisitos é o meio através do qual é possível descrever as restrições quanto às características do produto e quanto ao processo de desenvolvimento, a interface com outras aplicações, a descrição sobre o domínio e as informações de suporte ao conhecimento do problema.

A documentação dos requisitos do sistema pode ser realizada tanto através da linguagem natural [AMBRIOLA, 1997], bem como através da lógica [CASTRO, 1996] [ANTONIOU, 1998]. De acordo com Prereira [PEREIRA, 2007], na escolha da melhor forma de documentação dos requisitos, deve-se levar “[...] *em consideração, fundamentalmente, a natureza do sistema de software a ser desenvolvido, bem como a frequência de mudanças e/ou a volatilidade dos requisitos*”.

Qualquer que seja a forma de documentação utilizada é necessário não apenas especificar os requisitos de forma entendível, mas também permitir que eles sejam rastreados e gerenciados ao longo do tempo [GOTEL, 1994]. Segundo Toranzo [TORANZO, 1999], o rastreamento de requisitos é um fator importante para prover uma documentação completa e íntegra, bem como ajudar no processo de gerenciamento de mudanças nesses requisitos.

Logo, a documentação dos requisitos deve, além de servir como contrato entre os desenvolvedores e os *stakeholders*, dar suporte ao gerenciamento e monitoramento dos requisitos, auxiliando no processo de controle de mudanças dos mesmos.

3.1.4 Validação de Requisitos

Segundo Loucopoulus [LOUCOPOULUS, 1995], o processo de validação de requisitos é definido como a atividade na qual certifica que o documento de requisitos é consistente com as necessidades dos usuários. De acordo com Karlsson [KARLSSON, 2003], esta atividade é afetada consideravelmente pela ausência de cliente e usuários finais, principalmente quando um produto é desenvolvido pela primeira vez.

A validação de requisitos está relacionada com a dificuldade de obter um consenso entre diferentes usuários com objetivos conflitantes [LAMSWEEERDE, 1998], logo, pode ser considerada uma atividade um tanto complicada. Segundo Pereira [PEREIRA, 2007], *“técnicas comumente utilizadas pelas empresas para validação de requisitos são prototipação e versões Beta geralmente liberadas em sites proprietários ou sites parceiros”*.

3.1.5 Gerenciamento de Requisitos

Apesar do modelo espiral [KOTONYA, 1997], tomado como base neste trabalho, não considerar o gerenciamento dos requisitos, esta é uma etapa importante no processo de requisitos e será abordada nesta seção. O gerenciamento de requisitos é o processo de compreender e controlar as mudanças nos requisitos do sistema. De acordo com Sommerville [SOMMERVILLE, 2004], durante o estágio de gerenciamento dos requisitos, temos que decidir sobre os seguintes aspectos:

- **Identificação de requisitos:** cada requisito precisa ser identificado de modo único, para que possa ser feita a referência cruzada deste com os outros requisitos e para que ele possa ser utilizado nas avaliações de facilidade de rastreamento.
- **Gerenciamento de mudanças:** trata-se do conjunto de atividades que avalia o impacto e o custo das mudanças.
- **Políticas de facilidade de rastreamento:** políticas que definem as relações entre os requisitos, as relações entre os requisitos e o projeto de sistema que devem ser registrados e também como esses registros devem ser mantidos.

- **Suporte de ferramenta CASE:** o gerenciamento de requisitos envolve processar uma grande quantidade de informações sobre os mesmos. As ferramentas que podem ser utilizadas vão, desde sistemas especializados de gerenciamento de requisitos, até planilhas de cálculo e sistemas simples de bancos de dados.

Segundo Kotonya e Sommerville [KOTONYA, 1997] [SOMMERVILLE, 2004], o gerenciamento de requisitos é uma atividade importante no Processo de Engenharia de Requisitos, pois permite que os requisitos sejam rastreados e gerenciados ao longo da evolução do sistema de software. No entanto, esta atividade não é trivial à medida que necessita de ferramentas específicas para tratar adequadamente as alterações e informações importantes referente à evolução dos requisitos.

4 UMA PROPOSTA DE ANÁLISE SISTÊMICA DOS PROBLEMAS ENFRENTADOS POR EMPRESAS DE SOFTWARE DURANTE O PROCESSO DE REQUISITOS

Os problemas com requisitos de software são amplamente conhecidos por reduzirem a qualidade do software e diminuir a efetividade dos processos de desenvolvimento de software [SOMMERVILLE, 2004]. Segundo Kotonya e Sommerville [KOTONYA, 1997], os problemas ocorridos na etapa de requisitos se transformam nos mais caros de correção nas fases subsequentes do desenvolvimento de software, além de terem influência direta no atraso do prazo de entrega dos projetos, nas estratégias de negócio das empresas e, por decorrência, na satisfação dos clientes.

A literatura oferece pesquisas e artigos que levantam problemas freqüentes relacionados ao processo de requisitos nas empresas de software [HALL, 2002] [ALVES, 2007] [PEREIRA, 2007]. Portanto, é possível realizar um levantamento na literatura dos principais problemas enfrentados pelas empresas de software durante o processo de requisitos e, com base nas pesquisas encontradas, realizar uma análise destes problemas a fim de entendê-los nos detalhes para poder minimizá-los. Esta pesquisa se propõe a realizar uma análise, apoiada na Dinâmica de Sistemas, de alguns dos principais problemas enfrentados pelas empresas de software durante o processo de Engenharia de Requisitos.

Justamente para ampliar e melhorar o espectro de visão das pessoas com relação aos inúmeros problemas da Engenharia de Requisitos, a Dinâmica de Sistemas foi escolhida para embasar este estudo, pois é possível apresentar circuitos de causa-efeito representando o comportamento do sistema como um todo. A formulação conceitual do ocidente sobre o pensamento humano é, por padrão, de natureza linear, apresentando uma visão unilateral e limitada de causalidade. Por outro lado, com a emergência da Dinâmica de Sistemas, houve um enriquecimento nessa formulação conceitual, com a leitura de diagramas de loops de causalidade, com a qual se pode entender a dinâmica que rege o comportamento do sistema como um todo, verificando, não somente, as causas diretas, como também os loops de *feedback* e sua influência de reforço ou balanceamento.

4.1 Problemas Típicos Enfrentados Durante o Processo de Engenharia de Requisitos

As empresas de software já estão bem cientes da relação entre processos de requisitos inadequados e o impacto disto nas organizações, como por exemplo, custos elevados de manutenção [HALL, 2001]. Segundo o Chaos Report 1994 [STANDISH, 1994], um dos principais fatores de sucesso de projetos é a **clareza dos requisitos** (13%), dois dos principais fatores que tornam um projeto desafiador são **especificação e requisitos incompletos** (12,3%) e **mudanças nos requisitos e na especificação** (11,8%), e por fim, o principal fator que leva ao cancelamento do projeto é a **incompletude dos requisitos** (13,1%).

Hall e Beecham [HALL, 2002] realizaram um estudo empírico com 12 empresas de software baseadas no Reino Unido para analisar os problemas enfrentados pelas mesmas durante seus processos de requisitos. Neste estudo, foram identificados problemas com requisitos associados a fatores externos ao processo de requisitos, classificados como **problemas organizacionais**, e problemas com requisitos associados a fatores inerentes ao seu processo de requisitos, classificados como **problemas inerentes ao processo de requisitos**.

A tabela 2 apresenta os principais problemas com requisitos identificados a partir de 45 grupos focais nas 12 organizações.

Tabela 2 Principais problemas com requisitos identificados no estudo de Hall e Beecham.
Fonte: Adaptado de [HALL, 2002]

	Problemas
Problemas inerentes ao processo de requisitos	Requisitos iniciais vagos
	Falta de um processo de requisitos definido
	Crescimento dos requisitos
	Complexidade da aplicação
	Pobre compreensão das necessidades do usuário
	Rastreamento dos requisitos inadequado
Problemas organizacionais	Comunicação dos colaboradores

	Habilidades inapropriadas
	Inadequação de recursos
	Falta de treinamento
	Cultura da organização

De fato, os problemas associados ao processo de requisitos listados no quadro 4.1 acima são muito comuns no desenvolvimento de software. Cada vez mais, atividades complexas têm sido sistematizadas através de aplicações de software e, nem sempre, é possível identificar todos os requisitos envolvidos diante de tal complexidade. Dificuldades de gerenciar requisitos são muito comuns, principalmente quando estes requisitos crescem ao longo do projeto e quando o rastreamento dos requisitos é feito de forma inadequada. Podemos identificar, de certa forma, alguma relação de causalidade entre estes problemas: quanto mais vagos são os requisitos iniciais, maior a possibilidade de haver um crescimento dos requisitos no andamento do projeto, como também, maior a dificuldade em rastreá-los.

Há um estudo empírico realizado por um grupo de pesquisa do Centro de Informática da UFPE que analisa as práticas de Engenharia de Requisitos de 13 empresas de pacotes de software sediadas em Recife-Pernambuco [ALVES, 2007] [PEREIRA, 2007]. Nesta pesquisa foram levantados os principais problemas enfrentados pelas organizações durante o Processo de Engenharia de Requisitos. A tabela 3 apresenta os principais problemas diagnosticados na pesquisa juntamente com suas respectivas descrições.

Tabela 3 Principais problemas no Processo de Engenharia de Requisitos identificados no estudo de Alves e Pereira. Fonte: Adaptado de [PEREIRA, 2007]

Problema	Descrição
Falta de um Processo de Engenharia de Requisitos Definido	As empresas de software não possuem um Processo de Engenharia de Requisitos bem definido, seguindo um processo <i>ad-hoc</i> , subestimando (ou mesmo ignorando) os benefícios trazidos por um processo bem definido.

Dificuldade de entender as reais necessidades do usuário	As empresas de software sentem dificuldade em extrair, de fato, o que o cliente quer, seja por deficiência na comunicação ou por inexperiência dos analistas em descrever as reais necessidades do cliente.
Dificuldade de Interação com Clientes e Usuários	As empresas de software sentem dificuldade em interagir com os clientes e usuários em função da distância física dos mesmos, forçando, muitas vezes, a um contato virtual que, nem sempre, é eficaz.
Dificuldade em Gerenciar Requisitos	As empresas de software muita dificuldade no gerenciamento e controle dos requisitos devido às grandes e complexas demandas, ou por não utilizarem alguma ferramenta específica.

Apesar das pesquisas acima citadas terem sido realizadas em períodos distintos, com empresas de características diferentes e baseadas em locais de realidades culturais e mercadológicas completamente distintas, podemos observar que há, em ambas, problemas em comum citados como críticos durante o processo de Engenharia de Requisitos. Neste contexto, essas pesquisas serão tomadas como base para nossa análise e abordaremos adiante alguns destes problemas sob a ótica da Dinâmica de Sistemas.

4.2 Adaptação da Modelagem *Soft* da Dinâmica de Sistemas

Na realização desta análise, realizaremos uma adaptação do método proposto por Senge [SENGE, 2001], explicado na seção 2.4.1, para a realização da modelagem *Soft* da Dinâmica de Sistemas. A figura 19, mostra as etapas realizadas na análise.



Figura 19 Etapas do método de análise dos problemas com requisitos. Fonte: Autor

A primeira etapa consiste na Definição do Problema a ser analisado, portanto, a partir dos resultados das pesquisas citadas na seção anterior, definimos como problemas interessantes de análise: (I) Falta de um Processo de Engenharia de Requisitos Definido e (II) Dificuldade em Gerenciar Requisitos. Estes problemas foram os escolhidos para análise por serem muito críticos e muito freqüentes nas organizações. Uma modelagem dos mesmos sob a ótica da Dinâmica de Sistemas pode gerar insights e reflexões interessantes sobre os mesmos, de modo que a os problemas sejam mais bem compreendidos e que o aprendizado gerado a partir das reflexões possam embasar estratégias de ação mais eficazes para a minimização e resolução dos mesmos.

Na proposta de Senge [SENGE, 2001], a etapa posterior à definição da situação seria a de Apresentação da História dos Eventos. Principalmente, em virtude do tempo necessário para se fazer esta pesquisa, essa etapa foi substituída pela **Pesquisa Exploratória**, ou seja, pesquisa bibliográfica em livros, periódicos e principalmente nas fontes citadas [ALVES, 2007] [PEREIRA, 2007] e [HALL, 2002]. O objetivo dessa Pesquisa Exploratória é identificar os eventos (fenômenos, fatos, comportamentos, etc.) relevantes relacionados com a situação, citados nas pesquisas apresentadas na literatura.

A terceira etapa consiste na **Identificação das Variáveis** a partir das pesquisas tomadas como base. Consiste, portanto, na identificação de pensamentos-chave e construtos presentes nas descrições dos problemas e no discurso dos entrevistados, quando disponíveis na pesquisa. Após a definição destas variáveis, faz-se necessária a **Identificação das Relações Causais** entre elas, segundo Senge, a Identificação das Influências. Nesta etapa, assim como as variáveis, identificamos estas relações a partir das pesquisas tomadas como base para este estudo.

A última etapa é a **Identificação de Modelos Mentais**, cujo objetivo é identificar diagramas de loops de causalidade e arquétipos que reflitam a forma de enxergar a realidade dos atores envolvidos. Os diagramas e arquétipos foram identificados intuitivamente e

refletem situações recorrentes com um comportamento específico associado. A partir da leitura dos arquétipos é possível ter uma compreensão da dinâmica da situação, gerar reflexão e aprendizado.

Nas próximas seções trazemos reflexões sobre os problemas escolhidos para análise sob a ótica da Dinâmica de Sistemas, utilizando este método de análise.

4.3 Análise do Problema 1 - Falta de um Processo de Engenharia de Requisitos Definido

Diante da gama de problemas relacionados à Engenharia de Requisitos, este se trata, sem dúvidas, de um dos mais críticos. Na pesquisa de Alves e Pereira [PEREIRA, 2007], 53% das empresas participantes afirmaram não possuir um processo bem definido de Engenharia de Requisitos. Já na pesquisa de Hall [HALL, 2002], tal problema representa 24% do total de problemas inerentes ao Processo de Engenharia de Requisitos, um número bem relevante considerando que as empresas participantes já possuíam ou estavam em processo de certificação CMM, nos níveis de 1 a 3.

Partindo da origem do problema, um fator de grande influência sobre o mesmo é a crescente demanda por entrega rápida de novas versões dos produtos, ou seja, os clientes exercem uma pressão sobre as empresas de software para verem resolvidos seus problemas rapidamente (através do sistema). Isto leva à promessa por prazos curtos e entrega rápida por parte das empresas de software, com a intenção de diminuir a pressão sofrida. Ao tomarem esta atitude, as empresas de software resolvem parcialmente o problema, diminuindo a pressão pela solução momentaneamente, porém, com o tempo, começam a surgir consequências não intencionadas, resultantes da solução rápida tomada.

Este tipo de situação é caracterizado na Dinâmica de Sistemas pelo Arquétipo da Solução Quebra-Galho, citado na seção 2.5.4. Na situação acima, podemos imediatamente identificar as seguintes variáveis:

1. Pressão do Cliente
2. Promessa de Prazos Curtos e Entregas Rápidas

Entre estas variáveis há relacionamentos de causalidade de modo que quanto maior é a *Pressão do Cliente*, maior é a *Promessa de Prazos Curtos e Entregas Rápidas*, caracterizando

um relacionamento de mesmo sentido. Por outro lado, quanto maior a *Promessa de Prazos Curtos e Entregas Rápidas*, menor é a *Pressão do Cliente*, caracterizando um relacionamento no sentido oposto. Diante disto, temos um ciclo de balanceamento entre estas duas variáveis, como mostra a figura 20.

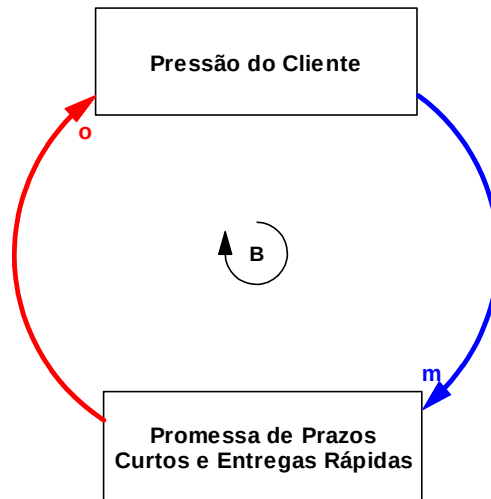


Figura 20 Ciclo de Balanceamento da Solução Quebra-Galho. Fonte: Autor

Diante deste raciocínio as empresas de software conseguem, momentaneamente, diminuir a pressão exercida pelos clientes. Dizemos momentaneamente porque esta solução rápida leva, através do tempo, a consequências não intencionadas como *Má Especificação de Requisitos, Falhas nas Funcionalidades e Baixa Aceitação do Sistema*, uma vez que o esforço para realizar atividades iniciais de entendimento do problema e planejamento são reduzidos. A promessa feita pelas empresas causa um impacto direto no desenvolvimento do sistema, causando estes problemas não previstos. Acontece que o efeito bola-de-neve não pára por aí, estas consequências não intencionadas acabam por gerar mais pressão por parte cliente. Uma vez que ele implanta o sistema e, durante a utilização, identifica tais problemas, torna a fazer pressão para que os problemas dele sejam resolvidos.

Logo, podemos identificar mais algumas relações de causalidade de modo que, a *Promessa de Prazos Curtos e Entregas Rápidas*, incrementa os resultados de *Má Especificação de Requisitos, Falhas nas Funcionalidades e Baixa Aceitação do Sistema*, embora leve algum tempo, caracterizando um relacionamento de mesmo sentido com *delay*. A outra relação, que nem sempre as empresas conseguem identificar, é que quanto mais ocorrem *Má Especificação de Requisitos, Falhas nas Funcionalidades e Baixa Aceitação do Sistema*,

mais acontece *Pressão do Cliente*, caracterizando um relacionamento de mesmo sentido. Como já identificamos anteriormente o relacionamento de mesmo sentido entre *Pressão do Cliente* e *Promessa de Prazos Curtos e Entregas Rápidas*, formamos um ciclo de reforço, como mostra a figura 21.

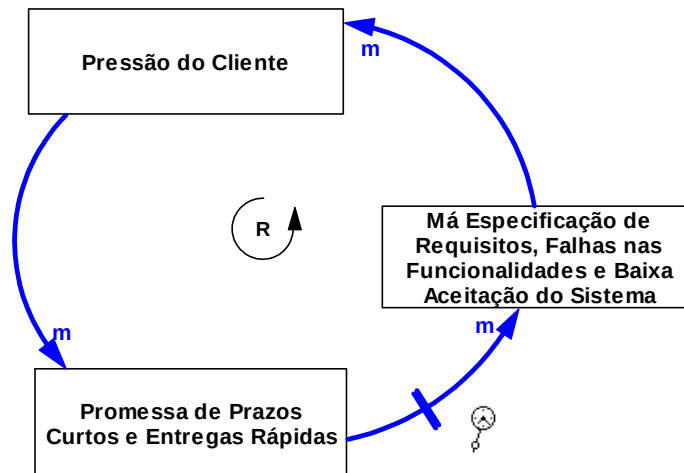


Figura 21 Ciclo de Reforço da Solução Quebra-Galho. Fonte: Autor

Apesar de isoladamente os dois ciclos apresentarem sua dinâmica própria, é a partir da junção destes, que identificamos o Arquétipo da Solução Quebra-Galho, figura 22. Neste tipo de arquétipo, há uma estrutura que apresenta um ciclo de balanceamento que revela a tendência a resolver um Problema (*Pressão do Cliente*) através de uma Solução Rápida (*Promessa de Prazos Curtos e Entregas Rápidas*), e um ciclo de reforço que, com o passar do tempo, gera Conseqüências Não Intencionais (*Má Especificação de Requisitos, Falhas nas Funcionalidades e Baixa Aceitação do Sistema*) e faz ressurgir o Problema (*Pressão do Cliente*), geralmente de forma mais grave.

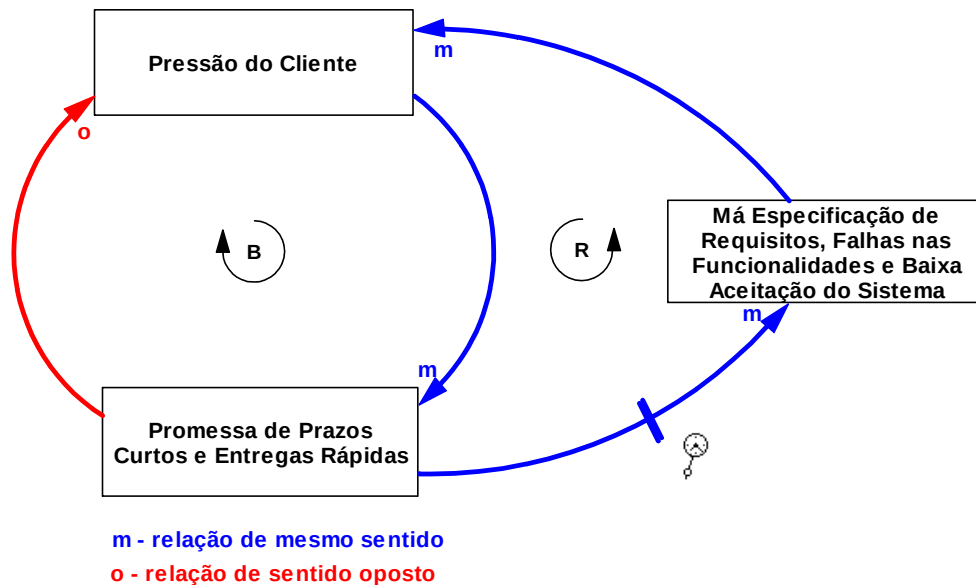


Figura 22 Arquétipo da Solução Quebra-Galho na Falta de Processo de ER Definido. Fonte: Autor

O que torna essa estrutura preocupante é que o tempo de espera entre a Solução Rápida (*Promessa de Prazos Curtos e Entregas Rápidas*) e as Consequências Não Intencionadas (*Má Especificação de Requisitos, Falhas nas Funcionalidades e Baixa Aceitação do Sistema*) pode ser grande o suficiente para desassociar o efeito da causa, tornando, portanto, difícil a identificação do agente causador dessas consequências. Logo, antes de decidir por um curso de ação, deve-se procurar identificar quais as consequências das ações e quais as consequências das consequências. Algumas vezes, consequências imprevistas só aparecem após várias simulações ou planejamento de cenários [VALENÇA, 1999].

De certa forma, a *Má Especificação de Requisitos, Falhas nas Funcionalidades e Baixa Aceitação do Sistema* são consequências um pouco previsíveis desta *Promessa de Prazos Curtos e Entregas Rápidas*, porém não é tão fácil de identificar que estas consequências podem agravar a *Pressão do Cliente* após certo tempo. Segundo Argyris e Schön [ARGYRIS e SCHON, 1978], “as soluções de ontem são os novos problemas de hoje”. De fato, quando não são consideradas as variáveis mais relevantes no processo de tomada de decisão, a possibilidade de uma aparente solução atual se tornar um problema futuro são maiores. Logo, através de um planejamento de cenários e realização de simulações, talvez estas consequências das consequências pudessem ter sido previstas e esta situação poderia ter sido evitada.

Alves e Pereira [PEREIRA, 2007], na sua pesquisa, consideram alguns outros fatores importantes sobre este problema:

"[...] Entretanto, a alta competitividade do setor juntamente com fatores como prazo reduzido e crescente demanda por entrega rápida de novas versões do produto impactam diretamente o processo de desenvolvimento de software. Em particular, a atividade de Engenharia de Requisitos é bastante afetada, pois a estratégia freqüente de empresas de produtos de software é iniciar o desenvolvimento o mais rápido possível, limitando o esforço nas fases de entendimento do problema e planejamento. Se, por um lado, esta estratégia pode garantir que o produto seja entregue dentro do prazo, por outro lado, o produto desenvolvido pode simplesmente não atender às necessidades dos clientes e usuários [...]".

Partindo das considerações acima, vamos agora fazer um alerta às empresas que adotam ou pretendem adotar estratégias de crescimento diante da alta competitividade do setor. Consideremos, inicialmente, as seguintes variáveis para fazermos uma análise da situação:

1. Demanda por Entrega Rápida de Novas Versões do Produto
2. Redução do Prazo de Entrega

Em virtude da alta competitividade, para aumentar sua demanda, as empresas estão reduzindo o seu prazo de entrega. Com isso identificamos as seguintes relações de causalidade: quanto mais ocorre *Redução do Prazo de Entrega*, maior a *Demanda por Entrega Rápida de Novas Versões do Produto*, o que caracteriza um relacionamento de mesmo sentido. Dando continuidade ao raciocínio, neste cenário de alta competitividade, quanto maior a *Demanda por Entrega Rápida de Novas Versões do Produto*, mais haverá *Redução do Prazo de Entrega*, caracterizando um relacionamento de mesmo sentido. Portanto, com a identificação destes relacionamentos, obtivemos um ciclo de reforço que simboliza o crescimento das empresas, como mostra a figura 23.

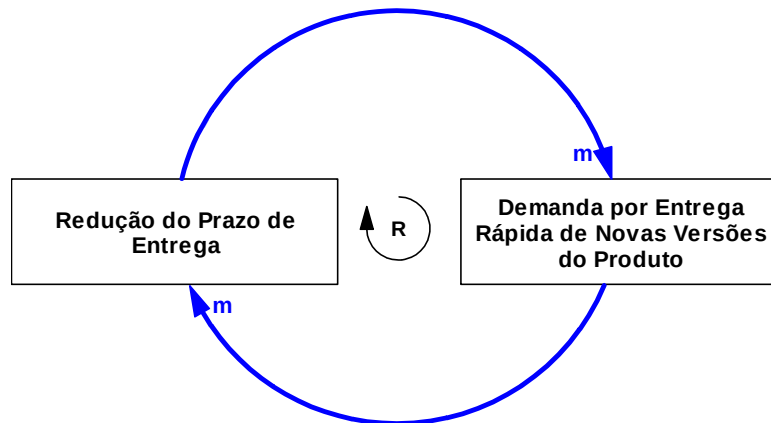


Figura 23 Ciclo de Reforço do Arquétipo de Crescimento e Subinvestimento. Fonte: Autor

Como identificado na pesquisa, a freqüente estratégia das empresas de reduzir os esforços para as atividades iniciais do ciclo de desenvolvimento tem afetado bastante a etapa de Requisitos. Esta redução de esforços pode levar, com o tempo, ao não atendimento das necessidades dos clientes, como consequência de uma etapa de requisitos mal realizada. A partir disto, identificamos um novo relacionamento, pois quanto mais ocorre *Demanda por Entrega Rápida de Novas Versões do Produto*, menos *Satisfação das Necessidades dos Clientes*, uma relação no sentido oposto justificada pela estratégia de redução de esforços para a etapa de Requisitos. Esta perda de desempenho representada pela diminuição na satisfação das necessidades dos clientes, por sua vez, leva a uma perda de demanda, justificada pela insatisfação do cliente diante do não atendimento das suas necessidades e expectativas. Com isto, identificamos um novo relacionamento, desta vez de mesmo sentido, pois quanto menor a *Satisfação das Necessidades dos Clientes*, menor será a *Demanda por Entrega Rápida de Novas Versões do Produto*. Uma vez explicitados estes dois relacionamentos, formamos um ciclo de balanceamento, mostrado na figura 24, que representa um limite ao crescimento das empresas.

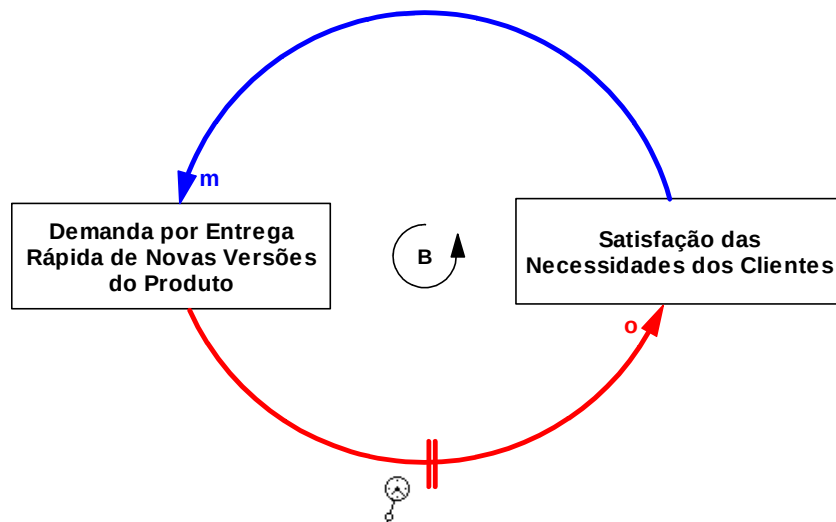


Figura 24 Primeiro Ciclo de Balanceamento do Arquétipo de Crescimento e Subinvestimento. Fonte: Autor

Como vimos anteriormente, a perda do desempenho afeta a demanda das empresas de software. Portanto, isto gera nas mesmas a necessidade de investimento para a melhoria do desempenho, caracterizado pela *Satisfação das Necessidades dos Clientes*, e, conseqüentemente, aumento da demanda. Logo, quanto menos ocorre o *Atendimento das Necessidades dos Clientes*, mais ocorre *Necessidade de Investimento em Atividades de Entendimento do Problema e Planejamento*. Temos, com isto, um relacionamento de sentido oposto entre estas duas variáveis. O investimento mais prudente a fim de melhorar o desempenho das empresas de software se trata da definição de um processo de Engenharia de Requisitos que se adéqüe à sua realidade. Logo, identificamos um relacionamento de mesmo sentido, onde quanto mais *Necessidade de Investimento em Atividades de Entendimento do Problema e Planejamento*, maior a intenção de ter um *Processo de Engenharia de Requisitos Definido*. A partir da definição de um processo específico para as atividades de Engenharia de Requisitos, com o tempo necessário para implantação e assimilação do mesmo pela organização, a capacidade da empresa em identificar as reais necessidades dos clientes será melhorada. Com isto, identificamos mais um relacionamento, onde a *Definição de um Processo Engenharia de Requisitos* aumenta a *Capacidade de Percepção das Necessidades dos Clientes*. Por fim, quanto maior a *Capacidade de Percepção das Necessidades dos Clientes*, maior a *Satisfação das Necessidades dos Clientes*, caracterizando um relacionamento de mesmo sentido entre as duas variáveis. Com estes relacionamentos

identificados, obtemos um novo ciclo de balanceamento, obtido em função da necessidade de investimento para a melhoria do desempenho, como mostra a figura 25.

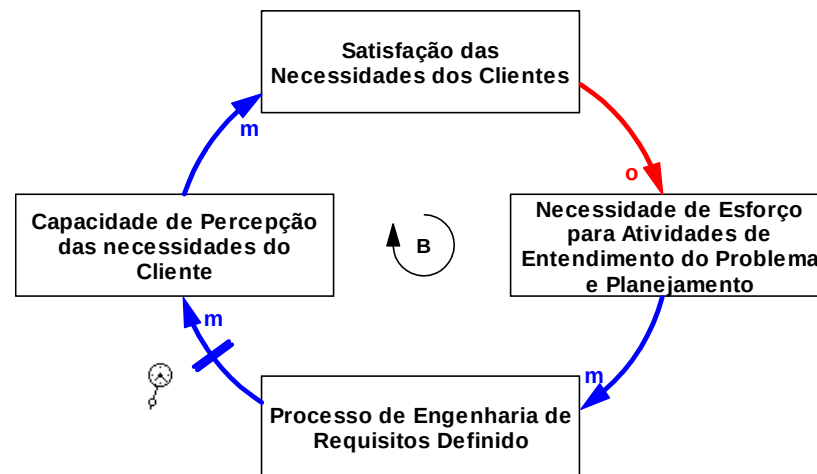


Figura 25 Segundo Ciclo de Balanceamento do Arquétipo de Crescimento e Subinvestimento. Fonte: Autor

Unindo os circuitos de *feedback* das figuras 23, 24 e 25 através das variáveis que eles possuem em comum, identificamos um arquétipo conhecido, na Dinâmica de Sistemas, por Crescimento e Subinvestimento, citado na seção 2.5.8. Neste tipo de arquétipo, há um ciclo de reforço associado ao crescimento (circuito da figura 23), um ciclo de balanceamento que representa um limite ao crescimento pela perda de desempenho (circuito da figura 24), e um ciclo de balanceamento que está associado a uma ação para a melhoria do desempenho (circuito da figura 25). A figura 26 traz a representação gráfica do arquétipo.

O risco apresentado para as empresas de software é que se os investimentos não forem feitos em tempo, as suas vendas podem estar comprometidas a tal ponto que não seja mais possível recuperar a sua imagem, nem tampouco os investimentos realizados. A decisão de investir parece simples, mas, dependendo do nível de comprometimento do desempenho, pode haver o seguinte dilema: se não vende não há dinheiro para investir, se não investe não define um processo de Engenharia de Requisitos, não atende às necessidades dos clientes e, finalmente, não consegue atingir o volume de vendas que precisa para tomar a decisão de investir.

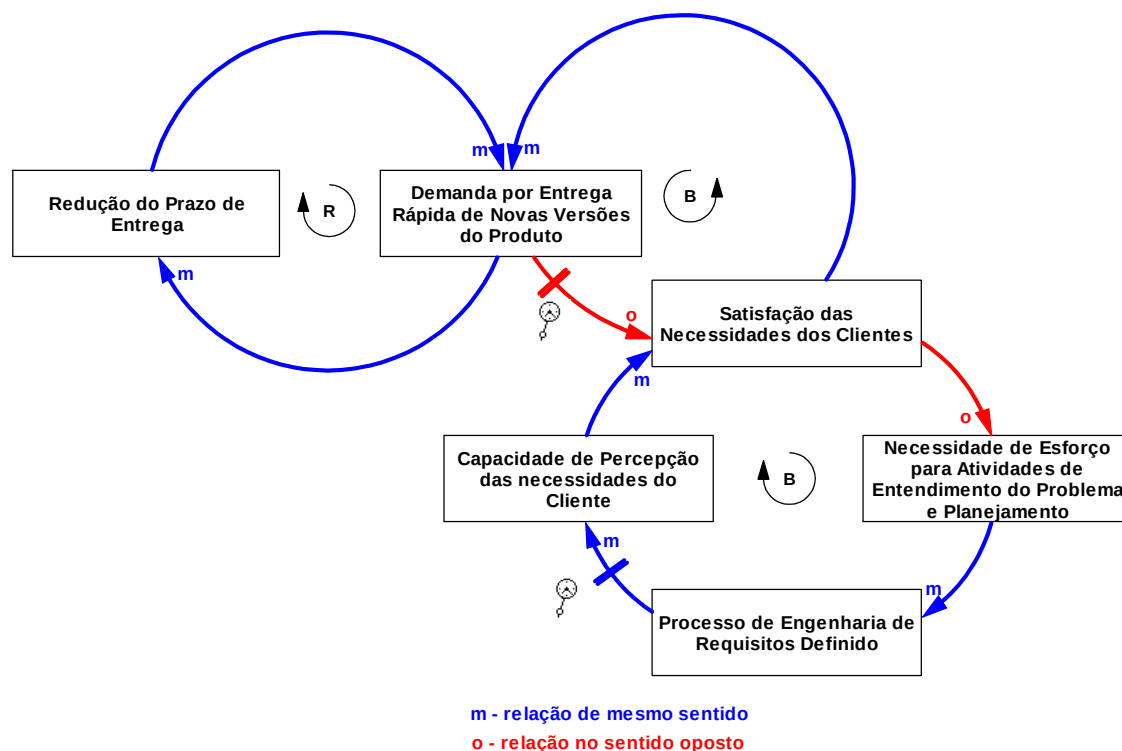


Figura 26 Arquétipo de Crescimento e Subinvestimento na Falta de Processo de ER Definido. Fonte: Autor

Um dos motivos capitais que levam as empresas de software a não adotarem um processo bem definido é porque os analistas consideram as técnicas muito complexas e dispendiosas de tempo. Vários modelos de qualidade de software e normas apresentam um conjunto de boas práticas (e.g. desenvolvimento de procedimentos que possibilitem re-uso de requisitos) em Engenharia de Requisitos que podem ser usadas pelas organizações para enfrentar esses problemas, porém, não há muita disseminação destas boas práticas dentro das empresas de software. Logo, é necessário que se definam processos adequados à realidade e à cultura das empresas, estabelecendo técnicas, ferramentas e procedimentos através dos quais a relação custo x benefício leve a baixo esta idéia distorcida considerada por alguns analistas.

4.4 Análise do Problema 2 - Dificuldade em Gerenciar Requisitos

A atividade de gerenciar requisitos é considerada uma das mais importantes durante o desenvolvimento de software, pois permite que os requisitos sejam rastreados e gerenciados ao longo da evolução do sistema de software. Portanto, um fator de fundamental importância para o gerenciamento dos requisitos trata-se da documentação dos mesmos.

Apesar desta notável importância, Alves e Pereira [ALVES, 2007] [PEREIRA, 2007] identificaram em sua pesquisa que 53,8% das empresas participantes tratam a documentação de requisitos de maneira informal. Uma justificativa é que esta atividade demanda considerável tempo, o que a torna inviável devido ao fato de que as empresas desenvolvedoras de produtos de software focam em práticas ágeis. Um dado também muito crítico da pesquisa é que 30,8% destas empresas acredita que a carência de um documento de requisitos não traz impactos negativos durante o desenvolvimento e/ou venda do software.

Diante desta situação, consideremos as seguintes variáveis:

1. Tempo
2. Alocação de Tempo para o Desenvolvimento
3. Finalização e Venda de Software
4. Alocação de Tempo para Documentação dos Requisitos
5. Gerenciamento dos Requisitos

Primeiramente, podemos notar que as empresas carecem de um recurso precioso que é o *Tempo*. Há a necessidade da divisão deste tempo segundo as suas necessidades e prioridades. Como vimos acima, a maior prioridade é dada para a *Finalização e Venda de Software*, fato que amplia a *Alocação de Tempo para o Desenvolvimento* do software e implica na limitação da *Alocação de tempo para a Documentação dos Requisitos*.

Portanto, identificamos algumas relações de causalidade entre estas variáveis considerando a alocação do recurso limitado de *Tempo*. Quanto mais *Tempo* disponível, maior a *Alocação de Tempo para o Desenvolvimento* do software, caracterizando uma relação de mesmo sentido. Quanto maior a *Alocação de Tempo para o Desenvolvimento*, mais acontece *Finalização e Venda de Software*, o que caracteriza uma relação de mesmo sentido. E, por fim, quanto mais acontece *Finalização e Venda de Software*, mais *Tempo* disponível as empresas possuem. Notemos que, com estas relações identificadas, formamos um ciclo de reforço que representa a alocação do tempo para o desenvolvimento ágil e o sucesso com a finalização e venda de software, como mostra a figura 27.

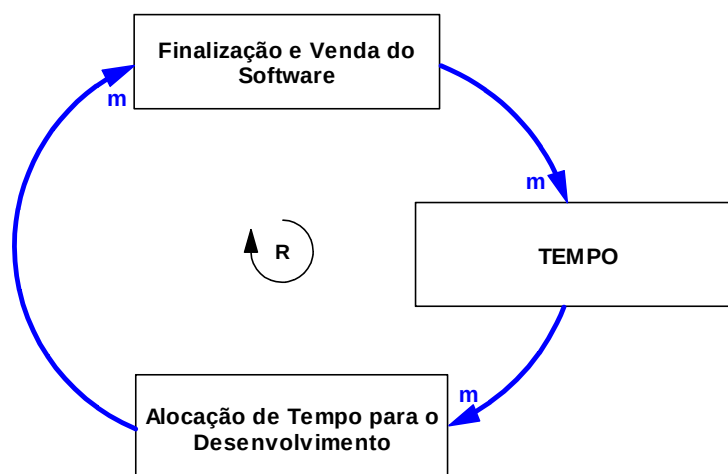


Figura 27 Primeiro Ciclo de Reforço do Arquétipo de Sucesso para os Bem Sucedidos.
Fonte: Autor

Por outro lado, se há prioridade de alocação do recurso de tempo para o desenvolvimento do software, outras atividades carecem deste recurso. Portanto, em virtude da *Alocação de Tempo para Desenvolvimento*, quanto mais *Tempo*, menos *Alocação de Tempo para Documentação dos Requisitos*, caracterizando uma relação no sentido oposto. Dando prosseguimento ao raciocínio, quanto menor a *Alocação de Tempo para Documentação dos Requisitos*, menor será o *Gerenciamento dos Requisitos*, uma vez que havendo uma carência de documentação de requisitos, menor será o material de suporte ao gerenciamento e, conseqüentemente, menor o *Gerenciamento dos Requisitos*. Por fim, quanto menos acontecer o *Gerenciamento dos Requisitos*, mais *Tempo* disponível as empresas possuirão. Com isto formamos um novo ciclo de reforço que representa o insucesso do gerenciamento dos requisitos em virtude da limitada alocação de tempo para a documentação dos requisitos, como mostra a figura 28.

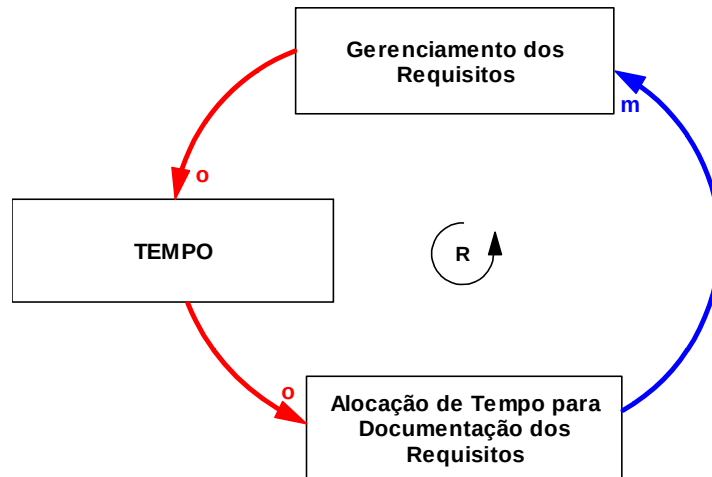


Figura 28 Segundo Ciclo de Reforço do Arquétipo de Sucesso para os Bem Sucedidos.
Fonte: Autor

A partir dos dois ciclos de balanceamento das figuras 27 e 28, unindo-os pela variável comum *Tempo*, identificamos um novo arquétipo da Dinâmica de Sistemas conhecido por Sucesso para os Bem Sucedidos, citado na seção 2.5.3. Neste tipo de arquétipo, devemos considerar que há recursos limitados (*Tempo*), de modo que quando alocados para A (*Desenvolvimento de Software*) em vez de para B (*Documentação dos Requisitos*), implica em mais Sucesso (*Finalização e Venda de Software*) de A (*Desenvolvimento de Software*) e mais Alocação de Recursos (*Tempo*) para A (*Desenvolvimento de Software*) em vez de para B (*Documentação dos Requisitos*). Mesmo obtendo algum recurso (*Tempo*), B (*Documentação dos Requisitos*) recebe menos que A (*Desenvolvimento de Software*), logo tem menos Sucesso (*Gerenciamento dos Requisitos*) e terá cada vez menos recursos (*Tempo*) alocados.

Logo, a competição pelo recurso é o que torna esta estrutura preocupante. A estratégia mais eficaz neste caso é dar uma melhor distribuição aos recursos e desconectar os dois ciclos, de modo que mais sucesso de um não signifique mais redução do sucesso do outro [VALENÇA, 1999].

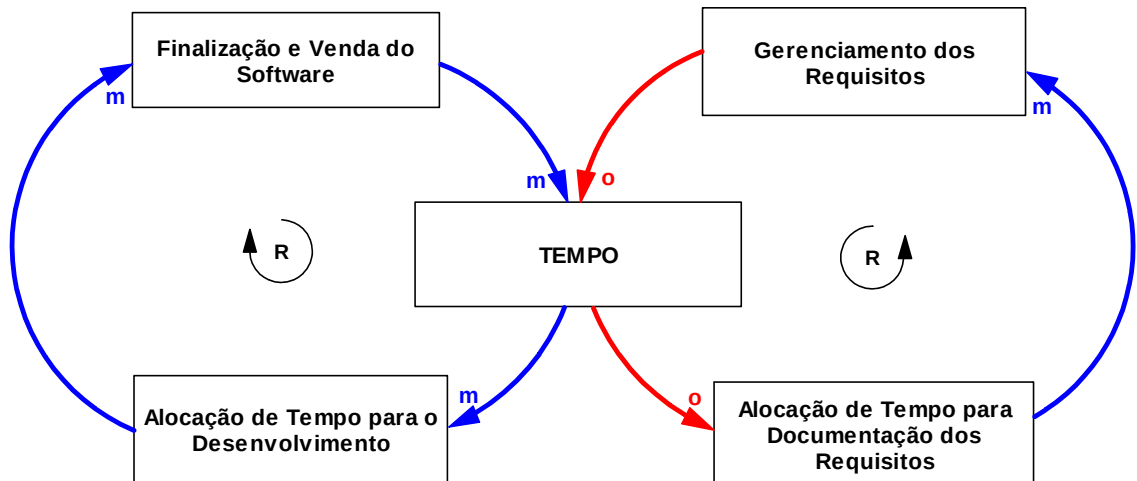


Figura 29 Arquétipo de Sucesso para os Bem Sucedidos na Dificuldade em Gerenciar Requisitos. Fonte: Autor

Como estamos tratando de pequenas e médias empresas neste caso, algo que pode estar causando essa situação de disputa pelo recurso é a superposição de papéis durante o desenvolvimento de software, de modo que a mesma pessoa esteja exercendo outros papéis além do de analista de requisitos. Neste caso, uma estratégia para evitar o insucesso do gerenciamento dos requisitos seria ter a alocação dos papéis para pessoas diferentes, de modo que o recurso de tempo não seja mais um conector entre os dois ciclos. Outra possibilidade seria a devida priorização do tempo para as atividades de requisitos de modo que não prejudique a finalização e comercialização do software, tendo em vista todas as complicações e falhas que podem surgir a partir de uma etapa de requisitos mal realizada.

5 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho procurou realizar uma análise sistêmica dos problemas enfrentados por empresas de software durante o processo de Engenharia de Requisitos, identificando fatores críticos de influência, as relações de causa-efeito entre eles e os arquétipos sistêmicos que melhor representassem a dinâmica das situações em análise. Partindo da análise de pesquisas presentes na literatura sobre tais problemas, este estudo buscou mostrar que:

- a Dinâmica de Sistemas, através das suas técnicas e dos seus conceitos, apresenta-se como uma ferramenta útil para ajudar na compreensão e entendimento dos problemas de requisitos;
- a partir da leitura dos diagramas e arquétipos resultantes da modelagem dos problemas utilizando o método sistêmico, é possível refletir sobre o comportamento das estruturas e entender a dinâmica das relações entre os fatores influentes sobre os problemas, gerando aprendizado e resultando em possíveis estratégias de ação para a resolução dos mesmos;

Pudemos observar que a estratégia de promessa de redução de prazo e tempo de entrega adotada pelas empresas de software, diante da competitividade do mercado e da pressão dos clientes por entregas rápidas, trata-se de uma Solução Quebra-Galho para a Dinâmica de Sistemas. Isto porque, apesar de resolver o problema a curto prazo, finda por agravar o problema em virtude do impacto causado pelas conseqüências não intencionadas relacionadas a redução de esforços para as atividades de requisitos. Pudemos observar também que as empresas podem estar diante de uma situação de Crescimento e Subinvestimento, na qual o limite ao crescimento das mesmas trata-se da satisfação das necessidades dos clientes, afetada em virtude da redução dos esforços para as atividades de requisitos. Para que este limitante não agrave ainda mais os problemas das organizações, é necessário investimento em atividades de Engenharia de Requisitos, mais especificamente a definição de um processo adequado à realidade das mesmas.

5.1 Principais Dificuldades e Limitações Encontradas

A aplicação da modelagem com Dinâmica de Sistemas, que é utilizada em trabalhos relacionados à Engenharia de Produção, mostrou-se desafiadora pelos seguintes motivos:

- Há pouca disponibilidade de trabalhos semelhantes para abordar problemas relacionados a empresas de software.
- A atividade intuitiva de identificação dos arquétipos requer bastante experiência na aplicação dos conceitos, portanto houve bastantes dificuldades na extração dos arquétipos para análise dos problemas.

Outra dificuldade encontrada está relacionada com a identificação de fatores críticos para os problemas e das relações de causalidade entre os mesmos. Baseamo-nos em pesquisas presentes na literatura, porém não tivemos acesso às informações “brutas” dessas pesquisas. Tivemos que identificar estes fatores a partir de informações das pesquisas resultantes de alguma análise primária já realizada. O ideal para este estudo seria, inicialmente, a realização de uma pesquisa empírica para o levantamento destes fatores, para então a realização da modelagem a partir dos dados “brutos”. Porém, em virtude das limitações de tempo o escopo foi reduzido à análise a partir de pesquisas já realizadas.

5.2 Oportunidades para Trabalhos Futuros

Neste estudo utilizamos uma adaptação do método sistêmico para a realização da modelagem dos problemas relacionados ao processo de Engenharia de Requisitos. Uma ótima oportunidade futura trata-se da aplicação do método sistêmico tal e qual sugerido por Peter Senge, para a realização de um diagnóstico com algumas empresas do Porto Digital, na cidade do Recife. A partir deste diagnóstico sistêmico, melhorias no processo de Engenharia de Requisitos poderiam ser propostas, ou até um processo de requisitos adequado para pequenas e médias empresas poderia ser proposto.

De uma maneira geral, a aplicação deste método de modelagem e análise pode ser utilizada em qualquer situação problemática complexa, inclusive em outras áreas referentes ao desenvolvimento de software.

6 REFERÊNCIAS BIBLIOGRÁFICAS

- [ACKOFF, 1974] ACKOFF, R. **General System Theory and Systems Research**, John Wiley & Sons. (1974)
- [ALENCAR, 1999] ALENCAR, F. M. R. **Mapeando a Modelagem Organizacional em Especificações Precisas**. Tese de Doutorado, Centro de Informática, Universidade Federal de Pernambuco. (1999)
- [ALVES, 2001] ALVES, C. **Seleção de Produtos de Software Utilizando uma Abordagem Baseada em Engenharia de Requisitos**. Dissertação de Mestrado em Ciências da Computação. Centro de Informática, Universidade Federal de Pernambuco (2001)
- [ALVES, 2007] Alves, C. Pereira, S. Valença, G. Pimentel, J. Andrade, R. 2007. **Preliminary Results from an Empirical Study in Market-Driven Software Companies**. 10th Workshop of Requirements Engineering, Toronto, Canada.
- [AMBRIOLA, 1997] AMBRIOLA, V. e GERVASI, V. **Processing Natural Language Requirements**. 12th International Conference on Automated Software Engineering. Lake Tahoe, EUA. Nov. (1997)
- [ANDRADE, 2006] ANDRADE, A. L., SELEME, A., RODRIGUES, L. H., SOUTO, R. **Pensamento Sistêmico – Caderno de Campo**. Porto Alegre: Bookman. (2006)
- [ANTONIOU, 1998] ANTONIOU, G. The Role of Non-monotonic Representations in Requirements Engineering. International Journal of Software Engineering and Knowledge Engineering, vol 8. (1998)
- [BELL, 1996] BELL, T. E. e THAYER, T. A. **Software Requirements: Are They Really a Problem?**. Second International Conference on Software Engineering. San Francisco. (1996)
- [BERTALANFFY, 1975] BERTALANFFY, L. V. **Teoria Geral dos Sistemas**, 2ª, Editora Vozes, Petrópolis, RJ. (1975)
- [CASTRO, 1995] CASTRO, J. **The Process of Requirements Formalization: The FORMLAB Project**. Proceedings of Focus Symposium on Information Systems Analysis and Synthesis – ISAS 95. This event was part of the 5th International Symposium on Systems Research, Informatics and Cybernetics – INTERSYMP'95. Aug. 1995, Baden-Baden, Germany, pp. 1-5. (1995)
- [CASTRO, 1996] CASTRO, J. GAUTREAU, J. e TORANZO, M. **Tool Support for Requirements Formalization. Proceedings of the ACM SIGSOFT Viewpoints 96**: International Workshop on

Multiple Perspective Software Development - ACM VP 96. San Francisco, USA, Oct. 1996, pp. 202-206. (1996)

[CHECKLAND, 1994] CHECKLAND, P. **Varieties of Systems Thinking: the Case of Soft Systems Methodology**. System Dynamic Review, Vol. 10, pp. 189-197. (1994)

[CONSTANTINE, 1999] CONSTANTINE, L. e LOCKWOOD, L. A. D. **Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design**. Addison Wesley: Reading, MA. (1999)

[COULIN, 2005] COULIN C. e ZOWGHI D. **What Do Experts Think About Elicitation? – A State of Practice Survey**. Proceedings of the 10th Australian Workshop on Requirements Engineering, Melbourne, Australia, November. (2005)

[DAV92] DAVIS, A. **Operational Prototyping: A New Developme nt Approach**. Software (1992)

[FERNANDES, 2003] FERNANDES, A.C. **Scorecard Dinâmico – Em direção à Intergração Dinâmica de Sistemas com o Balanced Scorecard**. COPPE, UFRJ, RJ. (2003)

[FORRESTER, 1961] FORRESTER, J. W. **Industrial Dynamics**. MIT Press. Cambridge. MA. (1961)

[FORRESTER, 1969] FORRESTER, J. W. **Urban Dynamics**. MIT Press. Cambridge. MA. (1969)

[FORRESTER, 1973] FORRESTER, J. W. **World Dynamics**. MIT Press. Cambridge. MA. (1973)

[FORRESTER, 1976] FORRESTER, J. W. **Principles of Systems**. 2a, Wright Allen Press. Cambridge. MA. (1976)

[GOODMAN, 1989] GOODMAN, M.R. **Study Notes in System Dynamics**. Toolbox Reprint Series, Pegasus Communications, Inc., (2000)

[GORDON, 1969] GORDON G. **System Simulation**. 1a, Prentice-Hall, Nova Jersey (1969)

[GOTEL, 1994] GOTEL, O. e FINKELNSTEIN, A. **An Analysis of the Requireme nts Traceability Problem**. 1st International Conference on Requirements Engineering, Colorado Springs, EUA. Apr. (1994)

[HALL, 2001] HALL, T. BEECHAM, S., BADDOO, N. e RAINER, A. **An Empirical Study of Maintenance Issues within Process Improvement Programmers in the Software Industry**. IEEE Conf Soft. Maint, 2001, in review

- [HALL, 2002] HALL, T. BEECHAM, S. e RAINER, A. **Requirements problems in twelve software companies: an empirical analysis**. IEEE Proceedings: Softw. (2002)
- [IEEE STD830, 1984] IEEE STD. 830. **IEEE Guide to Software Requirements Specification**. The Institute of Electrical and Electronics Engineers, New York, EUA. (1984)
- [KARLSSON, 2003] KARLSSON, L. **Improving Requirements Selection Quality in Market-Driven Software Development**. PhD Thesis. Department of Communication Systems, Lund Institute of Technology. (2003)
- [KIRKWOOD, 1998] KIRKWOOD, C. W. **System Dynamics Methods: A Quick Introduction** (1998) www.public.asu.edu/~kirkwood/sysdyn/SDIntro/SDIntro.htm Acessado em: Outubro de 2007
- [KOTONYA, 1997] KOTONYA, G. e SOMMERVILLE, I. **Requirements Engineering – Processes and Techniques**. John Willy & Sons. (1997)
- [KUJALA, 2001] KUJALA, S., KAUPPINEN, M. e REKOLA, S. **Bridging the Gap between User Needs and User Requirements**. www.cs.tut.fi/~kujala/PCHCI.pdf. Acessado em: Novembro de 2007. (2001)
- [LAMSWEERDE, 1998] LAMSWEERDE, A. DARIMONT, R. e LETIER, E. **Managing Conflicts in Goal-Driven Requirements Engineering**. IEEE Transactions on Software Engineering, Special Issue on Managing Inconsistency in Software Development. Nov. (1998)
- [LEITE, 1997] LEITE, J.C.S.P.; ROSSI, G.; MAIORANA, V.; BALAGUER, F.; KAPLAN, G.; HADAD, G.; OLIVEROS, A. **Enhancing a Requirements Baseline with Scenarios**. Third IEEE International Symposium on Requirements Engineering - IEEE Computer Society Press. Los Alamitos, Ca, USA. 1997. pp. 44 -53. (1997)
- [LOUCOPOULOS, 1995] LOUCOPOULOS, P. e KARAKOSTAS, V. **System Requirements Engineering**. McGraw-Hill Book Company. (1995)
- [MAANI e CAVANA, 2000] MAANI K. E CAVANA, R.Y. **System Dynamic and Modeling: Understanding Change and Complexity**. Pearson Education, Nova Zelândia. (2000)
- [MAIDEN, 1998] MAIDEN, N. **CREWS-SAVRE: Scenarios for Acquiring and Validating Requirements**. Automated Software Engineering. (1998)
- [MAMANI, 1999] MAMANI M. LEITE, J.C.S.P. **Elicit@99 um Protótipo de Ferramenta para a Elicitação de Requisitos**. WER99 - Workshop en Requerimientos / Workshop de Engenharia de Requisitos - II. SADIO. Buenos Aires. PP. 45-55. (1999)

- [MOHAPATRA, 1994] MOHAPATRA P.K.J. MANDAL, P. E., BORA M.C. **Introdução a Modelagem de Dinâmica de Sistemas**, Depto de Engenharia Industrial e Gerenciamento, Instituto de Tecnologia da Índia, Índia. (1994)
- [MONTENEGRO, 2006] MONTENEGRO, A. **A Aplicação de um Modelo de Simulação para o Gerenciamento de Projetos: Um estudo de caso utilizado a Dinâmica de Sistemas**. Departamento de Engenharia Naval e Oceânica, Escola Politécnica da USP, SP. (2006)
- [NUSEIBEH, 2000] NUSEIBEH, B. e EASTERBROOK, S. **Requirements Engineering: A Roadmap**. Proceedings of the 22nd International Conference on Software Engineering. Limerick, Ireland. Jun. (2000)
- [PATRICIO, 2004] PATRICIO, L., CUNHA, J.F., FISK, R.P. e NUNES, N.J. **Customer Experience Requirements for Multi-Platform Service Interaction: Bringing Services Marketing to the Elicitation of User Requirements**. Proceedings of the 12th IEEE International Requirements Engineering Conference, Pages: 26 – 35. September. (2004)
- [PEREIRA, 2007] PEREIRA, S. C. **Um Estudo Empírico sobre Engenharia de Requisitos em Empresas de Produtos de Software**. Dissertação de Mestrado em Ciências da Computação. Centro de Informática, Universidade Federal de Pernambuco (2007)
- [PIDD, 1997] PIDD, M. **Modelagem Empresarial: Ferramentas para tomada de decisão**. Porto Alegre, Bookman. (1997)
- [POTTS, 1997] POTTS, C. e NEWSTETTER, W. C. **Naturalistic Inquiry and Requirements Engineering: Reconciling their Theoretical Foundations**. Proceedings of the Third IEEE International Symposium on Requirements Engineering, Pages 118-127. January. (1997)
- [PRESSMAN, 2006] PRESSMAN, R. S. **Engenharia de Software**. Makron Books. 6ª Edição. (2006)
- [RADZICKI, 1997] RADZICKI, M.J. **Introduction to System Dynamic: A system approach to understanding Complex Policy Issues** (Versão 1.0), US Department of Energy, Disponível em: <http://www.systemdynamics.org/DL-IntroSysDyn/index.html> Acessado em: Outubro de 2007 (1997)
- [RYAN, 1993] RYAN, M. **Defaults in Specifications**. Proceedings of IEEE International Symposium on Requirements Engineering. IEEE Computer Society Press. San Diego, EUA. Jan. (1993)
- [SCHNEIDER, 1998] SCHNEIDER, G. e WINTERS, J. **Applying Use Cases: a practical guide**. Addison- Wesley. (1998)

- [SENGE, 1997] Senge, P. M.; Kleiner, A.; Roberts, C.; Ross, R. e Smith, B. **A quinta disciplina caderno de campo: estratégias e ferramentas para construir uma organização que aprende**. Rio de Janeiro: Qualitymark Ed. (1997)
- [SENGE, 2001] Senge, P. M. **A quinta disciplina. A arte e prática da organização que aprende**. São Paulo: Best Seller. (2001)
- [SHAW, 1996] SHAW, M. e GAINES, B. **Requirements Acquisition**. Software Engineering Journal, vol 11, (1995)
- [SOMMERVILLE, 1997] SOMMERVILLE, I. e SAWYER, P. **Requirements Engineering - A Good Practice Guide**. John Wiley & Sons. (1997)
- [SOMMERVILLE, 2004] SOMMERVILLE, I. **Engenharia de Software**. Addison-Wesley, 6a. edição. (2004)
- [STANDISH, 1994] STANDISH GROUP, **The Chaos Report (1994)**, disponível através da URL http://www.standishgroup.com/sample_research/chaos_1994_1.php, acessado pela última vez em 21/01/2008
- [STERMAN, 2000] STERMAN, J.D. **Business Dynamics – Systems Thinking and Modeling for a Complex World**, McGraw-Hill, 2000.
- [TORANZO, 1999] TORANZO, M. e CASTRO, J. **A Comprehensive Traceability Model to Support the Design of Interactive Systems**. International Workshop on Interactive System Development and Object Models - WISDOM 99. Lisboa, Portugal. Jun. (1999)
- [VALENÇA, 1995] VALENÇA, A. C., **Consultores em Ação – uma Pesquisa sobre Aprendizagem Organizacional**. Recife, Edições Bagaço. (1995)
- [VALENÇA, 1999] VALENÇA, A. C. e GRATULIANO, J., **Pensamento Sistêmico: 25 Aplicações Práticas**. Recife, Edições Bagaço. (1999)
- [VILLER, 1999] VILLER, S. e SOMMERVILLE, I. **Social Analysis in the Requirements Engineering Process: from ethnography to method**. 4th International Symposium on Requirements Engineering, Limerick, Ireland. Jun. (1999)
- [WEIRINGA, 1996] WEIRINGA, R. J. **Requirements Engineering – Frameworks for Understanding**. New York, John Wiley e Sons Ltd. (1996)
- [YU, 1995] YU, E. **Modeling Strategic Relationship for Process Reengineering**. PhD thesis, Computer Science Department, University of Toronto. Toronto, Canada. (1995)

[ZAVE, 1997] ZAVE, P. **Classification of Research Efforts in Requirements Engineering**. ACM Computer Surveys, vol 29, n ° 4. (1997)

[ZOWGHI, 2005] Zowghi, D. and Coulin C. **Requirements Elicitation: A Survey of Techniques, Approaches, and Tools**. Book Chapter in: Engineering and Managing Software Requirements. Edited by Aybuke Aurum and Claes Wohlin. Published by Springer. (2005)